Chosen-Ciphertext Secure Attribute-Hiding Non-Zero Inner Product Encryptions and Its Applications

Tapas Pal, Ratna Dutta
Department of Mathematics,
Indian Institute of Technology Kharagpur,
Kharagpur-721302, India
tapas.pal@iitkgp.ac.in,ratna@maths.iitkgp.ernet.in

Abstract

Non-zero inner product encryption (NIPE) allows a user to encrypt a message with an attribute vector and a receiver holding a secret-key associated to a predicate vector can recover the message from the ciphertext if the inner product between the attribute and predicate vectors is non-zero. The main focus is to hide messages in most of the existing NIPEs and the associated attribute is trivially included in the ciphertext. In this work, we investigate the design of NIPEs that are capable of hiding attributes along with messages and secure against active adversaries. In particular, we describe a generic transformation of an attribute-hiding chosen-ciphertext attack (CCA) secure NIPE from an inner product functional encryption (IPFE) and a quasi-adaptive non-interactive zero-knowledge (QANIZK) proof system. This leads us to a set of attribute-hiding NIPEs (AHNIPE) with security based on several assumptions such as plain Decisional Diffie-Hellman (DDH), Learning With Errors (LWE) and Decision Composite Reciprocity (DCR). Furthermore, we build a more efficient and concrete construction of a CCA secure AHNIPE the security of which can be based on DDH and Kernel Matrix Diffie-Hellman (KerMDH) assumptions. As DDH implies the computational KerMDH assumption, the latter construction achieves a CCA secure AHNIPE from minimal assumption to date. We explore a few applications of AHNIPE. More specifically, we show that AHNIPE directly implies an anonymous identity-based revocation (IBR) scheme. Consequently, we get the first CCA secure IBR solely based on plain DDH assumption in the standard model, improving the security of any previous anonymous CCA secure IBR scheme which is proven secure relying on pairing-based assumptions in the random oracle model. Moreover, we add a tracing algorithm to our anonymous IBR scheme to convert it into an efficient anonymous trace and revoked scheme with CCA security.

Keywords. non-zero inner product encryptions, attribute-hiding, chosen-ciphertext.

1 Introduction

To remedy *all-or-nothing* type encryption, plain public-key encryptions are refined over the years into more advanced primitives like *identity-based encryption*, *broadcast encryption*, *attribute-based encryption* [15, 19]. All these primitives can be combined into a single class of encryptions called *functional encryption* (FE) introduced much later by Boneh et al. [10]. Realizing FE for general class of functions employs heavy cryptographic tools [16], and as a result, existing constructions are inefficient for day-to-day use. However, FEs for certain type of functionalities such as Boolean formulae, inner product predicate, keyword search [19, 24] are built from standard and well-understood assumptions, hence are eligible for practical implementation.

In attribute-based encryption, a secret-key $\mathsf{sk}_{\boldsymbol{y}}$ is generated corresponding to a predicate $\boldsymbol{y}$ and a ciphertext CT for a message M is associated with an attribute $\boldsymbol{x}$. Using a secret-key $\mathsf{sk}_{\boldsymbol{y}}$, the decryption successfully recovers the message M from $\mathsf{CT}_{\boldsymbol{x}}$ if a relation $R(\boldsymbol{x}, \boldsymbol{y})$ holds. This paper studies a primitive called *non-zero inner product encryption* (NIPE) [5] that considers the predicate and attribute space to be $\mathbb{Z}^\ell$ (resp. $\mathbb{Z}_p^\ell$ for some prime p) for a natural number ℓ and the relation R is defined as $R(\boldsymbol{x}, \boldsymbol{y}) = 1$ if and only if $\langle \boldsymbol{x}, \boldsymbol{y} \rangle \neq 0$ over $\mathbb{Z}$ (resp. over $\mathbb{Z}_p$). In recent years inner product encryptions have emerged with several applications in identity-based encryption, polynomial evaluation, disjunctions/conjunctions equality test, proxy-re-encryption [24, 11, 25] etc. Since NIPE is a *negated* subclass of IPE, the above primitives with negation (such as identity-based revocation (IBR), polynomial non-equality and so on) are captured in applications of NIPEs [5, 4].

Mostly the security of a NIPE scheme is considered in *payload-hiding* (PHNIPE) setting where the challenge ciphertext is required to hide only the message associated with a single challenge attribute. The attributes are assumed to be a part of ciphertexts in a PHNIPE system. In many applications, for example, anonymous identity-based revocation (ANON-IBR) or broadcast schemes [9, 30, 39, 28], the attributes may contain user-specific sensitive information leaking of which is a strict violation of users privacy. Therefore, such applications demand to hide the attribute along with messages while encryption. This additional security feature is guaranteed by *attribute-hiding* NIPE (AHNIPE) where the adversary is asked to submit two attribute-message pairs $(\boldsymbol{x}_b, M_b)$ for $b \in \{0, 1\}$. Given encryption for a pair $(\boldsymbol{x}_b, M_b)$, it is required that for any PPT adversary the probability guessing the bit b is at most $1/2$. The secret-key queries for the predicate vectors $\boldsymbol{y}$ are restricted to satisfy that $\langle \boldsymbol{x}_0, \boldsymbol{y} \rangle = \langle \boldsymbol{x}_1, \boldsymbol{y} \rangle = 0$ if $M_0 \neq M_1$, else $\langle \boldsymbol{x}_0 - \boldsymbol{x}_1, \boldsymbol{y} \rangle = 0$. This is slightly weaker than the *full attribute-hiding* notion of [33] as the case $\langle \boldsymbol{x}_0, \boldsymbol{y} \rangle \neq \langle \boldsymbol{x}_1, \boldsymbol{y} \rangle$ is not captured in our model. But, it defines stronger security than the *weak attribute-hiding* model of [33] where the case $M_0 = M_1$ is totally excluded and our notion of attribute-hiding is sufficient for many applications discussed latter in this section.

Background. The first NIPE construction was given by Attrapadung and Libert [5]. The scheme is co-selectively secure under the Decision Linear (DLIN) and Decision Bilinear Diffie-Hellman (DBDH) assumptions. In co-selective model the adversary $\mathcal{A}$ declares its secret-key queries before the setup phase, but $\mathcal{A}$ can select the challenge attribute based on the information gained from the secret-key queries. Therefore, co-selective is a dual of the selective model where $\mathcal{A}$ picks the challenge attribute before seeing the master public-key and asks for secret-keys adaptively. Both of these security models are weaker than the desirable adaptive security in which $\mathcal{A}$ has the freedom to choose the challenge attribute as well as the predicate vectors (to be queried for the secret-keys) after the setup phase. As an application of NIPE, [5] built an IBR scheme [29] with constant size ciphertext. Despite its involvement in realizing many useful primitives, the security of NIPEs has not much improved in standard models. Most of the prior works [5, 32, 6, 38, 14, 13] have focused on reducing the size of ciphertexts or secret-keys (or both), but they end up with a paring based system that is secure either in co-selective or selective model. Okamoto and Takashima [33] gave the first adaptively secure NIPE from DLIN assumption. Recently, a learning with errors (LWE) based NIPE is proposed in [23] which is selectively secure and capable of one-bit encryption. In the multi-bit variant of the scheme, sizes of the master public-keys, ciphertexts and secret-keys increase at least linearly with the bit-length of the message. The NIPE also suffers from a complex parameter selection where the noise to modulus ratio is exponentially large in the dimension of attribute vectors and the ciphertext-size is greater than the square of this dimension. Although the generic construction of [23] delivers adaptively secure NIPEs via inner product functional encryptions (IPFE) of [3] in standard models, they are only payload-hiding and chosen-plaintext attack (CPA) secure like all previously known NIPEs.

In literature, hiding attribute in ABE is termed as *predicate encryption* (PE) [11, 24, 20]. The notion of AHNIPE corresponds to a particular function class of a PE scheme and hence a PE for all circuits such as [20] readily gives an indirect construction of AHNIPE. However, the LWE-based PE of [20] uses a fully homomorphic encryption (FHE) scheme [18] to evaluate predicate circuits on attributes which are encrypted under the FHE. Consequently, the resulting scheme becomes complex and expensive for simple function classes such as AHNIPE. Overcoming this limitation, Patranabis et al. [34] built a subset non-membership encryption (SNME) relying on the DDH-based IPFE of [3] which includes the function class needed for AHNIPE. The scheme is CPA secure under Matrix DDH (MDDH) assumption. Therefore, a direct construction of AHNIPE hardly exists and the efficiency of existing indirect schemes has been compromised in order to support a broader class of predicates. While PEs are mostly proved secure in CPA model, recently Koppula and Waters [27] provided a generic and black box transformation to achieve chosen-ciphertext attack (CCA) secure one-sided[1] PEs. The transformation additionally needs to utilize a signature scheme, a public-key encryption and a special pseudorandom generator and loses practical efficiency when applied to simple function classes.

A Motivating Example. To explain the importance of attribute-hiding property of a NIPE system, we consider a practical scenario where our AHNIPE based ANON-IBR can fulfil users' requirement. Suppose in a defence organization of a country the director wants to pass a message to the senior officers of all the departments such as army, navy, marines and air-force working at the post of General or Major. The director utilizes an ANON-IBR to encrypt the message with the set of all revoked users in the system. Note that, the set of revoked users contains the identities of all the junior officers in the organization working under a senior officer holding the post of a General or Major. It is natural to protect not only the message as well as the identities of each user in the organization since identities may contain code-names (or other delicate credentials) of the officers revealing which bring essential threat to the security of the nation. On the other hand, the significance of the message highly depends on the receivers identity, that is, the message sent to the high-rank officers contains much more sensitive information. Therefore, users' anonymity is necessary for this application and AHNIPE provides an efficient solution to it.

Contribution. Our contribution is mainly two-fold.

