Improved (Hierarchical) Inner-Product Encryption from Lattices[★]

Keita Xagawa

NTT Secure Platform Laboratories
xagawa.keita@lab.ntt.co.jp

March 17, 2015

Abstract. Inner-product encryption (IPE) provides fine-grained access control and has attractive applications. Agrawal, Freeman, and Vaikuntanathan (Asiacrypt 2011) proposed the first IPE scheme from lattices by twisting the identity-based encryption (IBE) scheme by Agrawal, Boneh, and Boyen (Eurocrypt 2010). Their IPE scheme supports inner-product predicates over R^{μ}, where the ring is $R = \mathbb{Z}_q$. Several applications require the ring R to be exponentially large and, thus, they set $q = 2^{O(n)}$ to implement such applications. This choice results in the AFV IPE scheme with public parameters of size $O(\mu n^2 \lg^3 q) = O(\mu n^5)$ and ciphertexts of size $O(\mu n \lg^3 q) = O(\mu n^4)$, where n is the security parameter. Hence, this makes the scheme impractical, as they noted.

We address this efficiency issue by "untwisting" their twist and providing another twist. Our scheme supports inner-product predicates over R^{μ} where $R = \mathrm{GF}(q^n)$ instead of $\mathbb{Z}_q$. Our scheme has public parameters of size $O(\mu n^2 \lg^2 q)$ and ciphertexts of size $O(\mu n \lg^2 q)$. Since the cardinality of $\mathrm{GF}(q^n)$ is inherently exponential in n, we have no need to set q as the exponential size for applications.

As side contributions, we extend our IPE scheme to a hierarchical IPE (HIPE) scheme and propose a fuzzy IBE scheme from IPE. Our HIPE scheme is more efficient than that developed by Abdalla, De Caro, and Mochetti (Latincrypt 2012). Our fuzzy IBE is secure under a much weaker assumption than that employed by Agrawal et al. (PKC 2012), who constructed the first lattice-based fuzzy IBE scheme.

Keywords. predicate encryption, (hierarchical) inner-product encryption, lattices, learning with errors, invertible difference encoding, pseudo-commutativity.

1 Introduction

Background: Predicate encryption (PE) gives fine-grained access control beyond identity-based encryption (IBE). In a PE scheme, a receiver corresponding to a *key attribute* v can decrypt a ciphertext corresponding to a *ciphertext attribute* w if and only if $\mathcal{P}(v, w) = 1$, where $\mathcal{P}$ is a predicate.

Katz, Sahai, and Waters [KSW08] introduced inner-product encryption (IPE), which is PE that supports the inner-product predicate: that is, predicate $\mathcal{P}^{\mathsf{IPE}} : R^{\mu} \times R^{\mu} \to \{0, 1\}$, where R is a finite ring, defined as $\mathcal{P}^{\mathsf{IPE}}(\vec{v}, \vec{w}) = 1$ if and only if $\vec{w}^{\top} \vec{v} = 0$. They showed that several predicates, for example, equalities, hidden-vector predicates, polynomial evaluations, and CNF/DNF formulae, can be encoded as an inner product that exemplifies the serviceability of IPE. Following their work and by exploiting the properties of the pairing on composite-number or prime order groups, recent studies on IPE have enhanced security or introduced compact schemes [KSW08,OT09,LOS$^+$10,OT10,AL10,Par11,OT11,OT12], and have left an open problem of constructing IPE from other assumptions, say, factoring, decisional Diffie-Hellman (DDH), or the learning with errors (LWE) assumptions.

In 2011, Agrawal, Freeman, and Vaikuntanathan [AFV11] overcame the hurdle, i.e., the problem of constructing IPE *without pairing*. They proposed the first IPE scheme based on the LWE assumption [Reg09] and left three open problems: improving security, efficiency, and functionality.

Let us focus on the second problem, the efficiency issue. Their scheme supports an inner-product predicate over $R = \mathbb{Z}_q$, and has public parameters of size $\Theta(\mu n^2 \lg^3 q)$ and ciphertexts of size $\Theta(\mu n \lg^3 q + \ell \lg q)$, where n is the security parameter, μ is the dimension of the vector space, and ℓ is the length of a message. (In what follows, we will ignore $\Theta(\ell \lg q)$.) This seems satisfactory for actual use.

[★] This article is based on an earlier article: Improved (Hierarchical) Inner-Product Encryption from Lattices. In PKC 2013, pages 235–252. © IACR 2013, DOI:10.1007/978-3-642-36362-7_15

In several applications of IPE, we require exponentially large R (see below). To implement such applications, Agrawal et al. set $q = 2^{O(n)}$ [AFV11, Section 6]. This setting results in the length of ciphertext $\Theta(\mu n^4)$, which shows the impracticality of the scheme in the real world.

Motivated by applications: We were motivated to improve the efficiency by applications of IPE that require large R, which we discuss here. Roughly speaking, we require the ring R to be exponentially large in order to implement an application when we have to take AND (logical conjunction) of predicates by using the technique proposed by Katz, Sahai, and Waters [KSW08][1]. Since the existing pairing-based IPE serves inner products over the ring $R = \mathbb{Z}_q$ or $\mathbb{Z}_N$, where q, N is an exponential of a security parameter, there are no problematic issues. Unfortunately, the exponential magnitude of R makes the AFV IPE scheme impractical, since the length of the ciphertext is the cubic order of $\lg(\#R) = \lg q$.

We have several attractive applications that are implemented by IPE with logical conjunctions. These include CNF formulae [KSW08], hidden vector encryption [BW07], which serves a comparison and a range query on a small set, and wild-carded IBE [ABC$^+$11]. We will review and discuss the applications of IPE schemes in Appendix E.

Moreover, for a realistic scenario, we will treat a colossal set as a domain of the predicate, e.g., one billion users ($10^9 \approx 2^{30}$), addresses of IPv6 (2^{128}), verification keys of one-time signature (2^{128}–), and hash values of SHA3 candidates (2^{256}–2^{512}). In such a situation, even the equality predicate requires logical conjunctions to split them into chunks in R.

Hence, we should make IPE efficient even for exponentially large R for the IPE applications.

1.1 Our Contribution

Our main contribution is to improve the efficiency of the AFV IPE scheme. More formally, we construct an IPE scheme under the LWE assumption, which supports an inner-product predicate over the field $\mathrm{GF}(q^n)$ instead of $\mathbb{Z}_q$ and has public parameters of size $\Theta(\mu n^2 \lg^2 q)$ and ciphertexts of size $\Theta(\mu n \lg^2 q + \ell \lg q)$. Since the cardinality of $\mathrm{GF}(q^n)$ is $q^n = 2^{\Omega(n)}$, and $\mathrm{GF}(q^n)$ is a field, we can set $q = \mathrm{poly}(n)$ even for the above applications. We note that Agrawal et al. [AFV11, Section 6] expected the *ring-LWE* assumption [LPR10] to resolve the issue, but we solve it *without the ring-LWE assumption*.

In addition, we have two side contributions; One is an extension of *hierarchical inner-product encryption* (HIPE) [OT09], which implies spatial encryption (SE) [BH08,Ham11,CLLW11]. We apply our techniques to again drastically improve the existing HIPE scheme from lattices [ADCM12] in the case of exponentially large R. The other is a fuzzy IBE (FIBE) scheme over a small universe $\{0, 1\}$ from IPE under the LWE assumption with *conservative* parameters, whereas the existing fuzzy IBE scheme from lattices are under the LWE assumption with *sub-exponential* parameters [ABV$^+$12].

Comparison: Since the description size of the public parameters is n times that of ciphertexts, we compare the efficiency of the schemes by the length of the ciphertext. For simplicity, we let L_{ours} and L_{AFV} denote the lengths of ciphertexts of our scheme and the AFV scheme, respectively.

When $q = \mathrm{poly}(n)$, our scheme improves the size by only a factor of $\lg q = O(\lg n)$ ($L_{\mathrm{ours}} = \Theta(\mu n \lg^2 q)$ and $L_{\mathrm{AFV}} = \Theta(\mu n \lg^3 q)$). Moreover, if we restrict $\vec{v}$ in a small domain, say, $\{0, 1\}^\mu$, then $L_{\mathrm{AFV}} = \Theta(\mu n \lg^2 q)$, and there is no improvement.

On the other hand, if we set $\#R = 2^{\Theta(n)}$ to implement applications, the improvement is drastic: $L_{\mathrm{ours}} = \Theta(\mu n \lg^2 q)$ since $\#\,\mathrm{GF}(q^n) = 2^{\Omega(n)}$ and $L_{\mathrm{AFV}} = \Theta(\mu n \lg^3 q) = \Theta(\mu n^4)$ since they need to set $q = 2^{\Theta(n)}$. In this case, efficiency is improved by a factor of $\tilde{O}(n^3)$.

[1] Suppose that we have two implementations of two predicates f and g; one is embedded as $\vec{v}_f$ and $\vec{w}_f$, and the other is embedded as $\vec{v}_g$ and $\vec{w}_g$. The Katz-Sahai-Waters (KSW) technique embeds $f \wedge g$ into two vectors $\vec{v}_{f \wedge g} = (\vec{v}_f, \vec{w}_g)$ and $\vec{w}_{f \wedge g} = (r_f \vec{v}_f, r_g \vec{w}_g)$, where r_f, r_g are chosen uniformly at random from R. The inner product of $\vec{v}_{f \wedge g}$ and $\vec{w}_{f \wedge g}$ is $r_f \vec{w}_f^{\top} \vec{v}_f + r_g \vec{w}_g^{\top} \vec{v}_g$. If the two inner products are 0, that is, two predicates f and g are true, then the inner product becomes 0. The inversion is not true; if not, then the inner product is not 0 without probability, say, $1/\#R$. This shows that R should be exponentially large, say, at least 2^{80} from the security requirement.

Next, we compare the FIBE schemes with a small universe; that is, their identities are binary vectors of length N. Agrawal, Boyen, Vaikuntanathan, Voulgaris, and Wee [ABV$^+$12] proposed a FIBE scheme based on the LWE assumption with *sub-exponential* parameters. They restricted $N = n^{\epsilon}$ with $\epsilon \in (0, 1/2)$ in order to obtain the security under the hardness of lattice problems. Their scheme is based on the worst-case hardness of lattice problems of approximation factor $2^{O(N)} = 2^{O(n^{\epsilon})}$ with a subexponential-time algorithm. The length of their ciphertext is $\Theta(Nm \lg q) = \Theta(N^2 n \lg^2 n)$.

On the contrary, our scheme enjoys flexible $N = \mathrm{poly}(n)$ and a weaker assumption, which is the worst-case hardness of lattice problem of approximation factor $\tilde{O}(n^{4.5})$. The length of our ciphertext is $O(N^2 n \lg^2 n)$, which is the same as theirs.

We note that Agrawal et al. also extended their scheme to support identity space $(\mathbb{Z}_q^n)^N$ without changing parameters or the assumption (see [ABV$^+$12, Appendix B]).

On the ring-LWE assumption: We finally note that there are the variants of the ABB IBE schemes based on the ring-LWE assumption and the variants of the AFV IPE scheme also [MP12b,MP12a], which yield certain exponentially large R. Our technique is orthogonal to their techniques and improves the variants of the AFV IPE scheme by a factor of $\lg q$. We will describe concrete schemes in Appendix D.

1.2 Related Works

IPE was introduced by Katz, Sahai, and Waters [KSW08], who gave a fully attribute-hiding but selectively secure IPE scheme based on the composite-order pairings. Following them, several researchers proposed (H)IPE schemes based on the pairings [OT09,LOS$^+$10,OT10,AL10,OT11,OT12].

On IPE based on lattices, Agrawal et al. [AFV11] constructed the first IPE scheme which is selectively secure and weakly attribute hiding under the LWE assumption. Another study on a lattice-based HIPE scheme was done by Abdalla, De Caro, and Mochetti [ADCM12], who extended the AFV IPE scheme to a HIPE scheme. They also proposed two extensions of the HIPE scheme, a wild-carded IBE scheme and a CCA secure HIPE scheme. The CCA2 construction exemplifies the requirement of large R, since, in the construction, the attribute space of the basic scheme includes a one-time verification key as required for the CHK conversion [BCHK06].

Another line of study of IPE is initiated as spatial encryption (SE) defined by Boneh and Hamburg [BH08]. Hamburg [Ham11] observed that HIPE and SE are strongly related, and Chen, Lim, Ling, and Wang [CLLW11] gave explicit property-preserving conversions between them, which enable us to treat SE schemes as (H)IPE schemes. For SE, see Hamburg's thesis [Ham11].

From the perspective of lattice-based encryption beyond IBE, we refer to fuzzy IBE schemes by Agrawal et al. [ABV$^+$12], a revocable IBE scheme by Chen, Lim, Ling, Wang, and Ngyuen [CLL$^+$12], and an attribute-based encryption scheme by Boyen [Boy12].

1.3 Overview of Our Construction

We give an overview of our construction. For simplicity, we focus on the construction of IPE and omit HIPE. After briefly explaining the basics and the AFV IPE, we present our ideas for "half untwisting" and "half twisting."

The basics: We first review the "dual" public-key encryption (PKE) scheme proposed by Gentry, Peikert, and Vaikuntanathan [GPV08] (or the Peikert KEM [Pei09]). Their public key is a random matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n \times m}$. The ciphertext is a vector close to the lattice $\Lambda_q(\boldsymbol{A}) = \{\boldsymbol{z} \in \mathbb{Z}^m : \boldsymbol{z} \equiv \boldsymbol{A}^{\top}\boldsymbol{s} \text{ for some } \boldsymbol{s} \in \mathbb{Z}_q^n\}$. The secret key is a short basis of $\Lambda_q^{\perp}(\boldsymbol{A}) = \{\boldsymbol{z} \in \mathbb{Z}^m : \boldsymbol{A}\boldsymbol{z} \equiv \boldsymbol{0}\}$, which enables us to recover a lattice vector in $\Lambda_q(\boldsymbol{A})$ from a vector close to $\Lambda_q(\boldsymbol{A})$. Cash, Hofheinz, Kiltz, and Peikert [CHKP10] proposed the first IBE scheme based on the lattices in the standard model.

After that, Agrawal, Boneh, and Boyen [ABB10] proposed a lattice analogue of the Boneh–Boyen IBE [BB11] (and that of the Waters IBE [Wat05]). Let $id = \mathfrak{w} = w_0 + w_1 X + \cdots + w_{n-1} X^{n-1} \in \mathrm{GF}(q^n)$. Let H be an invertible (or full-rank) difference encoding [CD09] that maps a polynomial in $\mathrm{GF}(q^n)$ to an n by n matrix of elements in $\mathbb{Z}_q$. [2] In the ABB IBE scheme, the public parameters consist of $\boldsymbol{A}_0$, $\boldsymbol{A}_1$, and $\boldsymbol{B}$, and the encryption lattice for id is

$$\Lambda_{id} = \Lambda_q(\boldsymbol{A}_0 \mid \boldsymbol{A_1} + H(id) \cdot \boldsymbol{B}).$$

The master has a short basis for $\Lambda_q^{\perp}(\boldsymbol{A}_0)$, and it can generate secret keys for $\Lambda_{id}^{\perp}$ using the basis sampling techniques as in [CHKP10]. For the security proof, we require H to be invertible difference.

The "twist" by Agrawal, Freeman, and Vaikuntanathan: Agrawal, Freeman, and Vaikuntanathan [AFV11] gave a novel twist on the ABB IBE [ABB10] and obtained an IPE scheme.

In the AFV IPE scheme, the encryption lattice for ciphertext-attribute vector $\vec{w} \in \mathbb{Z}_q^{\mu}$ is defined as

$$\Lambda_{\vec{w}} = \Lambda_q\left(\boldsymbol{A}_0 \mid \boldsymbol{A}_1 + w_1 \boldsymbol{B} \mid \cdots \mid \boldsymbol{A}_{\mu} + w_{\mu} \boldsymbol{B}\right).$$

The ciphertext is a vector $\boldsymbol{c} = (\boldsymbol{c}_0, \ldots, \boldsymbol{c}_{\mu}) \in (\mathbb{Z}_q^m)^{\mu+1}$ close to $\Lambda_{\vec{w}}$.

They define the mapping $F_{\vec{v}} : (\mathbb{Z}_q^m)^{\mu+1} \to (\mathbb{Z}_q^m)^2$ as

$$F_{\vec{v}}(\boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_{\mu}) = \left(\boldsymbol{c}_0, \textstyle\sum_{i=1}^{\mu} v_i \boldsymbol{c}_i\right) \in \mathbb{Z}_q^{2m}$$

for decryption, where F means "fold." Notice that, if $\vec{v}$ is a *short* vector, e.g., $\vec{v} \in \{0,1\}^{\mu} \subset \mathbb{Z}_q^{\mu}$, then $F_{\vec{v}}(\boldsymbol{c})$ is a vector close to the lattice

$$\Lambda_{\vec{v},\vec{w}} = \Lambda_q\left(\boldsymbol{A}_0 \,|\, \textstyle\sum_{i=1}^{\mu} v_i (\boldsymbol{A}_i + w_i \boldsymbol{B})\right) = \Lambda_q\left(\boldsymbol{A}_0 \,|\, \textstyle\sum_{i=1}^{\mu} v_i \boldsymbol{A}_i + (\vec{w}^{\top} \vec{v}) \boldsymbol{B}\right).$$

If $\vec{w}^{\top} \vec{v} = 0$ then the masking term, $(\vec{w}^{\top} \vec{v})\boldsymbol{B}$, vanishes. The secret key for $\vec{v} \in \mathbb{Z}_q^{\mu}$ is defined as a short basis of $\Lambda_q^{\perp}(\boldsymbol{A}_0 \mid \sum_{i=1}^{\mu} v_i \boldsymbol{A}_i)$.

They also gave a binary decomposition technique for $\vec{v}$ of long norm, which expands the public parameters and ciphertext by a factor of $\lg q$: they replaced $\vec{w}$ and $\vec{v}$ with $\vec{w}' = (1, 2, \ldots, 2^{k-1}) \otimes \vec{w}$, where $k = \lceil \lg q \rceil$ and $\otimes$ denotes the standard tensor product, and $\vec{v} \in \mathbb{Z}_q^{\mu}$ with $\vec{v}' \in \{0,1\}^{\mu k}$ such that $\vec{w}^{\top} \vec{v} = (\vec{w}')^{\top} \vec{v}'$. This technique is already exploited in the constructions of fully homomorphic encryption [BV11,BGV12,Bra12].

Our Ideas: Here, we present our two ideas for changing R from $\mathbb{Z}_q$ to $\mathrm{GF}(q^n)$.

Half untwist: We first change the domain of attributes $\vec{w}$ from $\mathbb{Z}_q^{\mu}$ to $\mathrm{GF}(q^n)^{\mu}$, while $\vec{v}$'s domain is the same as the original, $\mathbb{Z}_q^{\mu} \subset \mathrm{GF}(q^n)^{\mu}$.

Let us turn back to the invertible difference encoding H, which appeared in the ABB IBE but was omitted from the AFV IPE. We have the following facts on the typical construction of $H : \mathrm{GF}(q^n) \to \mathbb{Z}_q^{n \times n}$:

- Fact 1: H maps $w \in \mathbb{Z}_q \subseteq \mathrm{GF}(q^n)$ to $H(w) = w\boldsymbol{I}_n$, where $\boldsymbol{I}_n$ is the n-dimensional identity matrix.
- Fact 2: H is $\mathbb{Z}_q$-linear and is an isomorphism from $\mathrm{GF}(q^n)$ to a field contained in $\mathbb{Z}_q^{n \times n}$.

