Compact (Targeted Homomorphic) Inner Product Encryption from LWE

Jie Li[1,2,3], Daode Zhang[1,2,3](✉), Xianhui Lu[1,2,3], and Kunpeng Wang[1,2,3]

[1] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

[2] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{lijie,zhangdaode,luxianhui,wangkunpeng}@iie.ac.cn

[3] Science and Technology on Communication Security Laboratory, Beijing, China

Abstract. Inner product encryption (IPE) is a public-key encryption mechanism that supports fine-grained access control. Agrawal et al. (ASIACRYPT 2011) proposed the first IPE scheme from the Learning With Errors (LWE) problem. In their scheme, the public parameter size and ciphertext size are $O(un^2 \log^3 n)$ and $O(un \log^3 n)$, respectively. Then, Xagawa (PKC 2013) proposed the improved scheme with public parameter of size $O(un^2 \log^2 n)$ and ciphertext of size $O(un \log^2 n)$.

In this paper, we construct a more compact IPE scheme under the LWE assumption, which has public parameter of size $O(un^2 \log n)$ and ciphertext of size $O(un \log n)$. Thus our scheme improves the size of Xagawa's IPE scheme by a factor of $\log n$.

Inspired by the idea of Brakerski et al. (TCC 2016), we propose a targeted homomorphic IPE (THIPE) scheme based on our IPE scheme. Compared with Brakerski et al.'s scheme, our THIPE scheme has more compact public parameters and ciphertexts. However, our scheme can only apply to the inner product case, while in their scheme the predicate f can be any efficiently computable polynomial.

Keywords: Inner product encryption · Homomorphic encryption
Learning with errors

1 Introduction

Predicate encryption (PE) is a subclass of functional encryption that supports fine-grained access control. In the PE schemes, a receiver corresponding to the secret key sk_f which is associated with predicate f can decrypt the ciphertext c which is associated with the private attribute x if and only if $f(x) = 0$.

The inner product encryption (IPE) was firstly introduced by Katz et al. [10], which is a special case of PE. In the IPE scheme, the attribute x and predicate f are expressed as vectors $\boldsymbol{x}$ and $\boldsymbol{v}$, and $f(\boldsymbol{x}) = 0$ if and only if $\langle \boldsymbol{x}, \boldsymbol{v} \rangle = 0$. IPE has many useful application scenarios, such as it can support subset, conjunction

S. Qing et al. (Eds.): ICICS 2017, LNCS 10631, pp. 132–140, 2018.
https://doi.org/10.1007/978-3-319-89500-0_11

and range queries on encrypted data [8] and polynomial evaluation, CNF/DNF formulas [10].

At first, the IPE constructions [4,10–16] were based on bilinear groups and constructing IPE scheme from other assumption was left as an open problem. Until 2011, Agrawal et al. [2] proposed the first IPE scheme (denoted by AFV11) from the LWE assumption. One of the drawbacks of the scheme is that it has large sizes of public parameter (i.e., $O(un^2 \log^3 n)$) and ciphertext (i.e., $O(un \log^3 n)$) for $q = poly(n)$, where u is the dimension of the attribute vector, n is the security parameter. For efficiency, Xagawa[1] [17] improved the AFV11 IPE scheme and obtained a more compact IPE scheme (denoted by Xag13) with public parameter of size $O(un^2 \log^2 n)$ and ciphertext of size $O(un \log^2 n)$. Whether we can further compress the public parameter and ciphertext size to get a more compact IPE scheme is an interesting problem.

1.1 Our Contribution

In this paper, we mainly focus on the efficiency of the IPE scheme. We construct a selective security IPE scheme from the LWE assumption with compact parameters. Our scheme has smaller public parameter size (i.e., $O(un^2 \log n)$) and ciphertext size (i.e., $O(un \log n)$) for $q = poly(n)$ and improves both the public parameter size and the ciphertext size by a factor of $O(\log n)$ when compared with Xag13.

In addition, we further note that we can add homomorphic property to our IPE scheme. More formally, by using the technique proposed by Brakerski et al. [6], we obtain a targeted homomorphic IPE (THIPE) scheme which has more compact public parameters and ciphertexts than the scheme in [6] when only consider the inner product case. Note that, in Brakerski et al.'s scheme, the predicate f can be any efficiently computable polynomial.

