Functional Encryption for Inner Product: Achieving Constant-Size Ciphertexts with Adaptive Security or Support for Negation

Nuttapong Attrapadung[1] and Benoît Libert[2⋆]

[1] Research Center for Information Security, AIST (Japan)
[2] Université catholique de Louvain, Crypto Group (Belgium)

Abstract. In functional encryption (FE) schemes, ciphertexts and private keys are associated with attributes and decryption is possible whenever key and ciphertext attributes are suitably related. It is known that expressive realizations can be obtained from a simple FE flavor called inner product encryption (IPE), where decryption is allowed whenever ciphertext and key attributes form orthogonal vectors. In this paper, we construct (non-anonymous) IPE systems with *constant-size* ciphertexts for the zero *and* non-zero evaluations of inner products. These schemes respectively imply an adaptively secure identity-based broadcast encryption scheme and an identity-based revocation mechanism that both feature short ciphertexts and rely on simple assumptions in prime order groups. We also introduce the notion of *negated spatial encryption*, which subsumes non-zero-mode IPE and can be seen as the revocation analogue of the spatial encryption primitive of Boneh and Hamburg.

Keywords. Functional encryption, identity-based broadcast encryption, revocation, efficiency.

1 Introduction

Ordinary encryption schemes usually provide coarse-grained access control since, given a ciphertext, only the holder of the private key can obtain the plaintext. In many applications such as distributed file systems, the need for fine-grained and more complex access control policies frequently arises. To address these concerns, several kinds of *functional public key encryption* schemes have been studied.

Functional encryption can be seen as a generalization of identity-based encryption (IBE) [24, 8]. In IBE schemes, the receiver's ability to decrypt is merely contingent on his knowledge of a private key associated with an identity that matches a string chosen by the sender. In contrast, functional encryption (FE) systems make it possible to decrypt using a private key $\mathsf{sk}_{\boldsymbol{x}}$ corresponding to a set $\boldsymbol{x}$ of atomic elements, called *attributes*, that is suitably related – according to some well-defined relation R – to another attribute set $\boldsymbol{y}$ specified by the sender.

⋆ This author acknowledges the Belgian National Fund for Scientific Research (F.R.S.-F.N.R.S.) for their support and the BCRYPT Interuniversity Attraction Pole.

The goal of this paper is to describe new (pairing-based) functional encryption constructions providing short ciphertexts (ideally, their length should not depend on the size of attribute sets) while providing security against adaptive adversaries or supporting negation (e.g. decryption should be disallowed to holders of private keys $\mathsf{sk}_{\boldsymbol{x}}$ for which $R(\boldsymbol{x}, \boldsymbol{y}) = 1$).

Related Work. The first flavor of functional encryption traces back to the work of Sahai and Waters [22] that was subsequently extended in [16, 21]. Their concept, called *attribute-based encryption* (ABE), allows a sender to encrypt data under a set of attributes ω while an authority generates private keys for access control policies $\mathcal{T}$. Decryption rights are granted to anyone holding a private key for a policy $\mathcal{T}$ such that $\mathcal{T}(\omega) = 1$. Identity-based broadcast encryption (IBBE) [2, 23, 13, 9] and revocation (IBR) [19] schemes can also be thought of as functional encryption systems where ciphertexts are encrypted for a set of identities $S = \{\mathsf{ID}_1, \ldots, \mathsf{ID}_n\}$: in IBBE (resp. IBR) systems, decryption requires to hold a private key $\mathsf{sk}_{\mathsf{ID}}$ for which $\mathsf{ID} \in S$ (resp. $\mathsf{ID} \notin S$).

The above kinds of functional encryption systems are only *payload hiding* in that they keep encrypted messages back from unauthorized parties but ciphertexts do not hide their underlying attribute set. *Predicate encryption* schemes [10, 18, 26, 25] additionally provide *anonymity* as ciphertexts also conceal the attribute set they are associated with, which enables [7, 1] efficient searches over encrypted data. In [18], Katz, Sahai and Waters devised a predicate encryption scheme for inner products: a ciphertext encrypted for the attribute vector $\vec{Y}$ can be opened by any key $\mathsf{sk}_{\vec{X}}$ such that $\vec{X} \cdot \vec{Y} = 0$. As shown in [18], inner product encryption (IPE) suffices to give functional encryption for a number of relations corresponding to the evaluation of polynomials or CNF/DNF formulae.

Our Contributions. While quite useful, the IPE scheme of [18] strives to anonymize ciphertexts, which makes it difficult to break through the linear complexity barrier (in the vector length n) in terms of ciphertext size. It indeed seems very hard to avoid such a dependency as long as anonymity is required: for instance, anonymous FE constructions [10, 17] suffer from the same overhead. A similar problem appears in the context of broadcast encryption, where the only known scheme [3] that conceals the receiver set also has $O(n)$-size ciphertexts.

This paper focuses on applications of IPE schemes, such as identity-based broadcast encryption and revocation systems, where the anonymity property is not fundamental. Assuming public ciphertext attributes rather than anonymity may be useful in other contexts. For instance, suppose that a number of ciphertexts are stored with varying attributes $\boldsymbol{y}$ on a server and we want to decrypt only those for which $R(\boldsymbol{x}, \boldsymbol{y}) = 1$. Anonymous ciphertexts require to decrypt all of them whereas public attributes $\boldsymbol{y}$ make it possible to test whether $R(\boldsymbol{x}, \boldsymbol{y})$ (which is usually faster than decrypting) and only decrypt appropriate ones.

At the expense of sacrificing anonymity, we thus describe IPE schemes where the ciphertext overhead reduces to $O(1)$ as long as the description of the ciphertext attribute vector is not considered as being part of the ciphertext, which is a common assumption in the broadcast encryption/revocation applications. In addition, the number of pairing evaluations to decrypt is also constant, which

significantly improves upon $O(n)$, since pairings calculations still remain costly.

Our first IPE system achieves adaptive security, as opposed to the selective model, used in [18], where the adversary has to choose the target ciphertext vector $\vec{Y}$ upfront. To acquire adaptive security, we basically utilize the method used in the Waters' fully secure IBE [27], albeit we also have to introduce a new trick called "n-equation technique" so as to deal with the richer structure of IPE. Our system directly yields the first adaptively secure identity-based broadcast encryption scheme with constant-size ciphertexts in the standard model. Previous IBBE with $O(1)$-size ciphertexts were either only selective-ID secure [2, 13, 9, 23] or in the random oracle model [15]. Among IBBE systems featuring compact ciphertexts (including selective-ID secure ones), ours is also the first one relying on simple assumptions (*i.e.*, no q-type assumption) in prime order groups.

It is worth mentioning that techniques developed by Lewko and Waters [20] can be applied to the construction of Boneh and Hamburg [9] to give fully secure IBBE with short ciphertexts in composite order groups. However, it was not previously known how to obtain such a scheme in prime order groups (at least without relying on the absence of computable isomorphism in asymmetric pairing configurations). Indeed, despite recent progress [14], there is still no black-box way to translate pairing-based cryptosystems from composite to prime order groups. In particular, Freeman's framework [14] does not apply to [20].

Our second contribution is an IPE system for non-zero inner products: ciphertexts encrypted for vector $\vec{Y}$ can only be decrypted using $\mathsf{sk}_{\vec{X}}$ if $\vec{X} \cdot \vec{Y} \neq 0$, which – without retaining anonymity – solves a question left open by Katz, Sahai and Waters [18][Section 5.4]. The scheme implies the first identity-based revocation (IBR) mechanism [19] with $O(1)$-size ciphertexts. Like the schemes of Lewko, Sahai and Waters [19], its security is analyzed in a non-adaptive model where the adversary has to choose which users to corrupt at the outset of the game[3]. In comparison with [19] where ciphertexts grow linearly with the number of revoked users and public/private keys have constant size, our basic IBR construction performs in the dual way since key sizes depend on the maximal number of revoked users. Depending on the application, one may prefer one scheme over the other one. We actually show how to generalize both implementations and obtain a tradeoff between ciphertext and key sizes (and without assuming a maximal number of revoked users): the second scheme of [19] and ours can be seen as lying at opposite extremities of the spectrum.

On a theoretical side, our non-zero IPE realization turns out to be a particular case of a more general primitive, that we call *negated spatial encryption*, which we define as a negated mode for the spatial encryption primitive of Boneh and Hamburg [9]. Namely, keys correspond to subspaces and can decrypt ciphertexts encrypted under points that lie *outside* the subspace. This generalized primitive turns out to be non-trivial to implement and we had to use a fully

[3] We indeed work in a slightly stronger model, called *co-selective-ID*, where the adversary chooses which parties to corrupt at the beginning – before seeing the public key – but is not required to announce the target revoked set until the challenge phase.

generalized form of our new "n-equation" technique. The proposed scheme is proven secure under a non-standard assumption defined in [19].

Our Techniques. The core technique of our *non-zero* IPE scheme will be used throughout the paper, including in our adaptively secure *zero* IPE scheme. This can be viewed analogously to fact that Waters' fully secure IBE [27] uses the revocation technique of [19]. Our non-zero IPE also builds on [19]. However, the fact that non-zero IPE has much richer structure than revocation scheme and the pursued goal of achieving constant ciphertext size together prevent us from using their techniques directly. To describe the difficulties that arise, we first outline the Lewko-Sahai-Waters revocation scheme in its simplified form where security proof is not provided and where only one user is revoked.

Construction 1. (A simplified revocation scheme)

▶ Setup: lets $(\mathbb{G}, \mathbb{G}_T)$ be bilinear groups of prime order p and picks $g \xleftarrow{\$} \mathbb{G}$, $\alpha, \alpha_1, \alpha_2 \xleftarrow{\$} \mathbb{Z}_p$. The public key is $\big(g, g^{\alpha_1}, g^{\alpha_2}, e(g,g)^{\alpha}\big)$. The master key is g^{α}.

▶ KeyGen: chooses $t \xleftarrow{\$} \mathbb{Z}_p$ and outputs a private key for an identity $\mathsf{ID} \in \mathbb{Z}_p$ as $(K_0 = g^t,\ K_1 = g^{\alpha+\alpha_1 t},\ K_2 = g^{t(\alpha_1 \mathsf{ID}+\alpha_2)})$.

▶ Encrypt: encrypts M and specifies a revoked ID' by choosing $s \xleftarrow{\$} \mathbb{Z}_p$ and computing $(E_0 = \mathsf{M} \cdot e(g,g)^{\alpha s},\ E_1 = g^{s(\alpha_1 \mathsf{ID}'+\alpha_2)},\ E_2 = g^s)$.