[2] Originally, H is called as "full-rank difference" encoding [CD09]. Recently, this concept was generalized for composite q and others [MP12b,ASP12]: The one of reviews suggested to call it "invertible difference" and the author follows. We say that $H : R \to \mathbb{Z}_q^{n \times n}$ is invertible difference if for any distinct $\mathfrak{w} \neq \mathfrak{w}' \in R$, matrix $H(\mathfrak{w}) - H(\mathfrak{w}') \in \mathbb{Z}_q^{n \times n}$ is invertible (rather than the matrix has rank n). See Section 4 for a concrete construction.

From Fact 1, we have $w\boldsymbol{B} = w\boldsymbol{I}_n\boldsymbol{B} = H(w) \cdot \boldsymbol{B}$. We can rewrite the encryption lattice of the AFV IPE for $\vec{w} \in \mathbb{Z}_q^\mu \subseteq \mathrm{GF}(q^n)^\mu$ as

$$\Lambda_{\vec{w}} = \Lambda_q(\boldsymbol{A}_0 \mid \boldsymbol{A}_1 + H(w_1 + 0X + \cdots + 0X^{n-1})\boldsymbol{B} \mid \cdots \mid \boldsymbol{A}_\mu + H(w_\mu + 0X + \cdots + 0X^{n-1})\boldsymbol{B}).$$

We discover the hidden H in the AFV IPE and find $(n-1)$ empty slots for each $i \in [\mu]$.

Now, we can change the domain of ciphertext-attribute vector $\vec{w}$ from $\mathbb{Z}_q^\mu$ to $\mathrm{GF}(q^n)^\mu$. We naturally define $\Lambda_{\vec{w}}$ for $\vec{w} = (\mathfrak{w}_1, \ldots, \mathfrak{w}_\mu)^\top \in \mathrm{GF}(q^n)^\mu$ as

$$\Lambda_{\vec{w}} = \Lambda_q(\boldsymbol{A}_0 \mid \boldsymbol{A}_1 + H(\mathfrak{w}_1)\boldsymbol{B} \mid \cdots \mid \boldsymbol{A}_\mu + H(\mathfrak{w}_\mu)\boldsymbol{B}).$$

For a short key vector $\vec{v} \in \mathbb{Z}_q^\mu \subset \mathrm{GF}(q^n)^\mu$ and a ciphertext $\boldsymbol{c} = (\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\mu)$ close to $\Lambda_{\vec{w}}$, we observe that $F_{\vec{v}}(\boldsymbol{c}) = (\boldsymbol{c}_0, \sum_{i=1}^\mu v_i \boldsymbol{c}_v)$ is close to

$$\Lambda_{\vec{w},\vec{v}} = \Lambda_q\Big(\boldsymbol{A}_0 \,|\, \textstyle\sum_{i=1}^\mu v_i(\boldsymbol{A}_i + H(\mathfrak{w}_i) \cdot \boldsymbol{B})\Big) = \Lambda_q\Big(\boldsymbol{A}_0 \,|\, \textstyle\sum_{i=1}^\mu v_i\boldsymbol{A}_i + H(\vec{w}^\top\vec{v}) \cdot \boldsymbol{B}\Big),$$

where the latter equality follows from the linearity of H (Fact 2). If $\vec{w}^\top\vec{v} = 0$ then $H(\vec{w}^\top\vec{v}) = \boldsymbol{O}$. By using a short basis of $\Lambda_q^\perp(\boldsymbol{A}_0 \mid \sum_{i=1}^\mu v_i\boldsymbol{A}_i)$, one can decrypt the ciphertext if the inner product is 0. Otherwise, the masking matrix $H(\vec{w}^\top\vec{v}) \cdot \boldsymbol{B}$ survives and $H(\vec{w}^\top\vec{v})$ is invertible.

Half twist: We next change the domain of $\vec{v}$ from $\mathbb{Z}_q^\mu$ to $\mathrm{GF}(q^n)^\mu$.

We observe that the proof of security by Agrawal et al. [AFV11] does not require randomness of $\boldsymbol{B}$. Hence, we can safely replace a random matrix $\boldsymbol{B}$ with a very structured matrix $\boldsymbol{G} = \boldsymbol{I}_n \otimes (1, 2, 2^2, \ldots, 2^{k-1})$ as in Micciancio and Peikert [MP12b], where $\otimes$ denotes the Kronecker product, and $k = \lceil \lg q \rceil$,

We exploit the structure of $\boldsymbol{G}$ and define a new encoding, $H' : \mathrm{GF}(q^n) \to \{0,1\}^{m \times m}$ (see Section 4), which gives *pseudo-commutativity* with respect to $\boldsymbol{G}$ and H, that is, for any $\mathfrak{v} \in \mathrm{GF}(q^n)$, it holds that $\boldsymbol{G} \cdot H'(\mathfrak{v}) = H(\mathfrak{v}) \cdot \boldsymbol{G}$.

We apply the above idea and new encoding to the half untwist version of the AFV IPE. The encryption lattice for $\vec{w}$ is $\Lambda_{\vec{w}} = \Lambda_q(\boldsymbol{A}_0 \mid \boldsymbol{A}_1 + H(\mathfrak{w}_1) \cdot \boldsymbol{G} \mid \cdots \mid \boldsymbol{A}_\mu + H(\mathfrak{w}_\mu) \cdot \boldsymbol{G})$ as in the previous version. We modify the key-extraction and decryption algorithms for $\vec{v} = (\mathfrak{v}_1, \ldots, \mathfrak{v}_\mu) \in \mathrm{GF}(q^n)^\mu$. In decryption, a ciphertext $(\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\mu)$ is folded up by $H'(\mathfrak{v}_i)$ instead of v_i, that is,

$$F'_{\vec{v}}(\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\mu) = \Big(\boldsymbol{c}_0, \textstyle\sum_{i=1}^\mu \boldsymbol{c}_i \cdot H'(\mathfrak{v}_i)\Big).$$

By the pseudo-commutativity, the sum $F'_{\vec{v}}(\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\mu)$ is a vector close to the lattice

$$\begin{aligned}\Lambda_{\vec{w},\vec{v}} &= \Lambda_q\Big(\boldsymbol{A}_0 \,|\, \textstyle\sum_{i=1}^\mu (\boldsymbol{A}_i + H(\mathfrak{w}_i) \cdot \boldsymbol{G}) \cdot H'(\mathfrak{v}_i)\Big) \\ &= \Lambda_q\Big(\boldsymbol{A}_0 \,|\, \textstyle\sum_{i=1}^\mu \boldsymbol{A}_i \cdot H'(\mathfrak{v}_i) + \textstyle\sum_{i=1}^\mu H(\mathfrak{w}_i) \cdot H(\mathfrak{v}_i) \cdot \boldsymbol{G}\Big) \\ &= \Lambda_q\Big(\boldsymbol{A}_0 \,|\, \textstyle\sum_{i=1}^\mu \boldsymbol{A}_i \cdot H'(\mathfrak{v}_i) + H(\vec{w}^\top\vec{v}) \cdot \boldsymbol{G}\Big),\end{aligned}$$

since the matrix norm of $H'(\mathfrak{v}_i) \in \{0,1\}^{m \times m}$ is at most m. The secret key is a short basis of lattice $\Lambda_q^\perp(\boldsymbol{A}_0 \mid \sum_{i=1}^\mu \boldsymbol{A}_i \cdot H'(\mathfrak{v}_i))$.

We note that the binary-decomposition technique is built into our new encoding H' and the structured matrix $\boldsymbol{G}$. Therefore, we can save the $\lg q$ factor introduced by the binary decomposition in the AFV IPE scheme.

2 Preliminaries

A security parameter is denoted by κ. We use the standard O-notations, O, Θ, Ω, and ω. We use capital bold symbols $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ for matrices. In particular, $\boldsymbol{I}_n$ denotes an n by n identity matrix. We use lower-case bold symbols $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$ for vectors. In addition, we use over-arrows to denote ciphertext- and key-attribute vectors as $\vec{w}$, $\vec{v}$. We use lower-case fraktur symbols $\mathfrak{a}, \mathfrak{b}, \mathfrak{c}$ for polynomials and elements of $\mathrm{GF}(q^n)$. The abbreviations DPT and PPT stand for deterministic polynomial time and probabilistic polynomial time. For any integer $q \geq 3$, we write $\mathbb{Z}_q$ for the ring $\{-(q-1)/2, \dots, -1, 0, 1, \dots, (q-1)/2\}$ with addition and multiplication modulo q.

A function $f(\kappa)$ is said to be negligible if $f(\kappa) = \kappa^{-\omega(1)}$. We denote a set of negligible functions by $\mathsf{negl}(\kappa)$. For a positive integer n, $[n]$ denotes $\{1, 2, \dots, n\}$. For $x \in \mathbb{R}$, we define $\lfloor x \rceil = \lceil x - 1/2 \rceil$ as the integer closest to x. For $\boldsymbol{x} = (x_1, \dots, x_\ell) \in \mathbb{R}^\ell$, we define $\lfloor \boldsymbol{x} \rceil$ as $(\lfloor x_1 \rceil, \dots, \lfloor x_\ell \rceil) \in \mathbb{Z}^\ell$. For two matrices $\boldsymbol{X} \in \mathbb{R}^{m \times n_1}$ and $\boldsymbol{Y} \in \mathbb{R}^{m \times n_2}$, $[\boldsymbol{X} \mid \boldsymbol{Y}] \in \mathbb{R}^{m \times (n_1 + n_2)}$ is the concatenation of the columns of $\boldsymbol{X}$ and $\boldsymbol{Y}$. For two matrices $\boldsymbol{X} \in \mathbb{R}^{m_1 \times n}$ and $\boldsymbol{Y} \in \mathbb{R}^{m_2 \times n}$, $[\boldsymbol{X}; \boldsymbol{Y}] \in \mathbb{R}^{(m_1+m_2) \times n}$ is the concatenation of the rows of $\boldsymbol{X}$ and $\boldsymbol{Y}$. For a vector $\boldsymbol{x} \in \mathbb{R}^m$, $\|\boldsymbol{x}\|_p$ denotes the ℓ_p norm of $\boldsymbol{x}$. For ease of notation, we omit the subscript if $p = 2$.

For matrix $\boldsymbol{X} = [\boldsymbol{x}_1 \dots \boldsymbol{x}_n]$, $\tilde{\boldsymbol{X}}$ denotes the Gram-Schmidt orthogonalization of $\boldsymbol{X}$. For a matrix $\boldsymbol{X} = [\boldsymbol{x}_1; \dots; \boldsymbol{x}_m] \in \mathbb{R}^{m \times n}$, $\|\boldsymbol{X}\|_{\mathrm{row}} = \max_i \|\boldsymbol{x}_i\|$. For a matrix $\boldsymbol{X} \in \mathbb{R}^{m \times n}$, $s_1(\boldsymbol{X})$ denotes the largest singular value of $\boldsymbol{X}$; we have that $s_1(\boldsymbol{X}) = \sup_{\boldsymbol{u} \in \mathbb{R}^n, \|\boldsymbol{u}\|=1} \|\boldsymbol{X}\boldsymbol{u}\| = \sup_{\boldsymbol{u}' \in \mathbb{R}^m, \|\boldsymbol{u}'\|=1} \|\boldsymbol{X}^\top \boldsymbol{u}'\|$. For two matrices $\boldsymbol{X} \in \mathbb{R}^{n \times m}$ and $\boldsymbol{Y} \in \mathbb{R}^{m \times k}$, we have $s_1(\boldsymbol{X}\boldsymbol{Y}) \leq s_1(\boldsymbol{X}) \cdot s_1(\boldsymbol{Y})$. We also have for any $\boldsymbol{X} \in \mathbb{R}^{n \times m}$, $\|\boldsymbol{X}\|_{\mathrm{row}}, \|\boldsymbol{X}^\top\|_{\mathrm{row}} \leq s_1(\boldsymbol{X})$. Finally, for ring R and positive integer n, $\mathrm{GL}_n(R)$ denotes the set of n by n invertible matrices whose entries in R.

Distribution: We recall distributions in the lattice-based cryptography. For a distribution χ, we often write $x \leftarrow \chi$, which indicates that we take a sample x from χ. For a finite set S, $U(S)$ denotes the uniform distribution over S. The Gaussian distribution with mean 0 and variance s^2, denoted by $N(0, s^2)$, is defined by density function $(1/s\sqrt{2\pi}) \cdot \exp(-x^2/2s^2)$ over $\mathbb{R}$. For $\alpha \in (0, 1)$ and positive integer q, we define the *discretized* Gaussian $\bar{\Psi}_\alpha$ as: take sample x from $N(0, \alpha^2/2\pi)$ and output $\lfloor qx \rceil \bmod q$. For positive real s, the n-dimensional Gaussian function is defined as $\rho_s(\boldsymbol{x}) = \exp(-\pi\|\boldsymbol{x}\|^2/s^2)$. For positive real s and countable set A, the *discrete* Gaussian distribution $D_{A,s}$ is defined by $D_{A,s}(\boldsymbol{x}) = \frac{\rho_s(\boldsymbol{x})}{\sum_{\boldsymbol{y} \in A} \rho_s(\boldsymbol{y})}$.

2.1 Lattices

A (full-rank) lattice in $\mathbb{R}^n$ is $\Lambda = \{\sum_{i=1}^n x_i \boldsymbol{b}_i : x_i \in \mathbb{Z}\}$, where $\boldsymbol{b}_1, \dots, \boldsymbol{b}_n \in \mathbb{R}^n$ are linearly independent over $\mathbb{R}^n$. Matrix $\boldsymbol{B} = [\boldsymbol{b}_1 \dots \boldsymbol{b}_n]$ is a basis of lattice Λ. For $\boldsymbol{A} \in \mathbb{Z}_q^{n \times m}$ and $\boldsymbol{u} \in \mathbb{Z}_q^n$, we define lattices and their shift:

$$\begin{aligned}
\Lambda_q(\boldsymbol{A}) &= \{\boldsymbol{y} \in \mathbb{Z}^m : \exists \boldsymbol{s} \in \mathbb{Z}_q^n \text{ such that } \boldsymbol{y} \equiv \boldsymbol{A}^\top \boldsymbol{s} \pmod{q}\}, \\
\Lambda_q^\perp(\boldsymbol{A}) &= \{\boldsymbol{e} \in \mathbb{Z}^m : \boldsymbol{A}\boldsymbol{e} \equiv \boldsymbol{0} \pmod{q}\}, \\
\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A}) &= \{\boldsymbol{e} \in \mathbb{Z}^m : \boldsymbol{A}\boldsymbol{e} \equiv \boldsymbol{u} \pmod{q}\}.
\end{aligned}$$

We recall the property of very structured matrix $\boldsymbol{G}$.

Theorem 2.1 (Adapted version of [MP12b, Theorem 4.1]). *Let $q \geq 2$, $n \geq 1$, $k = \lceil \lg q \rceil$, and $\bar{m} = nk$ be integers. Let $\boldsymbol{g} = (1, 2, \dots, 2^{k-1}) \in \mathbb{Z}^k$ and $\boldsymbol{G} = \boldsymbol{I}_n \otimes \boldsymbol{g}$. Then the lattice $\Lambda_q^\perp(\boldsymbol{G})$ has a known basis $\boldsymbol{S} \in \mathbb{Z}^{\bar{m} \times \bar{m}}$ with $\|\tilde{\boldsymbol{S}}\| \leq \sqrt{5}$ and $\|\boldsymbol{S}\| \leq \max\{\sqrt{5}, \sqrt{k}\}$.*

Recently, Micciancio and Peikert introduced a new notion of "trapdoors" for lattices. Let $m = \bar{m} + nk$, where $k = \lceil \lg q \rceil$. We review their notion of trapdoors.

Definition 2.1 (Adapted, [MP12b, Definition 5.2]). *Let $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and $\boldsymbol{G} \in \mathbb{Z}_q^{n\times w}$ be matrices with $m \geq w \geq n$. We say a matrix $\boldsymbol{R} \in \mathbb{Z}^{(m-w)\times w}$ is a $\boldsymbol{G}$-trapdoor with tag $\boldsymbol{H} \in \mathrm{GL}_n(\mathbb{Z}_q) \subseteq \mathbb{Z}_q^{n\times n}$ if $\boldsymbol{A}\,[\boldsymbol{R};\boldsymbol{I}_w] = \boldsymbol{H}\boldsymbol{G}$. The* quality *of the trapdoor is measured by $s_1(\boldsymbol{R})$.*

Theorem 2.2. *We borrow the following algorithms in [MP12b], which are improvements of those in the literature [Ajt99,GPV08,AP09,ABB10,Pei10]. We set $k = \lceil \lg q \rceil$ and $m = \bar{m} + nk$ for simplicity of notation.*

$\mathsf{GenTrap}^D(\bar{\boldsymbol{A}},\boldsymbol{H})$**:** *Given a matrix $\bar{\boldsymbol{A}} \in \mathbb{Z}_q^{n\times\bar{m}}$, an invertible matrix $\boldsymbol{H} \in \mathrm{GL}_n(\mathbb{Z}_q)$, and a distribution D over $\mathbb{Z}_q$, it outputs $\boldsymbol{A} = [\bar{\boldsymbol{A}} \mid \boldsymbol{H}\boldsymbol{G} - \bar{\boldsymbol{A}}\boldsymbol{R}] \in \mathbb{Z}_q^{n\times(\bar{m}+nk)}$ and its trapdoor $\boldsymbol{R} \in \mathbb{Z}_q^{\bar{m}\times nk}$ with tag $\boldsymbol{H}$, where $\boldsymbol{R}$ is chosen from distribution D.*
In particular, we often set q as an odd prime, $\bar{m} = n \lg q + \omega(\lg \kappa)$, $D = U(\{-1,+1\})$, and choose $\bar{\boldsymbol{A}}$ from $\mathbb{Z}_q^{n\times\bar{m}}$ uniformly at random. These settings yield the obtained matrix $\boldsymbol{A}$ as $\mathsf{negl}(\kappa)$-uniform and $s_1(\boldsymbol{R}) \leq C(\sqrt{\bar{m}} + \sqrt{nk})$ with overwhelming probability.

$\mathsf{SampleD}(\boldsymbol{R},\boldsymbol{A},\boldsymbol{H},\boldsymbol{u},s)$**:** *The input is $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, its trapdoor $\boldsymbol{R} \in \mathbb{Z}_q^{\bar{m}\times nk}$ with tag $\boldsymbol{H} \in \mathrm{GL}_n(\mathbb{Z}_q)$, and a target vector $\boldsymbol{u}$, and Gaussian parameter $s > \sqrt{s_1(\boldsymbol{R})^2+1}\cdot\sqrt{7}\cdot\omega(\sqrt{\lg n})$. It outputs $\boldsymbol{x}$ according to a distribution statistically close to $D_{\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A}),s}$; roughly speaking, it samples $\boldsymbol{x}$ from $D_{\mathbb{Z},s}^m$ conditioned on $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{u}$.*

2.2 Assumption

The learning with errors (LWE) problem proposed by Regev [Reg09] is a generalization of the learning parity noise (LPN) problem.

For vector $\boldsymbol{s} \in \mathbb{Z}_q^n$ and distribution χ over $\mathbb{Z}_q$, let $A(\boldsymbol{s},\chi)$ be a distribution over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ defined by taking samples $\boldsymbol{a} \leftarrow \mathbb{Z}_q^n$ and $x \leftarrow \chi$, and outputting $(\boldsymbol{a},\boldsymbol{a}^\top\boldsymbol{s} + x)$.