In Table 1, we give a rough comparison of the sizes of public parameter and ciphertext, the modulus q, the approximate factor among the existing IPE schemes from LWE.

Table 1. Comparison of IPE schemes based on LWE.

IPE	Public parameter size	Ciphertext size	q	Approximation factor
AFV11 [2]	$O(un^2 \log^3 n)$	$O(un \log^3 n)$	$u^2 n^{3.5+5\delta} log^{2.5+2\delta} n$	$u^2 n^{4+5\delta} log^{1.5+\delta} n$
Xag13 [17]	$O(un^2 \log^2 n)$	$O(un \log^2 n)$	$u^2 n^{4.5+4\delta} log^{2.5+2\delta} n$	$u^2 n^{5+4\delta} log^{1.5+\delta} n$
This work	$O(un^2 \log n)$	$O(un \log n)$	$u^2 n^{6.5+4\delta} log^{0.5+2\delta} n$	$u^2 n^{7+4\delta} log^{-0.5+\delta} n$

Where u is the dimension of the attribute vector and $\delta > 0$ is a small constant.

1.2 Overview of Our Construction

Here we give the overview of our scheme. We first review the previous IPE scheme for $u = k\ell$ dimension attribute vector $\boldsymbol{x} = (x_{1,1}, \ldots, x_{1\ell}, \ldots, x_{k,1}, \ldots, x_{k,\ell})$ and

[1] Note that, when only consider the inner product case, the scheme in [7] is just as same as the scheme of Xagawa [17].

predicate vector $\boldsymbol{v} = (v_{1,1}, \ldots, v_{1\ell}, \ldots, v_{k,1}, \ldots, v_{k,\ell})$. We give a brief description of them and then we present our construction. For simplicity, we use the special case of $k = 1$ to demonstrate, that is $\boldsymbol{x} = (x_1, \ldots, x_\ell)$ and $\boldsymbol{v} = (v_1, \ldots, v_\ell)$.

Our Construction. We construct a compact IPE scheme based on [2,17] by using the technique of [1]. Let $\mathbf{G}_{n,2,m}$ be the gadget matrix with base 2 and matrix size $n \times m$. In our construction, we use two gadget matrices $\mathbf{G}_{n\ell,\ell',m}$ and $\mathbf{G}_{n,2,m}$ with different bases and matrix sizes as the critical tool to improve the efficiency.

In our construction, every public matrix can encode ℓ components of $\boldsymbol{x}$, where $\ell = O(\log n)$. That is, for $\boldsymbol{x} = (x_1, \ldots, x_\ell)$ and the corresponding $\mathbf{X}_i = x_i \mathbf{I}_n$ defined as before, let $\mathbf{X} = [\mathbf{X}_1 | \ldots | \mathbf{X}_\ell] \in \mathbb{Z}_q^{n \times n\ell}$, the encryption lattice is defined as

$$\Lambda_{\boldsymbol{x}} = \Lambda_q(\mathbf{A} | \mathbf{A}_1 + \mathbf{X}\mathbf{G}_{n\ell,\ell',m})$$

The corresponding ciphertext is a vector $CT = (\boldsymbol{c}, \boldsymbol{c}_1) \in (\mathbb{Z}_q^m)^2$.

For predicate vector $\boldsymbol{v} = (v_1, \ldots, v_\ell)$ and the corresponding $\mathbf{V}_i = v_i \mathbf{I}_n$ as before, let $\mathbf{V} = \begin{pmatrix} v_1 \mathbf{I}_n \\ v_2 \mathbf{I}_n \\ \vdots \\ v_\ell \mathbf{I}_n \end{pmatrix} \in \mathbb{Z}_q^{n\ell \times n}$, we define the mapping $T_{\boldsymbol{v}} : (\mathbb{Z}_q^m)^2 \to (\mathbb{Z}_q^m)^2$ by

$$T_{\boldsymbol{v}}(\boldsymbol{c}, \boldsymbol{c}_1) = (\boldsymbol{c}, \boldsymbol{c}_1 \mathbf{G}^{-1}_{n\ell,\ell',m}(\mathbf{V}\mathbf{G}_{n,2,m}))$$