- Firstly, we give a generic transformation to achieve a chosen-ciphertext attack (CCA) secure tag-based AHNIPE from an indistinguishability based CPA secure (IPFE) [3] and a quasi-adaptive non-interactive zero-knowledge (QANIZK) proof system [1, 26]. We introduce tag-based AHNIPE where the encryption algorithm takes a tag as an additional input along with an attribute and a message. Note that decryption with a tag is successful only if the same tag is used for encryption. However, we can always avoid the tag through a generic transformation by using a one-time signature on the tags. We show that the classic Naor-Yung dual encryption technique [31] can be applied in the setting of inner product encryption. We replace the PKE with IPFE in the transformation of [31] to achieve a CCA secure AHNIPE scheme. The generic NIPE of [23] is also based on IPFE and provides payload-hiding CPA security whereas our transformation delivers stronger security of attribute-hiding and additionally, we get CCA security with the help of a QANIZK proof system. If we drop QANIZK our transformation, generalizing the MDDH-based AHNIPE of [34], leads to the *first* CPA secure AHNIPE schemes based on various assumptions such as DDH, LWE, DCR, DDH-f and HSM when equipped with the IPFEs of [3, 12]. We note that any simulation sound NIZK scheme based on either paring or LWE [37, 21, 35] can be used in our transformation instead of QANIZK. Alternatively, one may avoid the use of QANIZK by considering CCA secure IPFEs of [7] in our transformation to achieve

[1] One-sided security corresponds to weak-attribute hiding, that is, the adversary is not allowed to get a secret-key which can decrypt the challenge ciphertext.

Table 1: Comparison with existing adaptively secure AHNIPEs where ℓ denotes the length of an attribute or predicate. The columns $|\mathsf{MSK}|, |\mathsf{MPK}|$, $|\mathsf{sk}_{\boldsymbol{y}}|$ and $|\mathsf{CT}|$ refer to the number of group elements in a cyclic group $\mathbb{G}$ of prime order or the number of $\mathbb{Z}$ elements. The row PMR19 corresponds to $k = 1$ of the SNME scheme of [34]. We instantiate our generic AHNIPE with DDH-based IPFE of [3] and KerMDH-based QANIZK of [26].

scheme	$\|\mathsf{MSK}\|$	$\|\mathsf{MPK}\|$	$\|\mathsf{sk}_{\boldsymbol{y}}\|$	$\|\mathsf{CT}\|$	assumption	CCA
PMR19 [34]	$4\ell\|\mathbb{Z}\|$	$(2\ell+2)\|\mathbb{G}\|$	$(\ell+4)\|\mathbb{G}\|$	$(2\ell+4)\|\mathbb{G}\|$	MDDH	✗
Ours generic	$(2\ell+8)\|\mathbb{Z}\|$	$(\ell+12)\|\mathbb{G}\|$	$2\|\mathbb{Z}\|$	$(2\ell+8)\|\mathbb{G}\|$	DDH + KerMDH	✓
Ours concrete	$(4\ell+8)\|\mathbb{Z}\|$	$(2\ell+12)\|\mathbb{G}\|$	$4\|\mathbb{Z}\|$	$(2\ell+4)\|\mathbb{G}\|$	DDH + KerMDH	✓

CCA secure AHNIPE, but this would require additional MDDH assumption and the resulting AHNIPE can not be completely based on LWE assumption. However, the Naor-Yung transformation naturally doubles the ciphertext size of our CCA secure AHNIPE which needs more storage and communicational power. To overcome this inefficiency we require different approach compatible with existing IPFE schemes.

- Next, we give a concrete instantiation of a CCA secure AHNIPE based on plain DDH assumption. Our generic transformation needs four ciphertexts of an IPFE and the QANIZK proof adds more elements to it. For example, a ciphertext of our DDH-based AHNIPE contains at least $4\ell + 16$ group elements when using the DDH-based IPFE of [3] and the Kernel Matrix Diffie-Hellman (KerMDH) based QANIZK of [26]. Note that the IPFE contributes $4\ell + 8$ elements to the ciphertext and the rest are coming from the QANIZK proof. We show how to reduce the ciphertext size to only $2\ell + 4$ elements using a technique proposed by Biagioni et al. [8]. Main idea is to use a shared randomness in Naor-Yung dual encryptions. This helps us to reduce the ciphertext and public-key sizes significantly. More precisely, we present a CCA secure AHNIPE based on the DDH-based IPFE of [3] and the KerMDH-based QANIZK of [26]. Interestingly, DDH implies KerMDH which is a computational assumption [26], and hence the AHNIPE is solely based on plain DDH assumption. The ciphertext of the MDDH-based AHNIPE of [34] also contains $2\ell+4$ group elements but achieves only CPA security. In addition to CCA security, our AHNIPEs are well comparable with the work of [34] in terms of ciphertext size and hardness assumption as shown in Table 1.

There are interesting implications of our results. Following the blueprint of [5], we show that any AHNIPE system directly implies an anonymous identity-based revocation (ANON-IBR) (or anonymous identity-based broadcast encryption [39]) scheme. Recall that an IBR allows one to encrypt messages with respect to a list of revoked users and only the users lying outside the revoked list can decrypt the ciphertext. We call the IBR anonymous if the ciphertext does not reveal revoked users identities. Our DDH-based CCA secure AHNIPE yields the *first* CCA secure ANON-IBR from plain DDH assumption in the *standard model*. Prior work [22] achieves anonymity and CCA security based on BDDH assumptions in the random oracle model. Inspired from the IBTR scheme of Agrawal et al. [2], we extend the IBR to efficient CPA secure anonymous identity-based trace and revoke (ANON-IBTR) schemes where the security can be based on DDH, LWE and DCR assumptions.

2 Preliminaries

Notation. We denote by $x \leftarrow \mathcal{D}$ the process of sampling a value x according to the distribution of $\mathcal{D}$. We consider $x \leftarrow S$ as the process of random sampling a value x according to the uniform distribution over a finite set S. We assume that the predicate and attribute vectors are of same length ℓ. The inner product between two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{Z}^\ell$ is written as $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \sum_{i=1}^{\ell} x_i y_i = \boldsymbol{x}^T \boldsymbol{y}$. For any $\lambda > \lambda_0$, if a non-negative function negl satisfies $\mathsf{negl}(\lambda) < 1/\lambda^c$, c is a constant, then negl is called a *negligible* function over the positive integers.

2.1 Pairing Groups and Hardness Assumptions

Let GGen be a probabilistic polynomial time (PPT) algorithm that on input 1^λ returns a description $\mathcal{PG} = \{\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, p, g_1, g_2, e\}$ of asymmetric pairing groups where $\mathbb{G}_s$ be a cyclic group of order p (for a λ-bit prime p) with a generator g_s for each $s \in \{1, 2, T\}$, and $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an efficiently computable (non-degenerate) bilinear map such that $g_T = e(g_1, g_2)$. We use implicit representation of group elements as $[a]_s = g_s^a \in \mathbb{G}_s$ for any $a \in \mathbb{Z}_p$ and $s \in \{1, 2, T\}$. More generally, for a matrix $\mathbf{A} = (a_{ij}) \in \mathbb{Z}_p^{n \times m}$ we define $[\mathbf{A}]_s$ as the implicit representation of $\mathbf{A}$ in $\mathbb{G}_s$:

$$[\mathbf{A}]_s = \begin{pmatrix} g_s^{a_{11}} & \cdots & g_s^{a_{11}} \\ \\ g_s^{a_{n1}} & \cdots & g_s^{a_{nm}} \end{pmatrix}$$

Given $[a]_1$ and $[b]_2$ one can efficiently compute $[a \cdot b]_T$ using the pairing e. For matrices $\mathbf{A}$ and $\mathbf{B}$ of matching dimensions, we define $[\mathbf{AB}]_T = e([\mathbf{A}]_1, [\mathbf{B}]_2)$. We now recall the DDH and KerMDH assumptions.

Definition 1. (Decisional Diffie-Hellman assumption) *Let $s \in \{1, 2, T\}$. We say that decisional Diffie-Hellman (DDH) assumption holds relative to GGen in group $\mathbb{G}_s$ (GGen_s), if for all PPT adversary $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathsf{DDH}}_{\mathcal{A},\mathsf{GGen}_s}(\lambda) = |\Pr[\mathcal{A}(\mathcal{G}_s, [\mathbf{a}]_s, [\mathbf{a}r]_s) = 1] - \Pr[\mathcal{A}(\mathcal{G}_s, [\mathbf{a}]_s, [\mathbf{u}]_s) = 1]|$$

is negligible in λ where the probability is taken over $\mathcal{G}_s = (\mathbb{G}_s, g_s, p) \leftarrow \mathsf{GGen}_s(1^\lambda)$, $(a, r) \leftarrow \mathbb{Z}_p^2, \mathbf{u} \leftarrow \mathbb{Z}_p^2$ and $\mathbf{a} = (1, a)$.

Definition 2. (Kernel Diffie-Hellman assumption)*[26] Let $k \in \mathbb{N}$ and $\mathcal{D}_k$ be a matrix distribution which outputs matrices in $\mathbb{Z}_p^{(k+1) \times k}$ of full rank k in polynomial time. Let $s \in \{1, 2\}$. We say that $\mathcal{D}_k$-Kernel Diffie-Hellman ($\mathcal{D}_k$-KerMDH) assumption holds relative to GGen_s in group $\mathbb{G}_s$, if for all PPT adversary $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathcal{D}_k\text{-}\mathsf{KerMDH}}_{\mathcal{A},\mathsf{GGen}_s}(\lambda) = \Pr[\mathbf{c}^\top \mathbf{A} = 0 \wedge \mathbf{c} \neq \mathbf{0} : [\mathbf{c}]_{3-s} \leftarrow \mathcal{A}(\mathcal{G}_s, [\mathbf{A}]_s)]$$

is negligible in λ where the probability is taken over $\mathcal{G}_s = (\mathbb{G}_s, g_s, p) \leftarrow \mathsf{GGen}_s(1^\lambda)$, $\mathbf{A} \leftarrow \mathcal{D}_k$. If $k = 1$, we simply denote it by KerMDH where $\mathcal{D}_k$ is assumed to output non-zero vectors from $\mathbb{Z}_p^2$.