Definition 2.2 (The LWE problem and assumption). *For integer $q = q(n)$ and distribution χ over $\mathbb{Z}_q$, the learning with errors problem, $\mathrm{LWE}(q,\chi)$, distinguishes oracle $A(\boldsymbol{s},\chi)$ from oracle $U(\mathbb{Z}_q^n \times \mathbb{Z}_q)$ for uniformly random $\boldsymbol{s} \in \mathbb{Z}_q^n$.*

We say the LWE assumption holds if for any PPT adversary $\mathcal{A}$, its advantage

$$\mathsf{Adv}_{\mathcal{A},\mathrm{LWE}(q,\chi)}(n) = \left|\Pr[\mathcal{A}^{A(s,\chi)}(1^n) = 1] - \Pr[\mathcal{A}^{U(\mathbb{Z}_q^n\times\mathbb{Z}_q)}(1^n) = 1]\right| = \mathsf{negl}(n),$$

where $\boldsymbol{s} \leftarrow \mathbb{Z}_q^n$.

It is well-known that under (quantum) reductions, solving the LWE problem *on average* is as hard as the *worst* case of the approximation version of the shortest independent vector problem, SIVP_γ, and the decision version of the shortest vector problem, GapSVP_γ, where γ is an approximation factor for appropriate parameters. In particular, we have a reduction with parameter $\chi = \bar{\Psi}_\alpha$, $\alpha q \geq 2\sqrt{n}$, and $\gamma = \tilde{O}(n/\alpha)$. See [Reg09,Pei09] for details.

3 Predicate Encryption

We review the syntax of predicate encryption.

Definition 3.1. *Let $\mathcal{P} : \Phi \times \Sigma \to \{0,1\}$ be a predicate where Φ and Σ denote "key attribute" and "ciphertext attribute" spaces. A predicate encryption scheme for $\mathcal{P}$ is a fourtuplet of algorithms.*

$\mathsf{Setup}(1^\kappa) \to (pp,msk)$**:** *The setup algorithm takes as input security parameter 1^κ and outputs public parameters pp and master secret key msk.*

$\mathsf{Extract}(msk,\phi) \to dk_\phi$**:** *The extraction algorithm takes as input msk and key attribute $\phi \in \Phi$. It outputs decryption key dk_ϕ.*

$\mathsf{Enc}(pp, \sigma, M) \to ct$**:** *The encryption algorithm takes as input pp, ciphertext attribute $\sigma \in \Sigma$, and message $M \in \mathcal{M}$. It outputs ciphertext ct.*

$\mathsf{Dec}(pp, dk_\phi, ct) \to M$ **or** $\perp$**:** *The decryption algorithm takes as input decryption key dk_ϕ and ciphertext ct. It outputs either $M \in \mathcal{M}$ or rejection symbol $\perp$.*

We define slightly weak correctness for decryption. For any $\phi \in \Phi$, $\sigma \in \Sigma$, and $M \in \mathcal{M}$, if $\mathcal{P}(\phi, \sigma) = 1$ then

$$\Pr\left[M = \tilde{M} : \begin{array}{l} (pp, msk) \leftarrow \mathsf{Setup}(1^\kappa); dk_\phi \leftarrow \mathsf{Extract}(msk, \phi); \\ ct \leftarrow \mathsf{Enc}(pp, \sigma, M); \tilde{M} \leftarrow \mathsf{Dec}(pp, dk_\phi, ct); \end{array}\right]$$

is overwhelming probability and if $\mathcal{P}(\phi, \sigma) = 0$ then

$$\Pr\left[\tilde{M} = \perp : \begin{array}{l} (pp, msk) \leftarrow \mathsf{Setup}(1^\kappa); dk_\phi \leftarrow \mathsf{Extract}(msk, \phi); \\ ct \leftarrow \mathsf{Enc}(pp, \sigma, M); \tilde{M} \leftarrow \mathsf{Dec}(pp, dk_\phi, ct); \end{array}\right]$$

is overwhelming probability. As in [AFV11], our construction satisfies the different correctness condition: the latter condition is replaced with the condition that if $\mathcal{P}(\phi, \sigma) = 0$ then $\mathsf{Dec}(pp, dk_\phi, ct)$ is computationally indistinguishable from a uniformly random element in $\mathcal{M}$. One can use a suitable message padding to obtain the original correctness, if an IPE scheme has message space $\{0,1\}^\ell$ for sufficiently large ℓ.

We next review the security definition of predicate encryption. Roughly speaking, we say that a PE scheme is weakly attribute hiding in a selective attribute setting against chosen-plaintext attacks (wAH-sA-CPA), if any adversary cannot distinguish $\mathsf{Enc}(pp, \sigma_0, M_0)$ or $\mathsf{Enc}(pp, \sigma_1, M_1)$, where σ_0 and σ_1 are declared at the initialization, even if the adversary can query the decryption key dk_ϕ for $\mathcal{P}(\sigma_0, \phi) = \mathcal{P}(\sigma_1, \phi) = 0$. The precise definition follows:

Definition 3.2 (wAH-sA-CPA security). *Let PE be a predicate encryption scheme, $\mathcal{A}$ an adversary, and κ a security parameter. The experiment between a challenger and adversary $\mathcal{A}$, $\mathsf{Expt}^{\mathsf{wah\text{-}sa\text{-}cpa}}_{\mathcal{A},\mathsf{PE}}(1^\kappa)$, is defined as follows:*

Initialization: *Given security parameter 1^κ, run adversary $\mathcal{A}$ with 1^κ. Receive two ciphertext attributes $\sigma_0, \sigma_1 \in \Sigma$ from $\mathcal{A}$. Run $(pp, msk) \leftarrow \mathsf{Setup}(1^\kappa)$. Flip a coin $b \leftarrow \{0,1\}$.*

Learning Phase: *Feed pp to adversary $\mathcal{A}$. Adversary $\mathcal{A}$ could issue queries to the following oracles in any order and many times except for the constraint regarding oracle* Challenge.

- *Oracle* Extract *receives key attribute $\phi \in \Phi$ subject to the restriction that $\mathcal{P}(\phi, \sigma_0) = \mathcal{P}(\phi, \sigma_1) = 0$. If so, it obtains $dk_\phi \leftarrow \mathsf{Extract}(msk, \phi)$ and provides $\mathcal{A}$ with dk_ϕ.*
- *Oracle* Challenge *receives two messages M_0 and M_1. It obtains $C \leftarrow \mathsf{Enc}(pp, \sigma_b, M_b)$ and provides $\mathcal{A}$ with C.*

Eventually, $\mathcal{A}$ halts after it outputs its decision, $b' \in \{0,1\}$.

Finalization: *Output 1 if $b' = b$. Otherwise, output 0.*

We define the advantage of $\mathcal{A}$ as

$$\mathsf{Adv}^{\mathsf{wah\text{-}sa\text{-}cpa}}_{\mathcal{A},\mathsf{PE}}(\kappa) = \left|\Pr[\mathsf{Expt}^{\mathsf{wah\text{-}sa\text{-}cpa}}_{\mathcal{A},\mathsf{PE}}(1^\kappa) = 1 \mid b = 0] - \Pr[\mathsf{Expt}^{\mathsf{wah\text{-}sa\text{-}cpa}}_{\mathcal{A},\mathsf{PE}}(1^\kappa) = 1 \mid b = 1]\right|.$$

We say that PE is weakly attribute hiding against chosen-plaintext attacks in selective attribute setting (wAH-sA-CPA-secure) if $\mathsf{Adv}^{\mathsf{wah\text{-}sa\text{-}cpa}}_{\mathcal{A},\mathsf{PE}}(\kappa)$ is negligible for every PPT adversary $\mathcal{A}$.

4 Pseudo-Commutativity of Invertible Difference Encoding

In this section, we define H and $H_{\boldsymbol{g}}$ such that, for any $\mathfrak{a}$, $H(\mathfrak{a}) \cdot \boldsymbol{G} = \boldsymbol{G} \cdot H_{\boldsymbol{g}}(\mathfrak{a})$ holds and $s_1(H_{\boldsymbol{g}}(\mathfrak{a}))$ is small. We first recall the polynomial rings. After a reminder of the invertible difference encoding, we define its companion $H_{\boldsymbol{g}}$.

4.1 Quick Reminder of Rings

Consider a finite ring $R = \mathbb{Z}_q[X]/\langle \mathfrak{g} \rangle$, where $\mathfrak{g} \in \mathbb{Z}_q[X]$ is monic and of degree n. If q is prime and $\mathfrak{g}$ is irreducible over $\mathbb{Z}_q$, ring R is the field $\mathrm{GF}(q^n)$.

We define the mapping $\tau : R \to \mathbb{Z}_q^n$ by $\mathfrak{a} = a_0 + a_1 X + \dots a_{n-1} X^{n-1} \mapsto (a_0, \dots, a_{n-1})^\top$. By this mapping (as known as "coefficient embedding"), we can identify a polynomial in R with a vector in $\mathbb{Z}_q^n$. We next define $\mathrm{Rot} : R \to \mathbb{Z}_q^{n \times n}$ by

$$\mathfrak{a} = a_0 + a_1 X + \dots a_{n-1} X^{n-1} \mapsto [\tau(\mathfrak{a})\, \tau(\mathfrak{a}X)\, \dots\, \tau(\mathfrak{a}X^{n-1})],$$

which is borrowed from Micciancio [Mic07]. We note that

$$\mathrm{Rot}(\mathfrak{a}) \cdot \tau(\mathfrak{b}) = \tau(\mathfrak{a}\mathfrak{b}), \mathrm{Rot}(\mathfrak{a}) \cdot \mathrm{Rot}(\mathfrak{b}) = \mathrm{Rot}(\mathfrak{a}\mathfrak{b}), \text{ and } \mathrm{Rot}(\mathfrak{a}) + \mathrm{Rot}(\mathfrak{b}) = \mathrm{Rot}(\mathfrak{a} + \mathfrak{b}),$$

and, thus, Rot is a ring-homomorphism from R into $\mathbb{Z}_q^{n \times n}$.

4.2 Invertible Difference Encoding *H*

Lattice-based cryptography often employs an encoding $H : \mathrm{GF}(q^n) \to \mathbb{Z}_q^{n \times n}$ for prime q, e.g., [PW08, due to Micciancio] and [ABB10]. Hereafter we stick to prime q.

We say that H is an invertible difference if for any two distinct polynomials $\mathfrak{a} \neq \mathfrak{a}' \in \mathrm{GF}(q^n)$, the difference of outputs, $H(\mathfrak{a}) - H(\mathfrak{a}')$, is always invertible.

In this paper, we employ explicit H defined by $H(\mathfrak{a}) := \mathrm{Rot}(\mathfrak{a})$. It holds that $H(\mathfrak{a}) - H(\mathfrak{a}') = H(\mathfrak{a} - \mathfrak{a}')$ for any $\mathfrak{a} \neq \mathfrak{a}'$. If $\mathfrak{a} - \mathfrak{a}'$ is a unit, that is, $\mathfrak{a} \neq \mathfrak{a}' \in \mathrm{GF}(q^n)$, then $H(\mathfrak{a} - \mathfrak{a}')$ is also a unit in $\mathbb{Z}_q^{n \times n}$. In addition, we note that for any constant $a \in \mathbb{Z}_q \subset \mathrm{GF}(q^n)$, $H(a) = a\boldsymbol{I}_n$.

4.3 New Encoding $H_{\boldsymbol{g}}$

We define a new encoding, denoted by $H_{\boldsymbol{g}}$, that maps an element in $\mathrm{GF}(q^n)$ to matrices in $\{0, 1, \dots, b-1\}^{nk \times nk}$ and gives pseudo-commutativity with $\boldsymbol{G}$ and H.

Let $b \geq 2$ be a positive integer and let B be the range $\{0, 1, \dots, b-1\} \subset \mathbb{Z}_q$. We define $k = \lceil \log_b q \rceil$ and $\boldsymbol{g} = (1, b, \dots, b^{k-1})$. The gadget matrix $\boldsymbol{G}$ in [MP12b] is defined by

$$\boldsymbol{G} = \boldsymbol{I}_n \otimes \boldsymbol{g} = \begin{bmatrix} -\boldsymbol{g}- & & & \\ & -\boldsymbol{g}- & & \\ & & \ddots & \\ & & & -\boldsymbol{g}- \end{bmatrix} = \begin{bmatrix} 1\ b \dots\ b^{k-1} & & & \\ & 1\ b \dots\ b^{k-1} & & \\ & & \ddots & \\ & & & 1\ b \dots\ b^{k-1} \end{bmatrix} \in \mathbb{Z}_q^{n \times nk}.$$

For $a \in \mathbb{Z}_q$, we define b-ary decomposition of a by $d_{\boldsymbol{g}}(a) = (a_1, \dots, a_k)^\top \in B^k$, on which we have that $\boldsymbol{g} \cdot d_{\boldsymbol{g}}(a) = \sum_{i=1}^{k} a_i \cdot b^{i-1} = a$.

We define $H_{\boldsymbol{g}}(\mathfrak{a})$ as the b-ary decomposition of $H(\mathfrak{a})$. More formally, we first define the mapping $D_{\boldsymbol{g}}$ by

$$D_{\boldsymbol{g}} : a \in \mathbb{Z}_q \mapsto [d_{\boldsymbol{g}}(a)\ d_{\boldsymbol{g}}(ba)\ \dots\ d_{\boldsymbol{g}}(b^{k-1}a)] \in B^{k \times k}.$$

By the definition of $D_{\boldsymbol{g}}$, we have that $\boldsymbol{g} \cdot D_{\boldsymbol{g}}(a) = (a, ba, \dots, b^{k-1}a) = a \cdot \boldsymbol{g}$, which is a source of the pseudo-commutativity. Next, we extend the domain of $D_{\boldsymbol{g}}$ to any matrix $\boldsymbol{A} = \{a_{i,j}\} \in \mathbb{Z}_q^{n \times m}$ as follows:

$$D_{\boldsymbol{g}}(\boldsymbol{A}) = \begin{bmatrix} D_{\boldsymbol{g}}(a_{1,1}) & D_{\boldsymbol{g}}(a_{1,2}) & \dots & D_{\boldsymbol{g}}(a_{1,m}) \\ D_{\boldsymbol{g}}(a_{2,1}) & D_{\boldsymbol{g}}(a_{2,2}) & \dots & D_{\boldsymbol{g}}(a_{2,m}) \\ \vdots & \vdots & \ddots & \vdots \\ D_{\boldsymbol{g}}(a_{n,1}) & D_{\boldsymbol{g}}(a_{n,2}) & \dots & D_{\boldsymbol{g}}(a_{n,m}) \end{bmatrix} \in B^{nk \times mk}.$$

Finally, we define $H_{\boldsymbol{g}}$ that maps a polynomial into a matrix as follows:

$$\mathfrak{a} = a_0 + a_1 x + \cdots + a_{n-1}x^{n-1} \in \mathrm{GF}(q^n) \mapsto D_{\boldsymbol{g}}(\mathrm{Rot}(\mathfrak{a})) \in B^{nk \times nk}.$$

The mapping $H_{\boldsymbol{g}}$ has two properties that are crucial for our construction. One is pseudo-commutativity with $\boldsymbol{G}$ and H and the other is a small matrix norm.

Lemma 4.1. *Let $\boldsymbol{G} = \boldsymbol{I}_n \otimes \boldsymbol{g} \in \mathbb{Z}_q^{n \times nk}$. It holds that, for any $\mathfrak{a} \in \mathrm{GF}(q^n)$, $\boldsymbol{G} \cdot H_{\boldsymbol{g}}(\mathfrak{a}) = H(\mathfrak{a}) \cdot \boldsymbol{G}$.*

Proof. We show that for any matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n \times n}$, $\boldsymbol{G} \cdot D_{\boldsymbol{g}}(\boldsymbol{A}) = \boldsymbol{A} \cdot \boldsymbol{G}$, where $\boldsymbol{A} = [\boldsymbol{a}_1 \mid \cdots \mid \boldsymbol{a}_n]$. We divide the matrices into k submatrices; $\boldsymbol{G} \cdot D_{\boldsymbol{g}}(\boldsymbol{A}) = [\boldsymbol{L}_1 \mid \cdots \mid \boldsymbol{L}_k]$ and $\boldsymbol{A} \cdot \boldsymbol{G} = [\boldsymbol{R}_1 \mid \cdots \mid \boldsymbol{R}_k]$, where $\boldsymbol{L}_i, \boldsymbol{R}_i \in \mathbb{Z}_q^{n \times n}$. It is easy to check that $\boldsymbol{L}_i = \boldsymbol{a}_i \otimes \boldsymbol{g} = \boldsymbol{R}_i$ for any i. □

Lemma 4.2. *For any $\mathfrak{a} \in \mathrm{GF}(q^n)$, $\|H_{\boldsymbol{g}}(\mathfrak{a})\|_{\mathrm{row}} \le (b-1) \cdot \sqrt{nk}$ and $s_1(H_{\boldsymbol{g}}(\mathfrak{a})) \le (b-1)nk$.*

Proof. Since $H_{\boldsymbol{g}} \in B^{nk \times nk}$, the maximal length of the rows is at most $(b-1)\sqrt{nk}$. The latter bound is obtained by the upper bound on the length of $(b-1) \cdot \mathbf{1} \cdot \boldsymbol{u}$, where $\mathbf{1}$ is an nk-dimensional all-1 matrix and $\boldsymbol{u}$ is a unit vector. □

4.4 On the Case Composite q

Although we have stuck to prime q here, lattice-based cryptography often employs $q = p^e$, say $q = 2^k$, or $q = \prod_i p_i$ for small prime p_i for the sake of easiness and speed of implementations. Therefore, one would extend our technique into such cases.

Micciancio and Peikert [MP12b, Section 6.1 of the ePrint version] and Alperin-Sheriff and Peikert [ASP12, Section 5.1] defined an encoding $H : \mathbb{Z}_q[X]/\langle \mathfrak{g} \rangle \to \mathbb{Z}_q^{n \times n}$, where $\mathfrak{g}$ is a monic degree-n polynomial in $\mathbb{Z}[X]$ and irreducible modulo every prime p dividing q. In their constructions, H is a ring homomorphism from $R = \mathbb{Z}_q[X]/\langle \mathfrak{g} \rangle$ into $\mathbb{Z}_q^{n \times n}$. Thus, if $\mathfrak{u} \in R$ is a unit, then $H(\mathfrak{u})$ is invertible. In general, $H(\mathfrak{u})$ is not invertible even for non-zero $\mathfrak{u} \in R \setminus R^*$.

This property suffices for public-key encryption, IBE, and signature, but, may trouble designers of predicates. If one can ensure that the inner product results in either a unit or zero of R, one can employ the above techniques.

4.5 On the Ring-LWE Setting

When q is a prime, we can extend our new encoding into the ring-LWE setting [LPR10]. Let us consider the cyclotomic ring $\mathcal{R} = \mathbb{Z}[X]/\langle \Phi_m(X) \rangle$, where $\Phi_m(X)$ denotes the m-th cyclotomic polynomial. Let n be the degree of $\Phi_m(X)$. For any poly(n)-bounded prime q, we let $\mathcal{R}_q = \mathcal{R}/q\mathcal{R}$.

The Micciancio–Peikert algorithm in the ring-LWE setting: Let $\boldsymbol{g} = (1, b, \ldots, b^{k-1}) \in \mathcal{R}_q^k$. In the ring setting, we will use $\boldsymbol{g}$ directly instead of $\boldsymbol{G}$. Let us set $R = \mathbb{Z}_q[X]/\langle \mathfrak{g} \rangle \simeq \mathrm{GF}(q^n)$ as in the LWE case.