We denote $w = \langle \boldsymbol{x}, \boldsymbol{v} \rangle$ and let $\mathbf{W} = w\mathbf{I}_n$. And $T_{\boldsymbol{v}}(\boldsymbol{c}, \boldsymbol{c}_1)$ is a vector close to the lattice

$$\Lambda_{\boldsymbol{v},\boldsymbol{x}} = \Lambda_q(\mathbf{A} | \mathbf{A}_1 \mathbf{G}^{-1}_{n\ell,\ell',m}(\mathbf{V}\mathbf{G}_{n,2,m}) + \mathbf{W}\mathbf{G}_{n,2,m})$$

The secret key $\boldsymbol{r}$ is defined as a short basis of $\Lambda_q^{\perp}(\mathbf{A} | \mathbf{A}_1 \mathbf{G}^{-1}_{n\ell,\ell',m}(\mathbf{V}\mathbf{G}_{n,2,m}))$, so if $\langle \boldsymbol{x}, \boldsymbol{v} \rangle = 0$, then $\mathbf{W} = \boldsymbol{0}$, and thus the secret key $\boldsymbol{r}$ can decrypt the corresponding ciphertext.

Due to the fact that $n\ell \log_{\ell'} q = O(m) = O(n \log q)$, then $\ell = O(\log \ell')$. And ℓ' is a bit decomposition base of modulus $q = poly(n)$, thus $\ell' = O(n)$ and $\ell = O(\log n)$. So it's obvious that our IPE scheme improves the public parameter and ciphertext size by a factor of $\ell = O(\log n)$.

2 Preliminaries

2.1 Predicate Encryption

Predicate Encryption ([10]). For the set of attribute Σ and the class of the predicate $\mathcal{F}$, a predicate encryption scheme consists four algorithm Setup, KeyGen, Enc, Dec which are PPT algorithms such that:

- Setup uses the security parameter λ and outputs the master public key mpk and master secret key msk.
- KeyGen uses the master secret key msk and a predicate $f \in \mathcal{F}$ and outputs a secret key sk_f for f.
- Enc uses the master public key mpk and a attribute $I \in \Sigma$, outputs a ciphertexts C for message $\mu \in \mathcal{M}$.
- Dec takes as input the ciphertexts C and secret key sk_f. If $f(I) = 0$, it outputs μ; if $f(I) = 1$, it outputs a distinguished symbol $\perp$ with all but negligible probability.

Security. We say a PE scheme is weakly attribute hiding in the selective attribute setting if the adversary can't distinguish $\mathsf{Enc}(mpk, I_1, \mu_1)$ and $\mathsf{Enc}(mpk, I_2, \mu_2)$.

The definition of the weakly attribute hiding security is given in [10].

2.2 Lattices

For positive integers n, m, q, and a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, the m-dimensional integer lattices are defined as: $\Lambda_q(\mathbf{A}) = \{\mathbf{y} : \mathbf{y} = \mathbf{A}^{\mathrm{T}}\mathbf{s}$ for some $\mathbf{s} \in \mathbb{Z}^n\}$ and $\Lambda_q^{\perp}(\mathbf{A}) = \{\mathbf{y} : \mathbf{A}\mathbf{y} = \mathbf{0} \mod q\}$.

For $\mathbf{x} \in \Lambda$, define the Gaussian function $\rho_{s,\mathbf{c}}(\mathbf{x})$ over $\Lambda \subseteq \mathbb{Z}^m$ centered at $\mathbf{c} \in \mathbb{R}^m$ with parameter $s > 0$ as $\rho_{s,\mathbf{c}}(\mathbf{x}) = \exp(-\pi||\mathbf{x} - \mathbf{c}||/s^2)$. Let $\rho_{s,\mathbf{c}}(\Lambda) = \sum_{\mathbf{x}\in\Lambda} \rho_{s,\mathbf{c}}(\mathbf{x})$, and define the discrete Gaussian distribution over Λ as $\mathcal{D}_{\Lambda,s,\mathbf{c}}(\mathbf{x}) = \frac{\rho_{s,\mathbf{c}}(\mathbf{x})}{\rho_{s,\mathbf{c}}(\Lambda)}$, where $\mathbf{x} \in \Lambda$. For simplicity, $\rho_{s,\mathbf{0}}$ and $\mathcal{D}_{\Lambda,s,\mathbf{0}}$ are abbreviated as ρ_s and $\mathcal{D}_{\Lambda,s}$, respectively.