2.2 Inner Product Functional Encryption [3]

Definition 3. (Inner product functional encryption) *An inner product functional encryption (IPFE) scheme for a predicate space $\mathcal{P}$, an attribute space $\mathcal{Q}$ and an inner product space $\mathcal{I}$ consists of four PPT algorithms* $\mathsf{IPFE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *satisfying the following requirement:*

- $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$*: A trusted authority runs the setup algorithm taking inputs a security parameter λ, a vector length parameter ℓ (a natural number that is a polynomial in λ) and outputs a master public-key* MPK *and a master secret-key* MSK.
- $\mathsf{sk}_{\boldsymbol{y}} \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, \boldsymbol{y})$*: A predicate holder submits a vector $\boldsymbol{y} \in \mathcal{P}$ to an authority that runs the key generation algorithm providing inputs as a master public-key* MPK*, a master secret-key* MSK*, a vector $\boldsymbol{y}$ and outputs a secret key* $\mathsf{sk}_{\boldsymbol{y}}$ *corresponding to the predicate vector $\boldsymbol{y}$.*

```
$(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$
$(\boldsymbol{x}_0, \boldsymbol{x}_1) \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{KG}}(\cdot)}(1^\lambda)$
$\mathsf{ct}^* \leftarrow \mathsf{Enc}(\mathsf{MPK}, \boldsymbol{x}_b)$
$b' \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{KG}}(\cdot)}(\mathsf{ct}^*)$
return $b'$

$\underline{O_{\mathsf{KG}}(\cdot)}$:
    input: $\boldsymbol{y} \in \mathcal{P}$
    return $\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, \boldsymbol{y})$
```

Figure 1: $\mathsf{Expt}_{\mathcal{A}}^{\mathsf{IND\text{-}IPFE}}(1^\lambda, b)$

- $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{MPK}, \boldsymbol{x})$*: An encrypter runs the encryption algorithm that takes as input a master public-key* MPK*, an attribute vector* $\boldsymbol{x} \in \mathcal{Q}$ *and publishes the ciphertext* ct *corresponding to the attribute* $\boldsymbol{x}$*.*
- $\perp$ *or* $\zeta \leftarrow \mathsf{IPFE.Dec}(\mathsf{MPK}, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{ct})$*: A decrypter runs the decryption algorithm taking as input a master public-key* MPK*, a secret-key* $\mathsf{sk}_{\boldsymbol{y}}$*, a ciphertext* ct *and outputs either a message* $\zeta \in \mathcal{I}$ *or a symbol* $\perp$ *indicating failure.*

Correctness: *For any* $\lambda, \ell \in \mathbb{N}$*,* $\boldsymbol{y} \in \mathcal{P}$*,* $\boldsymbol{x} \in \mathcal{Q}$*,* $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$*,* $\mathsf{sk}_{\boldsymbol{y}} \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, \boldsymbol{y})$*,* $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{MPK}, \boldsymbol{x})$ *we have*

$$\Pr\big[\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \mathsf{Dec}(\mathsf{MPK}, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{ct})\big] = 1 - \mathsf{negl}(\lambda)$$

Definition 4. (Indistinguishability-based security for IPFE) *An inner product functional encryption scheme* $\mathsf{IPFE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Dec})$ *for a predicate space* $\mathcal{P}$*, an attribute space* $\mathcal{Q}$ *and an inner product space* $\mathcal{I}$ *is said to be adaptively secure under chosen-plaintext attacks (*$\mathsf{IND\text{-}IPFE}$*) if, for any PPT adversary* $\mathcal{A}$*, for any* $\lambda \in \mathbb{N}$*, the advantage*

$$\mathsf{Adv}_{\mathcal{A},\mathsf{CPA}}^{\mathsf{IND\text{-}IPFE}}(\lambda) = \left| \Pr[\mathsf{Expt}_{\mathcal{A},\mathsf{CPA}}^{\mathsf{IND\text{-}IPFE}}(1^\lambda, 0) = 1] - \Pr[\mathsf{Expt}_{\mathcal{A},\mathsf{CPA}}^{\mathsf{IND\text{-}IPFE}}(1^\lambda, 1) = 1] \right|$$

is negligible in λ *where* $\mathsf{Expt}_{\mathcal{A},\mathsf{CPA}}^{\mathsf{IND\text{-}IPFE}}(1^\lambda, b)$ *is defined in Fig. 4 with the restriction that all secret-key queries* $\{\boldsymbol{y}\}$ *made to the key generation oracle* $\mathcal{O}_{\mathsf{KG}}(\cdot)$ *should satisfy* $\langle \boldsymbol{x}_0, \boldsymbol{y} \rangle = \langle \boldsymbol{x}_0, \boldsymbol{y} \rangle = 0$*.*

2.3 Non-zero Inner Product Encryption [5, 33]

Definition 5. (Non-zero inner product encryption) *A non-zero inner product functional encryption (*NIPE*) scheme for a predicate space* $\mathcal{P}$*, an attribute space* $\mathcal{Q}$*, an inner product space* $\mathcal{I}$*, a tag space* $\mathcal{T}$ *and a message space* $\mathcal{M}$ *consists of four probabilistic polynomial time (PPT) algorithms* $\mathsf{NIPE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Dec})$ *operating as follows:*

- $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$*: A trusted authority runs the setup algorithm which takes as input a security parameter* λ*, a vector length parameter* ℓ *(a natural number that is a polynomial in* λ*) and outputs a master public-key* MPK *and a master secret-key* MSK*.*
- $\mathsf{sk}_{\boldsymbol{y}} \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, \boldsymbol{y})$*: A predicate holder submits a vector* $\boldsymbol{y} \in \mathcal{P}$ *to an authority that runs the key generation algorithm providing inputs as a master public-key* MPK*, a master secret-key* MSK*, a vector* $\boldsymbol{y}$ *and outputs a secret key* $\mathsf{sk}_{\boldsymbol{y}}$ *corresponding to the predicate vector* $\boldsymbol{y}$*.*
- $\mathsf{CT} \leftarrow \mathsf{Enc}(\mathsf{MPK}, \tau, \boldsymbol{x}, M)$*: An encrypter runs this algorithm that takes as input a master public-key* MPK*, a tag* $\tau \in \mathcal{T}$*, an attribute vector* $\boldsymbol{x} \in \mathcal{Q}$*, a message* $M \in \mathcal{M}$ *and publishes the ciphertext* CT *corresponding to the attribute* $\boldsymbol{x}$*.*
- $\perp$ *or* $\zeta \leftarrow \mathsf{Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{CT}_{\boldsymbol{x}})$*: A user runs the decryption algorithm that takes as input a master public-key* MPK*, a tag* τ*, a secret-key* $\mathsf{sk}_{\boldsymbol{y}}$*, a ciphertext* $\mathsf{CT}_{\boldsymbol{x}}$*, and outputs either a message* $\zeta \in \mathcal{M}$ *or a symbol* $\perp$*.*

Correctness: *For any security parameter* $\lambda, \ell \in \mathbb{N}$*, any tag* $\tau \in \mathcal{T}$*,* $\boldsymbol{y} \in \mathcal{P}$*,* $\boldsymbol{x} \in \mathcal{Q}$*,* $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$*,* $\mathsf{sk}_{\boldsymbol{y}} \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, \boldsymbol{y})$ *and* $\mathsf{CT} \leftarrow \mathsf{Enc}(\mathsf{MPK}, \tau, \boldsymbol{x}, M)$ *we have:*

```
1. $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$
2. $(\tau^*, (\boldsymbol{x}_0, M_0), (\boldsymbol{x}_1, M_1)) \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{KG}}(\cdot), \mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)}(1^\lambda)$
3. $\mathsf{CT}^* \leftarrow \mathsf{Enc}(\mathsf{MPK}, \tau^*, \boldsymbol{x}_b, M_b)$
4. $b' \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{KG}}(\cdot), \mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)}(\mathsf{CT}^*)$
5. return $b'$

$O_{\mathsf{KG}}(\cdot)$:
    1. input: $\boldsymbol{y} \in \mathcal{P}$
    2. return $\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, \boldsymbol{y})$
$O_{\mathsf{Dec}}(\cdot,\cdot,\cdot)$:
    1. input: $\tau \in \mathcal{T}, \mathsf{CT}, \boldsymbol{y} \in \mathcal{P}$
    2. $\mathsf{sk_y} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, \boldsymbol{y})$
    3. return $\mathsf{Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{CT})$
```

Figure 2: $\mathsf{Expt}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, b)$

$$\textit{1. } \Pr\big[M = \mathsf{Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{CT}) : \langle \boldsymbol{x}, \boldsymbol{y} \rangle \neq 0\big] = 1 - \mathsf{negl}(\lambda)$$

$$\textit{2. } \Pr\big[\perp = \mathsf{Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{CT}) : \langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0\big] = 1 - \mathsf{negl}(\lambda)$$

Definition 6. (Adaptively attribute-hiding CCA security for NIPE) *A non-zero inner product encryption scheme* NIPE = (Setup, Keygen, Enc, Dec) *for a predicate space $\mathcal{P}$, an attribute space $\mathcal{Q}$, a tag space $\mathcal{T}$, an inner product space $\mathcal{I}$ and a message space $\mathcal{M}$ is said to be adaptively attribute-hiding secure under chosen-ciphertext attacks (*AHNIPE*) if, for any PPT adversary $\mathcal{A}$, for any $\lambda \in \mathbb{N}$, the advantage*

$$\mathsf{Adv}^{\mathsf{AH\text{-}NIPE}}_{\mathcal{A},\mathsf{CCA}}(\lambda) = \left| \Pr[\mathsf{Expt}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, 0) = 1] - \Pr[\mathsf{Expt}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, 1) = 1] \right|$$

is negligible in λ, where $\mathsf{Expt}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, b)$ is defined in Fig. 2 with the following restriction on $\mathcal{A}$'s queries:

- *All secret-key queries $\{\boldsymbol{y}\}$ to the key generation oracle $\mathcal{O}_{\mathsf{KG}}(\cdot)$ should satisfy $\langle \boldsymbol{x}_0, \boldsymbol{y} \rangle = \langle \boldsymbol{x}_0, \boldsymbol{y} \rangle = 0$ if $M_0 \neq M_1$ and $\langle \boldsymbol{x}_0 - \boldsymbol{x}_1, \boldsymbol{y} \rangle = 0$ if $M_0 = M_1$.*
- *All decryption queries $\{(\tau, \mathsf{CT}, \boldsymbol{y})\}$ to the decryption oracle $\mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)$ should satisfy that $\tau \neq \tau^*$.*

2.4 Quasi-Adaptive Non-Interactive Zero-Knowledge Proof [26]

A quasi-adaptive non-interactive zero knowledge argument (QANIZK) is a type of NIZK where the common reference string (crs) is allowed to depend on the specific parameter defined by the language for which proofs have to be generated. For public parameters par, let $\mathcal{D}_{par}$ be a probability distribution over a collection of relations $\mathcal{R} = \{R_\rho\}$ parameterized by ρ with as associated language $L_\rho = \{x : \exists\ w \text{ s.t. } R_\rho(x, w) = 1\}$.