▶ Decrypt: decryption computes $e(K_2, E_2)^{\frac{1}{\mathsf{ID}-\mathsf{ID}'}} e(E_1, K_0)^{-\frac{1}{\mathsf{ID}-\mathsf{ID}'}} = e(g,g)^{\alpha_1 t s}$ if $\mathsf{ID} \neq \mathsf{ID}'$. It then computes $e(g,g)^{\alpha s}$ as $e(K_1, E_2)/e(g,g)^{\alpha_1 t s} = e(g,g)^{\alpha s}$.

The scheme can be explained by viewing a key and a ciphertext as forming a linear system of 2 equations in the exponent of $e(g,g)$ with variables $\alpha_1 ts, \alpha_2 ts$.

$$M_{\mathsf{ID},\mathsf{ID}'} \begin{pmatrix} \alpha_1 ts \\ \alpha_2 ts \end{pmatrix} := \begin{pmatrix} \mathsf{ID} & 1 \\ \mathsf{ID}' & 1 \end{pmatrix} \begin{pmatrix} \alpha_1 ts \\ \alpha_2 ts \end{pmatrix} = \begin{pmatrix} \log(e(K_2, E_2)) \\ \log(e(E_1, K_0)) \end{pmatrix}.$$

Computing $e(g,g)^{\alpha_1 ts}$ amounts to solve the system, which is possible when $\det(M_{ID,\mathsf{ID}'}) \neq 0$ (and thus $\mathsf{ID} \neq \mathsf{ID}'$, as required). In particular, decryption computes a linear combination (in the exponent) with coefficients from the first row of $M^{-1}_{\mathsf{ID},\mathsf{ID}'}$ which is $(\frac{1}{\mathsf{ID}-\mathsf{ID}'}, \frac{-1}{\mathsf{ID}-\mathsf{ID}'})$. In [19], this is called "2-equation technique". The scheme is extended to n-dimension, *i.e.,* the revocation of n users $\{\mathsf{ID}'_1, \ldots, \mathsf{ID}'_n\}$, by utilizing n local independent systems of two equations

$$M_{\mathsf{ID},\mathsf{ID}'_j} \Big(\alpha_1 ts_j,\ \alpha_2 ts_j\Big)^{\top} = \Big(\log(e(K_2, E_{2,j})),\ \log(e(E_{1,j}, K_0))\Big)^{\top} \text{ for } j \in [1,n],$$

that yield $2n$ ciphertext components $(E_{1,j}, E_{2,j})$, each one of which corresponds to a share s_j of s such that $s = \sum_1^n s_j$. The decryption at j-th system returns $e(g,g)^{\alpha_1 t s_i}$ if $\mathsf{ID} \neq \mathsf{ID}'_j$. Combining these results finally gives $e(g,g)^{\alpha_1 ts}$.

We aim at *constant-size* ciphertexts for non-zero IPE schemes of dimension n. When trying to use the 2-equation technique with n dimensions, the following difficulties arise. First, the "decryptability" condition $\vec{X} \cdot \vec{Y} \neq 0$ cannot be decomposed as simply as that of the revocation scheme, which is decomposable as the conjunction of $\mathsf{ID} \neq \mathsf{ID}'_j$ for $j \in [1,n]$. Second, the ciphertext size was $O(n)$.

Towards solving these, we introduce a technique called "n-equation technique". First, we utilize n secret exponents $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)^\top$ and let α_1 function as the "master" exponent while $\alpha_2, \ldots, \alpha_n$ serve as the "perturbed" factors. Intuitively, we will set up a system of n linear equations of the form:

$$M_{\vec{X},\vec{Y}}(\alpha_1 ts, \ldots, \alpha_n ts)^\top = \big(\log(e(K_{i_1}, E_{j_1})), \ldots, \log(e(K_{i_n}, E_{j_n}))\big)^\top \quad (1)$$

where $\{K_{i_k}\}$ and $\{E_{j_k}\}$ are elements of $\mathbb{G}$ defined for a key for $\vec{X}$ and a ciphertext for $\vec{Y}$ respectively. At first, this generalized system seems to require linear-size ciphertexts $(E_{j_1}, \ldots, E_{j_n})$. A trick to resolve this is to reuse ciphertext elements throughout the system: we let $E_{j_k} = E_2 = g^s$ for $k \in [1, n-1]$. This effectively yields a constraint $M_{\vec{X},\vec{Y}} = \begin{pmatrix} Q_{\vec{X}}^\top & R^\top \end{pmatrix}^\top$, where $Q_{\vec{X}}$ is a $(n-1) \times n$ matrix parameterized only by $\vec{X}$ and R is a $1 \times n$ matrix. The remaining problem is then to choose $M_{\vec{X},\vec{Y}}$ in such a way that the system has a solution if $\vec{X} \cdot \vec{Y} \neq 0$ (the decryptability condition). To this end, we define

$$M_{\vec{X},\vec{Y}} := \begin{pmatrix} -\frac{x_2}{x_1} & 1 & & \\ -\frac{x_3}{x_1} & & 1 & \\ \vdots & & & \ddots \\ -\frac{x_n}{x_1} & & & 1 \\ y_1 & y_2 & y_3 & \ldots\ y_n \end{pmatrix}, \quad (2)$$

where it holds that $\det(M_{\vec{X},\vec{Y}}) = (-1)^{n+1} \vec{X} \cdot \vec{Y} / x_1$. By translating this conceptual view back into algorithms, we obtain a basic non-zero IPE scheme. From this, we propose two schemes for non-zero IPE: the first one is a special case of negated spatial encryption scheme in section 5.1, while the second one is proven secure under simple assumptions and given in section 5.2.

Organization. In the forthcoming sections, the syntax and the applications of functional encryption are explained in sections 2 and 3. We describe our zero mode IPE system in section 4. Our negated schemes are detailed in section 5.

2 Definitions

2.1 Syntax and Security Definition for Functional Encryption

Let $R : \Sigma_{\mathsf{k}} \times \Sigma_{\mathsf{e}} \to \{0,1\}$ be a boolean function where Σ_{k} and Σ_{e} denote "key attribute" and "ciphertext attribute" spaces. A functional encryption (FE) scheme for R consists of the following algorithms.

- $\mathsf{Setup}(1^\lambda, des) \to (\mathsf{pk}, \mathsf{msk})$: takes as input a security parameter 1^λ and a scheme description des (which usually describes the dimension n), and outputs a master public key pk and a master secret key msk.
- $\mathsf{KeyGen}(\boldsymbol{x}, \mathsf{msk}) \to \mathsf{sk}_{\boldsymbol{x}}$: takes as input a key attribute $\boldsymbol{x} \in \Sigma_{\mathsf{k}}$ and the master key msk. It outputs a private decryption key $\mathsf{sk}_{\boldsymbol{x}}$.

- $\mathsf{Encrypt}(\boldsymbol{y}, \mathsf{M}, \mathsf{pk}) \to C$: takes as input a ciphertext attribute $\boldsymbol{y} \in \Sigma_{\mathsf{e}}$, a message $\mathsf{M} \in \mathcal{M}$, and public key pk. It outputs a ciphertext C.
- $\mathsf{Decrypt}(C, \boldsymbol{y}, \mathsf{sk}_{\boldsymbol{x}}, \boldsymbol{x}) \to \mathsf{M}$: given a ciphertext C with its attribute $\boldsymbol{y}$ and the decryption key $\mathsf{sk}_{\boldsymbol{x}}$ with its attribute $\boldsymbol{x}$, it outputs a message M or $\perp$.

We require the standard correctness of decryption, that is, for all λ, all $(\mathsf{pk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$, all $\boldsymbol{x} \in \Sigma_{\mathsf{k}}$, all $\mathsf{sk}_{\boldsymbol{x}} \leftarrow \mathsf{KeyGen}(\boldsymbol{x}, \mathsf{msk})$, and all $\boldsymbol{y} \in \Sigma_{\mathsf{e}}$,

- If $R(\boldsymbol{x}, \boldsymbol{y}) = 1$, then $\mathsf{Decrypt}(\mathsf{Encrypt}(\boldsymbol{y}, \mathsf{M}, \mathsf{pk}), \mathsf{sk}_{\boldsymbol{x}}) = \mathsf{M}$.
- If $R(\boldsymbol{x}, \boldsymbol{y}) = 0$, $\mathsf{Decrypt}(\mathsf{Encrypt}(\boldsymbol{y}, \mathsf{M}, \mathsf{pk}), \mathsf{sk}_{\boldsymbol{x}}) = \perp$ with probability nearly 1.

Terminology and Variants. We refer to any encryption primitive A that can be casted as a functional encryption by specifying its corresponding function $R^{\mathsf{A}} : \Sigma_{\mathsf{k}}^{\mathsf{A}} \times \Sigma_{\mathsf{e}}^{\mathsf{A}} \to \{0, 1\}$. For a FE primitive A, we can define two variants:

- **Dual Variant**, denoted by $\mathsf{Dual}(\mathsf{A})$, is defined by setting $\Sigma_{\mathsf{k}}^{\mathsf{Dual}(\mathsf{A})} := \Sigma_{\mathsf{e}}^{\mathsf{A}}$ and $\Sigma_{\mathsf{e}}^{\mathsf{Dual}(\mathsf{A})} := \Sigma_{\mathsf{k}}^{\mathsf{A}}$ and $R^{\mathsf{A}}(\boldsymbol{x}, \boldsymbol{y}) = R^{\mathsf{Dual}(\mathsf{A})}(\boldsymbol{y}, \boldsymbol{x})$. In a dual variant, the roles of key and ciphertext attributes are swapped from those of its original primitive.
- **Negated Variant**, denoted by $\mathsf{Neg}(\mathsf{A})$, is defined by using the same domains as A and setting $R^{\mathsf{Neg}(\mathsf{A})}(\boldsymbol{x}, \boldsymbol{y}) = 1 \Leftrightarrow R^{\mathsf{A}}(\boldsymbol{x}, \boldsymbol{y}) = 0$. The condition is thus the opposite of the original primitive.

Security Definition. A FE scheme for a function $R : \Sigma_{\mathsf{k}} \times \Sigma_{\mathsf{e}} \to \{0, 1\}$ is fully secure if no PPT adversary $\mathcal{A}$ has non-negligible advantage in this game.

Setup. The challenger runs $\mathsf{Setup}(\mathrm{n})$ and hands the public key pk to $\mathcal{A}$.

Query Phase 1. The challenger answers private key queries for $\boldsymbol{x} \in \Sigma_{\mathsf{k}}$ by returning $\mathsf{sk}_{\boldsymbol{x}} \leftarrow \mathsf{KeyGen}(\boldsymbol{x}, \mathsf{msk})$.