Lemma 1. *Let p, q, n, m be positive integers with $q \geq p \geq 2$ and q prime. There exists PPT algorithms such that*

- [3,5]: $\mathsf{TrapGen}(n, m, q)$ *a randomized algorithm that, when $m \geq 6n\lceil \log q \rceil$, outputs a pair $(\mathbf{A}, \mathbf{T_A}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}^{m\times m}$ such that $\mathbf{A}$ is statistically close to uniform in $\mathbb{Z}_q^{n\times m}$ and $\mathbf{T_A}$ is a basis of $\Lambda_q^{\perp}(\mathbf{A})$, satisfying $\|\widetilde{\mathbf{T_A}}\| \leq \mathcal{O}(\sqrt{n\log q})$ with overwhelming probability.*
- [9]: $\mathsf{SampleLeft}(\mathbf{A}, \mathbf{B}, \mathbf{T_A}, \mathbf{u}, s)$ *a randomized algorithm that, given a full rank matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, a matrix $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$, a basis $\mathbf{T_A}$ of $\Lambda_q^{\perp}(\mathbf{A})$, a vector $\mathbf{u} \in \mathbb{Z}_q^n$ and $\sigma \geq \|\widetilde{\mathbf{T_A}}\| \cdot \omega(\sqrt{\log(2m)})$, then outputs a vector $\mathbf{r} \in \mathbb{Z}_q^{2m}$ distributed statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{F}),s}$ where $\mathbf{F} = [\mathbf{A}|\mathbf{B}]$.*

3 Compact Inner Product Encryption from LWE

In this section, we propose a compact IPE scheme from LWE problem. For attribute vector $\boldsymbol{x} = (\mathbb{Z}_q^{\ell})^k$ and predicate vector $\boldsymbol{v} = (\mathbb{Z}_q^{\ell})^k$, we use $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k)$ and $\boldsymbol{v} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$ to denote them respectively and each $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,\ell}), \boldsymbol{v}_i = (v_{i,1}, \ldots, v_{i,\ell}) \in \mathbb{Z}_q^{\ell}$.

3.1 The Construction

Let λ be the security parameter and $u = k\ell$ be the dimension of predicate and attribute vectors. Set lattice parameters $n = n(\lambda), m = m(\lambda), q = q(\lambda)$ and Gaussian parameters $\alpha = \alpha(\lambda), s = s(\lambda)$, define $\ell^{'} = 2^{\ell}$.

- $\mathsf{IPE.Setup}(1^{\lambda})$**:** On input the security parameter λ, do:
  1. Use the algorithm $\mathsf{TrapGen}(n, m, q)$ to generate a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and its trapdoor $\mathbf{T_A}$.
  2. Choose k uniformly random matrix $\mathbf{A}_i \in \mathbb{Z}_q^{n \times m}$ for $i = 1, \ldots, k$ and sample a uniformly random matrix $\mathbf{P} \in \mathbb{Z}_q^{n \times m}$.

  Output $mpk = (\mathbf{A}, \{\mathbf{A}_i\}_{i \in \{1,\ldots,k\}}, \mathbf{P})$ and $msk = \mathbf{T_A}$.
- $\mathsf{IPE.KeyGen}(mpk, msk, \boldsymbol{x})$**:** On input the master public key mpk and master secret key msk, and a predicate vector $\boldsymbol{v} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k) \in (\mathbb{Z}_q^{\ell})^k$ where $\boldsymbol{v}_i = (v_{i,1}, \ldots, v_{i,\ell}) \in \mathbb{Z}_q^{\ell}$, do:
  1. For $i = 1, \ldots, \ell$, compute the matrices $\mathbf{V}_i^{'} := \begin{pmatrix} v_{i,1}\mathbf{I}_n \\ v_{i,2}\mathbf{I}_n \\ \vdots \\ v_{i,\ell}\mathbf{I}_n \end{pmatrix} \in \mathbb{Z}_q^{\ell n \times n}$, and let $\mathbf{V}_i := \mathbf{G}^{-1}_{n\ell,\ell',m}(\mathbf{V}_i^{'} \cdot \mathbf{G}_{n,2,m})$
  2. Define the matrices:
  $$\mathbf{B} := \sum_{i=1}^{k} \mathbf{A}_i \mathbf{V}_i \in \mathbb{Z}_q^{n \times m}$$
  3. Using msk to compute $\mathbf{U} \leftarrow \mathsf{SampleLeft}(\mathbf{A}, \mathbf{B}, \mathbf{T_A}, \mathbf{P}, s)$, it holds that $[\mathbf{A}|\mathbf{B}] \cdot \mathbf{U} = \mathbf{P} \mod q$, for $\mathbf{U} \in \mathbb{Z}_q^{2m \times m}$.