Definition 7. (quasi-adaptive non-interactive zero knowledge argument) *A Quasi-adaptive non-interactive zero knowledege argument (QANIZK) for a language distribution $\mathcal{D}_{\mathsf{par}}$ consists of five PPT algorithms QANIZK = (Gen$_{\mathsf{par}}$, Gen$_{\mathsf{crs}}$,Prv, Sim, Vrfy) working as follows:*

- *$\mathsf{par} \leftarrow \mathsf{Gen}_{\mathsf{par}}(\lambda)$: It is a probabilistic algorithm which on input a security parameter λ outputs public parameters* par.
- *$(\mathsf{crs}, \mathsf{trap}) \leftarrow \mathsf{Gen}_{\mathsf{crs}}(\mathsf{par}, \rho)$: It is a probabilistic algorithm which takes as input* par *and a string ρ, an outputs* crs *and a trapdoor* trap*. We assume that* crs *implicitly contains* par *and ρ, and that it defines a tag space $\mathcal{T}$.*
- *$\pi \leftarrow \mathsf{Prv}(\mathsf{crs}, \tau, x, w)$: It is a deterministic algorithm which on input a* crs*, a tag $\tau \in \mathcal{T}$, a statement $x \in L_\rho$ and a witness w outputs a proof π.*
- *1 or 0 $\leftarrow \mathsf{Vrfy}(\mathsf{crs}, \tau, x, \pi)$: It is a deterministic algorithm which on input a* crs*, a tag τ, a statement x and a proof π outputs 1 if π is a valid proof that $x \in L_\rho$; otherwise returns 0.*
- *$\pi \leftarrow \mathsf{Sim}(\mathsf{crs}, \mathsf{trap}, \tau, x)$: It is a deterministic algorithm which on input a* crs*, a trapdoor* trap*, a tag $\tau \in \mathcal{T}$ an a statement x (not necessarily in L_ρ) outputs a simulated proof π.*

We require that the algorithms satisfy the following properties:

Perfect completeness. *For all λ, all par output by $\mathsf{Gen_{par}}(\lambda)$, all ρ output by $\mathcal{D}_{par}$, all (x, w) with $R_\rho(x, w) = 1$, all $\tau \in \mathcal{T}$, we have*

$$\Pr\left[\mathsf{Vrfy}(\mathsf{crs}, \tau, x, \pi) = 1 \;\middle|\; \begin{array}{c} (\mathsf{crs}, \mathsf{trap}) \leftarrow \mathsf{Gen_{crs}}(\mathsf{par}, \rho) \\ \pi \leftarrow \mathsf{Prv}(\mathsf{crs}, \tau, x, w) \end{array}\right] = 1$$

Perfect zero-knowledge. *For all λ, all par output by $\mathsf{Gen_{par}}(\lambda)$, all ρ output by $\mathcal{D}_{par}$, all $(\mathsf{crs}, \mathsf{trap})$ output by $\mathsf{Gen_{crs}}(\mathsf{par}, \rho)$, all (x, w) with $R_\rho(x, w) = 1$, all $\tau \in \mathcal{T}$, the distributions*

$$\mathsf{Prv}(\mathsf{crs}, \tau, x, w) \text{ and } \mathsf{Sim}(\mathsf{crs}, \mathsf{trap}, \tau, x)$$

are the same (where the coin tosses are taken over Prv and Sim).

Simulation soundness. *For all PPT adversary $\mathcal{A}$ and any QANIZK the following advantage*

$$\mathsf{Adv}^{\mathsf{SS}}_{\mathcal{A}}(\lambda) = \Pr\left[\begin{array}{c} \mathsf{Vrfy}(\mathsf{crs}, \tau^*, x^*, \pi^*) = 1 \\ \wedge x^* \notin L_\rho \wedge \tau^* \notin \mathcal{T}_{\mathsf{sim}} \end{array} \;\middle|\; \begin{array}{l} \mathsf{par} \leftarrow \mathsf{Gen_{par}}(\lambda); \rho \leftarrow \mathcal{D}_{\mathsf{par}}; \\ (\mathsf{crs}, \mathsf{trap}) \leftarrow \mathsf{Gen_{crs}}(\mathsf{par}, \rho); \\ (\tau^*, x^*, \pi^*) \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{sim}}(\cdot,\cdot)}(\mathsf{crs}) \end{array}\right]$$

is negligible, where $\mathcal{O}_{\mathsf{sim}}(\tau, x)$ returns $\pi \leftarrow \mathsf{Sim}(\mathsf{crs}, \mathsf{trap}, \tau, x)$ and $\mathcal{T}_{\mathsf{sim}}$ is the set of all tags queried by $\mathcal{A}$. We call QANIZK to satisfy one-time simulation soundness (OTSS) if $\mathcal{A}$ is allowed to make only one query to $\mathcal{O}_{\mathsf{sim}}(\cdot, \cdot)$, and the corresponding advantage is denoted as $\mathsf{Adv}^{\mathsf{OTSS}}_{\mathcal{A}}(\lambda)$.

Lemma 1. *(core lemma for one-time soundness of QANIZK)[26] Let $n, t, k \in \mathbb{N}$. For any $\mathbf{M} \in \mathbb{Z}_p^{n \times t}, \mathbf{A} \in \mathbb{Z}_p^{(k+1) \times k}$ and any (possibly unbounded) adversary $\mathcal{A}$,*

$$\Pr\left[\begin{array}{l} \boldsymbol{y} \notin \mathit{Span}(\mathbf{M}) \wedge \tau \neq \hat{\tau} \\ \wedge \boldsymbol{z}^\top = \boldsymbol{y}^\top(\mathbf{K}_0 + \hat{\tau}\mathbf{K}_1) \end{array} \middle| \begin{array}{c} \mathbf{K}_0, \mathbf{K}_1 \leftarrow \mathbb{Z}_p^{n \times (k+1)}; \\ (\boldsymbol{z}, \boldsymbol{y}, \tau) \leftarrow \mathcal{A}^{\mathcal{O}(\cdot)}(\mathbf{M}^\top \mathbf{K}_0, \mathbf{M}^\top \mathbf{K}_1, \mathbf{K}_0\mathbf{A}, \mathbf{K}_1\mathbf{A}) \end{array}\right] \leq \frac{1}{p}$$

where $\mathcal{O}(\hat{\tau})$ may be called one time and returns $\mathbf{K}_0 + \hat{\tau}\mathbf{K}_1$.

3 Generic Construction: AHNIPE from IPFE and QANIZK

We describe how to use the indistinguishability-based security of a IPFE [3] to achieve the attribute-hiding security for a NIPE through a generic transformation. Our technique is compatible with the CCA transformation given by Sahai [37] which obtains CCA security of a public-key encryption via NIZK proofs. However, we use QANIZK proofs in our transformation to achieve CCA security. Let us consider an $\mathsf{IPFE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ with a predicate space $\mathcal{P}'$, an attribute space $\mathcal{Q}'$ and an inner product space $\mathcal{I}'$. We construct a $\mathsf{NIPE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ with the same predicate space $\mathcal{P} = \mathcal{P}'$, the attribute space $\mathcal{Q}$, the inner product space $\mathcal{I} = \mathcal{I}'$ and a message space $\mathcal{M}$ such that $\mathcal{P}, \mathcal{Q}, \mathcal{Q}' \subseteq \mathcal{I}^l$, $\mathcal{M} \subset \mathcal{I}$ and for any $\boldsymbol{x} = (x_1, \ldots, x_l) \in \mathcal{Q}$, $M \in \mathcal{M}$ it holds that $M \cdot \boldsymbol{x} \in \mathcal{Q}'$ where $M \cdot \boldsymbol{x} = (Mx_1, \ldots, Mx_l)$. It is also required that the division operation can be efficiently executed in $\mathcal{I}$, that is for any product value $\alpha \cdot \beta \in \mathcal{I}$, one can easily compute β if α is known. We also consider a QANIZK = ($\mathsf{Gen_{par}}$, $\mathsf{Gen_{crs}}$,Prv, Sim, Vrfy) for the language