Micciancio and Peikert [MP12a] define $\mathcal{R}$ to be R-module [3] by extending the ideas in [MP12b, Section 6.1 of the ePrint version] and [ASP12, Section 5.1]. Formally speaking, for $\mathfrak{a} \in R$ and $b \in \mathcal{R}$, scalar multiplication $\mathfrak{a} \odot b \in \mathcal{R}$ is defined by $\sigma^{-1}(\mathfrak{a} \cdot \sigma(b)) \in \mathcal{R}$, where $\sigma : \mathcal{R} \to R$ is an additive isomorphism. (Notice that R and $\mathcal{R}$ are additively isomorphic to $\mathbb{Z}_q^n$.) By the construction, $\mathcal{R}$ is an R-module and $\mathfrak{a}$ acts as the linear transformation over $\mathcal{R}$. Now, the trapdoor of $\boldsymbol{a} \in \mathcal{R}_q^{\bar{m}+k}$ with tag $\mathfrak{h} \in R$ is short $\boldsymbol{R} \in \mathcal{R}_q^{\bar{m}+k}$ satisfying $\boldsymbol{a} = [\bar{\boldsymbol{a}} \mid \mathfrak{h} \odot \boldsymbol{g} - \bar{\boldsymbol{a}}\boldsymbol{R}]$. We can define the new encoding $h_{\boldsymbol{g}} : R \to R^{k \times k}$ in a similar way to the LWE case.

[3] We say $\mathcal{R}$ is R-module if for any $\mathfrak{r}, \mathfrak{s} \in R$ and any $x, y \in \mathcal{R}$, we have $\mathfrak{r}(x+y) = \mathfrak{r}x + \mathfrak{r}y$, $(\mathfrak{r}+\mathfrak{s})x = \mathfrak{r}x + \mathfrak{s}x$, $(\mathfrak{r}\mathfrak{s})x = \mathfrak{r}(\mathfrak{s}x)$, and $1_R x = x$.

Langlois and Stehlé [Ste12] also pointed out another way. Let us consider the case that n is even and $\Phi_m(X)$ is split into two polynomials $\mathfrak{f}_1$ and $\mathfrak{f}_2$ of degree $n/2$ which are irreducible over $\mathbb{Z}_q$. In such a case, we have $\mathcal{R}_q \simeq \mathbb{Z}_q[X]/\langle \mathfrak{f}_1 \rangle \times \mathbb{Z}_q[X]/\langle \mathfrak{f}_2 \rangle \simeq \mathrm{GF}(q^{n/2})^2$. We let $R = \mathrm{GF}(q^{n/2})$ and consider $H : R \to \mathcal{R}_q$ as follows:

We first define a duplicating function $\mathrm{dp} : \mathrm{GF}(q^{n/2}) \to \mathrm{GF}(q^{n/2})^2$ as $\mathfrak{a} \mapsto (\mathfrak{a}, \mathfrak{a})$. By the Chinese remainder theorem, we have invertible mapping $\tau : \mathbb{Z}_q[X]/\langle \Phi_m(X) \rangle \to \mathbb{Z}_q[X]/\langle \mathfrak{f}_1 \rangle \times \mathbb{Z}_q[X]/\langle \mathfrak{f}_2 \rangle$ as $\mathfrak{a} \mapsto (\mathfrak{a} \bmod \mathfrak{g}_1, \mathfrak{a} \bmod \mathfrak{g}_2)$. Then, we define the full-rank difference encoding from $\mathrm{GF}(q^{n/2})$ to R_q as $H = \tau^{-1} \circ \mathrm{dp}$. By the construction, H is an isomorphism from $\mathrm{GF}(q^{n/2})$ to a sub-ring of $\mathcal{R}_q$, which is a field.

Now, the trapdoor of $\boldsymbol{a} \in \mathcal{R}_q^{\bar{m}+k}$ with tag $\mathfrak{h} \in R$ is $\boldsymbol{R} \in \mathcal{R}_q^{\bar{m}+k}$ satisfying $\boldsymbol{a} = [\bar{\boldsymbol{a}} \mid H(\mathfrak{h})\boldsymbol{g} - \bar{\boldsymbol{a}}\boldsymbol{R}]$. We can define the new encoding $\boldsymbol{H_g} : R \to \mathcal{R}_q^{k \times k}$ in a similar way to the LWE case.

We defer the details to Appendix D.

5 Our Construction

We describe our IPE scheme that supports inner-product predicates over $\mathrm{GF}(q^n)^\mu$. The scheme is obtained by applying our ideas in the introduction to the AFV IPE scheme. The extension to HIPE is deferred to Appendix C.

Let $\kappa \in \mathbb{N}$ be a security parameter. Let μ be the length of predicate and attribute vectors. Let n be a dimension of lattices and let q and m be the parameters that define the matrices. Let $\mathfrak{g} = \mathfrak{g}(x) \in \mathbb{Z}_q[x]$ be a monic, irreducible polynomial of degree n that explicitly defines $\mathrm{GF}(q^n)$.

For simplicity, we set $b = 2$, $B = \{0,1\}$, and $k = k(\kappa,\mu) = \lceil \lg q \rceil$. Other choices are possible. For simplicity, we let $\zeta = \zeta(n)$ denote a fixed $\omega(\sqrt{\lg n})$ function. Let $s = s(\kappa,\mu)$ and $\alpha = \alpha(\kappa,\mu)$ be positive reals that define the Gaussians.

5.1 Construction

$\mathsf{Setup}(1^\kappa, n, q, m, \ell, s, \alpha, \mathfrak{g}, k)$**:** On input a security parameter 1^κ and additional parameters:

1. Generate a random matrix with a trapdoor by running $(\boldsymbol{A}, \boldsymbol{R_A}) \leftarrow \mathsf{GenTrap}(1^\kappa, q, n, m)$.
2. Choose μ uniformly random matrices $\boldsymbol{B}_i \leftarrow \mathbb{Z}_q^{n \times nk}$ for $i \in [\mu]$.
3. Choose a random matrix $\boldsymbol{U} = [\boldsymbol{u}_1 \mid \cdots \mid \boldsymbol{u}_\ell] \leftarrow \mathbb{Z}_q^{n \times \ell}$.

Output $pp = ((n, q, m, \ell, s, \alpha, \mathfrak{g}, k), \boldsymbol{A}, \{\boldsymbol{B}_i\}, \boldsymbol{U})$ and $msk = (\boldsymbol{R_A}, pp)$.

$\mathsf{Extract}(pp, msk, \vec{v})$**:** On input a key-attribute vector $\vec{v} = (\mathfrak{v}_1, \ldots, \mathfrak{v}_\mu)^\top \in \mathrm{GF}(q^n)^\mu$:

1. Define the matrices $\boldsymbol{B}_{\vec{v}} = \sum_{i=1}^{\mu} \boldsymbol{B}_i \cdot \boldsymbol{H_g}(\mathfrak{v}_i) \in \mathbb{Z}_q^{n \times nk}$ and $\boldsymbol{A}_{\vec{v}} = [\boldsymbol{A} \mid \boldsymbol{B}_{\vec{v}}] \in \mathbb{Z}_q^{n \times (m+nk)}$.
2. Sample vectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_\ell$ by using the master secret key $\boldsymbol{R_A}$; Formally, for $i = 1, \ldots, \ell$, take sample $\boldsymbol{e}_i \leftarrow \mathsf{SampleD}(\boldsymbol{R_A}, \boldsymbol{A}_{\vec{v}}, \boldsymbol{I}, \boldsymbol{u}_i, s)$.
3. Set $\boldsymbol{E}_{\vec{v}} = [\boldsymbol{e}_1 \mid \cdots \mid \boldsymbol{e}_\ell]$. (Notice that $\boldsymbol{A}_{\vec{v}} \cdot \boldsymbol{E}_{\vec{v}} = \boldsymbol{U}$.)

Output $dk_{\vec{v}} = \boldsymbol{E}_{\vec{v}}$.

$\mathsf{Enc}(pp, \vec{w}, \boldsymbol{m})$**:** On input pp, a ciphertext-attribute vector $\vec{w} = (\mathfrak{w}_1, \ldots, \mathfrak{w}_\mu)^\top \in \mathrm{GF}(q^n)^\mu$, and a message $\boldsymbol{m} \in \{0,1\}^\ell$:

1. Choose a random vector $\boldsymbol{s} \leftarrow \mathbb{Z}_q^n$.
2. Set $\boldsymbol{c}_0 \leftarrow \boldsymbol{A}^\top \boldsymbol{s} + \boldsymbol{x}_0$, where $\boldsymbol{x}_0 \leftarrow \chi^m$.
3. Set $\boldsymbol{c}' \leftarrow \boldsymbol{U}^\top \boldsymbol{s} + \boldsymbol{x}' + \lfloor q/2 \rfloor \boldsymbol{m}$, where $\boldsymbol{x}' \leftarrow \chi^\ell$.
4. For $i = 1, \ldots, \mu$; sample $\boldsymbol{R}_i \leftarrow \{-1,+1\}^{m \times nk}$ and set $\boldsymbol{c}_i \leftarrow (\boldsymbol{B}_i + H(\mathfrak{w}_i) \cdot \boldsymbol{G})^\top \boldsymbol{s} + \boldsymbol{R}_i^\top \boldsymbol{x}_0 \in \mathbb{Z}_q^{nk}$.
5. Output $ct = (\boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_\mu, \boldsymbol{c}')$.

$\mathsf{Dec}(pp, dk_{\vec{v}}, ct)$**:** On input pp, a decryption key $\boldsymbol{E}_{\vec{v}}$, and $ct = (\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\mu, \boldsymbol{c}')$:

1. Compute $\boldsymbol{c}_{\vec{v}} \leftarrow \sum_{i=1}^{\mu} \boldsymbol{H_g}(\mathfrak{v}_i) \cdot \boldsymbol{c}_i$.
2. Let $\boldsymbol{c} \leftarrow [\boldsymbol{c}_0; \boldsymbol{c}_{\vec{v}}] \in \mathbb{Z}_q^{m+\bar{m}}$.
3. Compute $\boldsymbol{d} \leftarrow \boldsymbol{c}' - \boldsymbol{E}_{\vec{v}}^\top \boldsymbol{c} \bmod q$ and output $\lfloor (2/q)\boldsymbol{d} \rceil \bmod 2$.

Remark 5.1. In the following, we will take the noise of $\boldsymbol{c}_0$ and $\boldsymbol{c}'$ from $\chi = \bar{\Psi}_\alpha$. We note that $\bar{\Psi}_\alpha$ has a good tail bound on the inner product and this is why we employ the conservative distribution $\bar{\Psi}_\alpha$.

We can replace the distribution $\bar{\Psi}_\alpha$ with $D_{\mathbb{Z},\sigma}$. We then change the noises $\boldsymbol{x}_i$ with $\boldsymbol{x}_i \leftarrow D_{\mathbb{Z},r}^{nk}$ where $r = \sqrt{\|\boldsymbol{x}_0\|^2 + nk\sigma^2}\cdot\zeta$ as in the CCA2 secure PKE scheme in [MP12b]. The problem $\mathrm{LWE}(n, q, D_{\mathbb{Z},\sqrt{2}\cdot\alpha q})$ is as hard as $\mathrm{LWE}(n, q, \bar{\Psi}_\alpha)$, which is shown by Gordon, Katz, and Vaikuntanathan [GKV10, Lemma 1] employing [Pei10, Theorem 3.1]. Hence, even if we change the noise distribution from $\bar{\Psi}_\alpha$ to $D_{\mathbb{Z},\sqrt{2}\alpha q}$, we can reduce the security to the lattice problems.

5.2 Correctness, Security, and Parameters

The scheme is correct and secure as the following theorems.

Theorem 5.1. *Let $\chi = \bar{\Psi}_\alpha$. Suppose that $s > 4Cm \cdot \omega(\sqrt{\lg n})$ and $(\alpha q \cdot \omega(\sqrt{\lg \kappa}) + \sqrt{m}/2) \cdot 4C\mu s m^2 < q/5$. Then our scheme is correct.*

Theorem 5.2. *Let $m = 2n \lg q + \omega(\lg \kappa)$ and $s \geq 3C\mu m^{1.5} \cdot \omega(\sqrt{\lg n})$. Suppose that the* LWE$(n, q, \chi)$ *assumption holds. Then, the scheme is selectively and weakly attribute hiding.*

The proofs are obtained by merging those of [AFV11] and [MP12b]. We defer the proofs to Appendix B.

Parameter settings: Let us summarize the constraints on the parameters:

- To satisfy the correctness (Theorem 5.1), we require that $\chi = \bar{\Psi}_\alpha$, $s > 4Cm\omega(\sqrt{\lg n})$, and $(\alpha q \cdot \omega(\sqrt{\lg \kappa}) + \sqrt{m}/2) \cdot 4C\mu s m^2 < q/5$. For example, we can take $q = \Omega(\mu m^{5/2} s)$ and $\alpha \leq \left(\mu m^2 s \cdot \omega(\sqrt{\lg \kappa})\right)^{-1}$ to satisfy the above condition with $q\alpha \cdot \omega(\sqrt{\lg \kappa}) = \sqrt{m}/2$.
- From the security (Theorem 5.2), we obtain the bound that $m = 2n \lg q + \omega(\lg \kappa)$ and $s \geq 3C\mu m^{1.5} \cdot \omega(\sqrt{\lg n})$.
- In order to reduce the security to the worst-case hardness of lattice problems, we require that $q\alpha > 2\sqrt{n}$ and $1/\alpha = \mathrm{poly}(n)$.

For example, the following setting fulfills the above requirements:

$$\begin{aligned} k &= \lceil \lg q \rceil, & \zeta &= \omega(\sqrt{\lg(2m)}), & m &= 3n \lg q, \\ s &= 3C\mu m^{1.5} \cdot \zeta, & q &= 60C^2\mu^2 \cdot m^4 \cdot \zeta, & \alpha &= (120C^2\mu^2 \cdot m^{3.5} \cdot \zeta^2)^{-1}. \end{aligned}$$

By these settings, the security is based on the worst-case hardness of GapSVP$_\gamma$ or SIVP$_\gamma$, where $\gamma = \tilde{\Omega}(\mu^2 n^{4.5})$, while the AFV scheme is based on that with $\gamma = \tilde{\Omega}(\mu^2 n^4)$. (We note that if the AFV scheme also employs the Micciancio–Peikert trapdoor [MP12b] as we did, γ is reduced to $\tilde{\Omega}(\mu^2 n^{3.5})$.)

In our scheme, the size of the public parameter is $nm \lg q + \mu n^2 k \lg q + \ell n \lg q = \Theta(\mu n^2 \lg^2 q) = \tilde{\Theta}(\mu n^2)$, and the size of the ciphertext is $m \lg q + \mu nk \lg q + \ell \lg q = \Theta(\mu n \lg^2 q)$, where ℓ denotes the length of plaintexts.

6 Fuzzy Identity-based Encryption

In this section, we construct a FIBE scheme from an IPE scheme without changing R. We first review the embedding of exact threshold by Katz, Sahai, and Waters. If the IPE scheme hides attribute weakly, we can take logical disjunction in a lazy way.

Exact threshold: For binary vector $\vec{x} \in \{0,1\}^N$, $H_w(\vec{x})$ denotes a Hamming weight of $\vec{x}$, that is, the number of 1 in $\vec{x}$. For binary vectors $\vec{a}, \vec{x} \in \{0,1\}^u$, the exact threshold predicate is denoted by $\mathcal{P}^{\mathsf{th}}_{=t}(\vec{a}, \vec{x})$ and outputs 1 if and only if $H_w(\vec{a} \mathbin{\&} \vec{x}) = t$, where & denotes the logical conjunction. Suppose that $t < q$. Katz, Sahai, and Waters [KSW08] gave an embedding $\mathcal{P}^{\mathsf{th}}_{=t}$ into $\mathcal{P}^{\mathsf{ipe}}$ as follows:

$$\mu = N + 1, \vec{v} = (\vec{a}, 1) \in \mathbb{Z}_q^{\mu}, \text{ and } \vec{w} = (\vec{x}, -t) \in \mathbb{Z}_q^{\mu}.$$

We have that $\vec{w}^{\top}\vec{v} = 0$ if and only if $H_w(\vec{a} \mathbin{\&} \vec{x}) = t$.

6.1 Construction

Now, we implement FIBE with a small universe from IPE. Let $\{0,1\}^N$ be a space of identities. The threshold predicate over $\{0,1\}^N$ is defined by $\mathcal{P}^{\mathsf{th}}_{\geq t}(\vec{a}, \vec{x}) = 1$ if and only if $H_w(\vec{a} \mathbin{\&} \vec{x}) \geq t$.

We observe that the above predicate can be written as $\bigvee_{i=t}^{N} \mathcal{P}^{\mathsf{th}}_{=i}(\vec{a}, \vec{x})$. Hence, repeating ciphertexts of an IPE scheme that supports the relations $\mathcal{P}^{\mathsf{th}}_{\geq i}$ for $i = t, \ldots, N$, we can implement a FIBE scheme by the relation $\mathcal{P}^{\mathsf{th}}_{\geq t}$.

When we employ our IPE scheme, the obtained scheme has a ciphertext of length $(N - t + 1) \cdot O(Nm \lg q) = O(N^2 n \lg^2 q)$ and enjoys the security reduced to the worst-case hardness of lattice problems with approximation factor $\tilde{O}(n^{4.5})$.

Comparison: Agrawal et al. already presented FIBE schemes from lattices [ABV$^+$12]. Their small-universe construction is defined with the identity space $\{0,1\}^N$ as in our case. They gave concrete parameter settings for $\epsilon \in (0, 1/2)$ as follows:

$$N = n^{\epsilon}, q \in [n^6 2^{5N}, 2n^6 2^{5N}], m = n^{1.5} \geq 5n \lg q, \text{ and } \alpha = 2\sqrt{m}/q = 1/(2^{5n^{\epsilon}} \cdot \mathrm{poly}(n)).$$

The length of the ciphertext is $N \cdot O(m \lg q) = O(n^{1.5+2\epsilon} \lg n)$. The security is reduced to the worst-case hardness of $2^{O(n^{\epsilon})}$-approximating gapSVP or SIVP using $2^{O(n^{\epsilon})}$-time algorithms, which is stronger assumption than that we employ.

We note that, their scheme allows identity space $(\mathbb{Z}_q^n)^u$ without drastic changes of parameters whereas our scheme cannot.

Acknowledgments

The author thanks Damien Stehlé, Chris Peikert, Daniele Micciancio, and reviewers for helpful discussions and comments.

References

ABC$^+$11. Michel Abdalla, James Birkett, Dario Catalano, Alexander W. Dent, John Malone-Lee, Gregory Neven, Jacob C. N. Schuldt, and Nigel P. Smart. Wildcarded identity-based encryption. *Journal of Cryptology*, 24(1):42–82, 2011. Combined and extended of two papers [ACD$^+$06,BDNS07].

ACD$^+$06. Michel Abdalla, Dario Catalano, Alexander W. Dent, John Malone-Lee, Gregory Neven, and Nigel P. Smart. Identity-based encryption gone wild. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *ICALP 2006, Part II*, volume 4052 of *LNCS*, pages 300–311. Springer, Heidelberg, 2006. The full version is available at `http://eprint.iacr.org/2006/304`.

ADCM12. Michel Abdalla, Angelo De Caro, and Karina Mochetti. Lattice-based hierarchical inner product encryption. In Alejandro Hevia and Gregory Neven, editors, *LATINCRYPT 2012*, volume 7533 of *LNCS*, pages 121–138. Springer, Heidelberg, 2012.

ABB10. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (H)IBE in the standard model. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 553–572. Springer, Heidelberg, 2010.

ABV$^+$12. Shweta Agrawal, Xavier Boyen, Vinod Vaikuntanathan, Panagiotis Voulgaris, and Hoeteck Wee. Functional encryption for threshold functions (or, fuzzy IBE) from lattices. In Marc Fischlin, Johannes Buchmann, and Mark Manulis, editors, *PKC 2012*, volume 7293 of *LNCS*, pages 280–297. Springer, Heidelberg, 2012.