  Output the secret key $sk_{\boldsymbol{v}} = \mathbf{U}$.
- $\mathsf{IPE.Enc}(mpk, \boldsymbol{x}, \mu)$**:** On input the master public key mpk, the attribute vector $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k) \in (\mathbb{Z}_q^{\ell})^k$, and a message $\mu \in \{0, 1\}$, do:
  1. For $i = 1, \ldots, k$, set the matrices $\mathbf{X}_i = [x_{i,1}\mathbf{I}_n | x_{i,2}\mathbf{I}_n | \ldots | x_{i,\ell}\mathbf{I}_n] \in \mathbb{Z}_q^{n \times n\ell}$.
  2. Choose a uniformly random vector $\boldsymbol{s} \in \mathbb{Z}_q^n$, and sample two noise vectors $\boldsymbol{e}, \boldsymbol{e}^{'} \leftarrow \mathcal{D}_{\mathbb{Z}_q^m}$.
  3. For $i = 1, \ldots, k$, choose these random matrices $\mathbf{R}_i \in \{-1, 1\}^{m \times m}$. Then define noise vectors $\boldsymbol{e}_i^{\mathrm{T}} := \boldsymbol{e}^{\mathrm{T}} \mathbf{R}_i$.
  4. For $i = 1, \ldots, k$, compute the ciphertext
  $$\boldsymbol{c} := \boldsymbol{s}^{\mathrm{T}}\mathbf{A} + \boldsymbol{e}^{\mathrm{T}}, \boldsymbol{c}_i := \boldsymbol{s}^{\mathrm{T}}(\mathbf{A}_i + \mathbf{X}_i\mathbf{G}_{n\ell,\ell',m}) + \boldsymbol{e}_i^{\mathrm{T}}, \boldsymbol{c}^{'} := \boldsymbol{s}^{\mathrm{T}}\mathbf{P} + \boldsymbol{e}^{'} + (0, \ldots, 0, \lfloor \frac{q}{2} \rceil \mu)$$

  Output the ciphertext $CT := (\boldsymbol{c}, \{\boldsymbol{c}_i\}_{i \in \{1,\ldots k\}}, \boldsymbol{c}^{'})$
- $\mathsf{IPE.Dec}(mpk, CT, sk_{\boldsymbol{v}})$**:** On input the master public key, a secret key $sk_{\boldsymbol{v}} = \mathbf{U}$ for predicate vector $\boldsymbol{v}$ and the ciphertext $CT := (\boldsymbol{c}, \{\boldsymbol{c}_i\}_{i \in \{1,\ldots k\}}, \boldsymbol{c}^{'})$, do:
  1. For $i = 1, \ldots, k$, compute the vector $\boldsymbol{c}_{\boldsymbol{v}} = \sum_{i=1}^{k} \boldsymbol{c}_i \mathbf{V}_i$.
  2. Compute $\boldsymbol{z} \leftarrow \boldsymbol{c}^{'} - [\boldsymbol{c}|\boldsymbol{c}_{\boldsymbol{v}}] \cdot \mathbf{U} \mod q$.

  Output $\lfloor \frac{z_m}{q/2} \rceil \in \{0, 1\}$, if $\|(z_1, \ldots, z_{m-1})\|_{\infty} < q/4$; otherwise, output $\perp$.

3.2 Parameters

In Table 2, we set the parameters of the IPE scheme above.

Table 2. IPE parameters setting.