$$L_{\mathsf{mpk}} = \left\{ (\{\mathsf{ct}_{1,i}, \mathsf{ct}_{2,i}\}_{i=1}^2) : \begin{array}{r} \exists(\boldsymbol{x}, M, r_1, s_1, r_2, s_2) \text{ s.t.} \\ \wedge_{i=1,2}(\mathsf{ct}_{1,i} \leftarrow \mathsf{IPFE.Enc}(\mathsf{mpk}_i, \boldsymbol{x}; r_i) \wedge \\ \mathsf{ct}_{2,i} \leftarrow \mathsf{IPFE.Enc}(\mathsf{mpk}_i, M \cdot \boldsymbol{x}; s_i)) \end{array} \right\} \tag{1}$$

```
$\underline{\mathsf{Setup}(1^\lambda, 1^\ell)}$:
    $(\mathsf{msk}_i, \mathsf{mpk}_i) \leftarrow \mathsf{IPFE.Setup}(1^\lambda, 1^\ell)$ for $i = 1, 2$
    $(\mathsf{crs}, \mathsf{trap}) \leftarrow \mathsf{QANIZK.Gen}_{\mathsf{crs}}(\mathsf{par}, \mathsf{mpk})$
    set $\mathsf{MSK} := \mathsf{msk}_1$, $\mathsf{MPK} := (\mathsf{mpk}_1, \mathsf{mpk}_2, \mathsf{crs})$
    return $(\mathsf{MSK}, \mathsf{MPK})$

$\underline{\mathsf{Enc}(\mathsf{MPK}, \tau, \boldsymbol{x}, M)}$:
    parse $\mathsf{MPK} = (\mathsf{mpk}_1, \mathsf{mpk}_2, \mathsf{crs})$
    for $i = 1, 2$
      choose $r_i, s_i \leftarrow \{0,1\}^{l(\lambda)}$                    // $l(\lambda)$ is a polynomial in $\lambda$
      $\mathsf{ct}_{1,i} \leftarrow \mathsf{IPFE.Enc}(\mathsf{mpk}_i, \boldsymbol{x}; r_i)$
      $\mathsf{ct}_{2,i} \leftarrow \mathsf{IPFE.Enc}(\mathsf{mpk}_i, M \cdot \boldsymbol{x}; s_i)$
    $\pi \leftarrow \mathsf{QANIZK.Prv}(\mathsf{crs}, \tau, (\{\mathsf{ct}_{1,i}, \mathsf{ct}_{2,i}\}_{i=1}^2), (\boldsymbol{x}, M, r_1, s_1, r_2, s_2))$
    return $\mathsf{CT} := (\{\mathsf{ct}_{1,i}, \mathsf{ct}_{2,i}\}_{i=1}^2, \pi)$

$\underline{\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, \boldsymbol{y})}$:
    parse $\mathsf{MSK} = \mathsf{msk}_1$, $\mathsf{MPK} = (\mathsf{mpk}_1, \mathsf{mpk}_2, \mathsf{crs})$
    $\mathsf{sk}_{\boldsymbol{y}} \leftarrow \mathsf{IPFE.KeyGen}(\mathsf{mpk}_1, \mathsf{msk}_1, \boldsymbol{y})$
    return $\mathsf{sk}_{\boldsymbol{y}}$

$\underline{\mathsf{Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{CT})}$:
    parse $\mathsf{MPK} = (\mathsf{mpk}_1, \mathsf{mpk}_2, \mathsf{crs})$
    parse $\mathsf{CT} = (\{\mathsf{ct}_{1,i}, \mathsf{ct}_{2,i}\}_{i=1}^2, \pi)$
    if $\mathsf{QANIZK.Vrfy}(\mathsf{crs}, \tau, (\{\mathsf{ct}_{1,i}, \mathsf{ct}_{2,i}\}_{i=1}^2), \pi) = 0$
      return $\perp$
    $\mu \leftarrow \mathsf{IPFE.Dec}(\mathsf{mpk}_1, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{ct}_{1,1})$
    if $\mu = 0$
      return $\perp$
    $\mu' \leftarrow \mathsf{IPFE.Dec}(\mathsf{mpk}_1, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{ct}_{2,1})$
    return $\mu' \cdot \mu^{-1}$
```

Figure 3: CCA secure AHNIPE from IPFE and QANIZK

and par is a part of the system parameters of IPFE. Our CCA secure attribute-hiding NIPE is described in Fig. 3. QANIZK is employed to prove that the two IPFE ciphertexts $\mathsf{ct}_{1,i}, \mathsf{ct}_{2,i}$, main part of the NIPE ciphertext, corresponds to the same attribute $\boldsymbol{x}$ for each $i = 1, 2$. If a ciphertext $\mathsf{CT} = (\{\mathsf{ct}_{1,i}, \mathsf{ct}_{2,i}\}_{i=1}^2, \pi)$ passes the verification, by the correctness of IPFE, $\mu = \langle \boldsymbol{x}, \boldsymbol{y} \rangle$ and We note that one can get rid of the tag from the AHNIPE if a one-time signature scheme is utilized. Specifically at the time of encryption, a tag is randomly chosen from $\mathcal{T}$ and include a signature of the tag into the final ciphertext. Decryption proceeds in the same as before except it first verifies the signature of the tag.

Theorem 1. *Assuming the underlying IPFE is indistinguishability-based secure under chosen plaintext attacks and QANIZK is a one-time simulation sound, the AHNIPE described in Fig. 3 is adaptively attribute-hiding secure under chosen-ciphertext attacks. More specifically, for any PPT adversary $\mathcal{A}$, there exists PPT adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ such that:*

$$\mathsf{Adv}_{\mathcal{A},\mathsf{CCA}}^{\mathsf{AHNIPE}}(\lambda) \leq 4 \cdot \mathsf{Adv}_{\mathcal{B}_1,\mathsf{CPA}}^{\mathsf{IND\text{-}IPFE}}(\lambda) + 3Q_{\mathsf{Dec}} \cdot \mathsf{Adv}_{\mathcal{B}_2}^{\mathsf{OTSS}}(\lambda)$$

where Q_{Dec} denotes the total number of decryption queries made by the adversary.

Proof. To prove this theorem, we consider a sequence of games $\{\text{Game}j\}_{j\in[6]}$ (overview given in Table. 2) where we denote $[6] = \{0, 1, \dots, 10\}$. Let G_j denotes the event $b = b'$ in game j where b' is the bit output by the adversary $\mathcal{A}$. Let $(\tau^*, (\boldsymbol{x}_0, M_0), (\boldsymbol{x}_1, M_1))$ be the challenge tuple submitted by the adversary $\mathcal{A}$.

Game 0: It is the standard security AHNIPE experiment $\mathsf{Expt}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, 0)$ (Def. 6). Let the challenge ciphertext be $\mathsf{CT}^* = (\{\mathsf{ct}^0_{1,i}, \mathsf{ct}^0_{2,i}\}^2_{i=1}, \pi^0)$ where $\mathsf{ct}^0_{1,i}$ and $\mathsf{ct}^0_{2,i}$ are the encryptions of $\boldsymbol{x}_0$ and $M_0 \cdot \boldsymbol{x}_0$ respectively under mpk_i and $\pi^0 \leftarrow \mathsf{QANIZK.Prv}(\mathsf{crs}, \tau^*, (\{\mathsf{ct}^0_{1,i}, \mathsf{ct}^0_{2,i}\}^2_{i=1}), (\boldsymbol{x}_0, M_0, r_1, s_1, r_2, s_2))$.

Game 1: In this game, we use the Sim algorithm of QANIZK to replace the proof π^0 by a simulated proof $\pi^{\mathsf{sim}} \leftarrow \mathsf{QANIZK.Sim}(\mathsf{crs}, \mathsf{trap}, \tau^*, (\{\mathsf{ct}^0_{1,i}, \mathsf{ct}^0_{2,i}\}^2_{i=1}))$. By the perfect zero-knowledge property of QANIZK, the distributions of the challenge ciphertext are identical in the games 0 and 1. Therefore, $\Pr[\mathsf{G}_0] = \Pr[\mathsf{G}_1]$.

Game 2: Here, we replace the ciphertext component $\mathsf{ct}^0_{1,1}$ by an encryption of $\boldsymbol{x}_1$, that is, $\mathsf{ct}^1_{1,1} \leftarrow \mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)$. Since all secret-key queries $\{\boldsymbol{y}\}$ satisfy the condition $\langle \boldsymbol{x}_0, \boldsymbol{y}\rangle = \langle \boldsymbol{x}_1, \boldsymbol{y}\rangle$, there exists an adversary $\mathcal{B}_1$ such that

$$|\Pr[\mathsf{G}_1] - \Pr[\mathsf{G}_2]| \leq \mathsf{Adv}^{\mathsf{IND\text{-}IPFE}}_{\mathcal{B}_1,\mathsf{CPA}}(\lambda)$$

Note that, $\mathcal{B}_1$ can simulate the decryption oracle using msk_2. In particular, for a ciphertext query $(\tau, \mathsf{CT} = (\{\bar{\mathsf{ct}}_{1,i}, \bar{\mathsf{ct}}_{2,i}\}^2_{i=1}, \bar{\pi}), \boldsymbol{y})$, $\mathcal{B}_1$ first verifies the proof $\bar{\pi}$. If the proof passes then it uses the secret-key $\mathsf{sk}_{\boldsymbol{y}} \leftarrow \mathsf{IPFE.KeyGen}(\mathsf{mpk}_2, \mathsf{msk}_2, \boldsymbol{y})$ to decrypt $\bar{\mathsf{ct}}_{1,2}$ and $\bar{\mathsf{ct}}_{2,2}$ and return the message according to the original decryption algorithm.

Game 3: In this game, we perform an additional check on all decryption queries $(\tau, \mathsf{CT} = (\{\bar{\mathsf{ct}}_{1,i}, \bar{\mathsf{ct}}_{2,i}\}^2_{i=1}, \bar{\pi}), \boldsymbol{y})$ using the following circuit with $i = 1$:

$\underline{\mathsf{C}[\mathsf{msk}_i](\bar{\mathsf{ct}}_{1,i}, \bar{\mathsf{ct}}_{2,i})}$:

```
for $j$ runs from 1 to $\ell$
    $\mathsf{sk}_{\boldsymbol{e}_j} \leftarrow \mathsf{IPFE.KeyGen}(\mathsf{mpk}_i, \mathsf{msk}_i, \boldsymbol{e}_j)$    // $\{\boldsymbol{e}_j\}_{j\in[\ell]}$ is the standard basis of $\mathcal{I}^\ell$
    $z_j \leftarrow \mathsf{IPFE.Dec}(\mathsf{mpk}_i, \mathsf{sk}_{\boldsymbol{e}_j}, \bar{\mathsf{ct}}_{1,i})$
    $z'_j \leftarrow \mathsf{IPFE.Dec}(\mathsf{mpk}, \mathsf{sk}_{\boldsymbol{e}_i}, \bar{\mathsf{ct}}_{2,i})$
set $\boldsymbol{z} \leftarrow (z_1, \dots, z_\ell)$ and $\boldsymbol{z}' \leftarrow (z'_1, \dots, z'_\ell)$
if $\exists M \in \mathcal{M}$ s.t. $\boldsymbol{z}' = M \cdot \boldsymbol{z}$, return 1
else return 0
```