AFV11. Shweta Agrawal, David Mandell Freeman, and Vinod Vaikuntanathan. Functional encryption for inner product predicates from learning with errors. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 21–40. Springer, Heidelberg, 2011. The full version is available at `http://eprint.iacr.org/2011/410`.

Ajt99. Miklós Ajtai. Generating hard instances of the short basis problem. In Jirí Wiedermann, Peter van Emde Boas, and Mogens Nielsen, editors, *ICALP '99*, volume 1644 of *LNCS*, pages 1–9, 1999.

ASP12. Jacob Alperin-Sheriff and Chris Peikert. Circular and KDM security for identity-based encryption. In Marc Fischlin, Johannes Buchmann, and Mark Manulis, editors, *PKC 2012*, volume 7293 of *LNCS*, pages 334–352. Springer, Heidelberg, 2012.

AP09. Joël Alwen and Chris Peikert. Generating shorter bases for hard random lattices. In Susanne Albers and Jean-Yves Marion, editors, *STACS 2009*, volume 3 of *LIPIcs*, pages 75–86, Germany, 2009. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

AL10. Nuttapong Attrapadung and Benoît Libert. Functional encryption for inner product: Achieving constant-size ciphertexts with adaptive security or support for negation. In Phong Q. Nguyen and David Pointcheval, editors, *PKC 2010*, volume 6056 of *LNCS*, pages 384–402. Springer, Heidelberg, 2010.

BDNS07. James Birkett, Alexander W. Dent, Gregory Neven, and Jacob C. N. Schuldt. Efficient chosen-ciphertext secure identity-based encryption with wildcards. In Josef Pieprzyk, Hossein Ghodosi, and Ed Dawson, editors, *ACISP 2007*, volume 4586 of *LNCS*, pages 274–292. Springer, Heidelberg, 2007.

BB11. Dan Boneh and Xavier Boyen. Efficient selective identity-based encryption without random oracles. *Journal of Cryptology*, 24(4):659–693, 2011. A preliminary version appeared in *EUROCRYPT 2004*, 2004.

BCHK06. Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. *SIAM Journal on Computing*, 36(5):1301–1328, 12 2006.

BH08. Dan Boneh and Michael Hamburg. Generalized identity based and broadcast encryption schemes. In Josef Pieprzyk, editor, *ASIACRYPT 2008*, volume 5350 of *LNCS*, pages 455–470. Springer, Heidelberg, 2008.

BW07. Dan Boneh and Brent Waters. Conjunctive, subset, and range queries on encrypted data. In Salil P. Vadhan, editor, *TCC 2007*, volume 4392 of *LNCS*, pages 535–554. Springer, Heidelberg, 2007.

Boy12. Xavier Boyen. Attribute-based encryption from lattices, 2012. To appear *TCC 2013*.

Bra12. Zvika Brakerski. Fully homomorphic encryption without modulus switching from classical GapSVP. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 868–886. Springer, Heidelberg, 2012. see also `http://eprint.iacr.org/2012/078`.

BGV12. Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. In Shafi Goldwasser, editor, *ITCS 2012*, pages 309–325. ACM, 2012. See also `http://eprint.iacr.org/2011/277`.

BV11. Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In *FOCS 2011*, pages 97–106. IEEE Computer Society, 2011. See also `http://eprint.iacr.org/2011/344`.

CHKP10. David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 523–552. Springer, Heidelberg, 2010.

CLLW11. Jie Chen, Hoon Wei Lim, San Ling, and Huaxiong Wang. The relation and transformation between hierarchical inner product encryption and spatial encryption. Cryptology ePrint Archive, Report 2011/455, 2011. Available at `http://eprint.iacr.org/2011/455`.

CLL$^+$12. Jie Chen, Hoon Wei Lim, San Ling, Huaxiong Wang, and Ta Toan Khoa Nguyen. Revocable identity-based encryption from lattices. In Willy Susilo, Yi Mu, and Jennifer Seberry, editors, *ACISP 2012*, volume 7372 of *LNCS*, pages 390–403. Springer, Heidelberg, 2012. The full version is available at `http://eprint.iacr.org/2011/583`.

CD09. Ronald Cramer and Ivan Damgård. On the amortized complexity of zero-knowledge protocols. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 177–191. Springer, Heidelberg, 2009.

DORS08. Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM Journal on Computing*, 38(1):97–139, 2008. A preliminary version appeared in *EUROCRYPT 2004*, 2004.

GPV08. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Cynthia Dwork, editor, *STOC 2008*, pages 197–206. ACM, 2008. see also `http://eprint.iacr.org/2007/432`.

GKV10. S. Dov Gordon, Jonathan Katz, and Vinod Vaikuntanathan. A group signature scheme from lattice assumptions. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 395–412. Springer, Heidelberg, 2010.

Ham11. Michael Hamburg. *Spatial Encryption*. PhD thesis, Stanford University, 2011. Available at `http://eprint.iacr.org/2011/389`.

IP08. Vincenzo Iovino and Giuseppe Persiano. Hidden-vector encryption with groups of prime order. In Steven D. Galbraith and Kenneth G. Paterson, editors, *Pairing 2008*, volume 5209 of *LNCS*, pages 75–88. Springer, Heidelberg, 2008.

KSW08. Jonathan Katz, Amit Sahai, and Brent Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 146–162. Springer, Heidelberg, 2008. The full version is available at `http://eprint.iacr.org/2007/404`.

LS12. Adeline Langlois and Damien Stehlé. Hardness of decision (R)LWE for any modulus. Cryptology ePrint Archive, Report 2012/091, 2012. Available at http://eprint.iacr.org/2012/091.

LOS$^+$10. Allison Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, and Brent Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 62–91. Springer, Heidelberg, 2010. The full version is available at http://eprint.iacr.org/2010/110.

LPRTJ05. Alexander Litvak, Alain Pajor, Mark Rudelson, and Nicole Tomczak-Jaegermann. Smallest singular value of random matrices and geometry of random polytopes. *Advances in Mathematics*, 195(2):491–523, 2005.

LPR10. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23. Springer, Heidelberg, 2010.

Mic07. Daniele Micciancio. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions. *Computational Complexity*, 16:365–411, 2007. A preliminary version appeared in *FOCS 2002*, 2002. See also ECCC TR04-095.

MP12a. Daniele Micciancio and Chris Peikert. Private communication, December 2012. 2012-12-10.

MP12b. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 700–718. Springer, Heidelberg, 2012. Available at http://eprint.iacr.org/2011/501.

MR07. Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on Gaussian measures. *SIAM Journal on Computing*, 37(1):267–302, 2007. A preliminary version appeared in *FOCS 2004*, 2004.

OT09. Tatsuaki Okamoto and Katsuyuki Takashima. Hierarchical predicate encryption for inner-products. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 214–231. Springer, Heidelberg, 2009.

OT10. Tatsuaki Okamoto and Katsuyuki Takashima. Fully secure functional encryption with general relations from the decisional linear assumption. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 191–208. Springer, Heidelberg, 2010. The full version is available at http://eprint.iacr.org/2010/563.

OT11. Tatsuaki Okamoto and Katsuyuki Takashima. Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption. In Dongdai Lin, Gene Tsudik, and Xiaoyun Wang, editors, *CANS 2011*, volume 7092 of *LNCS*, pages 138–159. Springer, Heidelberg, 2011. The full version is available at http://eprint.iacr.org/2010/648.

OT12. Tatsuaki Okamoto and Katsuyuki Takashima. Adaptively attribute-hiding (hierarchical) inner product encryption. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 591–608. Springer, Heidelberg, 2012. The full version is available at http://eprint.iacr.org/2010/543.

Par11. Jong Hwan Park. Inner-product encryption under standard assumptions. *Designs, Codes and Cryptography*, 58(3):235–257, March 2011.

Pei09. Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In Michael Mitzenmacher, editor, *STOC 2009*, pages 333–342. ACM, 2009.

Pei10. Chris Peikert. An efficient and parallel gaussian sampler for lattices. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 80–97. Springer, Heidelberg, 2010.

PW08. Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In Cynthia Dwork, editor, *STOC 2008*, pages 187–196. ACM, 2008.

Reg09. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*, 56(6):Article 34, 2009. A preliminary version appeared in *STOC 2005*, 2005.

SBC$^+$07. Elaine Shi, John Bethencourt, Hubert Chan, Dawn Song, and Adrian Perrig. Multi-dimensional range query over encrypted data. In *IEEE S&P 2007*, pages 350–367. IEEE Computer Society, 2007.

Ste12. Damien Stehlé. Private communication, December 2012. 2012-12-12.

Ver10. Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices, 2010. Available at http://arxiv.org/abs/1011.3027.

Wat05. Brent Waters. Efficient identity-based encryption without random oracles. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 114–127. Springer, Heidelberg, 2005. The full version is available at http://eprint.iacr.org/2004/180.

YNKF11. Reo Yoshida, Akira Nagai, Tetsutaro Kobayashi, and Hitoshi Fuji. An algorithm for inner-product batch searchable encryption. SCIS 2011 4C2-4 (in Japanese), 2011.

A Preliminaries for Proofs

We review the tools in the lattice-based cryptography.

A.1 Distributions

We use the following statistical properties.

Lemma A.1 ([GPV08]). *Let κ be the security parameter. Let z be an arbitrary vector in $\mathbb{Z}^m$. Let $\alpha \in (0,1)$ and let q be a positive integer. Let $\boldsymbol{e}$ be a random variable according to distribution $\bar{\Psi}_\alpha^m$. With overwhelming probability, we have*

$$|\langle \boldsymbol{e}, z\rangle| \leq \|z\|(q\alpha \cdot \omega(\sqrt{\lg \kappa}) + \sqrt{m}/2) \text{ and } \|\boldsymbol{e}\| \leq \sqrt{m}(q\alpha \cdot \omega(\sqrt{\lg \kappa}) + 1/2),$$

where $\langle \boldsymbol{e}, z\rangle$ is treated as integer in $[-(q-1)/2, (q-1)/2]$.

In addition, a sample from the distribution $\bar{\Psi}_\alpha$ has an absolute value at most $q\alpha \cdot \omega(\sqrt{\lg \kappa}) + 1/2$ with overwhelming probability.

We recall an adapted version for lattice of the *generalized* leftover hash lemma [DORS08].

Lemma A.2 (Adapted version of [ABB10]). *Let q be an odd prime. Suppose that $m = (n+t)\lg q + \omega(\lg \kappa)$. Let $\boldsymbol{R}$ be an $m \times \bar{m}$ matrix whose columns are chosen from a distribution of min-entropy at least m. Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be matrices chosen uniformly in $\mathbb{Z}_q^{n\times m}$ and $\mathbb{Z}_q^{n\times \bar{m}}$. Then, for any positive integer t and all matrix $\boldsymbol{W} \in \mathbb{Z}_q^{m\times t}$, we have that*

$$\Delta((\boldsymbol{A}, \boldsymbol{A}\boldsymbol{R}, \boldsymbol{R}^\top \boldsymbol{W}), (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{R}^\top \boldsymbol{W})) \leq \frac{\bar{m}}{2} \cdot 2^{-\frac{1}{2}(m-(n+t)\lg q)} = 2^{-\omega(\lg \kappa)}.$$

A.2 The Matrix Norm of Random Matrix

We review the bounds on $s_1(\boldsymbol{R})$ for random matrix $\boldsymbol{R}$ from the discrete Gaussian. We borrow the following lemmas whose source are summarized in Vershynin's monograph [Ver10]:

Lemma A.3 ([ABB10,LPRTJ05]). *Let $\boldsymbol{R} \leftarrow \{-1,+1\}^{n\times m}$. Then, there exists a universal constant C that, with overwhelming probability of n, $s_1(\boldsymbol{R}) \leq C(\sqrt{n} + \sqrt{m})$.*

Lemma A.4 (Adapted version of [MP12b, Lemmas 2.8 and 2.9]). *Let $\Lambda \subset \mathbb{R}^n$ be a lattice and $r \geq \eta_\epsilon(\Lambda)$ for some $\epsilon \in (0,1)$, where η_ϵ denotes the smoothing parameter [MR07]. Let $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_k \in$* span$(\Lambda)$. *Let $\boldsymbol{x}_i \leftarrow D_{\Lambda+\boldsymbol{c}_i, r}$. Then, there exists a universal constant $C > 0$ such that for any $t \geq 0$, $s_1([\boldsymbol{x}_1 \mid \cdots \mid \boldsymbol{x}_k]) \leq C \cdot r \cdot (\sqrt{n} + \sqrt{k} + t)$ except with probability at most $2\frac{1+\epsilon}{1-\epsilon}\exp(-\pi t^2)$.*

In particular, the following holds: Let n and q be positive integers with q prime, and let $m \geq 2n \lg q$. For all but negligible fraction of $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and arbitrary matrix $\boldsymbol{U} = [\boldsymbol{u}_1 \mid \cdots \mid \boldsymbol{u}_k] \in \mathbb{Z}_q^{n\times k}$, the followings hold: Let $\boldsymbol{r}_i \leftarrow D_{\Lambda_q^{\boldsymbol{u}_i}(\boldsymbol{A}), r}$ and set $\boldsymbol{R} = [\boldsymbol{r}_1 \mid \cdots \mid \boldsymbol{r}_k]$ (we have $\boldsymbol{A}\boldsymbol{R} = \boldsymbol{U}$). Then, there exists a universal constant $C > 0$ that, with overwhelming probability of n, $s_1(\boldsymbol{R}) \leq Cr(\sqrt{m} + \sqrt{k})$.

In the following, we use the upper bound, C, without explanation.

B Proofs

B.1 Correctness Proof

Proof. Since $s_1(\boldsymbol{R}_{\boldsymbol{A}}) \leq C\sqrt{2m}$ from the property of GenTrap (Theorem 2.2), we have $s > 4Cm \cdot \omega(\sqrt{\lg n}) > \sqrt{s_1(\boldsymbol{R})^2 + 1} \cdot \sqrt{7} \cdot \omega(\sqrt{\lg n})$. Hence, the master can use SampleD correctly.

In decryption, $\boldsymbol{c}_{\vec{v}}$ is computed as $\boldsymbol{c}_{\vec{v}} = \sum_{i=1}^{\mu} H_{\boldsymbol{g}}(\mathfrak{v}_i)^\top \cdot \boldsymbol{c}_i$. Expanding it, we have that

$$\begin{aligned}
\boldsymbol{c}_{\vec{v}} &= \sum_{i=1}^{\mu} H_{\boldsymbol{g}}(\mathfrak{v}_i)^\top \cdot [(\boldsymbol{B}_i + H(\mathfrak{w}_i) \cdot \boldsymbol{G})^\top \boldsymbol{s} + \boldsymbol{R}_i^\top \boldsymbol{x}_0] \\
&= \left(\sum_{i=1}^{\mu} \boldsymbol{B}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)\right)^\top \boldsymbol{s} + \left(\sum_{i=1}^{\mu} H(\mathfrak{w}_i) \cdot \boldsymbol{G} \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)\right)^\top \boldsymbol{s} + \sum_{i=1}^{\mu} (\boldsymbol{R}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i))^\top \boldsymbol{x}_0 \\
&= \boldsymbol{B}_{\vec{v}}^\top \boldsymbol{s} + \left(\sum_{i=1}^{\mu} H(\mathfrak{w}_i) \cdot H(\mathfrak{v}_i) \cdot \boldsymbol{G}\right)^\top \boldsymbol{s} + \sum_{i=1}^{\mu} (\boldsymbol{R}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i))^\top \boldsymbol{x}_0 \\
&= \boldsymbol{B}_{\vec{v}}^\top \boldsymbol{s} + \left(H(\vec{w}^\top \vec{v}) \cdot \boldsymbol{G}\right)^\top \boldsymbol{s} + \boldsymbol{R}_{\vec{v}}^\top \boldsymbol{x}_0,
\end{aligned}$$

where we set $\boldsymbol{R}_{\vec{v}} = \sum_{i=1}^{\mu} \boldsymbol{R}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)$. If $\vec{w}^\top \vec{v} = 0$ then, the middle term vanishes and we have

$$\boldsymbol{c}_{\vec{v}} = \boldsymbol{B}_{\vec{v}}^\top \boldsymbol{s} + \boldsymbol{R}_{\vec{v}}^\top \boldsymbol{x}_0 \text{ and } \boldsymbol{c} = [\boldsymbol{c}_0; \boldsymbol{c}_{\vec{v}}] = \boldsymbol{A}_{\vec{v}}^\top \boldsymbol{s} + [\boldsymbol{x}_0; \boldsymbol{R}_{\vec{v}}^\top \boldsymbol{x}_0].$$

Therefore, in decryption, we have that

$$\begin{aligned} d &= c' - \boldsymbol{e}_{\vec{v}}^\top \cdot \boldsymbol{c} \\ &= M\lfloor q/2 \rfloor + \boldsymbol{u}^\top \boldsymbol{s} + x' - \boldsymbol{e}_{\vec{v}}^\top \cdot \left(\boldsymbol{A}_{\vec{v}}^\top \boldsymbol{s} + \left[\boldsymbol{x}_0; \boldsymbol{R}_{\vec{v}}^\top \boldsymbol{x}_0\right]\right) \\ &= M\lfloor q/2 \rfloor + x' - \boldsymbol{e}_{\vec{v}}^\top \left[\boldsymbol{x}_0; \boldsymbol{R}_{\vec{v}}^\top \boldsymbol{x}_0\right] && \text{from } \boldsymbol{A}_{\vec{v}} \cdot \boldsymbol{e}_{\vec{v}} = \boldsymbol{u} \end{aligned}$$

In order to simplify notation, we set $\boldsymbol{e}_{\vec{v}} = [\boldsymbol{e}_1; \boldsymbol{e}_2] \in \mathbb{Z}^m \times \mathbb{Z}^{nk}$. Then, we have that

$$d = M\lfloor q/2 \rfloor + x' - (\boldsymbol{e}_1 + \boldsymbol{R}_{\vec{v}} \boldsymbol{e}_2)^\top \boldsymbol{x}_0.$$

Hence, it suffices to show that, with overwhelming probability,

$$|x'| + \left|(\boldsymbol{e}_1 + \boldsymbol{R}_{\vec{v}} \cdot \boldsymbol{e}_2)^\top \cdot \boldsymbol{x}_0\right| \le q/5.$$

Let $\boldsymbol{e} = \boldsymbol{e}_1 + \boldsymbol{R}_{\vec{v}} \cdot \boldsymbol{e}_2 = \boldsymbol{e}_1 + \left(\sum_{i=1}^{\mu} \boldsymbol{R}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)\right) \cdot \boldsymbol{e}_2 \in \mathbb{Z}^m$.

We here recall the norm bounds. We have that, with overwhelming probability, $\|\boldsymbol{e}_1\|, \|\boldsymbol{e}_2\| \le \|\boldsymbol{e}_{\vec{v}}\| \le s\sqrt{m+nk} \le s\sqrt{2m}$ from Lemma A.4, $s_1(\boldsymbol{R}_i) \le C(\sqrt{nk} + \sqrt{m}) \le 2C\sqrt{m}$ from Lemma A.3, and $s_1(H_{\boldsymbol{g}}(\mathfrak{v}_i)) \le nk \le m$ from Lemma 4.2. Therefore, we have that $s_1(\boldsymbol{R}_{\vec{v}}) = s_1(\sum_{i=1}^{\mu} \boldsymbol{R}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)) \le \sum_{i=1}^{\mu} s_1(\boldsymbol{R}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)) \le 2\mu C m^{3/2}$ with overwhelming probability and

$$\|\boldsymbol{e}\| = \|\boldsymbol{e}_1 + \boldsymbol{R}_{\vec{v}} \cdot \boldsymbol{e}_2\| \le (1 + 2\mu C m^{3/2}) \cdot s\sqrt{2m}.$$

From Lemma A.1 and the hypothesis of the theorem, we obtain the bound

$$\begin{aligned} |x'| + \left|\boldsymbol{e}^\top \boldsymbol{x}_0\right| &\le \alpha q \omega(\sqrt{\lg \kappa}) + 1/2 + (1 + 2\mu C m^{3/2}) \cdot s\sqrt{2m} \cdot (\alpha q \cdot \omega(\sqrt{\lg \kappa}) + \sqrt{m}/2) \\ &\le (\alpha q \cdot \omega(\sqrt{\lg \kappa}) + \sqrt{m}/2) \cdot 4C\mu s m^2 < q/5 \end{aligned}$$

with overwhelming probability. □

B.2 Security Proof

The security proof mainly follows the proof by Agrawal et al. [AFV11] with combination of that in Micciancio and Peikert [MP12b]. We include the security proof for completeness, since we change several parameters. We only show the security in the case $\ell = 1$ since extending the proof for general ℓ is easy.