Variable	Description	Parameters setting
λ	Security parameter	
n	Row dimension of PK matrix	$n = \lambda$
m	Column dimension of PK matrix	$m = n^{1+\delta}$
q	Modulus	$q = k^2 n^{6.5+4\delta} log^{2.5+2\delta} n$
$k\ell$	Dimension of attribute vector	$\ell = \log n$
$\ell^{'}$	Base of gadget matrix $\mathbf{G}_{n\ell,\ell',m}$	$\ell^{'} = n$
α	Gaussian parameter of error	$\alpha = \sqrt{n} log^{1+\delta} n$
s	Parameter of $\mathsf{SampleLeft}$ and $\mathsf{SampleRight}$	$s = kn^{2.5+1.5\delta} log^{1.5+\delta} n$

3.3 Security

Theorem 1. *Suppose that $m \geq 6n \log q$, assuming the hardness of the decisional LWE problem, then the above inner product encryption scheme is weakly attribute hiding.*

4 A Single Targeted Homomorphic Compact IPE Scheme

In this section, we propose our single targeted homomorphic compact inner product encryption scheme from LWE. Inspired by the idea of [6], we add homomorphic property to our IPE scheme and get compact ciphertext and public parameter size. The construction of the scheme is as follows:

4.1 The THIPE Construction

Let λ be the security parameter and $u = k\ell$ be the length of predicate and attribute vectors. Set lattice parameters $n = n(\lambda), m = m(\lambda), q = q(\lambda)$ and Gaussian parameters $\alpha = \alpha(\lambda), s = s(\lambda)$, define $\ell^{'} = 2^{\ell}$ and $M = (2m + 1)\lceil \log q \rceil$.

- $\mathsf{THIPE.Setup}(1^{\lambda})$**:** On input a security parameter λ, do:
  1. Use the algorithm $\mathsf{TrapGen}(n, m, q)$ to generate a matrix $\mathbf{A}$ and its trapdoor $\mathbf{T_A}$.
  2. Choose $k+1$ uniformly random matrix $\mathbf{A}_i \in \mathbb{Z}_q^{n \times m}$ for $i = 0, 1, \ldots, k$ and sample a uniformly random vector $\boldsymbol{u} \in \mathbb{Z}_q^n$.

  Output $mpk = (\mathbf{A}, \mathbf{A}_0, \{\mathbf{A}_i\}_{i \in \{1,\ldots,k\}}, \boldsymbol{u})$ and $msk = \mathbf{T_A}$.

- $\mathsf{THIPE.KeyGen}(mpk, msk, \boldsymbol{x})$**:** On input the master public key mpk and master secret key msk, and a predicate vector $\boldsymbol{v} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k) \in (\mathbb{Z}_q^{\ell})^k$ where $\boldsymbol{v}_i = (v_{i,1}, \ldots, v_{i,\ell}) \in \mathbb{Z}_q^{\ell}$, do:
  1. For $i = 1, \ldots, \ell$, compute the matrices $\mathbf{V}_i^{'} := \begin{pmatrix} v_{i,1}\mathbf{I}_n \\ v_{i,2}\mathbf{I}_n \\ \vdots \\ v_{i,\ell}\mathbf{I}_n \end{pmatrix} \in \mathbb{Z}_q^{\ell n \times n}$, and let $\mathbf{V}_i := \mathbf{G}^{-1}_{n\ell,\ell',m}(\mathbf{V}_i^{'} \cdot \mathbf{G}_{n,2,m})$
  2. Define the matrices:
  $$\mathbf{B} := \sum_{i=1}^{k} \mathbf{A}_i \mathbf{V}_i \in \mathbb{Z}_q^{n \times m}$$
  3. Using msk to compute $\boldsymbol{r}_1 \leftarrow \mathsf{SampleLeft}(\mathbf{A}, \mathbf{A}_0 + \mathbf{B}, \mathbf{T}_{\mathbf{A}}, \boldsymbol{u}, s)$, it holds that $[\mathbf{A}|\mathbf{A}_0 + \mathbf{B}] \cdot \boldsymbol{r}_1 = \boldsymbol{u} \mod q$. For $\boldsymbol{r}^{\mathrm{T}} = [-\boldsymbol{r}_1{}^{\mathrm{T}}, 1]$, we have that $[\mathbf{A}|\mathbf{A}_0 + \mathbf{B}|\boldsymbol{u}] \cdot \boldsymbol{r} = \boldsymbol{0}$.