Observe that, $\mathsf{C}[\mathsf{msk}_i]$ uses secret-keys $\{\mathsf{sk}_{\boldsymbol{e}_j}\}_{j\in[\ell]}$ to verify (without using the tag τ) that $\bar{\mathsf{ct}}_{1,1}, \bar{\mathsf{ct}}_{2,1}$ are encryptions of the vectors $\boldsymbol{z}, \boldsymbol{z}'$ such that $\boldsymbol{z}' = M \cdot \boldsymbol{z}$ for some $M \in \mathcal{M}$. If this additional check fails for a ciphertext $\mathsf{CT} = (\{\bar{\mathsf{ct}}_{1,i}, \bar{\mathsf{ct}}_{2,i}\}^2_{i=1}, \bar{\pi})$, but passes through the verification $\mathsf{QANIZK.Vrfy}(\mathsf{crs}, \tau, (\{\bar{\mathsf{ct}}_{1,i}, \bar{\mathsf{ct}}_{2,i}\}^2_{i=1}), \bar{\pi})$ then the tuple $(\tau, (\{\bar{\mathsf{ct}}_{1,i}, \bar{\mathsf{ct}}_{2,i}\}^2_{i=1}), \bar{\pi})$ violates the one-time simulation soundness of QANIZK (Def. 7). Hence, running through all the decryption queries we get a PPT adversary $\mathcal{B}_2$ such that

$$|\Pr[\mathsf{G}_2] - \Pr[\mathsf{G}_3]| \leq Q_{\mathsf{Dec}} \cdot \mathsf{Adv}^{\mathsf{OTSS}}_{\mathcal{B}_2}(\lambda)$$

Game 4: Here, we replace the ciphertext component $\mathsf{ct}^0_{2,1}$ by an encryption of $M_1 \cdot \boldsymbol{x}_1$, that is, $\mathsf{ct}^1_{2,1} \leftarrow \mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_1 \cdot \boldsymbol{x}_1; s_1)$. Since all secret-key queries $\{\boldsymbol{y}\}$ satisfy the condition $\langle M_0 \cdot \boldsymbol{x}_0, \boldsymbol{y}\rangle = \langle M_1 \cdot \boldsymbol{x}_1, \boldsymbol{y}\rangle$, we get the following

$$|\Pr[\mathsf{G}_3] - \Pr[\mathsf{G}_4]| \leq \mathsf{Adv}^{\mathsf{IND\text{-}IPFE}}_{\mathcal{B}_1,\mathsf{CPA}}(\lambda)$$

As described in game 3, here also $\mathcal{B}_1$ uses msk_2 in a similar fashion to answer the decryption queries of $\mathcal{A}$.

Game 5: In this game, we replace the ciphertext component $\mathsf{ct}^0_{1,2}$ by an encryption of $\boldsymbol{x}_1$, that is, $\mathsf{ct}^1_{1,2} \leftarrow \mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_1; r_2)$. Since all secret-key queries $\{\boldsymbol{y}\}$ satisfy the condition $\langle \boldsymbol{x}_0, \boldsymbol{y}\rangle = \langle \boldsymbol{x}_1, \boldsymbol{y}\rangle$, there exists an adversary $\mathcal{B}_1$ such that

$$|\Pr[\mathsf{G}_4] - \Pr[\mathsf{G}_5]| \leq \mathsf{Adv}^{\mathsf{IND\text{-}IPFE}}_{\mathcal{B}_1,\mathsf{CPA}}(\lambda)$$

Now, $\mathcal{B}_1$ uses msk_1 to simulate the decryption oracle as in the original scheme.

Game 6: It is similar to the previous game except that we perform an additional check on all decryption queries $(\tau, \mathsf{CT} = (\{\bar{\mathsf{ct}}_{1,i}, \bar{\mathsf{ct}}_{2,i}\}^2_{i=1}, \bar{\pi}), \boldsymbol{y})$ using the circuit $\mathsf{C}[\mathsf{msk}_2]$ (described in game 3) on the component $(\bar{\mathsf{ct}}_{1,2}, \bar{\mathsf{ct}}_{2,2})$ of the ciphertext. If this additional check fails we return $\perp$ in the decryption oracle. Using the similar argument as in game 3, we obtain a PPT adversary $\mathcal{B}_2$ such that

$$|\Pr[\mathsf{G}_5] - \Pr[\mathsf{G}_6]| \leq Q_{\mathsf{Dec}} \cdot \mathsf{Adv}^{\mathsf{OTSS}}_{\mathcal{B}_2}(\lambda)$$

Game 7: Here, we replace the ciphertext component $\mathsf{ct}^0_{2,2}$ by an encryption of $M_1 \cdot \boldsymbol{x}_1$, that is, $\mathsf{ct}^1_{2,2} \leftarrow \mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_1 \cdot \boldsymbol{x}_1; s_2)$. Since all secret-key queries $\{\boldsymbol{y}\}$ satisfy the condition $\langle M_0 \cdot \boldsymbol{x}_0, \boldsymbol{y}\rangle = \langle M_1 \cdot \boldsymbol{x}_1, \boldsymbol{y}\rangle$, there exists an adversary $\mathcal{B}_1$ such that

$$|\Pr[\mathsf{G}_6] - \Pr[\mathsf{G}_7]| \leq \mathsf{Adv}^{\mathsf{IND\text{-}IPFE}}_{\mathcal{B}_1,\mathsf{CPA}}(\lambda)$$

Game 8: In this game, we drop the additional checks defined by $\mathsf{C}[\mathsf{msk}_i]$ for $i = 1, 2$, on the decryption queries. By one-time simulation soundness of QANIZK, we have the following

$$|\Pr[\mathsf{G}_7] - \Pr[\mathsf{G}_8]| \leq Q_{\mathsf{Dec}} \cdot \mathsf{Adv}^{\mathsf{OTSS}}_{\mathcal{B}_2}(\lambda)$$

Game 9: We now compute the proof of the challenge ciphertext using Prv of the QANIZK. That is, $\pi^1 \leftarrow \mathsf{QANIZK.Prv}(\mathsf{crs}, \tau, (\{\mathsf{ct}^1_{1,i}, \mathsf{ct}^1_{2,i}\}^2_{i=1}), (\boldsymbol{x}_1, M_1, r_1, s_1, r_2, s_2))$. By the perfect zero-knowledge property of QANIZK we have $\Pr[\mathsf{G}_8] = \Pr[\mathsf{G}_9]$.

We can see that game 9 is eventually the standard experiment $\mathsf{Expt}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, 1)$ (Def. 6). Therefore, combining all the probabilities we complete the proof. □

Remark 1. *From the generic transformation it is clear that we need QANIZK for the CCA security of AHNIPE. Therefore, dropping the QANIZK and considering the IPFEs of [3] our transformation accomplishes CPA secure AHNIPEs based on various assumptions such as DDH, LWE and DCR. For CCA security of the AHNIPE, any one-time simulation sound NIZK (OTSS-NIZK) is sufficient. We have seen constructions of NIZK proof systems for any arbitrary NP language based on bilinear pairing [21] and (plain) LWE assumption [35]. A transformation from NIZK to OTSS-NIZK is also well know [37]. Using such OTSS-NIZK proof system we can get rid of the tag from our AHNIPE and the all decryption queries of the form $(CT^*, \boldsymbol{y})$ should satisfy that $\langle \boldsymbol{x}_b, \boldsymbol{y}\rangle = 0$ for $b \in \{0, 1\}$. However, the reason behind selecting QANIZK over OTSS-NIZK for our application is that QANIZK proofs [1, 26] for certain languages are much shorter than the existing OTSS-NIZK. Consequently, the ciphertext size is (significantly) reduced as shown in the next section.*

4 Concrete Construction: AHNIPE from DDH

In this section, we present a more efficient construction of AHNIPE from plain DDH assumption. First, we recall the DDH-based IPFE of [3]. Consider a cyclic group $\mathbb{G}$ of prime order p. Let $\mathsf{mpk} = ([\mathbf{a}], [\mathbf{Ua}])$, $\mathsf{msk} = \mathbf{U}$ where $\mathbf{a} = (1, a) \leftarrow \mathbb{Z}^2_p$ and $\mathbf{U} \leftarrow \mathbb{Z}^{\ell\times 2}_p$. The ciphertext and secret-key are computed as

$$\mathsf{Enc}(\mathsf{mpk}, \boldsymbol{x}) = \begin{bmatrix} \mathbf{a}r \\ \boldsymbol{x} + \mathbf{Ua}r \end{bmatrix} \in \mathbb{G}^{\ell+2} \text{ and } \mathsf{KeyGen}(\mathsf{msk}, \mathbf{y}) = \begin{pmatrix} -\mathbf{U}^\top\mathbf{y} \\ \mathbf{y} \end{pmatrix} \in \mathbb{Z}^{\ell+2}$$

Game	$\mathsf{ct}_{j,1}$ for $j = 1, 2$	$\mathsf{ct}_{j,2}$ for $j = 1, 2$	π	CHECK
0	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_0; r_1)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_0 \cdot \boldsymbol{x}_0; s_1)$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_0; r_2)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_0 \cdot \boldsymbol{x}_0; s_2)$	π^0	–
1	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_0; r_1)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_0 \cdot \boldsymbol{x}_0; s_1)$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_0; r_2)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_0 \cdot \boldsymbol{x}_0; s_2)$	$\boxed{\pi^{\mathsf{sim}}}$	–
2	$\boxed{\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)}$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_0 \cdot \boldsymbol{x}_0; s_1)$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_0; r_2)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_0 \cdot \boldsymbol{x}_0; s_2)$	π^{sim}	–
3	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_0 \cdot \boldsymbol{x}_0; s_1)$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_0; r_2)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_0 \cdot \boldsymbol{x}_0; s_2)$	π^{sim}	$\boxed{\mathsf{C}[\mathsf{msk}_1]}$
4	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)$ $\boxed{\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_1 \cdot \boldsymbol{x}_1; s_1)}$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_0; r_2)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_0 \cdot \boldsymbol{x}_0; s_2)$	π^{sim}	$\mathsf{C}[\mathsf{msk}_1]$
5	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_1 \cdot \boldsymbol{x}_1; s_1)$	$\boxed{\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_1; r_2)}$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_0 \cdot \boldsymbol{x}_0; s_2)$	π^{sim}	$\mathsf{C}[\mathsf{msk}_1]$
6	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_1 \cdot \boldsymbol{x}_1; s_1)$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_1; r_2)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_0 \cdot \boldsymbol{x}_0; s_2)$	π^{sim}	$\mathsf{C}[\mathsf{msk}_1]$ $\boxed{\mathsf{C}[\mathsf{msk}_2]}$
7	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_1 \cdot \boldsymbol{x}_1; s_1)$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_1; r_2)$ $\boxed{\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_1 \cdot \boldsymbol{x}_1; s_2)}$	π^{sim}	$\mathsf{C}[\mathsf{msk}_1]$ $\mathsf{C}[\mathsf{msk}_2]$
8	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_1 \cdot \boldsymbol{x}_1; s_1)$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_1; r_2)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_1 \cdot \boldsymbol{x}_1; s_2)$	π^{sim}	$\boxed{-}$
9	$\mathsf{IPFE.Enc}(\mathsf{mpk}_1, \boldsymbol{x}_1; r_1)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_1, M_1 \cdot \boldsymbol{x}_1; s_1)$	$\mathsf{IPFE.Enc}(\mathsf{mpk}_2, \boldsymbol{x}_1; r_2)$ $\mathsf{IPFE.Enc}(\mathsf{mpk}_2, M_1 \cdot \boldsymbol{x}_1; s_2)$	$\boxed{\pi^1}$	–