Challenge. $\mathcal{A}$ submits messages $\mathsf{M}_0, \mathsf{M}_1$ and a target ciphertext attribute vector $\boldsymbol{y}^\star \in \Sigma_{\mathsf{e}}$ such that $R(\boldsymbol{x}, \boldsymbol{y}^\star) = 0$ for all key attributes $\boldsymbol{x}$ that have been queried so far. The challenger then flips a bit $\beta \xleftarrow{\$} \{0, 1\}$ and computes the challenge ciphertext $C^\star \leftarrow \mathsf{Encrypt}(\boldsymbol{y}, \mathsf{M}_\beta, \mathsf{pk})$ which is given to $\mathcal{A}$.

Query Phase 2. The adversary is allowed to make further private key queries $\boldsymbol{x} \in \Sigma_{\mathsf{k}}$ under the same restriction as above, *i.e.,* $R(\boldsymbol{x}, \boldsymbol{y}^\star) = 0$.

Guess. The adversary $\mathcal{A}$ outputs a guess $\beta' \in \{0, 1\}$ and wins if $\beta' = \beta$. In the game, $\mathcal{A}$'s advantage is typically defined as $\mathbf{Adv}_{\mathcal{A}}(\lambda) = |\Pr[\beta = \beta'] - \frac{1}{2}|$.

(Co-)Selective Security. We also consider the notion of selective security [11, 4], where $\mathcal{A}$ has to choose the challenge attribute $\boldsymbol{y}^\star$ before the setup phase, but can adaptively choose the key queries for $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_q$. One can consider its "dual" notion where $\mathcal{A}$ must output the q key queries for attribute vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_q$ before the setup phase, but can adaptively choose the target challenge attribute $\boldsymbol{y}^\star$. We refer to this scenario as the *co-selective* security model, which is useful in some applications such as revocation. By definition, both notions are incomparable in general and we do not know about their relation yet.

We shall show how one FE primitive can be obtained from another. The following useful lemma from [9] describes a sufficient criterion for implication.

Proposition 1 (Embedding Lemma [9]**).** *Consider encryption primitives* A, B *that can be casted as functional encryption for functions* $R^{\mathsf{A}}, R^{\mathsf{B}}$*, respectively. Suppose there exists efficient injective mappings* $f_{\mathsf{k}} : \Sigma_{\mathsf{k}}^{\mathsf{A}} \to \Sigma_{\mathsf{k}}^{\mathsf{B}}$ *and* $f_{\mathsf{e}} : \Sigma_{\mathsf{e}}^{\mathsf{A}} \to \Sigma_{\mathsf{e}}^{\mathsf{B}}$ *such that* $R^{\mathsf{B}}(f_{\mathsf{k}}(\boldsymbol{x}), f_{\mathsf{e}}(\boldsymbol{y})) = 1 \Leftrightarrow R^{\mathsf{A}}(\boldsymbol{x}, \boldsymbol{y}) = 1$*. Let* Π_{B} *be a construction for primitive* B*. We then construct* Π_{A} *for primitive* A *from* Π_{B} *by applying mappings* $f_{\mathsf{k}}, f_{\mathsf{e}}$ *to all key attributes and ciphertext attributes, respectively. More precisely, we use exactly the same setup algorithm and define key generation and encryption procedures as* $\Pi_{\mathsf{A}}.\mathsf{KeyGen}(x, \mathsf{msk}) := \Pi_{\mathsf{B}}.\mathsf{KeyGen}(f_{\mathsf{k}}(x), \mathsf{msk})$ *and* $\Pi_{\mathsf{A}}.\mathsf{Encrypt}(y, M, \mathsf{pk}) := \Pi_{\mathsf{B}}.\mathsf{Encrypt}(f_{\mathsf{e}}(y), M, \mathsf{pk})$*, respectively. Then, if* Π_{B} *is secure, so is* Π_{A}*. This holds for adaptive, selective, co-selective security models. We denote this primitive implication by* $\mathsf{B} \xrightarrow{f_{\mathsf{k}}, f_{\mathsf{e}}} \mathsf{A}$*.*

We immediately obtain the next corollary stating that the implication applies to the negated (resp. dual) variant with the same (resp. swapped) mappings.

Corollary 1. $\mathsf{B} \xrightarrow{f_{\mathsf{k}}, f_{\mathsf{e}}} \mathsf{A}$ *implies* $\mathsf{Dual}(\mathsf{B}) \xrightarrow{f_{\mathsf{e}}, f_{\mathsf{k}}} \mathsf{Dual}(\mathsf{A})$ *and* $\mathsf{Neg}(\mathsf{B}) \xrightarrow{f_{\mathsf{k}}, f_{\mathsf{e}}} \mathsf{Neg}(\mathsf{A})$*.*

2.2 Complexity Assumptions in Bilinear Groups

We consider groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order p with an efficiently computable map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ such that $e(g^a, h^b) = e(g, h)^{ab}$ for any $(g, h) \in \mathbb{G} \times \mathbb{G}$ and $a, b \in \mathbb{Z}$ and $e(g, h) \neq 1_{\mathbb{G}_T}$ whenever $g, h \neq 1_{\mathbb{G}}$. In these groups, we assume the hardness of the Decision Bilinear Diffie-Hellman and Decision Linear [5] problems.

Definition 1. *The* **Decision Bilinear Diffie-Hellman Problem** *(DBDH) in* $(\mathbb{G}, \mathbb{G}_T)$ *is, given elements* $(g, g^{\theta_1}, g^{\theta_2}, g^{\theta_3}, \eta) \in \mathbb{G}^4 \times \mathbb{G}_T$ *with* $\theta_1, \theta_2, \theta_3 \xleftarrow{\$} \mathbb{Z}_p$*, to decide whether* $\eta = e(g, g)^{\theta_1 \theta_2 \theta_3}$ *or* $\eta \in_R \mathbb{G}_T$*.*

Definition 2. *The* **Decision Linear Problem** *(DLIN) in* $\mathbb{G}$ *consists in, given a tuple* $(g, f, \nu, g^{\theta_1}, f^{\theta_2}, \eta) \in \mathbb{G}^6$ *with* $\theta_1, \theta_2 \xleftarrow{\$} \mathbb{Z}_p$ *and* $f, g, \nu \xleftarrow{\$} \mathbb{G}$*, deciding whether* $\eta = \nu^{\theta_1 + \theta_2}$ *or* $\nu \in_R \mathbb{G}$*.*

3 Functional Encryption Instances and Their Implications

3.1 Inner Product Encryption and Its Consequences

We underline the power of IPE by showing its implications in this section. Each primitive is defined by describing the corresponding boolean function R. We then show how to construct one primitive from another by explicitly describing attribute mappings. In this way, correctness and security are consequences of the embedding lemma. Basically, the approach follows exactly the same way as [18]. A new contribution is that we also consider the negated variant of primitives, which will be useful for non-zero polynomial evaluation and revocation schemes. The implication for negated variants follows from Corollary 1.

Inner Product. An inner product encryption (IPE) scheme over $\mathbb{Z}_p^n$, for some prime p, is defined as follows. Both attribute domains are $\Sigma_{\mathsf{k}}^{\mathsf{IPE}_n} = \Sigma_{\mathsf{e}}^{\mathsf{IPE}_n} = \mathbb{Z}_p^n$.

We consider two distinct IPE modes here. The first one is zero-mode IPE where $R^{\mathsf{ZIPE}_n}(\vec{X}, \vec{Y}) = 1$ iff $\vec{X} \cdot \vec{Y} = 0$. The other one is its negated primitive, which we call the non-zero-mode IPE, where $R^{\mathsf{NIPE}_n}(\vec{X}, \vec{Y}) = 1$ iff $\vec{X} \cdot \vec{Y} \neq 0$.

Polynomial Evaluation. Functional encryption for the zero evaluation of polynomials of degree $\leq n$ is defined as follows. The ciphertext and key attribute domains are defined as $\Sigma_{\mathsf{e}}^{\mathsf{ZPoly}_{\leq n}} = \mathbb{Z}_p$ and $\Sigma_{\mathsf{k}}^{\mathsf{ZPoly}_{\leq n}} = \{P \in \mathbb{Z}_p[x] \mid \deg(P) \leq n\}$, respectively. The relation is defined by $R^{\mathsf{ZPoly}_{\leq n}}(P, x) = 1$ iff $P(x) = 0$. The non-zero evaluation mode can be defined as its negated primitive $\mathsf{Neg}(\mathsf{ZPoly}_{\leq n})$.

Given an IPE scheme over $\mathbb{Z}_p^{n+1}$, one obtain a functional encryption system for polynomial evaluation via the following embedding. For the key attribute, we map the polynomial $P[X] = \rho_0 + \rho_1 X + \cdots + \rho_n X^n$ to $\vec{X}_p = (\rho_0, \ldots, \rho_n)$. Regarding ciphertext attributes, each element $w \in \mathbb{Z}_p$ is mapped onto a vector $\vec{Y}_w = (1, w, w^2, \ldots, w^n)$. Correctness and security hold since $P(w) = 0$ whenever $\vec{X}_p \cdot \vec{Y}_w = 0$. The non-zero evaluation case can be analogously derived from the non-zero-mode IPE using the same mappings, due to Corollary 1.

We can also consider other variants such as a scheme that supports multivariate polynomials and a dual variant, where the key attribute corresponds to a fixed point and the ciphertext attribute corresponds to a polynomial, as in [18].

OR, AND, DNF, CNF Formulae. We now consider a FE scheme for some boolean formulae that evaluate disjunctions, conjunctions, and their extensions to disjunctive or conjunctive normal forms. As an example, a functional encryption scheme for boolean formula $R^{\mathsf{OR}_{\leq n}} : \mathbb{Z}_N^{\leq n} \times \mathbb{Z}_N \to \{0, 1\}$ can be defined by $R^{\mathsf{OR}_{\leq n}}((I_1, \ldots, I_k), z) \mapsto 1$ (for $k \leq n$) iff $(z = I_1)$ or $\cdots$ or $(z = I_k)$. This can be obtained from a functional encryption for the zero evaluation of a univariate polynomial of degree smaller than n by generating a private key for $f_{\mathsf{OR}, I_1, \ldots, I_k}(z) = (z - I_1) \cdots (z - I_k)$, and letting senders encrypting to z.