  Output the secret key $sk_{\boldsymbol{v}} = \boldsymbol{r}$.
- $\mathsf{THIPE.Enc}(mpk, \boldsymbol{x}, \mu)$**:** On input the master public key mpk, the attribute vector $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k) \in (\mathbb{Z}_q^{\ell})^k$, and a message $\mu \in \{0, 1\}$, do:
  1. For $i = 1, \ldots, k$, set the matrices $\mathbf{X}_i = [x_{i,1}\mathbf{I}_n | x_{i,2}\mathbf{I}_n | \ldots | x_{i,\ell}\mathbf{I}_n] \in \mathbb{Z}_q^{n \times n\ell}$.
  2. Choose a uniformly random vector $\mathbf{S} \in \mathbb{Z}_q^{n \times M}$, and sample a noise matrix $\mathbf{E} \leftarrow \mathcal{D}_{\mathbb{Z}_q^{m \times M}, \alpha}$ and a noise vector $\boldsymbol{e} \leftarrow \mathcal{D}_{\mathbb{Z}_q^{m}, \alpha}$.
  3. For $i = 0, 1, \ldots, k$, choose these random matrices $\mathbf{R}_i \in \{-1, 1\}^{m \times m}$. Then define noise vectors $\mathbf{E}_i := \mathbf{R}_i^{\mathrm{T}}\mathbf{E}$.
  4. Compute the ciphertext as follows:
  $$\begin{pmatrix} \mathbf{C}_{\mathbf{A}} \\ \mathbf{C}_0 \\ \mathbf{C}_{\boldsymbol{u}} \end{pmatrix} = \begin{pmatrix} \mathbf{A}^{\mathrm{T}} \\ \mathbf{A}_0^{\mathrm{T}} \\ \boldsymbol{u}^{\mathrm{T}} \end{pmatrix} \cdot \mathbf{S} + \begin{pmatrix} \mathbf{E} \\ \mathbf{E}_0 \\ \boldsymbol{e} \end{pmatrix} + \mu \mathbf{G}_{2m+1,2,M}$$
  And for all $i = 1, \ldots, k$, we compute:
  $$\mathbf{C}_i = (\mathbf{A}_i + \mathbf{X}_i \mathbf{G}_{n\ell,\ell',m})^{\mathrm{T}} \mathbf{S} + \mathbf{E}_i$$

  Output the ciphertext $CT := (\mathbf{C}_{\mathbf{A}}, \mathbf{C}_0, \mathbf{C}_{\boldsymbol{u}}, \{\mathbf{C}_i\}_{i \in \{1,\ldots,k\}})$.
- $\mathsf{THIPE.Trans}(mpk, CT, \boldsymbol{v})$**:** For predicate vector $\boldsymbol{v}$ and ciphertext CT which corresponds to attribute $\boldsymbol{x}$, such that $\langle \boldsymbol{x}, \boldsymbol{v} \rangle = 0$. The evaluator then computes:
  $$\mathbf{C}_{\boldsymbol{v}} = \sum_{i=1}^{k} \mathbf{V}_i^{\mathrm{T}} \mathbf{C}_i$$
  Then the evaluator sets:
  $$\mathbf{C} = \begin{pmatrix} \mathbf{C}_{\mathbf{A}} \\ \mathbf{C}_0 + \mathbf{C}_{\boldsymbol{v}} \\ \boldsymbol{c}_{\boldsymbol{u}} \end{pmatrix} \in \mathbb{Z}_q^{(2m+1) \times M}$$
  The ciphertext $\mathbf{C}$ is the final ciphertext that used to do homomorphic evaluation.

- $\mathsf{THIPE.TEval}(g, \mathbf{C}_1, \ldots, \mathbf{C}_t)$**:** The ciphertexts $\mathbf{C}_i$ which are the outputs of $\mathsf{THIPE.Trans}$ are corresponding to the same predicate vector $\boldsymbol{v}$ that the evaluator knows in advance, it outputs $\mathbf{C}_g = \mathsf{Eval}(g, \mathbf{C}_1, \ldots, \mathbf{C}_t)$. In the process of evaluation, it computes NAND gate as:

$$\mathrm{NAND}(\mathbf{C}_1, \mathbf{C}_2) = \mathbf{G}_{2m+1,2,M} - \mathbf{C}_1(\mathbf{G}^{-1}_{2m+1,2,M}\mathbf{C}_2)$$

- $\mathsf{THIPE.Dec}(mpk, \mathbf{C}_g, sk_{\boldsymbol{v}})$**:** On input the master public key, a secret key $sk_{\boldsymbol{v}} = \boldsymbol{r}$ for predicate vector $\boldsymbol{v}$ and the ciphertext $\mathbf{C}_g$, do:
  1. For $\boldsymbol{b} = (0, \ldots, 0, \lfloor q/2 \rceil)^{\mathrm{T}}$, compute $z \leftarrow \boldsymbol{r}^{\mathrm{T}}\mathbf{C}_g\mathbf{G}^{-1}_{2m+1,2,M}(\boldsymbol{b}) \mod q$
  2. Output 0, if $|z| < q/4$; otherwise, output 1.