Table 2: An overview of the games used in the proof of Th. 1

where $r \leftarrow \mathbb{Z}_p$. Let $\mathbf{c} = \mathbf{a}r$ and $\mathbf{c}_1 = \boldsymbol{x} + \mathbf{U}\mathbf{a}r$. The decryption computes a discrete log of the inner product of $(\mathbf{c}, \mathbf{c}_1)$ and $\mathsf{sk}_{\boldsymbol{y}}$. To instantiate our AHNIPE with this IPFE, one encrypts an attribute-message pair $(\boldsymbol{x}, M) \in \mathbb{Z}_p^\ell \times \mathbb{Z}_p$ as

$$\mathsf{Enc}(\mathsf{mpk}, \boldsymbol{x}) = \begin{bmatrix} \mathbf{c} = \mathbf{a}r \\ \mathbf{c}_1 = \boldsymbol{x} + \mathbf{U}\mathbf{a}r \end{bmatrix} \text{ and } \mathsf{Enc}(\mathsf{mpk}, M \cdot \boldsymbol{x}) = \begin{bmatrix} \mathbf{c}' = \mathbf{a}s \\ \mathbf{c}'_1 = M \cdot \boldsymbol{x} + \mathbf{U}\mathbf{a}r \end{bmatrix}$$

where $r, s \leftarrow \mathbb{Z}_p$ and computes a proof π that will verify that both $\mathbf{c}$ and $\mathbf{c}'$ belongs to spanning set of $\mathbf{a}$ (instead of proving the whole encryption process). In particular, we consider QANIZK for the language

$$L_{[\mathbf{a}]} = \{([\mathbf{c}], [\mathbf{c}']) : \exists (r, s) \in \mathbb{Z}_p^2 \text{ s.t. } \mathbf{c} = \mathbf{a}r \wedge \mathbf{c}' = \mathbf{a}s\}$$

If we employ the QANIZK of [26] based on KerMDH assumption (with $k = 1$), such a proof π consists of four group elements. Therefore, a ciphertext of the AHNIPE contains $2\ell + 8$ group elements.

```
$\underline{\mathsf{Setup}(1^\lambda, 1^\ell, \mathcal{PG})}$:
    1. $\mathbf{a} = (1, a) \leftarrow \mathbb{Z}_p^2$, $(\mathbf{U}_1, \mathbf{U}_2) \leftarrow (\mathbb{Z}_p^{\ell \times 2})$
    2. $\mathsf{msk} := (\mathbf{U}_1, \mathbf{U}_2)$, $\mathsf{mpk} := ([\mathbf{a}]_1, [\mathbf{U}_1\mathbf{a}]_1, [\mathbf{U}_2\mathbf{a}]_1)$
    3. $\boldsymbol{\alpha} \leftarrow \mathcal{D}_1$, $\mathbf{K}_1, \mathbf{K}_2 \leftarrow \mathbb{Z}_p^{2\times 2}$
    4. $\boldsymbol{\vartheta}_1 := \mathbf{K}_1\mathbf{a}$, $\boldsymbol{\vartheta}_2 := \mathbf{K}_2\mathbf{a}$, $\boldsymbol{\beta}_1 := \mathbf{K}_1\boldsymbol{\alpha}$, $\boldsymbol{\beta}_2 := \mathbf{K}_2\boldsymbol{\alpha}$
    5. $\mathsf{crs} := ([\boldsymbol{\vartheta}_1]_1, [\boldsymbol{\vartheta}_2]_1, [\boldsymbol{\beta}_1]_2, [\boldsymbol{\beta}_2]_2, [\boldsymbol{\alpha}]_2)$, $\mathsf{trap} := (\mathbf{K}_1, \mathbf{K}_2)$
    6. return $\mathsf{MSK} := (\mathsf{msk}, \mathsf{trap})$, $\mathsf{MPK} := (\mathsf{mpk}, \mathsf{crs})$

$\underline{\mathsf{Enc}(\mathsf{MPK} = (\mathsf{mpk}, \mathsf{crs}), \tau, \boldsymbol{x}, M)}$:
    1. $\mathsf{mpk} = ([\mathbf{a}]_1, [\mathbf{U}_1\mathbf{a}]_1, [\mathbf{U}_2\mathbf{a}]_1)$, $\mathsf{crs} := ([\boldsymbol{\vartheta}_1]_1, [\boldsymbol{\vartheta}_2]_1, [\boldsymbol{\beta}_1]_2, [\boldsymbol{\beta}_2]_2, [\boldsymbol{\alpha}]_2)$
    2. $r \leftarrow \mathbb{Z}_p$, $\mathbf{c} := \mathbf{a}r$, $\pi := [(\boldsymbol{\vartheta}_1 + \tau\boldsymbol{\vartheta}_2)r]_1 \in \mathbb{G}_1^2$
    3. $[\mathbf{ct}]_1 := \begin{bmatrix} \mathbf{c} \\ \boldsymbol{x} + \mathbf{U}_1\mathbf{c} \\ M \cdot \boldsymbol{x} + \mathbf{U}_2\mathbf{c} \end{bmatrix}_1 \in \mathbb{G}_1^{2\ell+2}$
    4. return $\mathsf{CT} := ([\mathbf{ct}]_1, \pi) \in \mathbb{G}_1^{2\ell+4}$

$\underline{\mathsf{KeyGen}(\mathsf{MSK} = (\mathsf{msk}, \mathsf{trap}), \boldsymbol{y})}$:
    1. $\mathsf{msk} = (\mathbf{U}_1, \mathbf{U}_2)$
    2. return $\mathsf{sk}_{\boldsymbol{y}} := \begin{pmatrix} -\mathbf{U}_1^\top \boldsymbol{y} \\ -\mathbf{U}_2^\top \boldsymbol{y} \\ \boldsymbol{y} \end{pmatrix} \in \mathbb{Z}^{\ell+4}$

$\underline{\mathsf{Dec}(\mathsf{MPK} = (\mathsf{mpk}, \mathsf{crs}), \tau, \mathsf{sk}_{\boldsymbol{y}}, \mathsf{CT} = ([\mathbf{ct}]_2, \pi))}$:
    1. $\mathsf{mpk} = ([\mathbf{a}]_1, [\mathbf{U}_1\mathbf{a}]_1, [\mathbf{U}_2\mathbf{a}]_1)$, $\mathsf{crs} := ([\boldsymbol{\vartheta}_1]_1, [\boldsymbol{\vartheta}_2]_1, [\boldsymbol{\beta}_1]_2, [\boldsymbol{\beta}_2]_2, [\boldsymbol{\alpha}]_2)$
    2. $[\mathbf{ct}]_1 = \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix}_1, [\boldsymbol{v}_1]_1 := \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_1 \end{bmatrix}_1, [\boldsymbol{v}_2]_1 := \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_2 \end{bmatrix}_1$
    3. if $e(\pi, [\boldsymbol{\alpha}]_2) \neq e([\mathbf{c}]_1, [\boldsymbol{\beta}_1 + \tau\boldsymbol{\beta}_2]_2)$, return $\perp$
    4. $\mathsf{sk}_{\boldsymbol{y}} = \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \boldsymbol{y} \end{pmatrix}, \varsigma_1 := \begin{pmatrix} \mathbf{s}_1 \\ \boldsymbol{y} \end{pmatrix}, \varsigma_2 := \begin{pmatrix} \mathbf{s}_2 \\ \boldsymbol{y} \end{pmatrix}$
    5. $\mu := [\langle \boldsymbol{v}_1, \varsigma_1 \rangle]_1$, $\mu' := [\langle \boldsymbol{v}_2, \varsigma_2 \rangle]_1$
    6. if $\mu = [0]_1$, return $\perp$
    7. return $\log_{g_1}(\mu' \cdot \mu^{-1})$
```

Figure 4: CCA secure AHNIPE from DDH assumption

We aim to construct more efficient version of the AHNIPE where the ciphertext consists of only $2\ell + 4$ group elements. The main observation is that, instead of considering two independent encryptions as above, we encrypt $\boldsymbol{x}$ and $M \cdot \boldsymbol{x}$ together. More precisely, a ciphertext corresponding to $(\boldsymbol{x}, M)$ and a secret-key associated to $\boldsymbol{y}$ become

$$\mathsf{Enc}(\mathsf{MPK}, \boldsymbol{x}, M) = \begin{bmatrix} \mathbf{a}r \\ \boldsymbol{x} + \mathbf{U}_1\mathbf{a}r \\ M \cdot \boldsymbol{x} + \mathbf{U}_2\mathbf{a}r \end{bmatrix} \text{ and } \mathsf{KeyGen}(\mathsf{MSK}, \mathbf{y}) = \begin{pmatrix} -\mathbf{U}_1^\top \mathbf{y} \\ -\mathbf{U}_2^\top \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

where $\mathsf{MPK} = ([\mathbf{a}], [\mathbf{U}_1\mathbf{a}], [\mathbf{U}_2\mathbf{a}]) \in \mathbb{G}^{2\ell+2}$ and $\mathsf{MSK} = (\mathbf{U}_1, \mathbf{U}_2) \in (\mathbb{Z}_p^{\ell\times 2})^2$. Consequently, the ciphertext includes a shorter QANIZK proof of a statement belongs to a language defined by

$$L_{[\mathbf{a}]} = \{[\mathbf{c}] : \exists r \in \mathbb{Z}_p \text{ s.t. } \mathbf{c} = \mathbf{a}r\} \tag{2}$$

and each proof consists of only two group elements. Hence, the ciphertext contains $2\ell+4$ group elements and a secret-key belongs to $\mathbb{Z}^4$ (excluding the predicate vector).