We define the following algorithms for simulation in the proof.

$\overline{\mathsf{Setup}}(1^\kappa, n, q, m, s, \alpha, \mathfrak{g}, k, \vec{w}^*)$**:** On input the parameters and an additional parameter $\vec{w}^* = (\mathfrak{w}_1, \dots, \mathfrak{w}_\mu)^\top \in \mathrm{GF}(q^n)^\mu$, which is the one of declared ciphertext attributes,

1. Choose a random matrix $\boldsymbol{A} \leftarrow \mathbb{Z}_q^{n \times m}$ and a random vector $\boldsymbol{u} \leftarrow \mathbb{Z}_q^n$.
2. Set $(\boldsymbol{G}, \boldsymbol{S})$ as Theorem 2.1.
3. For $i \in [\mu]$; Choose a random matrix $\boldsymbol{R}_i \leftarrow \{-1, +1\}^{m \times nk}$ and set $\boldsymbol{B}_i \leftarrow \boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}$.

Output $pp = (\boldsymbol{A}, \{\boldsymbol{B}_i\}, \boldsymbol{u})$ and $\overline{msk} = (\boldsymbol{S}, \{\boldsymbol{R}_i\}, pp)$.

$\overline{\mathsf{Extract}}(pp, \overline{msk}, \vec{v})$**:** On input $pp, \overline{msk} = (\boldsymbol{S}, \{\boldsymbol{R}_i\}, pp)$, and predicate vector $\vec{v} = (\mathfrak{v}_1, \dots, \mathfrak{v}_\mu) \in \mathrm{GF}(q^n)^\mu$ it outputs $dk_{\vec{v}}$ if $\vec{w}^\top \vec{v} \ne 0$:

1. Define the matrices $\boldsymbol{B}_{\vec{v}} = \sum_{i=1}^{\mu} \boldsymbol{A}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i) \in \mathbb{Z}_q^{n\times nk}$ and $\boldsymbol{A}_{\vec{v}} = [\boldsymbol{A} \mid \boldsymbol{B}_{\vec{v}}] \in \mathbb{Z}_q^{n\times(m+nk)}$. We note that
$$\boldsymbol{A}_{\vec{v}} = \left[\boldsymbol{A} \,\middle|\, \boldsymbol{A}\left(\sum_{i=1}^{\mu} \boldsymbol{R}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)\right) - \left(\sum_{i=1}^{\mu} H(\mathfrak{w}_i)\boldsymbol{G} \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)\right)\right] = \left[\boldsymbol{A} \mid \boldsymbol{A}\boldsymbol{R}_{\vec{v}} - H(\vec{w}^{\top}\vec{v})\boldsymbol{G}\right],$$
where we set $\boldsymbol{R}_{\vec{v}} = \sum_{i=1}^{\mu} \boldsymbol{R}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_i)$. We remark that $-\boldsymbol{R}_{\vec{v}}$ is a trapdoor for $\boldsymbol{A}_{\vec{v}}$ with tag $-H(\vec{w}^{\top}\vec{v})$.
2. Using the simulated master secret key, $\boldsymbol{S}$ and $-\boldsymbol{R}_{\vec{v}}$, sample $\boldsymbol{e}_{\vec{v}} \leftarrow \mathsf{SampleD}(\boldsymbol{A}, -H(\vec{w}^{\top}\vec{v}), \boldsymbol{G}, -\boldsymbol{R}_{\vec{v}}, \boldsymbol{S}, \boldsymbol{u}, s)$.

Output $dk_{\vec{v}} = \boldsymbol{e}_{\vec{v}}$.

$\overline{\mathsf{Enc}}(pp, \vec{w}^*, M, \overline{msk})$**:** On input pp, the one of declared attributes $\vec{w}^* = (\mathfrak{w}_1, \ldots, \mathfrak{w}_{\mu}) \in \mathrm{GF}(q^n)^{\mu}$, and a message $M \in \{0,1\}$:

1. Choose a random vector $\boldsymbol{s} \leftarrow \mathbb{Z}_q^n$.
2. Choose noises $\boldsymbol{x} \leftarrow \chi^m$, and $x \leftarrow \chi$.
3. Set $\boldsymbol{c}_0 \leftarrow \boldsymbol{A}^{\top}\boldsymbol{s} + \boldsymbol{x}$
4. Set $c' \leftarrow \boldsymbol{u}^{\top}\boldsymbol{s} + x' + M\lfloor q/2 \rfloor$.
5. For $i = 1, \ldots, \mu$; compute $\boldsymbol{c}_i \leftarrow (\boldsymbol{B}_i + H(\mathfrak{w}_i)\boldsymbol{G})^{\top}\boldsymbol{s} + \boldsymbol{R}_i^{\top}\boldsymbol{x} \in \mathbb{Z}_q^{nk}$. Note that $\boldsymbol{c}_i = \boldsymbol{R}_i^{\top}\boldsymbol{c}_0$ because we set $\boldsymbol{B}_i = \boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i)\boldsymbol{G}$.

Output $ct = (\boldsymbol{c}_0, \{\boldsymbol{c}_i\}, c')$ as a ciphertext.

In order to prove the security, we prepare the following six games and show that they are indistinguishable.

Real_0**:** Setup, $\mathsf{Extract}$, Enc, $\vec{w}_0$, and M_0.
Sim_0**:** $\overline{\mathsf{Setup}}$, $\overline{\mathsf{Extract}}$, $\overline{\mathsf{Enc}}$, $\vec{w}_0$, and M_0.
Rand_0**:** $\overline{\mathsf{Setup}}$, $\overline{\mathsf{Extract}}$, $\overline{\mathsf{Enc}}$, $\vec{w}_0$, and a random ciphertext.
Rand_1**:** $\overline{\mathsf{Setup}}$, $\overline{\mathsf{Extract}}$, $\overline{\mathsf{Enc}}$, $\vec{w}_1$, and a random ciphertext.
Sim_1**:** $\overline{\mathsf{Setup}}$, $\overline{\mathsf{Extract}}$, $\overline{\mathsf{Enc}}$, $\vec{w}_1$, and M_1.
Real_1**:** Setup, $\mathsf{Extract}$, Enc, $\vec{w}_1$, and M_1.

Lemma B.1. *Suppose that $m = 2n\lg q + \omega(\lg\kappa)$ and $s \geq 3C\mu m^{1.5} \cdot \omega(\sqrt{\lg(n)})$. For any $\beta \in \{0,1\}$, two games Real_{β} and Sim_{β} are statistically indistinguishable.*

Proof. From the hypothesis, we have that $s > 4Cm \cdot \omega(\sqrt{\lg n}) > \sqrt{s_1(\boldsymbol{R}_{\boldsymbol{A}})^2 + 1} \cdot \sqrt{7} \cdot \omega(\sqrt{\lg n})$ in the game Real.

In the simulation, we have that $s_1(\boldsymbol{R}_{\vec{v}}) \leq \sum_{i=1}^{\mu} s_1(\boldsymbol{R}) \cdot s_1(H_{\boldsymbol{g}}(\mathfrak{v}_i)) \leq \mu \cdot C(\sqrt{m} + \sqrt{nk}) \cdot nk \leq 2C\mu m^{1.5}$ with overwhelming probability (as in the analysis of correctness above). Hence, it is hold that $s > \sqrt{s_1(\boldsymbol{R}_{\vec{v}})^2 + 1} \cdot \sqrt{7} \cdot \omega(\sqrt{\lg n})$ in the game Sim with overwhelming probability.

If they hold, in both game the decryption key $\boldsymbol{e}_{\vec{v}}$ is chosen from $D_{\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A}_{\vec{v}}),s}$ from the property of $\mathsf{SampleD}$ (Definition 2.2). Hence, $\boldsymbol{A}$ and $\boldsymbol{e}_{\vec{v}}$ in Real_{β} and Sim_{β} are close within negligible distance.

We next consider, the distributions of $\boldsymbol{B}_i$ and target ciphertext $\boldsymbol{c}_i$. We have that
$$(\boldsymbol{A}, \{\boldsymbol{B}_i, \boldsymbol{c}_i\}) = \begin{cases} (\boldsymbol{A}, \{\boldsymbol{B}_i, (\boldsymbol{B}_i^{\top} + H(\mathfrak{w}_i)\boldsymbol{G})^{\top}\boldsymbol{s} + \boldsymbol{R}_i^{T}\boldsymbol{x}_0\}) & \text{in } \mathsf{Real}_{\beta} \\ (\boldsymbol{A}, \{\boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}, \boldsymbol{R}_i^{T}(\boldsymbol{A}^{T}\boldsymbol{s} + \boldsymbol{x}_0)\}) & \text{in } \mathsf{Sim}_{\beta} \end{cases}$$

From the generalized leftover hash lemma (Lemma A.2), we have that, for any $\boldsymbol{w}$, $(\boldsymbol{A}, \boldsymbol{B}_i, \boldsymbol{R}_i^{\top}\boldsymbol{w})$ and $(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{R}_i, \boldsymbol{R}_i^{\top}\boldsymbol{w})$ are statistically close. Since $\boldsymbol{B}_i$ is uniformly distributed in the game Real, we also have that for any fixed matrix $\boldsymbol{G}$ and $\vec{w}$, $(\boldsymbol{A}, \boldsymbol{B}_i, \boldsymbol{R}_i^{\top}\boldsymbol{w})$ and $(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}, \boldsymbol{R}_i^{\top}\boldsymbol{w})$ are also statistically close. Since the matrices $\boldsymbol{R}_i$ are chosen independently for every i, we obtain that
$$(\boldsymbol{A}, \{\boldsymbol{B}_i, \boldsymbol{R}_i^{\top}\boldsymbol{x}_0\}_{i\in[\mu]}) \approx_s (\boldsymbol{A}, \{\boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}, \boldsymbol{R}_i^{\top}\boldsymbol{x}_0\}_{i\in[\mu]}).$$

Adding $(\boldsymbol{B}_i + H(\mathfrak{w}_i) \cdot \boldsymbol{G})^\top \boldsymbol{s}$ to them, we have that

$$
\begin{aligned}
&(\boldsymbol{A}, \{\boldsymbol{B}_i, (\boldsymbol{B}_i + H(\mathfrak{w}_i) \cdot \boldsymbol{G})^\top \boldsymbol{s} + \boldsymbol{R}_i^\top \boldsymbol{x}_0\}_{i \in [\mu]}) \\
&\quad \approx_s (\boldsymbol{A}, \{\boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}, (\boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G} + H(\mathfrak{w}_i) \cdot \boldsymbol{G})^\top \boldsymbol{s} + \boldsymbol{R}_i^\top \boldsymbol{x}_0\}_{i \in [\mu]}) \\
&\quad = (\boldsymbol{A}, \{\boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}, (\boldsymbol{A}\boldsymbol{R}_i)^\top \boldsymbol{s} + \boldsymbol{R}_i^\top \boldsymbol{x}_0\}_{i \in [\mu]}).
\end{aligned}
$$

This shows that in both game the joint distributions of $(\boldsymbol{A}, \{\boldsymbol{B}_i, \boldsymbol{c}_i\}_{i \in [\mu]})$ are statistically close. □

Lemma B.2. *Suppose that $m \geq (n+1)\lg q + \omega(\lg \kappa)$. For any $\beta \in \{0,1\}$, two games Sim_β and Rand_β are computationally indistinguishable under the LWE assumption.*

Proof. We construct a LWE solver $\mathcal{B}$ using $\mathcal{A}$ who distinguishes Sim_β from Rand_β.

Given the LWE challenges $(\boldsymbol{a}_i, y_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ for $i = 0, \ldots, m$, define the matrices and vectors as

$$
\begin{aligned}
\boldsymbol{A} &= [\boldsymbol{a}_1 \ldots \boldsymbol{a}_m] \in \mathbb{Z}_q^{n \times m}, & \boldsymbol{u} &= \boldsymbol{a}_0, \\
\boldsymbol{c}_0 &= (y_1, \ldots, y_m)^T \in \mathbb{Z}_q^m, & c &= y_0 + M \lfloor q/2 \rfloor .
\end{aligned}
$$

$\mathcal{B}$ simulates the games as follows: For setup, it runs $\overline{\mathsf{Setup}}$ with $\vec{w} = \vec{w}_\beta$ and use $\boldsymbol{A}$ and $\boldsymbol{u}$ in the above. For private-key queries, it runs $\overline{\mathsf{Extract}}$. For challenge query, $\mathcal{B}$ sets $\boldsymbol{c}_i = \boldsymbol{R}_i^\top \boldsymbol{c}_0$, where $\{\boldsymbol{R}_i\}$ is a component of *msk*, and returns $(\boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_\mu, c)$.

If $\mathcal{B}$'s oracle is $A_{\boldsymbol{s},\chi}$, we have $\boldsymbol{c}_0 = \boldsymbol{A}^\top \boldsymbol{s} + \boldsymbol{x}_0$ and $c' = \boldsymbol{u}^\top \boldsymbol{s} + x'$ where $\boldsymbol{x}_0 \leftarrow \chi^m$ and $x' \leftarrow \chi$. Therefore, $\mathcal{B}$ perfectly simulates Sim_β.

If $\mathcal{B}$'s oracle is $U(\mathbb{Z}_q^n \times \mathbb{Z}_q)$, $\boldsymbol{A}$ and $\boldsymbol{c}_0$ are chosen uniformly at random. Therefore, $(\boldsymbol{A}, \boldsymbol{c}_0, \boldsymbol{A}\boldsymbol{R}_i, \boldsymbol{R}_i^\top \boldsymbol{c}_0)$ is statistically close to the uniform over $\mathbb{Z}_q^{(n+1) \times m} \times \mathbb{Z}_q^{(n+1) \times nk}$ from Lemma A.2 with parameters $m \geq (n+1)\lg q + \omega(\lg n)$ (we do not need the generalized version). Hence, $\mathcal{B}$'s output, i.e., $(\boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_\mu, c)$, is randomly distributed over $\mathbb{Z}_q^{m+nk\mu+1}$. This shows that $\mathcal{B}$ simulates Rand_β all but negligible probability. □

Lemma B.3. *Suppose that $m \geq n \lg q + \omega(\lg \kappa)$. Then two games Rand_0 and Rand_1 are statistically indistinguishable.*

Proof. In the two games, $\vec{w}$ only appears at public parameters $\{\boldsymbol{B}_i\}_{i \in [\mu]} = \{\boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}\}_{i \in [\mu]}$. From the leftover hash lemma (Lemma A.2), we have

$$
(\boldsymbol{A}, \boldsymbol{B}_1, \ldots, \boldsymbol{B}_\mu) \approx_s (\boldsymbol{A}, \boldsymbol{A}\boldsymbol{R}_1, \ldots, \boldsymbol{A}\boldsymbol{R}_\mu)
$$

where $\boldsymbol{A} \leftarrow \mathbb{Z}_q^{n \times m}$, $\boldsymbol{B}_i \leftarrow \mathbb{Z}_q^{n \times nk}$, and $\boldsymbol{R}_i \leftarrow \{-1,+1\}^{m \times nk}$ for $i \in [\mu]$. Since the left hand side is uniformly at random, we have

$$
(\boldsymbol{A}, \{\boldsymbol{B}_i\}_{i \in [\mu]}) \approx_s (\boldsymbol{A}, \{\boldsymbol{A}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}\}_{i \in [\mu]}) \approx_s (\boldsymbol{A}, \{\boldsymbol{A}\boldsymbol{R}_i - H(\mathfrak{w}_i) \cdot \boldsymbol{G}\}_{i \in [\mu]})
$$

for any fixed vector $\vec{w} \in \mathrm{GF}(q^n)^\mu$ and matrix $\boldsymbol{G}$. This hides $\vec{w}_0$ and $\vec{w}_1$ statistically. □

C Hierarchical Inner-Product Encryption

We follow the definition in Okamoto and Takashima [OT09]. We call a tuple of positive integers $\vec{\mu} = (d; \mu_1, \ldots, \mu_d)$ a format of hierarchy of depth d attributes spaces. For $i = 1, \ldots, d$, let Σ_i and Φ_i be the sets of attributes, where $\Sigma_i = \mathrm{GF}(q^n)^{\mu_i} \setminus \{\vec{0}\}$ and $\Phi_i = \mathrm{GF}(q^n)^{\mu_i}$. We additionally define $\Phi_0 = \{\top\}$. Let $\Sigma = \bigcup_{i=1}^d (\Sigma_1 \times \cdots \times \Sigma_i)$ and $\Phi = \bigcup_{i=1}^d (\Phi_1 \times \cdots \times \Phi_i)$. We define the relation as follows: For $\vec{V} = (\vec{v}_1, \ldots, \vec{v}_j)$ and $\vec{W} = (\vec{w}_1, \ldots, \vec{w}_h)$,

$$
\mathcal{P}^{\mathsf{HIPE}}(\vec{V}, \vec{W}) = \begin{cases} 1 & \text{if } j \leq h \text{ and for all } 1 \leq i \leq j,\ \vec{v}_i \cdot \vec{w}_i = 0 \\ 0 & \text{otherwise} \end{cases} .
$$

Definition C.1. *Let $\vec{\mu} = (d; \mu_1, \dots, \mu_d)$ be a format of hierarchy of depth d attribute and predicate spaces. A hierarchical inner-product encryption (HIPE) scheme for $\mathcal{P}^{\mathsf{HIPE}}$ is a tuple of algorithms.*

$\mathsf{Setup}(1^\kappa, \vec{\mu}) \rightarrow (pp, msk)$**:** *The setup algorithm takes as input security parameter 1^κ and format of hierarchy $\vec{\mu}$, and outputs public parameters pp and master secret key $msk = dk_\top$.*

$\mathsf{Extract}(msk, \vec{V}) \rightarrow dk_{\vec{V}}$**:** *The extraction algorithm takes as input msk and key attribute $\vec{V} = (\vec{v}_1, \dots, \vec{v}_j) \in \Phi$, where $j \in [d]$. It outputs decryption key $dk_{\vec{V}}$.*

$\mathsf{Enc}(pp, \vec{W}, M) \rightarrow ct$**:** *The encryption algorithm takes as input pp, ciphertext attribute $\vec{W} = (\vec{w}_1, \dots, \vec{w}_h) \in \Sigma$, where $h \in [d]$, and message $M \in \mathcal{M}$. It outputs ciphertext ct.*

$\mathsf{Dec}(pp, dk_{\vec{V}}, ct) \rightarrow M$ **or** $\perp$**:** *The decryption algorithm takes as input decryption key $dk_{\vec{V}}$ and ciphertext ct. It outputs either $M \in \mathcal{M}$ or rejection symbol $\perp$.*