5 Conclusion

In this work, we built a compact IPE scheme and a targeted homomorphic compact IPE scheme. We make use of two gadget matrix $\mathbf{G}_{n\ell,\ell',m}$ and $\mathbf{G}_{n,2,m}$ and decrease the public parameter size to $O(un^2 \log n)$, ciphertext size to $O(un \log n)$. Our IPE scheme improve the public parameters by a factor of $O(\log n)$ compared with [17].

Acknowledgments. We thank the anonymous ICICS'2017 reviewers for their helpful comments. This work is supported by the National Basic Research Program of China (973 project, No. 2014CB340603) and the National Nature Science Foundation of China (No. 61672030).

References

1. Apon, D., Fan, X., Liu, F.: Compact identity based encryption from LWE. http://eprint.iacr.org/2016/125
2. Agrawal, S., Freeman, D.M., Vaikuntanathan, V.: Functional encryption for inner product predicates from learning with errors. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 21–40. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25385-0_2
3. Ajtai, M.: Generating hard instances of the short basis problem. In: Wiedermann, J., van Emde Boas, P., Nielsen, M. (eds.) ICALP 1999. LNCS, vol. 1644, pp. 1–9. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48523-6_1
4. Attrapadung, N., Libert, B.: Functional encryption for inner product: achieving constant-size ciphertexts with adaptive security or support for negation. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 384–402. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13013-7_23
5. Alwen, J., Peikert, C.: Generating shorter bases for hard random lattices. Theory Comput. Syst. **48**, 535–553 (2011)
6. Brakerski, Z., Cash, D., Tsabary, R., Wee, H.: Targeted homomorphic attribute-based encryption. In: Hirt, M., Smith, A. (eds.) TCC 2016. LNCS, vol. 9986, pp. 330–360. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53644-5_13

7. Boneh, D., Gentry, C., Gorbunov, S., Halevi, S., Nikolaenko, V., Segev, G., Vaikuntanathan, V., Vinayagamurthy, D.: Fully key-homomorphic encryption, arithmetic circuit ABE and compact Garbled circuits. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 533–556. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-55220-5_30
8. Boneh, D., Waters, B.: Conjunctive, subset, and range queries on encrypted data. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 535–554. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-70936-7_29
9. Cash, D., Hofheinz, D., Kiltz, E., Peikert, C.: Bonsai trees, or how to delegate a lattice basis. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 523–552. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_27
10. Katz, J., Sahai, A., Waters, B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: Smart, N. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 146–162. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78967-3_9
11. Lewko, A., Okamoto, T., Sahai, A., Takashima, K., Waters, B.: Fully secure functional encryption: attribute-based encryption and (hierarchical) inner product encryption. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 62–91. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_4
12. Okamoto, T., Takashima, K.: Hierarchical predicate encryption for inner-products. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 214–231. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10366-7_13
13. Okamoto, T., Takashima, K.: Fully secure functional encryption with general relations from the decisional linear assumption. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 191–208. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14623-7_11
14. Okamoto, T., Takashima, K.: Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption. In: Lin, D., Tsudik, G., Wang, X. (eds.) CANS 2011. LNCS, vol. 7092, pp. 138–159. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25513-7_11
15. Okamoto, T., Takashima, K.: Adaptively attribute-hiding (hierarchical) inner product encryption. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 591–608. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29011-4_35
16. Park, J.-H.: Inner-product encryption under standard assumptions. Des. Codes Crypt. **58**, 235–257 (2011)
17. Xagawa, K.: Improved (Hierarchical) Inner-Product Encryption from Lattices. In: Kurosawa, K., Hanaoka, G. (eds.) PKC 2013. LNCS, vol. 7778, pp. 235–252. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36362-7_15