Other fundamental cases can be considered similarly as in [18] and are shown below. In [18] only non-negated policies (the first three cases below and their extensions) were considered. Schemes supporting negated policies (the latter three cases below and their extensions) are introduced in this paper. The negated case can be implemented by IPE for non-zero evaluation. One can combine these cases to obtain DNF, CNF formulae. Below, $r \xleftarrow{\$} \mathbb{Z}_p$ is chosen by KeyGen.[4]

Policy	Implementation
$(z = I_1)$ or $(z = I_2)$	$f_{\mathsf{OR}, I_1, I_2}(z) = (z - I_1)(z - I_2) = 0$
$(z_1 = I_1)$ or $(z_2 = I_2)$	$f_{\overline{\mathsf{OR}}, I_1, I_2}(z_1, z_2) = (z_1 - I_1)(z_2 - I_2) = 0$
$(z_1 = I_1)$ and $(z_2 = I_2)$	$f_{\mathsf{AND}, I_1, I_2}(z_1, z_2) = (z_1 - I_1)r + (z_2 - I_2) = 0$
$(z_1 \neq I_1)$ or $(z_2 \neq I_2)$	$f_{\mathsf{NOR}, I_1, I_2}(z_1, z_2) = (z_1 - I_1)r + (z_2 - I_2) \neq 0$
$(z \neq I_1)$ and $(z \neq I_2)$	$f_{\mathsf{NAND}, I_1, I_2}(z) = (z - I_1)(z - I_2) \neq 0$
$(z_1 \neq I_1)$ and $(z_2 \neq I_2)$	$f_{\overline{\mathsf{NAND}}, I_1, I_2}(z_1, z_2) = (z_1 - I_1)(z_2 - I_2) \neq 0$

ID-based Broadcast Encryption and Revocation. Let $\mathcal{I}$ be an identity space. An ID-based broadcast encryption scheme (IBBE) for maximum n re-

[4] As noted in [18], the AND (and NOR) case will not work in the adaptive security model since the information on r leaks.

ceivers per ciphertext is a functional encryption for $R^{\mathsf{IBBE}_{\leq n}} : \mathcal{I} \times 2^{\mathcal{I}} \to \{0,1\}$ defined by $R^{\mathsf{IBBE}_{\leq n}} : (\mathsf{ID}, S) \mapsto 1$ iff $\mathsf{ID} \in S$. An IBBE system can be constructed by a functional encryption for $R^{\mathsf{Dual}(\mathsf{OR}_{\leq n})}$. To encrypt a message for the receiver set $S = \{\mathsf{ID}_1, \ldots, \mathsf{ID}_k\}$, one encrypts using the policy $(z = \mathsf{ID}_1)$ or $\cdots$ or $(z = \mathsf{ID}_k)$.

Likewise, identity-based revocation (IBR) [19] for at most n revocations per ciphertext can be casted as a negated IBBE, *i.e.,* $R^{\mathsf{IBR}_{\leq n}} : (\mathsf{ID}, R) \mapsto 1$ iff $\mathsf{ID} \notin R$.

3.2 Spatial Encryption

We now recall the concept of spatial encryption [9]. For a $n \times d$ matrix M of which elements are in $\mathbb{Z}_p$ and a vector $\vec{c} \in \mathbb{Z}_p^n$, we define its corresponding affine space as $\mathrm{Aff}(M, \vec{c}) = \{M\vec{w} + \vec{c} \mid \vec{w} \in \mathbb{Z}_p^d\}$. Let $\mathcal{V}_n \subseteq 2^{(\mathbb{Z}_p^n)}$ be the collection of all affine spaces inside $\mathbb{Z}_p^n$. That is, $\mathcal{V}_n = \{\mathrm{Aff}(M, \vec{c}) \mid M \in \mathbb{M}_{n \times d}, c \in \mathbb{Z}_p^n, d \leq n\}$, where $\mathbb{M}_{n \times d}$ is the set of all $n \times d$ matrices in $\mathbb{Z}_p$.

A spatial encryption in $\mathbb{Z}_p^n$ is a functional encryption for a relation $R^{\mathsf{Spatial}} : \mathcal{V}_n \times \mathbb{Z}_p^n \to \{0,1\}$ defined by $R^{\mathsf{Spatial}} : (V, \vec{y}) \mapsto 1$ iff $\vec{y} \in V$.

The notion of spatial encryption was motivated by Boneh and Hamburg [9]. It has many applications as it notably implies broadcast HIBE and multi-authority schemes. Nevertheless, its connection to inner-product encryption has not been investigated so far. In section 4.1, we prove that spatial encryption implies inner product encryption by providing a simple attribute mapping.

As a result of independent interest, we also consider the negated spatial encryption primitive (namely, FE that is defined by $R^{\mathsf{Neg}(\mathsf{Spatial})} : (V, \vec{y}) \mapsto 1$ iff $\vec{y} \notin V$) and provide a construction in section 5.1. From this scheme and Corollary 1 together with our implication result of zero-mode IPE from spatial encryption, we then obtain a non-zero-mode IPE construction.

4 Functional Encryption for Zero Inner-Product

4.1 Warm-up: Selectively Secure Zero IPE from Spatial Encryption

We first show that spatial encryption implies zero IPE. For the key attribute, we map $\vec{X} = (x_1, \ldots, x_n)^\top \in \mathbb{Z}_p^n$ to an $(n-1)$-dimension affine space $V_{\vec{X}} = \mathrm{Aff}(M_{\vec{X}}, \vec{0}_n) = \{M_{\vec{X}}\vec{w} + \vec{0}_n \mid \vec{w} \in \mathbb{Z}_p^{n-1}\}$ with the matrix $M_{\vec{X}} \in \mathbb{Z}_p^{n \times (n-1)}$

$$M_{\vec{X}} = \begin{pmatrix} -\frac{x_2}{x_1}, -\frac{x_3}{x_1}, \cdots, -\frac{x_n}{x_1} \\ I_{n-1} \end{pmatrix}. \tag{3}$$

For any $\vec{Y} = (y_1, \ldots, y_n)^\top \in \mathbb{Z}_p^n$, we then have $\vec{X} \cdot \vec{Y} = 0 \Leftrightarrow \vec{Y} \in V_{\vec{X}}$ since $\vec{X} \cdot \vec{Y} = 0 \Leftrightarrow y_1 = y_2 \cdot (-\frac{x_2}{x_1}) + \cdots + y_n \cdot (-\frac{x_n}{x_1}) \Leftrightarrow \vec{Y} = M_{\vec{X}} \cdot (y_2, \ldots, y_n)^\top \Leftrightarrow \vec{Y} \in V_{\vec{X}}$. By the embedding lemma, we can therefore conclude its implication.

In [9], Boneh and Hamburg described a selectively secure construction of spatial encryption that achieves constant-size ciphertexts (by generalizing the Boneh-Boyen-Goh HIBE [6]). We thus immediately obtain a selectively secure zero IPE scheme with constant-size ciphertext as shown below.

We give some notations here. For a vector $\vec{a} = (a_1, \ldots, a_n)^\top \in \mathbb{Z}_p^n$, we write $g^{\vec{a}}$ to denote $(g^{a_1}, \ldots, g^{a_n})^\top$. Given $g^{\vec{a}}, \vec{z}$, one can easily compute $(g^{\vec{a}})^{\vec{z}} := g^{\langle \vec{a}, \vec{z} \rangle}$, where $\langle \vec{a}, \vec{z} \rangle$ denotes the inner product $\vec{a} \cdot \vec{z} = \vec{a}^\top \vec{z}$.

Construction 2. (Selectively secure zero IPE)

▶ $\mathsf{Setup}(1^\lambda, n)$: chooses bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^\lambda$ with a generator $g \xleftarrow{\$} \mathbb{G}$. It chooses $\alpha, \alpha_0, \ldots, \alpha_n \xleftarrow{\$} \mathbb{Z}_p$. Let $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$. The public key is $\mathsf{pk} = \big(g, g^{\alpha_0}, \vec{H} = g^{\vec{\alpha}}, Z = e(g,g)^\alpha\big)$. The master key is $\mathsf{msk} = g^\alpha$.

▶ $\mathsf{KeyGen}(\vec{X}, \mathsf{msk}, \mathsf{pk})$: chooses $t \xleftarrow{\$} \mathbb{Z}_p$ and parses $\vec{X}$ as $(x_1, \ldots, x_n)$ and returns $\perp$ if $x_1 = 0$. It outputs the private key as $\mathsf{sk}_{\vec{X}} = (D_0, D_1, K_2, \ldots, K_n)$ where

$$D_0 = g^t, \qquad D_1 = g^{\alpha + \alpha_0 t}, \qquad \{K_i = (g^{-\alpha_1 \frac{x_i}{x_1}} g^{\alpha_i})^t\}_{i=2,\ldots,n}.$$

▶ $\mathsf{Encrypt}(\vec{Y}, \mathsf{pk})$: the encryption algorithm first picks $s \xleftarrow{\$} \mathbb{Z}_p$. It parses $\vec{Y}$ as $(y_1, \ldots, y_n)$ and computes the ciphertext as

$$E_0 = \mathsf{M} \cdot e(g,g)^{\alpha s}, \qquad E_1 = (g^{\alpha_0} g^{\langle \vec{\alpha}, \vec{Y} \rangle})^s, \qquad E_2 = g^s.$$

▶ $\mathsf{Decrypt}(C, \vec{Y}, \mathsf{sk}_{\vec{X}}, \vec{X}, \mathsf{pk})$: to decrypt, the algorithm computes the message blinding factor as $\frac{e(D_1 K_2^{y_2} \cdots K_n^{y_n}, E_2)}{e(E_1, D_0)} = e(g,g)^{\alpha s}$.

The selective security of this scheme is a consequence of a result given in [9].

Theorem 1. *Construction 2 is selectively secure under the n-Decisional Bilinear Diffie-Hellman Exponent assumption* (see [9] for a description of the latter).

4.2 Adaptively Secure Zero IPE under Simple Assumptions

We extend the above selectively secure zero IPE to acquire adaptive security by applying the Waters' dual system method [27]. However, we have to use our "n-equation technique" as opposed to 2-equation technique used for IBE in [27]. The reason is that we have to deal with the difficulties arising from the richer structure of IPE and the aggregation of ciphertexts into a constant number of elements, analogously to what we described in section 1.

The scheme basically goes as follows. A ciphertext contains a random tag tagc in the element E_1 while each key contains $n-1$ tags (tagk_i for each K_i element) that are aggregated into $\mathsf{tagk} = \sum_{i=2}^n \mathsf{tagk}_i y_i$ upon decryption of a ciphertext intended for $\vec{Y}$. The receiver can decrypt if $\mathsf{tagk} \neq \mathsf{tagc}$ (and $\vec{X} \cdot \vec{Y} = 0$).