$\mathsf{Delg}_j(pp, dk_{\vec{V}}, \vec{v}_{j+1},) \rightarrow dk_{\vec{V}'}$**:** *The decryption algorithm takes as input decryption key $dk_{\vec{V}}$, where $\vec{V} = (\vec{v}_1, \dots, \vec{v}_j)$, and $(j+1)$-th level predicate $\vec{v}_{j+1}$. It outputs $dk_{\vec{V}'}$, where $\vec{V}' = (\vec{v}_1, \dots, \vec{v}_{j+1})$.*

Correctness: We define slightly weak correctness for decryption. For any $\vec{V} = (\vec{v}_1, \dots, \vec{v}_h) \in \Phi$, $\vec{W} \in \Sigma$, and $M \in \mathcal{M}$, if $\mathcal{P}(\vec{V}, \vec{W}) = 1$ then

$$\Pr\left[M = \tilde{M} : \begin{array}{l} (pp, msk) \leftarrow \mathsf{Setup}(1^\kappa, \vec{\mu}); dk_{\vec{V}} \leftarrow \mathsf{Extract}(msk, \vec{V}); \\ ct \leftarrow \mathsf{Enc}(pp, \vec{W}, M); \tilde{M} \leftarrow \mathsf{Dec}(pp, dk_{\vec{V}}, ct); \end{array} \right]$$

$$\text{and } \Pr\left[M = \tilde{M} : \begin{array}{l} (pp, msk) \leftarrow \mathsf{Setup}(1^\kappa, \vec{\mu}); \\ dk_{(\vec{v}_1, \dots, \vec{v}_{j+1})} \leftarrow \mathsf{Delg}_j(pp, dk_{(\vec{v}_1, \dots, \vec{v}_j)}, \vec{v}_{j+1}) \text{ for } j \in [0, h-1]; \\ ct \leftarrow \mathsf{Enc}(pp, \vec{W}, M); \tilde{M} \leftarrow \mathsf{Dec}(pp, dk_{\vec{V}}, ct); \end{array} \right]$$

is overwhelming probability and if $\mathcal{P}(\vec{V}, \vec{W}) = 0$ then

$$\Pr\left[\tilde{M} = \perp : \begin{array}{l} (pp, msk) \leftarrow \mathsf{Setup}(1^\kappa); dk_{\vec{V}} \leftarrow \mathsf{Extract}(msk, \vec{V}); \\ ct \leftarrow \mathsf{Enc}(pp, \sigma, M); \tilde{M} \leftarrow \mathsf{Dec}(pp, dk_{\vec{V}}, ct); \end{array} \right]$$

$$\text{and } \Pr\left[\tilde{M} = \perp : \begin{array}{l} (pp, msk) \leftarrow \mathsf{Setup}(1^\kappa, \vec{\mu}); \\ dk_{(\vec{v}_1, \dots, \vec{v}_{j+1})} \leftarrow \mathsf{Delg}_j(pp, dk_{(\vec{v}_1, \dots, \vec{v}_j)}, \vec{v}_{j+1}) \text{ for } j \in [0, h-1]; \\ ct \leftarrow \mathsf{Enc}(pp, \vec{W}, M); \tilde{M} \leftarrow \mathsf{Dec}(pp, dk_{\vec{V}}, ct); \end{array} \right]$$

is overwhelming probability.

Security: We consider weakly attribute hiding in selective attribute setting against chosen-plaintext attacks (wAH-sA-CPA).

Definition C.2 (wAH-sA-CPA security). *Let* HIPE *be a HIPE scheme, $\mathcal{A}$ an adversary, and κ a security parameter. The experiment between a challenger and adversary $\mathcal{A}$, $\mathsf{Expt}^{\mathsf{wah\text{-}sa\text{-}cpa}}_{\mathcal{A},\mathsf{HIPE}}(1^\kappa)$, is defined as follows:*

Initialization: *Given security parameter 1^κ and format of hierarchy $\vec{\mu} = (d; \mu_1, \dots, \mu_d)$, run adversary $\mathcal{A}$ with 1^κ and $\vec{\mu}$. Receive depth h and two ciphertext attributes $\vec{W}_0, \vec{W}_1 \in \Sigma_1 \times \cdots \times \Sigma_h$ from $\mathcal{A}$. Run $(pp, msk) \leftarrow \mathsf{Setup}(1^\kappa, \vec{\mu})$. Flip a coin $\beta \leftarrow \{0, 1\}$.*

Learning Phase: *Feed pp to adversary $\mathcal{A}$. Adversary $\mathcal{A}$ could issue queries to the following oracles in any order and many times except for the constraint regarding oracle* Challenge.

- *Oracle* Extract *receives key attribute $\vec{V} \in \Phi$ subject to the restriction that $\mathcal{P}^{\mathsf{HIPE}}(\vec{V}, \vec{W}_0) = \mathcal{P}^{\mathsf{HIPE}}(\vec{V}, \vec{W}_1) = 0$. Otherwise, it obtains $dk_{\vec{V}} \leftarrow \mathsf{Extract}(msk, \vec{V})$ and provides $\mathcal{A}$ with $dk_{\vec{V}}$.*
- *Oracle* Delegate *receives key attribute $\vec{V} = (\vec{v}_1, \dots, \vec{v}_j) \in \Phi$ and $\vec{v}_{j+1}$ subject to the restriction that $\mathcal{P}^{\mathsf{HIPE}}(\vec{V}', \vec{W}_0) = \mathcal{P}^{\mathsf{HIPE}}(\vec{V}', \vec{W}_1) = 0$, where $\vec{V}' = (\vec{v}_1, \dots, \vec{v}_{j+1})$. Otherwise, it obtains $dk_{\vec{V}} \leftarrow \mathsf{Extract}(msk, \vec{V})$, computes $dk_{\vec{V}'} \leftarrow \mathsf{Delg}_j(pp, dk_{\vec{V}}, \vec{v}_{j+1})$, and provides $\mathcal{A}$ with $dk_{\vec{V}'}$.*

- *Oracle* Challenge *receives two messages* $M_0, M_1 \in \mathcal{M}$. *It obtains* $C \leftarrow \mathsf{Enc}(pp, \vec{W}_\beta, M_\beta)$ *and provides* $\mathcal{A}$ *with* C.

Eventually, $\mathcal{A}$ *halts after it outputs its decision,* $\beta' \in \{0,1\}$.

Finalization: *Output* 1 *if* $\beta' = \beta$. *Otherwise, output* 0.

We define the advantage of $\mathcal{A}$ *as*

$$\mathsf{Adv}^{\text{wah-sa-cpa}}_{\mathcal{A},\mathsf{HIPE}}(\kappa) = \left|\Pr[\mathsf{Expt}^{\text{wah-sa-cpa}}_{\mathcal{A},\mathsf{HIPE}}(1^\kappa) = 1] - 1/2\right|.$$

We say that FE *is weakly attribute hiding against chosen-plaintext attacks in selective attribute setting if* $\mathsf{Adv}^{\text{wah-sa-cpa}}_{\mathcal{A},\mathsf{HIPE}}(\kappa)$ *is negligible for every polynomial-time adversary* $\mathcal{A}$.

Remark C.1. As in [OT11], we restrict that the target ciphertext attributes $\vec{W}_0$ and $\vec{W}_1$ should be the same depth. To remove the restriction, we follow the conversion in [LOS+10]: the ciphertext attributes vector $\vec{W} = (\vec{w}_1, \ldots, \vec{w}_h)$ are padded with random vectors $\vec{w}_{h+1}, \ldots, \vec{w}_d$ in the encryption algorithm.

C.1 Our HIPE Scheme

For $\delta \in [d]$, s_δ is the Gaussian parameter used in key extraction and delegation algorithms. Roughly speaking, we stack matrices $\boldsymbol{B}_{\vec{v}_\delta}$ to make IPE hierarchical. We interleave the matrices $\boldsymbol{B}_{\vec{v}_\delta}$ with $\bar{\boldsymbol{B}}_\delta$ for delegation.

$\mathsf{Setup}(1^\kappa, n, q, m, \ell, \vec{\mu}, \{s_\delta\}, \alpha, b, \mathfrak{g}, k)$**:** On input a security parameter 1^κ and parameters:

1. $(\boldsymbol{A}, \boldsymbol{R}_{\boldsymbol{A}}) \leftarrow \mathsf{GenTrap}(1^\kappa, q, n, m)$.
2. For $\delta \in [d]$, choose a random matrix $\bar{\boldsymbol{B}}_\delta \leftarrow \mathbb{Z}_q^{n \times nk}$.
3. For $\delta \in [d]$ and $i \in [\mu_\delta]$, choose uniformly random matrices $\boldsymbol{B}_{\delta,i} \leftarrow \mathbb{Z}_q^{n \times nk}$ for $i \in [\mu_\delta]$.
4. Choose a random matrix $\boldsymbol{U} \leftarrow \mathbb{Z}_q^{n \times \ell}$.

Output $pp = ((n, q, m, \ell, \vec{\mu}, \{s_\delta\}, \alpha, b, \mathfrak{g}, k), \boldsymbol{A}, \{\boldsymbol{B}_{\delta,i}\}, \{\bar{\boldsymbol{B}}_i\}, \boldsymbol{U})$ and $msk = (\boldsymbol{R}_{\boldsymbol{A}}, pp)$.

$\mathsf{Extract}(pp, msk, \vec{V})$**:** On input pp, msk, and predicate vectors $\vec{V} = (\vec{v}_1, \ldots, \vec{v}_j)$ where $\vec{v}_\delta = (\mathfrak{v}_{\delta,1}, \ldots, \mathfrak{v}_{\delta,\mu_\delta})$:

1. For $\delta \in [j]$, define the matrices $\boldsymbol{B}_{\vec{v}_\delta} = \sum_{i=1}^{\mu_\delta} \boldsymbol{B}_i \cdot H_{\boldsymbol{g}}(\mathfrak{v}_{\delta,i}) \in \mathbb{Z}_q^{n \times nk}$.
2. Set

$$\boldsymbol{A}'_{\vec{V}} = [\boldsymbol{A} \mid \boldsymbol{B}_{\vec{v}_1} \mid \bar{\boldsymbol{B}}_1 \mid \cdots \mid \boldsymbol{B}_{\vec{v}_j}] \in \mathbb{Z}_q^{n \times (m + (2j-1)nk)},$$
$$\boldsymbol{A}_{\vec{V}} = [\boldsymbol{A} \mid \boldsymbol{B}_{\vec{v}_1} \mid \bar{\boldsymbol{B}}_1 \mid \cdots \mid \boldsymbol{B}_{\vec{v}_j} \mid \bar{\boldsymbol{B}}_j] \in \mathbb{Z}_q^{n \times (m + 2jnk)},$$

3. Using the master secret key $\boldsymbol{R}_{\boldsymbol{A}}$, sample each column of $\boldsymbol{E}_{\vec{V}}$ independently from $D^{(2j-1)nk}_{\mathbb{Z}, s_j}$ conditioned on $\boldsymbol{A}'_{\vec{V}} \cdot \boldsymbol{E}_{\vec{V}} = \boldsymbol{U}$ and sample each column of $\boldsymbol{R}_{\vec{V}}$ independently from $D^{(2j-1)nk}_{\mathbb{Z}, s_j}$ conditioned on $\boldsymbol{A}'_{\vec{V}} \cdot \boldsymbol{R}_{\vec{V}} = \boldsymbol{G} - \bar{\boldsymbol{B}}_j$. ($\boldsymbol{R}_{\vec{V}}$ is a trapdoor of $\boldsymbol{A}_{\vec{V}}$ with tag $\boldsymbol{I}_n$.)

Output $dk_{\vec{V}} = (\boldsymbol{E}_{\vec{V}}, \boldsymbol{R}_{\vec{V}})$.

$\mathsf{Enc}(pp, \vec{W}, \boldsymbol{m})$**:** On input pp, a message $\boldsymbol{m} \in \{0,1\}^\ell$, and attribute vectors $\vec{W} = (\vec{w}_1, \ldots, \vec{w}_h)$ where $\vec{w}_\delta = (\mathfrak{w}_{\delta,1}, \ldots, \mathfrak{w}_{\delta,\mu_\delta})$:

1. Set $\boldsymbol{G} \leftarrow \boldsymbol{I}_n \otimes (1, b, \ldots, b^{k-1})$.
2. Choose a random vector $\boldsymbol{s} \leftarrow \mathbb{Z}_q^n$.
3. Set $\boldsymbol{c}_0 \leftarrow \boldsymbol{A}^\top \boldsymbol{s} + \boldsymbol{x}_0$, where $\boldsymbol{x}_0 \leftarrow \chi^m$.
4. Set $\boldsymbol{c}' \leftarrow \boldsymbol{U}^\top \boldsymbol{s} + \boldsymbol{x}' + \lfloor q/2 \rfloor \boldsymbol{m}$, where $\boldsymbol{x}' \leftarrow \chi^\ell$.
5. For $\delta \in [h-1]$, set $\boldsymbol{c}_i \leftarrow \bar{\boldsymbol{B}}_\delta^\top \boldsymbol{s} + \boldsymbol{x}_i$, where $\boldsymbol{x}_i \leftarrow \chi^{nk}$.
6. For $\delta \in [h]$ and $i \in [\mu_\delta]$, choose a random matrix $\boldsymbol{R}_{\delta,i} \leftarrow \{-1,+1\}^{m \times nk}$ and set $\boldsymbol{c}_{\delta,i} \leftarrow (\boldsymbol{B}_{\delta,i} + H(\mathfrak{w}_{\delta,i}) \cdot \boldsymbol{G})^\top \boldsymbol{s} + \boldsymbol{R}_{\delta,i}^\top \boldsymbol{x}_0 \in \mathbb{Z}_q^{nk}$.

Output $ct = (\boldsymbol{c}_0, \{\boldsymbol{c}_{\delta,i}\}, \{\boldsymbol{c}_i\}, \boldsymbol{c}')$ as a ciphertext.

$\mathsf{Dec}(pp, dk_{\vec{V}}, ct)$**:** On input pp, a decryption key $\boldsymbol{T}_{\vec{V}}$, where $\vec{V} = (\vec{v}_1, \ldots, \vec{v}_j)$, and a ciphertext $ct = (\boldsymbol{c}_0, \{\boldsymbol{c}_{\delta,i}\}, \{\boldsymbol{c}_i\}, \boldsymbol{c}')$:

1. For $\delta \in [j]$,
$$\boldsymbol{c}_{\vec{v}_\delta} \leftarrow \sum_{i=1}^{\mu_\delta} H_{\boldsymbol{g}}(\mathfrak{v}_{\delta,i}) \cdot \boldsymbol{c}_{\delta,i}.$$
2. Let $\boldsymbol{c} \leftarrow [\boldsymbol{c}_0; \boldsymbol{c}_{\vec{v}_1}; \boldsymbol{c}_1; \ldots; \boldsymbol{c}_{j-1}; \boldsymbol{c}_{\vec{v}_j}] \in \mathbb{Z}_q^{m+(2j-1)nk}$.
3. Compute $\boldsymbol{d} = \boldsymbol{c}' - \boldsymbol{E}_{\vec{V}}^\top \boldsymbol{c}$ and output $\lfloor (2/q)\boldsymbol{d} \rceil \bmod q$.

$\mathsf{Delg}(pp, dk_{\vec{V}}, \vec{V}')$**:** On input pp, a decryption key $(\boldsymbol{E}_{\vec{V}}, \boldsymbol{R}_{\vec{V}})$, where $\vec{V} = (\vec{v}_1, \ldots, \vec{v}_j)$, and $\vec{V}' = (\vec{v}_1, \ldots, \vec{v}_j, \vec{v}_{j+1}, \ldots, \vec{v}_t)$, do: Let $\vec{v}_\delta = (\mathfrak{v}_{\delta,1}, \ldots, \mathfrak{v}_{\delta,\mu_\delta}) \in \mathrm{GF}(q^n)^{\mu_\delta}$.

1. For $\delta \in [t]$, define the matrices
$$\boldsymbol{B}_{\vec{v}_\delta} = \sum_{i=1}^{\mu} \boldsymbol{B}_{\delta,i} \cdot H_{\boldsymbol{g}}(\mathfrak{v}_{\delta,i}) \in \mathbb{Z}_q^{n \times nk}.$$
2. Set
$$\boldsymbol{A}'_{\vec{V}'} = [\boldsymbol{A} \mid \boldsymbol{B}_{\vec{v}_1} \mid \bar{\boldsymbol{B}}_1 \mid \cdots \mid \boldsymbol{B}_{\vec{v}_t}] \in \mathbb{Z}_q^{n \times (m+(2t-1)nk)},$$
$$\boldsymbol{A}_{\vec{V}'} = [\boldsymbol{A} \mid \boldsymbol{B}_{\vec{v}_1} \mid \bar{\boldsymbol{B}}_1 \mid \cdots \mid \boldsymbol{B}_{\vec{v}_t} \mid \bar{\boldsymbol{B}}_t] \in \mathbb{Z}_q^{n \times (m+2tnk)},$$
3. Using the secret key $\boldsymbol{R}_{\vec{V}}$, sample each column of $\boldsymbol{E}_{\vec{V}'}$ independently from $D_{\mathbb{Z},s_t}^{(2t-1)nk}$ conditioned on $\boldsymbol{A}'_{\vec{V}'} \cdot \boldsymbol{E}_{\vec{V}'} = \boldsymbol{U}$ and sample each column of $\boldsymbol{R}_{\vec{V}'}$ independently from $D_{\mathbb{Z},s_t}^{(2t-1)nk}$ conditioned on $\boldsymbol{A}'_{\vec{V}'} \cdot \boldsymbol{R}_{\vec{V}'} = \boldsymbol{G} - \bar{\boldsymbol{B}}_t$. ($\boldsymbol{R}_{\vec{V}'}$ is a trapdoor of $\boldsymbol{A}_{\vec{V}'}$ with tag $\boldsymbol{I}_n$.)
4. Output $dk_{\vec{V}'} \leftarrow (\boldsymbol{E}_{\vec{V}'}, \boldsymbol{R}_{\vec{V}'})$.

Remark C.2. We note that we can delete $\boldsymbol{E}_{\vec{V}}$ if we employ the inversion algorithm for the LWE problem.

We omit a security proof, since it is very similar to ones in [AFV11,ADCM12] and ours for IPE. We also omit the detail of parameters.

D Construction from Ring-LWE

Langlois and Stehlé [LS12] showed They showed that the Ring-LWE problem over the ring $R_q = \mathbb{Z}_q[X]/\langle X^n + 1 \rangle \simeq \mathrm{GF}(q^{n/2}) \times \mathrm{GF}(q^{n/2})$ is hard on average if the corresponding ideal lattice problem is hard in the worst case. They also proposed the ring-LWE version of the ABB IBE scheme. It is easy to extend their IBE scheme to an IPE scheme. The obtained scheme supports $R = \mathrm{GF}(q^{n/2})$ and has public parameters and ciphertext of size $\Theta(\mu n \lg^3 q)$. Our idea ("half twisting") is applicable to the scheme and reduces the sizes by a $\lg q$ factor.

Full-rank difference encoding: Let us review the full-rank difference encoding in [LS12]. Let n be a power of 2 larger than 8. Let $q \equiv 3 \bmod 8$ be a prime. Then, $X^n + 1 = \mathfrak{g}_1 \mathfrak{g}_2$ over $\mathbb{Z}_q$, where $\mathfrak{g}_i = X^{n/2} + t_i X^{n/2} + 1$ is irreducible over $\mathbb{Z}_q$.