Construction 3. (Adaptively secure zero IPE)

▶ $\mathsf{Setup}(1^\lambda, n)$: chooses bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^\lambda$. It then picks generators $g, v, v_1, v_2 \xleftarrow{\$} \mathbb{G}$ and chooses $\alpha, \alpha_0, \alpha_1, \ldots, \alpha_n, a_1, a_2, b \xleftarrow{\$} \mathbb{Z}_p$. Let $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$ and $\vec{H} = (h_1, \ldots, h_n) = g^{\vec{\alpha}}$. The public key consists of

$$\mathsf{pk} = \left(\begin{array}{c} g,\ w = g^{\alpha_0},\ Z = e(g,g)^{\alpha \cdot a_1 \cdot b},\ \vec{H} = g^{\vec{\alpha}},\ A_1 = g^{a_1},\ A_2 = g^{a_2}, B = g^b, \\ B_1 = g^{b \cdot a_1},\ B_2 = g^{b \cdot a_2},\ \tau_1 = v \cdot v_1^{a_1},\ \tau_2 = v \cdot v_2^{a_2},\ T_1 = \tau_1^b,\ T_2 = \tau_2^b \end{array} \right)$$

The master key is defined to be $\mathsf{msk} = (g^{\alpha}, g^{\alpha a_1}, v, v_1, v_2)$.

▶ $\mathsf{Keygen}(\vec{X}, \mathsf{msk}, \mathsf{pk})$: parses $\vec{X}$ as $(x_1, \ldots, x_n)$ and returns $\perp$ if $x_1 = 0$. Otherwise, it picks $r_1, r_2 \xleftarrow{\$} \mathbb{Z}_p$, $z_1, z_2 \xleftarrow{\$} \mathbb{Z}_p$, $\mathsf{tagk}_2, \ldots, \mathsf{tagk}_n \xleftarrow{\$} \mathbb{Z}_p$, sets $r = r_1 + r_2$ and generates $\mathsf{sk}_{\vec{X}} = (D_1, \ldots, D_7, K_2, \ldots, K_n, \mathsf{tagk}_2, \ldots, \mathsf{tagk}_n)$ by computing

$$\mathsf{sk}_{\mathsf{core}} = \left\{K_i = \left(g^{-\alpha_1 \frac{x_i}{x_1}} \cdot g^{\alpha_i} \cdot g^{\alpha_0 \cdot \mathsf{tagk}_i}\right)^{r_1}\right\}_{i=2,\ldots,n},$$

$$\mathsf{sk}_{\mathsf{adapt}} = \begin{pmatrix} D_1 = g^{\alpha a_1} \cdot v^r, & D_2 = g^{-\alpha} \cdot v_1^r \cdot g^{z_1}, & D_3 = B^{-z_1}, & D_4 = v_2^r \cdot g^{z_2}, \\ D_5 = B^{-z_2}, & D_6 = B^{r_2}, & D_7 = g^{r_1} & \end{pmatrix}.$$

▶ $\mathsf{Encrypt}(\vec{Y}, \mathsf{M}, \mathsf{pk})$: to encrypt $\mathsf{M} \in \mathbb{G}_T$ under $\vec{Y} = (y_1, \ldots, y_n) \in (\mathbb{Z}_p)^n$, pick $s_1, s_2, t, \mathsf{tagc} \xleftarrow{\$} \mathbb{Z}_p$ and compute $C = (C_1, \ldots, C_7, E_0, E_1, E_2, \mathsf{tagc})$ where

$$C_{\mathsf{core}} = \left(E_0 = \mathsf{M} \cdot Z^{s_2}, \quad E_1 = (g^{\alpha_0 \cdot \mathsf{tagc}} \cdot g^{\langle \vec{\alpha}, \vec{Y} \rangle})^t, \quad E_2 = g^t\right),$$

$$C_{\mathsf{adapt}} = \begin{pmatrix} C_1 = B^{s_1 + s_2}, & C_2 = B_1^{s_1}, & C_3 = A_1^{s_1}, & C_4 = B_2^{s_2}, \\ C_5 = A_2^{s_2}, & C_6 = \tau_1^{s_1} \cdot \tau_2^{s_2}, & C_7 = T_1^{s_1} \cdot T_2^{s_2} \cdot w^{-t} & \end{pmatrix}.$$

▶ $\mathsf{Decrypt}(C, \vec{Y}, \mathsf{sk}_{\vec{X}}, \vec{X}, \mathsf{pk})$: computes $\mathsf{tagk} = \mathsf{tagk}_2 y_2 + \cdots + \mathsf{tagk}_n y_n$ and then $W_1 = \prod_{j=1}^{5} e(C_j, D_j) \cdot (\prod_{j=6}^{7} e(C_j, D_j))^{-1} = e(g, g)^{\alpha \cdot a_1 \cdot b \cdot s_2} \cdot e(g, w)^{r_1 t}$, as well as $W_2 = \left(\frac{e(K_2^{y_2} \cdots K_n^{y_n}, E_2)}{e(E_1, D_7)}\right)^{\frac{1}{\mathsf{tagk} - \mathsf{tagc}}} = e(g, w)^{r_1 t}$. It finally recovers the plaintext as $\mathsf{M} = E_0 / Z^{s_2} = E_0 / e(g, g)^{\alpha \cdot a_1 \cdot b \cdot s_2} \leftarrow E_0 \cdot W_2 \cdot W_1^{-1}$.

The correctness of W_2 is shown in appendix A.1, while the rest follows from [27]. As we can see, ciphertexts have the same size as in the IBE scheme of [27], no matter how large the vector $\vec{Y}$ is. Also, decryption entails a constant number of pairing evaluations (whereas ciphertexts cost $O(n)$ pairings to decrypt in [18]).

Theorem 2. *Construction 3 is adaptively secure under the DLIN and DBDH assumptions.*

Proof. The proof uses the dual system methodology similar to [27], which involves ciphertexts and private keys that can be normal or semi-functional.

- Semi-functional ciphertexts are generated by first computing a normal ciphertext $(C'_0, C'_1, \ldots, C'_7, E'_1, E'_7, \mathsf{tagc}')$ and then choosing $\chi \xleftarrow{\$} \mathbb{Z}_p$ before replacing (C'_4, C'_5, C'_6, C'_7), respectively, by

$$C_4 = C'_4 \cdot g^{b a_2 \chi}, \quad C_5 = C'_5 \cdot g^{a_2 \chi}, \quad C_6 = C'_6 \cdot v_2^{a_2 \chi}, \quad C_7 = C'_7 \cdot v_2^{a_2 b \chi}. \quad (4)$$

- From a normal key $(D'_1, \ldots, D'_7, K'_2, \ldots, K'_n, \mathsf{tagk}'_2, \ldots, \mathsf{tagk}'_n)$, semi-functional keys are obtained by choosing $\gamma \xleftarrow{\$} \mathbb{Z}_p$ and replacing (D'_1, D'_2, D'_4) by

$$D_1 = D'_1 \cdot g^{-a_1 a_2 \gamma}, \qquad D_2 = D'_2 \cdot g^{a_2 \gamma}, \qquad D_4 = D'_4 \cdot g^{a_1 \gamma}. \quad (5)$$

The proof proceeds with a game sequence starting from Game_{Real}, which is the actual attack game. The following games are defined below.

Game_0 is the real attack game but the challenge ciphertext is semi-functional.
Game_k (for $1 \le k \le q$) is identical to Game_0 except that the first i private key generation queries are answered by returning a semi-functional key.
Game_{q+1} is as Game q but the challenge ciphertext is a semi-functional encryption of a random element of $\mathbb{G}_T$ instead of the actual plaintext.

We prove the indistinguishability between two consecutive games under some assumptions. The sequence ends in Game_{q+1}, where the challenge ciphertext is independent of the challenger's bit β, hence any adversary has no advantage. □

The indistinguishability of Game_{Real} and Game_0 as well as that of Game_q and Game_{q+1} can be proved exactly in the same way as in [27] and the details are given in the full version of the paper.

Lemma 1. *If DLIN is hard,* Game_0 *is indistinguishable from* Game_{Real}*.*

Lemma 2. *For any* $1 \le k \le q$*, if an adversary* $\mathcal{A}$ *can distinguish* Game_k *from* Game_{k-1}*, we can build a distinguisher for the DLIN problem.*

This lemma is the most non-trivial part in the theorem. The main issue is that, in order to enable adaptive security, the reduction must be done in such a way that the simulator $\mathcal{B}$ can create semi-functional keys for any vector $\vec{X}$, including those for which $\vec{X} \cdot \vec{Y}^\star = 0$. However, the crucial point is that we must prevent $\mathcal{B}$ from directly deciding whether the k^{th} queried private key is normal or semi-functional by generating a semi-functional ciphertext for itself. Indeed, if this were possible, the reduction from $\mathcal{A}$ would not be established.

To resolve this, we use a secret exponent vector $\vec{\zeta} \in \mathbb{Z}_p^n$ and embed the DLIN instance so that $\mathcal{B}$ can simulate only the key at k^{th} query for $\vec{X}$ with tags $(\mathsf{tagk}_2, \ldots, \mathsf{tagk}_n)$ and the challenge ciphertext for $\vec{Y}^\star$ with $\mathsf{tagc}^\star$ that obey the relation: $(\mathsf{tagk}_2, \ldots, \mathsf{tagk}_n, \mathsf{tagc}^\star)^\top = -M_{\vec{X},\vec{Y}^\star}\vec{\zeta}$, where $M_{\vec{X},\vec{Y}}$ is the $n \times n$ matrix defined in Eq.(2). We thereby conceptually use the n-equation technique here. A particular consequence is that if we have $\vec{X} \cdot \vec{Y}^\star = 0$ then it holds that

$$\mathsf{tagk} = \sum_{i=2}^{n} \mathsf{tagk}_i y_i^\star = \zeta_1 \sum_{i=2}^{n} \frac{x_i}{x_1} y_i^\star - \sum_{i=2}^{n} \zeta_i y_i^\star = \zeta_1 \cdot (-y_1^\star) - \sum_{i=2}^{n} \zeta_i y_i^\star = \mathsf{tagc}^\star,$$

which is the exact condition that hampers the decryption, thus $\mathcal{B}$ cannot test by itself, as desired. We are now ready to describe the proof of Lemma 2.

Proof. The distinguisher $\mathcal{B}$ receives $(g, f, \nu, g^{\theta_1}, f^{\theta_2}, \eta)$ and decides if $\eta = \nu^{\theta_1+\theta_2}$.

Setup. Algorithm $\mathcal{B}$ picks $\alpha, a_1, a_2, \delta_{v_1}, \delta_{v_2} \xleftarrow{\$} \mathbb{Z}_p$ and sets $g = g$, $Z = e(f,g)^{\alpha a_1}$,

$$\begin{array}{llll} A_1 = g^{a_1}, & A_2 = g^{a_2}, & B = g^b = f, & v_1 = \nu^{a_2} \cdot g^{\delta_{v_1}} \\ B_1 = g^{ba_1} = f^{a_1}, & B_2 = g^{ba_2} = f^{a_2}, & v = \nu^{-a_1 a_2}, & v_2 = \nu^{a_1} \cdot g^{\delta_{v_2}}, \\ \tau_1 = v v_1^{a_1} = g^{\delta_{v_1} a_1}, & \tau_2 = v v_2^{a_2} = g^{\delta_{v_2} a_2}, & \tau_1^b = f^{\delta_{v_1} a_1}, & \tau_2^b = f^{\delta_{v_2} a_2}. \end{array}$$

Next, $\mathcal{B}$ chooses $\delta_w \xleftarrow{\$} \mathbb{Z}_p, \vec{\zeta} = (\zeta_1, \ldots, \zeta_n) \xleftarrow{\$} \mathbb{Z}_p^n$, $\vec{\delta} = (\delta_1, \ldots, \delta_n) \xleftarrow{\$} \mathbb{Z}_p^n$, then defines $w = g^{\alpha_0} = f \cdot g^{\delta_w}$, and $h_i = g^{\alpha_i} = f^{\zeta_i} \cdot g^{\delta_i}$ for $i = 1, \ldots, n$. Note that, as

in the proof of lemma 2 in [27] , $\mathcal{B}$ knows $\mathsf{msk} = (g^{\alpha}, g^{\alpha a_1}, v, v_1, v_2)$.

Key Queries. When $\mathcal{A}$ makes the j^{th} private key query, $\mathcal{B}$ does as follows.
[**Case** $j > k$] It generates a normal key, using the master secret key msk.
[**Case** $j < k$] It creates a semi-functional key, which it can do using $g^{a_1 a_2}$.
[**Case** $j = k$] It defines $\mathsf{tagk}_2, \ldots, \mathsf{tagk}_n$ as $\mathsf{tagk}_i = \zeta_1 \cdot \frac{x_i}{x_1} - \zeta_i$ for $i = 2, \ldots, n$, which implies that $(h_1^{-x_i/x_1} \cdot h_i \cdot w^{\mathsf{tagk}_i}) = g^{-\delta_1(x_i/x_1)+\delta_i+\delta_w \mathsf{tagk}_i}$, for $i = 2, \ldots, n$. Using these tags, it generates a normal private key $(D'_1, \ldots, D'_7, K'_2, \ldots, K'_n)$ using random exponents $r'_1, r'_2, z'_1, z'_2 \xleftarrow{\$} \mathbb{Z}_p$. Then, it sets

$$
\begin{aligned}
&D_1 = D'_1 \cdot \eta^{-a_1 a_2}, && D_2 = D'_2 \cdot \eta^{a_2} \cdot (g^{\theta_1})^{\delta_{v_1}}, && D_3 = D'_3 \cdot (f^{\theta_2})^{\delta_{v_1}},\\
&D_4 = D'_4 \cdot \eta^{a_1} \cdot (g^{\theta_1})^{\delta_{v_2}}, && D_5 = D'_5 \cdot (f^{\theta_2})^{\delta_{v_2}}, && D_6 = D'_6 \cdot f^{\theta_2},
\end{aligned}
$$

as well as $D_7 = D'_7 \cdot (g^{\theta_1})$ and $K_i = K'_i \cdot (g^{\theta_1})^{-\delta_1(x_i/x_1)+\delta_i+\delta_w \mathsf{tagk}_i}$ for $i = 2, \ldots, n$.

If $\eta = \nu^{\theta_1+\theta_2}$, $\mathsf{sk}_{\vec{X}} = (D_1, \ldots, D_7, K_2, \ldots, K_n, \mathsf{tagk}_2, \ldots, \mathsf{tagk}_n)$ is easily seen to form a normal key where $r_1 = r'_1 + \theta_1$, $r_2 = r'_2 + \theta_2$, $z_1 = z'_1 - \delta_{v_1}\theta_2$, $z_2 = z'_2 - \delta_{v_2}\theta_2$ are the underlying random exponents. If $\eta \in_R \mathbb{G}$, it can be written $\eta = \nu^{\theta_1+\theta_2} \cdot g^{\gamma}$ for some $\gamma \in_R \mathbb{Z}_p$, so that $\mathsf{sk}_{\vec{X}}$ is distributed as a semi-functional key. We note that $\mathsf{tagk}_2, \ldots, \mathsf{tagk}_n$ are independent and uniformly distributed since $\zeta_1, \ldots, \zeta_n$ (which are the solutions of a system of $n-1$ equations with n unknowns) are uniformly random and perfectly hidden from $\mathcal{A}$'s view.

Challenge. $\mathcal{A}$ outputs $M_0, M_1 \in \mathbb{G}_T$ along with a vector $\vec{Y}^{\star} = (y_1^{\star}, \ldots, y_n^{\star})$. $\mathcal{B}$ flips a coin $\beta \xleftarrow{\$} \{0,1\}$ and computes the tag $\mathsf{tagc}^{\star} = -\langle \vec{Y}^{\star}, \vec{\zeta} \rangle$ for which $\mathcal{B}$ will be able to prepare the semi-functional ciphertext. To this end, $\mathcal{B}$ first computes a normal encryption $(C'_0, C'_1, \ldots, C'_7, E'_1, E'_2, \mathsf{tagc}^{\star})$ of M_{β} using exponents s'_1, s'_2, t'. It then chooses $\chi \xleftarrow{\$} \mathbb{Z}_p$ and computes

$$
\begin{aligned}
&C_4 = C'_4 \cdot f^{a_2 \cdot \chi}, && C_5 = C'_5 \cdot g^{a_2 \cdot \chi}, && C_7 = C'_7 \cdot \nu^{-\delta_w \cdot a_1 \cdot a_2 \cdot \chi} \cdot f^{\delta_{v_2} \cdot a_2 \cdot \chi},\\
&C_6 = C'_6 \cdot v_2^{a_2 \cdot \chi}, && E_2 = E'_2 \cdot \nu^{a_1 \cdot a_2 \cdot \chi}, && E_1 = E'_1 \cdot (\nu^{\delta_w \cdot \mathsf{tagc}^{\star} + \langle \vec{Y}^{\star}, \vec{\delta} \rangle})^{a_1 \cdot a_2 \cdot \chi}.
\end{aligned}
$$

We claim that $(C'_0, C'_1, C'_2, C'_3, C_4, C_5, C_6, C_7, E_1, E_2, \mathsf{tagc}^{\star})$ is a semi-functional ciphertext with underlying exponents $\chi, s_1 = s'_1, s_2 = s'_2$ and $t = t' + \log_g(\nu) a_1 a_2 \chi$. To prove this, we observe that

$$
\begin{aligned}
C_7 &= T_1^{s_1} \cdot T_2^{s_2} \cdot w^{-t} \cdot v_2^{a_2 b \chi} = T_1^{s_1} \cdot T_2^{s_2} \cdot w^{-t' - \log_g(\nu) a_1 a_2 \chi} \cdot (\nu^{a_1} \cdot g^{\delta_{v_2}})^{a_2 b \chi}\\
&= T_1^{s_1} \cdot T_2^{s_2} \cdot w^{-t'} \cdot (f \cdot g^{\delta_w})^{-\log_g(\nu) a_1 a_2 \chi} \cdot (\nu^{a_1} \cdot g^{\delta_{v_2}})^{a_2 b \chi}\\
&= C'_7 \cdot \nu^{-\delta_w a_1 a_2 \chi} \cdot f^{\delta_{v_2} a_2 \chi},
\end{aligned}
$$

where the unknown term in $v_2^{a_2 b \chi}$ is canceled out by w^{-t}. Also,

$$
\begin{aligned}
E_1 &= E'_1 \cdot \left(h_1^{y_1^{\star}} \cdots h_n^{y_n^{\star}} \cdot w^{\mathsf{tagc}^{\star}}\right)^{\log_g(\nu) a_1 a_2 \chi}\\
&= E'_1 \cdot \left((f^{\zeta_1} g^{\delta_1})^{y_1^{\star}} \cdots (f^{\zeta_n} g^{\delta_n})^{y_n^{\star}} \cdot (f g^{\delta_w})^{-\langle \vec{Y}^{\star}, \vec{\zeta} \rangle}\right)^{\log_g(\nu) a_1 a_2 \chi}\\
&= E'_1 \cdot (\nu^{\langle \vec{Y}^{\star}, \vec{\delta} \rangle + \delta_w \cdot \mathsf{tagc}^{\star}})^{a_1 a_2 \chi},
\end{aligned}
$$

where the unknown $f^{\log_g(\nu)}$ vanishes due to our definition of $\mathsf{tagc}^\star$. It then remains to show that $\mathsf{tagc}^\star, \mathsf{tagk}_2, \ldots, \mathsf{tagk}_n$ are still n-wise independent. But this holds since their relations form a system

$$M \cdot \vec{\zeta} := \begin{pmatrix} -\frac{x_2}{x_1} & 1 & & \\ -\frac{x_3}{x_1} & & 1 & \\ \vdots & & & \ddots \\ -\frac{x_n}{x_1} & & & 1 \\ y_1^\star & y_2^\star & y_3^\star & \ldots\ y_n^\star \end{pmatrix} \begin{pmatrix} \zeta_1 \\ \zeta_2 \\ \vdots \\ \\ \zeta_n \end{pmatrix} = - \begin{pmatrix} \mathsf{tagk}_2 \\ \mathsf{tagk}_3 \\ \vdots \\ \mathsf{tagk}_n \\ \mathsf{tagc}^\star \end{pmatrix},$$

which has a solution in $\vec{\zeta}$ whenever $\det(M) = (-1)^{n+1}\vec{X} \cdot \vec{Y}^\star / x_1 \neq 0$.