We first define a duplicating function $\mathrm{dp} : \mathrm{GF}(q^{n/2}) \to \mathrm{GF}(q^{n/2}) \times \mathrm{GF}(q^{n/2})$ as
$$\mathfrak{a} = a_0 + a_1 X + \ldots a_{n/2-1} X^{n/2-1} \mapsto (\mathfrak{a}, \mathfrak{a}).$$

By CRT, we have invertible mapping $\tau : \mathbb{Z}_q[X]/\langle X^n + 1 \rangle \to \mathbb{Z}_q[X]/\langle \mathfrak{g}_1 \rangle \times \mathbb{Z}_q[X]/\langle \mathfrak{g}_1 \rangle$ as
$$\mathfrak{a} \mapsto (\mathfrak{a} \bmod \mathfrak{g}_1, \mathfrak{a} \bmod \mathfrak{g}_2).$$

Then, we define the full-rank difference encoding from $\mathrm{GF}(q^{n/2})$ to R_q as
$$H = \tau^{-1} \circ \mathrm{dp}.$$

By the construction, H is an isomorphism from $\mathrm{GF}(q^{n/2})$ to a sub-ring of R_q, which is a field.

The Micciancio–Peikert algorithm in the Ring-LWE setting: Let $\boldsymbol{g} = (1, b, \ldots, b^{k-1}) \in R_q^k$. In the ring setting, we will use $\boldsymbol{g}$ directly instead of $\boldsymbol{G}$. Roughly speaking, the trapdoor of $\boldsymbol{a} \in R_q^{\bar{m}+k}$ with tag $\mathfrak{h} \in R_q^*$ is $\boldsymbol{R} \in R_q^{\bar{m}+k}$ such that

$$\boldsymbol{a} = [\bar{\boldsymbol{a}} \mid \mathfrak{h}\boldsymbol{g} - \bar{\boldsymbol{a}}\boldsymbol{R}].$$

New encoding: We next define $H_{\boldsymbol{g}} : \mathrm{GF}(q^{n/2}) \to R_q^{k \times k}$.

As in Section 4, we define $d_{\boldsymbol{g}}(\mathfrak{a}) = (\mathfrak{a}_1, \ldots, \mathfrak{a}_k) \in B^k$ such that $\boldsymbol{g} \cdot d_{\boldsymbol{g}}(\mathfrak{a}) = \sum_{i=1}^{k} \mathfrak{a}_i \cdot b^{i-1} = \mathfrak{a}$. For a polynomial $\mathfrak{a}$, we define $D_{\boldsymbol{g}}$ as

$$D_{\boldsymbol{g}} : \mathfrak{a} \mapsto [d_{\boldsymbol{g}}(\mathfrak{a})\, d_{\boldsymbol{g}}(b\mathfrak{a})\, \ldots\, d_{\boldsymbol{g}}(b^{k-1}\mathfrak{a})] \in B^{k \times k} \subset R_q^{k \times k}.$$

We then define $H_{\boldsymbol{g}} : \mathrm{GF}(q^{n/2}) \to R_q^{k \times k}$ by $H_{\boldsymbol{g}} = D_{\boldsymbol{g}} \circ H$. By the construction, we have that $\boldsymbol{g} \cdot H_{\boldsymbol{g}}(\mathfrak{a}) = H(\mathfrak{a}) \cdot \boldsymbol{g}$.

D.1 Construction

For simplicity, we only consider $b = 2$, $B = \{0, 1\}$, and $k = \lceil \lg q \rceil$. In the description, Υ denotes a distribution over the elliptic Gaussians in R_q defined in [LPR10].

$\mathsf{Setup}(1^\kappa, n, q, m, \ell, \vec{\mu}, \{s_\delta\}, \alpha, b, k)$**:** On input security parameter 1^κ and parameters:

1. $(\boldsymbol{a}, \boldsymbol{R_a}) \leftarrow \mathsf{RingGenTrap}(1^\kappa, q, n, m)$.
2. For $\delta \in [d]$: choose uniformly random vectors $\bar{\boldsymbol{b}}_{\delta,i} \leftarrow R_q^k$
3. For $\delta \in [d]$ and $\imath \in [\mu_\delta]$: choose uniformly random vectors $\boldsymbol{b}_{\delta,i} \leftarrow R_q^k$
4. Choose random vector $\boldsymbol{u} \leftarrow R_q^\ell$.

Output $pp = ((n, q, m, \ell, \vec{\mu}, \{s_\delta\}, \alpha, b, \mathfrak{g}, k), \boldsymbol{a}, \{\boldsymbol{b}_{\delta,i}\}, \{\bar{\boldsymbol{b}}_\delta\}, \boldsymbol{u})$ and $msk = (\boldsymbol{R_a}, pp)$.

$\mathsf{Extract}(pp, msk, \vec{V})$**:** On input pp, msk, and predicate vectors $\vec{V} = (\vec{v}_1, \ldots, \vec{v}_j)$ where $\vec{v}_\delta = (\mathfrak{v}_{\delta,1}, \ldots, \mathfrak{v}_{\delta,\mu_\delta}) \in \mathrm{GF}(q^{n/2})^{\mu_\delta}$:

1. For $\delta \in [j]$, define the vectors $\boldsymbol{b}_{\vec{v}_\delta} = \sum_{i=1}^{\mu_\delta} \boldsymbol{b}_{\delta,i} \cdot H_{\boldsymbol{g}}(\mathfrak{v}_{\delta,i})$.
2. Set

$$\begin{aligned} \boldsymbol{a}'_{\vec{v}_\delta} &= [\boldsymbol{a} \mid \boldsymbol{b}_{\vec{v}_1} \mid \bar{\boldsymbol{b}}_1 \mid \cdots \mid \boldsymbol{b}_{\vec{v}_j}] \in R_q^{m+(2j-1)nk}, \\ \boldsymbol{a}_{\vec{v}_\delta} &= [\boldsymbol{a} \mid \boldsymbol{b}_{\vec{v}_1} \mid \bar{\boldsymbol{b}}_1 \mid \cdots \mid \boldsymbol{b}_{\vec{v}_j} \mid \bar{\boldsymbol{b}}_j] \in R_q^{m+2jnk}. \end{aligned}$$

3. Using the master secret key $\boldsymbol{R_a}$, sample each column of $\boldsymbol{E}_{\vec{V}}$ independently from $D_{\mathbb{Z},s_j}^{(2j-1)nk}$ conditioned on $\boldsymbol{a}'_{\vec{V}} \cdot \boldsymbol{E}_{\vec{V}} = \boldsymbol{U}$ and sample $\boldsymbol{R}_{\vec{V}}$ from $D_{\mathbb{Z},s_j}^{(2j-1)nk}$ conditioned on $\boldsymbol{a}'_{\vec{V}} \cdot \boldsymbol{R}_{\vec{V}} = \boldsymbol{g} - \bar{\boldsymbol{b}}_j$. ($\boldsymbol{b}_{\vec{V}}$ is a trapdoor of $\boldsymbol{a}_{\vec{V}}$ with tag $1 \in \mathrm{GF}(q^{n/2})$.)

Output $dk_{\vec{V}} = (\boldsymbol{E}_{\vec{V}}, \boldsymbol{R}_{\vec{V}})$.

$\mathsf{Enc}(pp, \vec{W}, \boldsymbol{m})$**:** On input pp, message $\boldsymbol{m} \in (\{0,1\}^n)^\ell \subset \mathrm{GF}(q^n)^\ell$, and attribute vector $\vec{W} = (\vec{w}_1, \ldots, \vec{w}_h)$ where $\vec{w}_\delta = (\mathfrak{w}_{\delta,1}, \ldots, \mathfrak{w}_{\delta,\mu_\delta}) \in \mathrm{GF}(q^{n/2})^{\mu_\delta}$:

1. $\chi \leftarrow \Upsilon$.
2. Set $\boldsymbol{g} \leftarrow (1, b, \ldots, b^{k-1}) \in R_q^k$.
3. Choose $\mathfrak{s} \leftarrow R_q$.
4. Set $\boldsymbol{c}_0 \leftarrow \mathfrak{s} \cdot \boldsymbol{a} + \boldsymbol{x}_0$, where $\boldsymbol{x}_0 \leftarrow \chi^m$.
5. Set $\boldsymbol{c}' \leftarrow \mathfrak{s} \cdot \boldsymbol{u} + \boldsymbol{x}' + \lfloor q/2 \rfloor\, \boldsymbol{m}$, where $\boldsymbol{x}' \leftarrow \chi^\ell$.
6. For $\delta \in [h-1]$, set $\boldsymbol{c}_i \leftarrow \mathfrak{s}\bar{\boldsymbol{b}}_\delta + \boldsymbol{x}_i$, where $\boldsymbol{x}_i \leftarrow \chi^k$.
7. For $\delta \in [h]$ and $i \in [\mu_\delta]$, choose a random matrix $\boldsymbol{R}_{\delta,i} \leftarrow (\{-1,+1\}^n)^{m \times k}$ and set $\boldsymbol{c}_{\delta,i} \leftarrow \mathfrak{s} \cdot (\boldsymbol{b}_{\delta,i} + H(\mathfrak{w}_{\delta,i}) \cdot \boldsymbol{g}) + \boldsymbol{x}\boldsymbol{R}_{\delta,i} \in R_q^k$.

Output $ct = (\boldsymbol{c}_0, \{\boldsymbol{c}_{\delta,i}\}, \{\boldsymbol{c}_i\}, \boldsymbol{c}')$ as a ciphertext.

$\mathsf{Dec}(pp, dk_{\vec{V}}, ct)$**:** On input pp, decryption key $\boldsymbol{E}_{\vec{V}}$, where $\vec{V} = (\vec{v}_1, \ldots, \vec{v}_j)$, $ct = (\boldsymbol{c}_0, \{\boldsymbol{c}_{\delta,i}\}, \{\boldsymbol{c}_i\}, \boldsymbol{c}')$:

1. For $\delta \in [j]$,

$$\boldsymbol{c}_{\vec{v}_\delta} \leftarrow \sum_{i=1}^{\mu_\delta} \boldsymbol{c}_{\delta,i} \cdot H_{\boldsymbol{g}}(\mathfrak{v}_{\delta,i}).$$

2. Let $\boldsymbol{c} \leftarrow [\boldsymbol{c}_0, \boldsymbol{c}_{\vec{v}_1}, \ldots, \boldsymbol{c}_{\vec{v}_j}] \in R_q^{m+j\bar{m}}$.
3. $\boldsymbol{d} \leftarrow \boldsymbol{c}' - \boldsymbol{c}\boldsymbol{E}_{\vec{V}}$ and output $\lfloor 2\boldsymbol{d}/q \rceil \bmod 2$.

$\mathsf{Delg}(pp, dk_{\vec{V}}, \vec{V}')$**:** On input pp, decryption key $\boldsymbol{R}_{\vec{V}}$, where $\vec{V} = (\vec{v}_1, \ldots, \vec{v}_j)$, and $\vec{V}' = (\vec{v}_1, \ldots, \vec{v}_j, \vec{v}_{j+1}, \ldots, \vec{v}_t)$: Let $\vec{v}_\delta = (\mathfrak{v}_{\delta,1}, \ldots, \mathfrak{v}_{\delta,\mu_\delta}) \in \mathrm{GF}(q^{n/2})^{\mu_\delta}$.

1. For $\delta \in [t]$, define the vectors

$$\boldsymbol{b}_{\vec{v}_\delta} = \sum_{i=1}^{\mu} \boldsymbol{a}_{\delta,i} \cdot H_{\boldsymbol{g}}(\mathfrak{v}_{\delta,i}) \in R_q^k.$$

2. Define
3. Set

$$\boldsymbol{a}'_{\vec{V}'} = [\boldsymbol{a} \mid \boldsymbol{b}_{\vec{v}_1} \mid \bar{\boldsymbol{b}}_1 \mid \cdots \mid \boldsymbol{b}_{\vec{v}_t}] \in R_q^{m+(2t-1)nk},$$
$$\boldsymbol{a}_{\vec{V}'} = [\boldsymbol{a} \mid \boldsymbol{b}_{\vec{v}_1} \mid \bar{\boldsymbol{b}}_1 \mid \cdots \mid \boldsymbol{b}_{\vec{v}_t} \mid \bar{\boldsymbol{b}}_t] \in R_q^{m+2tnk}.$$

4. Using the secret key $\boldsymbol{R}_{\vec{V}}$, sample each column of $\boldsymbol{E}_{\vec{V}'}$ independently from $D_{\mathbb{Z},s_t}^{(2t-1)nk}$ conditioned on $\boldsymbol{a}'_{\vec{V}'} \cdot \boldsymbol{E}_{\vec{V}'} = \boldsymbol{u}$ and sample each column of $\boldsymbol{R}_{\vec{V}'}$ independently from $D_{\mathbb{Z},s_t}^{(2t-1)nk}$ conditioned on $\boldsymbol{a}'_{\vec{V}'} \cdot \boldsymbol{R}_{\vec{V}'} = \boldsymbol{g} - \bar{\boldsymbol{b}}_t$. ($\boldsymbol{R}_{\vec{V}'}$ is a trapdoor of $\boldsymbol{a}_{\vec{V}'}$ with tag 1.)
5. Output $dk_{\vec{V}'} \leftarrow (\boldsymbol{E}_{\vec{V}'}, \boldsymbol{R}_{\vec{V}'})$.

Here, we only remark that we require the ring-LWE version of the leftover hash lemma in the security proof. The lemma is obtained by modifying Micciancio's regularity lemma [Mic07] and its improvements.

E Application requiring Large Ring

In what follows, we discuss several embeddings of useful predicates into our IPE scheme. Hereafter we denote key and ciphertext attributes by a and x, respectively.

Equality, polynomial, disjunction, and conjunction: The most basic predicate is equality, which on input a and x outputs 1 if and only if $a = x$. To generate a secret key for x, we let $\vec{v} = (x, -1)$. To encrypt a message for a, we set $\vec{w} = (1, a)$. The correctness and security follows from the fact that $a = x$ if and only if $\vec{w}^\top \vec{v} = 0$.

Katz, Sahai, Waters showed how to encode predicates corresponding to polynomial evaluations. For a univariate polynomial p of degree at most d, we define the corresponding predicate $\mathcal{P}_p(x)$ that is 1 if and only if $p(x) = 0$. To generate a secret key corresponding to a polynomial $p(X) = a_0 + \cdots + a_d X^d$, we set $\vec{v} = (a_0, \ldots, a_d)^\top$. To encrypt a message for x, we set $\vec{w} = (1, x, x^2, \ldots, x^d)^\top$. Since $\vec{w}^\top \vec{v} = 0$ if and only if $p(x) = 0$, the correctness and security follow. We can easily extend the above encoding to multi-variate polynomials. We note that we can change the role of keys and ciphertexts.

We next review how the basic tools on logic, the logical disjunction and conjunction, are encoded into polynomial and inner product [KSW08]. The following is a summary of implementations.

Policy	Implementation
$(x = a_1) \vee (x = a_2)$	$f(x) = (x - a_1)(x - a_2) = 0$
$(x_1 = a_1) \vee (x_2 = a_2)$	$f(x) = (x_1 - a_1)(x_2 - a_2) = 0$
$(x_1 = a_1) \wedge (x_2 = a_2)$	$f(x) = r_1(x_1 - a_1) + r_2(x_2 - a_2) = 0$

In the implementation of logical conjunction ($\wedge$), one should choose $r_1, r_2 \in R$ uniformly at random. To show the security, we require that for any two values $b_1 \in R$ and $b_2 \in R$, $r_1 b_1 + r_2 b_2 \neq 0$ with all but negligible probability. In our case, this follows from R is exponentially large field, $\mathrm{GF}(q^n)$.

E.1 Hidden Vector Encryption

Hidden vector encryption (HVE) is proposed by Boneh and Waters [BW07], which is a basis of range query over encrypted data and is extended by Shi et al. [SBC$^+$07] to multi-dimensional queries. Iovino and Persiano proposed a prime-order HVE scheme [IP08].

Let S be a finite set and let $*$ be a special symbol not in S. We define $S_* = S \cup \{*\}$. For two vectors $\vec{a} \in S_*^N$ and $\vec{x} \in S^N$, we define a hidden-vector predicate as $\mathcal{P}^{\mathsf{hve}}(\vec{a}, \vec{x}) = 1$ if and only if $(a_i = x_i$ or $*)$ for any $i \in [N]$. By using the KSW embeddings [KSW08], we can implement the hidden-vector predicate: The 1-dimensional $\mathcal{P}^{\mathsf{hve}}(a, x)$ is encoded by setting key-attribute vector $(\mathfrak{v}_1, \mathfrak{v}_2)$ as $(1, \mathfrak{a})$ if $\mathfrak{a}_i \neq *$ and $(0,0)$ if $\mathfrak{a}_i = *$, and setting ciphertext-attribute vector $(\mathfrak{w}_1, \mathfrak{w}_2) = (-\mathfrak{x}_i, 1)$. To extend it to an N-dimensional predicate, we take logical conjunction of 1-dimensional predicates. Since R is $\mathrm{GF}(q^n)$ in our case, we can implement it by our IPE as Katz et al. did, while setting $q = 2^{O(n)}$ is required if we employ the AFV IPE scheme.

In addition, we note that HVE can implement wild-carded identity-based encryption [ABC$^+$11] if we do not require hierarchical key delegation.

E.2 Subset Query in Large Universe

We additionally review subset queries that is the important application of the logical conjunction. HVE already serves a subset query by setting N as the cardinality of the universe and, thus, allows a subset query over the small universe [BW07]. We note that the logical conjunction and disjunction solves the problematic issues, by setting N as the size of the set in query instead of the universe.

Let $\mathcal{T} \subseteq R$ be the universe. Each user has a set of attributes $A = (a_1, \ldots, a_m) \subseteq \mathcal{T}$. Encryption is related to another set of attributes $X = (x_1, \ldots, x_n) \subseteq \mathcal{T}$. The subset predicate is denoted by $\mathcal{P}^{\mathsf{subset}}(A, X)$, which is 1 if and only if $A \subseteq X$. We can express this predicate as $\bigwedge_{j=1}^{m} \bigvee_{i=1}^{n} (x_i = a_j)$. By the KSW encoding, we can encode it into a randomized polynomial $P(X_1, \ldots, X_n, a_1, \ldots, a_m) = \sum_{j=1}^{m} r_j \prod_{i=1}^{n} (X_i - a_j)$ with $r_1, \ldots, r_m \leftarrow R$. From [YNKF11], we have an expansion

$$P(x_1, \ldots, x_n, a_1, \ldots, a_m) = \sum_{j=1}^{m} r_j \prod_{i=1}^{n} (x_i - a_j) = \sum_{k=0}^{n} \sigma_{n,k}(x_1, \ldots, x_n) \left(\sum_{j=0}^{m} r_j (-a_j)^{n-k} \right),$$

where $\sigma_{n,k}$ is the elementary symmetric equations of n variables and degree k, that is, $\sigma_{n,k}(X_1, \ldots, X_n) = \sum_{S \subseteq [n], \#S = k} \left(\prod_{i \in S} X_i \right)$. [4] We associate a set $A = \{a_1, \ldots, a_m\}$ with a key-attribute vector $\vec{v}_A = (\mathfrak{v}_0, \ldots, \mathfrak{v}_n)^\top \in R^{n+1}$ by setting $\mathfrak{v}_i = \sum_{j=1}^{m} r_j (-a_j)^{n-i}$ for $i = 0, \ldots, n$, where $r_j \leftarrow R$ for $j = 1, \ldots, m$. For ciphertext attribute, we associate a set $X = \{x_1, \ldots, x_n\}$ with a vector $\vec{w}_X = (\mathfrak{w}_0, \ldots, \mathfrak{w}_n)^\top \in R^{n+1}$ by setting $\mathfrak{w}_i = \sigma_{n,i}(x_1, \ldots, x_n)$. Since this encoding employs the logical conjunction, we require exponentially large R.

[4] Notice that, by expanding $\prod_{i=1}^{n} (X - x_i)$, we can compute a vector $(\sigma_{n,0}(x_1, \ldots, x_n), \ldots, \sigma_{n,n})$ in $O(n^2)$ multiplication and addition in R.