Eventually, $\mathcal{A}$ outputs a bit β' and $\mathcal{B}$ outputs 0 if $\beta = \beta'$. As in [27], we see that $\mathcal{A}$ is playing Game_{k-1} if $\eta = \nu^{\theta_1+\theta_2}$ and Game_k otherwise. □

Lemma 3. *If DBDH is hard,* Game_q *and* Game_{q+1} *are indistinguishable.*

5 Functional Encryption for Non-Zero Inner-Product

5.1 Negated Spatial Encryption

We begin this section by providing a co-selectively-secure construction of negated spatial encryption, which is motivated by its implication of non-zero IPE. At a high-level, our scheme can be viewed as a "negative" analogue of the Boneh-Hamburg spatial encryption [9], in very much the same way as the Lewko-Sahai-Waters revocation scheme [19] is a negative analogue of the Boneh-Boyen IBE [4]. The intuition follows exactly from section 1, where we have to use "n-equation technique". In spatial encryption, we have to deal with, in general, how we can set up a system of n equations similarly to Eq.(1). To this end, we confine the vector subspaces that we can use as follows. Our construction is a FE for $R^{\mathsf{Neg(Spatial)}} : \mathcal{W}_n \times \mathbb{Z}_p^n \to \{0,1\}$, where we define a collection $\mathcal{W}_n \subseteq \mathcal{V}_n$ of vector subspaces in $\mathbb{Z}_p^n$ as $\mathcal{W}_n = \{\mathrm{Aff}(M, \vec{0}) \in \mathcal{V}_n \mid \mathrm{rank}(M_{(-1)}) = n-1\}$, where we denote $M_{(-1)}$ as the matrix obtained by deleting the first row $M_1 \in \mathbb{Z}_p^{1\times d}$ of M.

Construction 4. (Co-selectively secure negated spatial encryption)

▶ $\mathsf{Setup}(1^\lambda, n)$: chooses a bilinear group $\mathbb{G}$ of prime order $p > 2^\lambda$ with a random generator $g \xleftarrow{\$} \mathbb{G}$. It randomly chooses $\alpha, \alpha_1, \ldots, \alpha_n \xleftarrow{\$} \mathbb{Z}_p$. Let $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$. The public key is $\mathsf{pk} = \left(g, g^{\vec{\alpha}}, g^{\alpha_1 \vec{\alpha}}, e(g,g)^\alpha\right)$. The master key is $\mathsf{msk} = (\alpha, \vec{\alpha})$.

▶ $\mathsf{KeyGen}(V, \mathsf{msk}, \mathsf{pk})$: suppose that $V = \mathrm{Aff}(M, \vec{0})$, from a matrix $M \in (\mathbb{Z}_p)^{n\times d}$. The algorithm picks $t \xleftarrow{\$} \mathbb{Z}_p$ and outputs $\mathsf{sk}_V = (D_0, D_1, \vec{K}) \in \mathbb{G}^{d+2}$ where

$$D_0 = g^t, \qquad D_1 = g^{\alpha + t\alpha_1^2}, \qquad \vec{K} = g^{tM^\top \vec{\alpha}}.$$

▶ $\mathsf{Encrypt}(\vec{y}, \mathsf{M}, \mathsf{pk})$: picks $s \xleftarrow{\$} \mathbb{Z}_p$ and computes (C_0, C_1, C_2, C_3) as

$$C_0 = \mathsf{M} \cdot e(g,g)^{\alpha s}, \qquad C_1 = g^{s\alpha_1 \langle \vec{y}, \vec{\alpha} \rangle}, \qquad C_2 = g^s, \qquad C_3 = g^{\alpha_1 s}.$$

▶ $\mathsf{Decrypt}(C, \vec{y}, \mathsf{sk}_V, V, \mathsf{pk})$: the algorithm first obtains M from V. We also recall the the notation of M_1, which is the vector of the first row of M. It first solves the system of equations in $\vec{w}$ from $M_{(-1)}\vec{w} = (y_2, \ldots, y_n)^\top$, which it can do since $V \in \mathcal{W}_n$. It computes the message blinding factor $e(g,g)^{\alpha s}$ as

$$e(D_1, C_2) \cdot \left(\frac{e(C_1, D_0)}{e(\vec{K}^{\vec{w}}, C_3)} \right)^{\frac{1}{M_1\vec{w} - y_1}} = e(g^{\alpha + t\alpha_1^2}, g^s) \cdot \left(\frac{e(g^{s\alpha_1\langle \vec{y}, \vec{\alpha}\rangle}, g^t)}{g^{t\vec{w}^\top M^\top \vec{\alpha}}, g^{\alpha_1 s})} \right)^{\frac{1}{M_1\vec{w} - y_1}}.$$

Computability. We claim that the decryption can be computed if $y \notin V$. Indeed, we prove that if $y \notin V$ then $M_1\vec{w} - y_1 \neq 0$ (and the above equation is well-defined). To prove the contrapositive, suppose that $M_1\vec{w} - y_1 = 0$. Then, we must have $\vec{y} \in V$ since $M\vec{w} = \begin{bmatrix} M_1 \\ M_{(-1)} \end{bmatrix} \vec{w} = \begin{bmatrix} M_1\vec{w} \\ M_{(-1)}\vec{w} \end{bmatrix} = \vec{y}$.

Correctness. We verify that decryption is correct as follows. First, we note that due to our definition of $\vec{w}$, we have $\langle M\vec{w} - \vec{y}, \vec{\alpha}\rangle = (M_1\vec{w} - y_1)\alpha_1$. Therefore, the correctness follows from the fact that

$$\left(\frac{e(g^{s\alpha_1\langle \vec{y}, \vec{\alpha}\rangle}, g^t)}{e(g^{t\vec{w}^\top M^\top \vec{\alpha}}, g^{\alpha_1 s})} \right)^{\frac{1}{M_1\vec{w} - y_1}} = \left(\frac{1}{e(g,g)^{ts\alpha_1\langle M\vec{w} - \vec{y}, \vec{\alpha}\rangle}} \right)^{\frac{1}{M_1\vec{w} - y_1}} = e(g,g)^{-st\alpha_1^2}.$$

Theorem 3. *Construction 4 is co-selectively secure under the q-Decisional Multi-Exponent Bilinear Diffie-Hellman assumption (q is the number of key queries).* (The proof is given in the full paper where the assumption [19] is also recalled).

Implications. For a vector $\vec{X} \in \mathbb{Z}_p^n$, the embedding $V_{\vec{X}} = \mathrm{Aff}(M_{\vec{X}}, \vec{0_n})$ defined in Eq.(3) is easily seen to be in the limited domain $\mathcal{W}_n$ since $(M_{\vec{X}})_{(-1)}$ is an identity matrix of size $n-1$ and hence $\mathrm{rank}((M_{\vec{X}})_{(-1)}) = n-1$. Therefore, from Corollary 1, the above scheme implies non-zero IPE.

5.2 Non-Zero IPE under Simple Assumptions

We prove the co-selective security of our negated spatial encryption scheme under a non-standard q-type assumption introduced in [19]. Here, we show that the dual system technique [27] makes it possible to rest on simple assumptions such as DBDH and DLIN. The scheme is very similar to the zero IPE scheme of section 4.2 and we only state the differences. The intuition again follows exactly from section 1 and the security proof uses similar techniques as in [19].

Construction 5. (Co-selectively secure non-zero IPE)

▶ $\mathsf{Setup}(1^\lambda, n)$: outputs pk exactly as in the construction 3 except that we define $w = g^{\alpha_1} (= h_1)$ in this scheme, instead of g^{α_0}.

▶ $\mathsf{Keygen}(\vec{X}, \mathsf{msk}, \mathsf{pk})$: outputs $\mathsf{sk}_{\vec{X}} = (\mathsf{sk}_{\mathsf{adapt}}, \mathsf{sk}_{\mathsf{core}})$ where $\mathsf{sk}_{\mathsf{adapt}}$ is the same as in the construction 3 (with $w = g^{\alpha_1}$) and $\mathsf{sk}_{\mathsf{core}} = \{K_i = \left(g^{-\alpha_1 \frac{x_i}{x_1}} \cdot g^{\alpha_i}\right)^{r_1}\}_{i=2,\ldots,n}$.

▶ $\mathsf{Encrypt}(\vec{Y}, \mathsf{M}, \mathsf{pk})$: outputs $C = (C_{\mathsf{adapt}}, C_{\mathsf{core}})$ where C_{adapt} is as in the construction 3 (with $w = g^{\alpha_1}$) and $C_{\mathsf{core}} = \left(E_0 = \mathsf{M} \cdot Z^{s_2},\ E_1 = (g^{\langle \vec{\alpha}, \vec{Y}\rangle})^t,\ E_2 = g^t\right)$.

▶ $\mathsf{Decrypt}(C, \vec{Y}, \mathsf{sk}_{\vec{X}}, \vec{X}, \mathsf{pk})$: computes W_1 as in the construction 3 and W_2 as $W_2 = \left(\frac{e(K_2^{y_2} \cdots K_n^{y_n}, E_2)}{e(E_1, D_7)}\right)^{-\frac{x_1}{\vec{X} \cdot \vec{Y}}} = e(g, w)^{r_1 t}$. (See appendix A.2).

Theorem 4. *Construction 5 is co-selectively secure under the DLIN and DBDH assumptions.* (The proof is deferred to the full version of the paper.)

5.3 A Generalization of the Scheme and Its Application

Extended Ciphertext Attribute Domain. The above scheme for the relation $R^{\mathsf{NIPE}_n} : \mathbb{Z}_p^n \times \mathbb{Z}_p^n \to \{0,1\}$ can be extended so as to support relations of the form $R^{\mathsf{NIPE}_n^*} : \mathbb{Z}_p^n \times (\mathbb{Z}_p^n)^d \to \{0,1\}$, for some $d \in \mathsf{poly}(\lambda)$, and defined as $R^{\mathsf{NIPE}_n^*}(\vec{X}, (\vec{Y}_1, \ldots, \vec{Y}_d)) = 1$ iff for all $i = 1, \ldots, d$: $\vec{X} \cdot \vec{Y}_i \neq 0$.

We construct this extended system by setting up exactly the same public and private keys (for $\vec{X}$) as in the original scheme. To encrypt to $(\vec{Y}_1, \ldots, \vec{Y}_d)$, the scheme generates $C_0, \ldots, C_7$ as usual with the underlying exponents s_1, s_2, t. Then, it chooses $t_1, \ldots, t_d \in \mathbb{Z}_p$ so that $t = t_1 + \cdots + t_d$ and for $i = 1, \ldots, d$, parses $\vec{Y}_i = (y_{i,1}, \ldots, y_{i,n})$ and computes $E_{1,i} = (g^{\langle \vec{\alpha}, \vec{Y}_i \rangle})^{t_i} = (h_1^{y_{i,1}} \cdots h_n^{y_{i,n}})^{t_i}$ and $E_{2,i} = g^{t_i}$, in such a way that the ciphertext is $(C_0, \ldots, C_7, \{E_{1,i}, E_{2,i}\}_{i=1,\ldots,d})$. Decryption requires to first compute

$$W_{2,i} = \left(\frac{e(K_2^{y_{i,2}} \cdots K_n^{y_{i,n}}, E_{2,i})}{e(E_{1,i}, D_7)}\right)^{-\frac{x_1}{\vec{X} \cdot \vec{Y}_i}} = e(g, w)^{r_1 t_i},$$

for $i = 1, \ldots, d$, from which the receiver obtains $W_2 = W_{2,1} \cdots W_{2,d} = e(g, w)^{r_1 t}$. The rest is then done as usual and we explain in the full version of the paper how the security proof must be adapted.

Applications. We can obtain an identity-based revocation scheme with parameter tradeoff from the aforementioned extension. The instantiation of ID-based revocation scheme ($\mathsf{IBR}_{\leq n}$) from our non-zero inner-product system NIPE_{n+1} yields a construction with $O(1)$-size ciphertexts and $O(n)$-size private keys, where n denotes the maximal number of revoked users.

From our extended scheme NIPE_{n+1}^*, we can obtain an ID-based revocation scheme $\mathsf{IBR}_{\mathsf{poly}(\lambda)}$, without a fixed maximal number of revoked users. To revoke the set R where $|R| = r$, we divide it into a disjointed union $R = R_1 \cup \cdots \cup R_{r/n}$, where $|R_i| = n$ for all i (we assume that n divides r). We then simply construct the vector $\vec{Y}_i$ from the revocation subset R_i for each $i \in [1, r/n]$, in the same way as we use NIPE_{n+1} to instantiate $\mathsf{IBR}_{\leq n}$. We then finally encrypt using the set of vectors $(\vec{Y}_1, \ldots, \vec{Y}_{r/n})$. The correctness and security properties hold since $R^{\mathsf{IBR}_{\leq n}}(\mathsf{ID}, R) = 1 \Leftrightarrow R^{\mathsf{IBR}_{\mathsf{poly}(\lambda)}}(\mathsf{ID}, (R_1, \ldots, R_{r/n})) = 1$. The construction has $O(r/n)$-size ciphertexts and $O(n)$-size private keys. Interestingly, we note that the second scheme described by Lewko, Sahai and Waters [19] (which indeed inspires ours) can be viewed as a special case of our scheme where $n = 1$.

References

1. M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, H. Shi. Searchable Encryption Revisited: Consistency Properties, Relation to Anonymous IBE, and Extensions. In *Crypto'05*, *LNCS* 3621, pp. 205–222, 2005.
2. M. Abdalla, E. Kiltz, G. Neven. Generalized Key Delegation for Hierarchical Identity-Based Encryption. In *ESORICS'07*, *LNCS* 4734, pp. 139–154. Springer, 2007.
3. A. Barth, D. Boneh, B. Waters. Privacy in Encrypted Content Distribution Using Private Broadcast Encryption. In *Financial Cryptography 2006*, *LNCS* 4107, pp. 52–64, 2006.
4. D. Boneh, X. Boyen. Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In *Eurocrypt'04*, *LNCS* 3027, pp. 223–238, 2004.
5. D. Boneh, X. Boyen, and H. Shacham. Short Group Signatures. In *Crypto'04*, *LNCS* 3152, pp. 41–55, 2004.
6. D. Boneh, X. Boyen, E.-J. Goh. Hierarchical Identity-Based encryption with Constant Size Ciphertext. In *Eurocrypt'05*, *LNCS* 3494, pp. 440–456, 2005.
7. D. Boneh, G. Di Crescenzo, R. Ostrovsky, G. Persiano. Public Key Encryption with Keyword Search. In *Eurocrypt'04*, *LNCS* 3027, pp. 506–522, 2004.
8. D. Boneh, M. Franklin. Identity-Based Encryption from the Weil Pairing. In *SIAM* Journal of Computing 32(3), pp. 586–615, 2003, earlier version in *Crypto'01*, *LNCS* 2139, pp. 213–229, 2001.
9. D. Boneh, M. Hamburg. Generalized Identity Based and Broadcast Encryption Schemes. In *Asiacrypt'08*, *LNCS* 5350, pp. 455–470, 2008
10. D. Boneh, B. Waters. Conjunctive, Subset, and Range Queries on Encrypted Data. In *4th Theory of Cryptography Conference* (TCC 2007), *LNCS* 4392, pp. 535–554, 2007.
11. R. Canetti, S. Halevi, J. Katz. A Forward-Secure Public-Key Encryption Scheme. In *Eurocrypt'03*, *LNCS* 2656, pp. 254–271, 2003.
12. R. Canetti, S. Halevi, J. Katz. Chosen-Ciphertext Security from Identity-Based Encryption. In *Eurocrypt'04*, *LNCS* 3027, pp. 207–222, 2004.
13. C. Delerablée. Identity-Based Broadcast Encryption with Constant Size Ciphertexts and Private Keys. In *Asiacrypt'07*, *LNCS* 4833, pp. 200–215, 2007.
14. D. Freeman. Converting Pairing-Based Cryptosystems from Composite-Order Groups to Prime-Order Groups. In *Eurocrypt'10*, *LNCS* series, to appear, 2010.
15. C. Gentry, B. Waters. Adaptive Security in Broadcast Encryption Systems (with Short Ciphertexts). In *Eurocrypt'09*, *LNCS* 5479, pp. 171–188, 2009.
16. V. Goyal, O. Pandey, A. Sahai, B. Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *ACM CCS'06*, pp. 89–98, 2006.
17. V. Iovino, G. Persiano. Hidden-Vector Encryption with Groups of Prime Order. In *Pairing'08*, *LNCS* 5209, pp. 75–88, 2008.
18. J. Katz, A. Sahai, B. Waters. Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products. In *Eurocrypt'08*, *LNCS* 4965, pp. 146-162, 2008.
19. A. Lewko, A. Sahai, B. Waters. Revocation Systems with Very Small Private Keys. In IEEE Symposium on Security and Privacy (S&P) 2010, to appear.
20. A. Lewko, B. Waters. New Techniques for Dual System Encryption and Fully Secure HIBE with Short Ciphertexts. In *TCC 2010*, LNCS 5978, pp. 455-479, Springer, 2010.

21. R. Ostrovsky, A. Sahai, B. Waters. Attribute-based encryption with non-monotonic access structures. In *ACM CCS'07*, pp. 195–203, 2007.
22. A. Sahai, B. Waters. Fuzzy Identity-Based Encryption In *Eurocrypt'05*, *LNCS* 3494, pp. 457–473, 2005.
23. R. Sakai, J. Furukawa. Identity-Based Broadcast Encryption. In Cryptology ePrint Archive: Report 2007/217, `http://eprint.iacr.org/2007/217`, 2007.
24. A. Shamir. Identity-Based Cryptosystems and Signature Schemes. In *Crypto'84*, *LNCS* 196, pp. 47–53, 1984.
25. E. Shen, E. Shi, B. Waters. Predicate Privacy in Encryption Systems. In *TCC'09*, *LNCS* 5444, pp. 457–473, 2009.
26. E. Shi, B. Waters. Delegating Capabilities in Predicate Encryption Systems. In *ICALP'08*, *LNCS* 5126, pp. 560–578, 2008.
27. B. Waters. Dual System Encryption: Realizing Fully Secure IBE and HIBE under Simple Assumptions. In *Crypto'09*, *LNCS* series, 2009.

A Verifying Correctness in Decryption

A.1 For the Zero IPE Scheme of Section 4.2

$$W_2 = \left(\frac{e(\prod_{i=2}^n K_i^{y_i}, E_2)}{e(E_1, D_7)}\right)^{\frac{1}{\mathsf{tagk}-\mathsf{tagc}}} = \left(\frac{e\Big(\prod_{i=2}^n (g^{-\alpha_1 \frac{x_i}{x_1}} g^{\alpha_i} w^{\mathsf{tagk}_i})^{r_1 y_i}, g^t\Big)}{e\Big(\big(g^{\langle \vec{\alpha}, \vec{Y}\rangle} \cdot w^{\mathsf{tagc}}\big)^t, g^{r_1}\Big)}\right)^{\frac{1}{\mathsf{tagk}-\mathsf{tagc}}}$$

$$= \left(\frac{e\Big((g^{-\alpha_1 \frac{x_2 y_2 + \cdots + x_n y_n}{x_1}} g^{\alpha_2 y_2 + \cdots + \alpha_n y_n} w^{\mathsf{tagk}_2 y_2 + \cdots + \mathsf{tagk}_n y_n})^{r_1}, g^t\Big)}{e\Big(\big(g^{\alpha_1 y_1 + \alpha_2 y_2 + \cdots + \alpha_n y_n} \cdot w^{\mathsf{tagc}}\big)^t, g^{r_1}\Big)}\right)^{\frac{1}{\mathsf{tagk}-\mathsf{tagc}}}$$

$$= e\Big(g^{-\alpha_1 (\frac{x_2 y_2 + \cdots x_n y_n}{x_1} + y_1)} w^{(\mathsf{tagk}-\mathsf{tagc})}, g\Big)^{\frac{r_1 t}{\mathsf{tagk}-\mathsf{tagc}}}$$

$$= e\Big(g^{-\alpha_1 \frac{\vec{X}\cdot\vec{Y}}{x_1}} w^{(\mathsf{tagk}-\mathsf{tagc})}, g\Big)^{\frac{r_1 t}{\mathsf{tagk}-\mathsf{tagc}}} = e(g, w)^{r_1 t}.$$

A.2 For the Non-Zero IPE Scheme of Section 5.2

$$W_2 = \left(\frac{e(\prod_{i=2}^n K_i^{y_i}, E_2)}{e(E_1, D_7)}\right)^{-\frac{x_1}{\vec{X}\cdot\vec{Y}}} = \left(\frac{e\Big(\prod_{i=2}^n (g^{-\alpha_1 \frac{x_i}{x_1}} g^{\alpha_i})^{r_1 y_i}, g^t\Big)}{e\Big(\big(g^{\alpha_1 y_1 + \cdots + \alpha_n y_n}\big)^t, g^{r_1}\Big)}\right)^{-\frac{x_1}{\vec{X}\cdot\vec{Y}}}$$

$$= \left(\frac{e\Big((w^{-\frac{x_2 y_2 + \cdots + x_n y_n}{x_1}} g^{\alpha_2 y_2 + \cdots + \alpha_n y_n})^{r_1}, g^t\Big)}{e\Big(\big(w^{y_1} \cdot g^{\alpha_2 y_2 + \cdots + \alpha_n y_n}\big)^t, g^{r_1}\Big)}\right)^{-\frac{x_1}{\vec{X}\cdot\vec{Y}}}$$

$$= e\Big(w^{\frac{\vec{X}\cdot\vec{Y}}{x_1}}, g\Big)^{r_1 t \cdot \frac{x_1}{\vec{X}\cdot\vec{Y}}} = e(g, w)^{r_1 t}.$$