We show how to utilize the shared randomness technique of [8] to enable much more efficient QANIZK proof where only four pairing operations are required to verify the proof. The main idea is that instead of two independent encryption, we encrypt the vectors $\boldsymbol{x}$ and $M \cdot \boldsymbol{x}$ with the same randomness which helps us to reduce the ciphertext size and achieve an efficient decryption procedure. In particular, a ciphertext will have only $(2\ell + 4)$ group elements. We consider the QANIZK of Kiltz and Wee [26] based on KerMDH assumption (with $k = 1$) for the language $L_{[\mathbf{a}]} = \{[\mathbf{c}] : \exists r \in \mathbb{Z}_p \text{ s.t. } \mathbf{c} = \mathbf{a}r\}$. We describe our AHNIPE for $\mathcal{P} = \mathcal{Q} = \mathbb{Z}_p^\ell$, $\mathcal{I} = \mathcal{T} = \mathbb{Z}_p$ and $\mathcal{M} \subset \mathcal{I}$, in Fig. 4 where $\mathcal{PG} = \{\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, p, g_1, g_2, e\} \leftarrow \mathsf{GGen}(1^\lambda)$. We assume that $\mathcal{M}$ is polynomially bounded so that messages can be recovered by discrete logarithm.

Correctness. For all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{Z}_p^\ell, \tau \in \mathbb{Z}_p, M \in \mathcal{M}$ we have

$$\begin{aligned} e(\pi, [\boldsymbol{\alpha}]_2) &= e([(\boldsymbol{\vartheta}_1 + \tau\boldsymbol{\vartheta}_2)r]_1, [\boldsymbol{\alpha}]_2) \\ &= e([(\mathbf{K}_1 + \tau\mathbf{K}_2)\mathbf{c}]_1, [\boldsymbol{\alpha}]_2) \qquad\qquad (\text{when } \mathbf{c} = \mathbf{a}r) \\ &= e([\mathbf{c}]_1, [(\mathbf{K}_1 + \tau\mathbf{K}_2)\boldsymbol{\alpha}]_2) \\ &= e([\mathbf{c}]_1, [\boldsymbol{\beta}_1 + \tau\boldsymbol{\beta}_2) \end{aligned}$$

which verifies the ciphertext component $\mathbf{c} = \mathbf{a}r$. Next, we note that

$$\langle \boldsymbol{v}_1, \varsigma_1 \rangle = \begin{pmatrix} \mathbf{c} \\ \boldsymbol{x} + \mathbf{U}_1\mathbf{c} \end{pmatrix}^\top \begin{pmatrix} -\mathbf{U}_1^\top \boldsymbol{y} \\ \boldsymbol{y} \end{pmatrix} = -(\mathbf{U}_1\mathbf{c})^\top \boldsymbol{y} + (\boldsymbol{x} + \mathbf{U}_1\mathbf{c})^\top \boldsymbol{y} = \boldsymbol{x}^\top \boldsymbol{y}.$$

Therefore, $\mu = [\langle \boldsymbol{x}, \boldsymbol{y} \rangle]_1$ and similarly one can show that $\mu = [M \cdot \langle \boldsymbol{x}, \boldsymbol{y} \rangle]_1$. If $\mu \neq [0]_1$, we recover the message as $M = \log_{g_1}(\mu' \cdot \mu^{-1})$.

Theorem 2. *Assuming the DDH and the KerMDH assumptions hold in the groups $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively, the AHNIPE described in Fig. 4 is adaptively attribute-hiding secure under chosen-ciphertext attacks. More specifically, for any PPT adversary $\mathcal{A}$, there exist PPT adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ such that:*

$$\mathsf{Adv}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(\lambda) \leq 2 \cdot \mathsf{Adv}^{\mathsf{DDH}}_{\mathcal{B}_1,\mathsf{GGen}_1}(\lambda) + 2Q_{\mathsf{Dec}} \cdot \mathsf{Adv}^{\mathsf{KerMDH}}_{\mathcal{B}_2,\mathsf{GGen}_2}(\lambda) + \mathsf{negl}(\lambda)$$

where Q_{Dec} denotes the total number of decryption queries made by the adversary.

Proof. We prove this theorem using a sequence of hybrid games $\{\text{Game}j\}_{j\in[7]}$ described in Fig. 5 where game 0 is the standard AHNIPE experiment $\mathsf{Expt}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, 0)$ (Def. 6). Let G_j denotes the event $b = b'$ in game j where b' is the bit output by the adversary $\mathcal{A}$. Further, we assume that $\mathcal{A}$'s queries are consistent with the restrictions described in Def. 6.

Game 1: In this game, we compute the proof π^* for the statement $[\mathbf{c}^*]_1$ without using the witness r, that is, we set $\pi^* := [(\mathbf{K}_1 + \tau^*\mathbf{K}_2)\mathbf{c}^*]_1$. The distributions of π^* in both the games 0 and 1 are identical since

$$\pi^* = \underbrace{[(\boldsymbol{\vartheta}_1 + \tau^*\boldsymbol{\vartheta}_2)r]_1}_{\text{(Game 0)}} = [(\mathbf{K}_1 + \tau^*\mathbf{K}_2)\mathbf{a}r]_1 = \underbrace{[(\mathbf{K}_1 + \tau^*\mathbf{K}_2)\mathbf{c}^*]_1}_{\text{(Game 1)}} \qquad (3)$$

Therefore, we have $\Pr[\mathsf{G}_0] = \Pr[\mathsf{G}_1]$.

Game 2: It is exactly same game 1 except that we choose $\mathbf{c}^*$ uniformly at random from $\mathbb{Z}_p^2$. For indistinguishability between games 1 and 2, we rely on DDH assumption in group $\mathbb{G}_1$.

Suppose, $\mathcal{B}_1$ be a DDH adversary which receives a tuple $([\mathbf{a}]_1, [\mathbf{c}^*]_1)$ from it's challenger. It then selects $(\mathbf{U}_1, \mathbf{U}_2) \leftarrow (\mathbb{Z}_p^{\ell\times 2})$, $\boldsymbol{\alpha} \leftarrow \mathcal{D}_1$, $\mathbf{K}_1, \mathbf{K_1} \leftarrow \mathbb{Z}_p^{2\times 2}$ and simulates $\mathcal{A}$ as defined in Fig. 5, using $[\mathbf{c}^*]_1$ to compute $[\mathbf{ct}^*]_1$. We note that, if $\mathbf{c}^* = \mathbf{a}r$ for some $r \in \mathbb{Z}_p$ then $\mathcal{B}_1$ plays the role

```
Game $\boldsymbol{j}$, $j \in [7] = \{0,1,2,3,4,5,6,7\}$

 1. $j \in [7] \setminus \{3,4\}$, $\mathbf{a} = (1, a) \leftarrow \mathbb{Z}_p^2$
    $j \in \{3,4\}$,    $\mathbf{a} = (1, a) \leftarrow \mathbb{Z}_p^2, \mathbf{a}^\perp \leftarrow \mathbb{Z}_p^2 \setminus \{\mathbf{0}\}$ s.t. $\mathbf{a}^\top \mathbf{a}^\perp = 0$
 2. $j \in [7]$,    $(\mathbf{U}_1, \mathbf{U}_2) \leftarrow (\mathbb{Z}_p^{\ell \times 2})^2$
 3. $j \in [7]$,    $\mathsf{msk} := (\mathbf{U}_1, \mathbf{U}_2)$, $\mathsf{mpk} := ([\mathbf{a}]_1, [\mathbf{U}_1\mathbf{a}]_1, [\mathbf{U}_2\mathbf{a}]_1)$
 4. $j \in [7]$,    $\boldsymbol{\alpha} \leftarrow \mathcal{D}_1$, $\mathbf{K}_1, \mathbf{K}_2 \leftarrow \mathbb{Z}_p^{2\times 2}$
 5. $j \in [7]$,    $\boldsymbol{\vartheta}_1 := \mathbf{K}_1\mathbf{a}$, $\boldsymbol{\vartheta}_2 := \mathbf{K}_2\mathbf{a}$, $\boldsymbol{\beta}_1 := \mathbf{K}_1\boldsymbol{\alpha}$, $\boldsymbol{\beta}_2 := \mathbf{K}_2\boldsymbol{\alpha}$
 6. $j \in [7]$,    $\mathsf{crs} := ([\boldsymbol{\vartheta}_1]_1, [\boldsymbol{\vartheta}_2]_1, [\boldsymbol{\beta}_1]_2, [\boldsymbol{\beta}_2]_2, [\boldsymbol{\alpha}]_2)$, $\mathsf{trap} := (\mathbf{K}_1, \mathbf{K}_2)$
 7. return $\mathsf{MSK} := (\mathsf{msk}, \mathsf{trap})$, $\mathsf{MPK} := (\mathsf{mpk}, \mathsf{crs})$
 8. $(\tau^*, (\boldsymbol{x}_0, M_0), (\boldsymbol{x}_1, M_1)) \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{KG}}(\cdot), \mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)}(\mathsf{MPK})$
 9. $\mathsf{CT}^* \leftarrow \mathcal{O}_{\mathsf{Enc}}(\tau^*, \{\boldsymbol{x}_b, M_b\}_{b \in \{0,1\}})$
10. $b' \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{KG}}(\cdot), \mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)}(\mathsf{CT}^*)$
11. return $b'$

$\mathcal{O}_{\mathsf{Enc}}(\tau^*, \{\boldsymbol{x}_b, M_b\}_{b\in\{0,1\}})$:
 1. $j \in \{0,1,6,7\}$,    $r \leftarrow \mathbb{Z}_p$, $\mathbf{c}^* := \mathbf{a}r$
    $j \in \{2,3,4,5\}$,    $\mathbf{c}^* \leftarrow \mathbb{Z}_p$
 2. $j \in \{0,7\}$,    $\pi^* := [(\boldsymbol{\vartheta}_1 + \tau^*\boldsymbol{\vartheta}_2)r]_1$
    $j \in [7] \setminus \{0,7\}$,    $\pi^* := [(\mathbf{K}_1 + \tau^*\mathbf{K}_2)\mathbf{c}^*]_1$
 3. $j \in \{0,1,2,3\}$,    $[\mathbf{ct}^*]_1 := \begin{bmatrix} \mathbf{c}^* \\ \boldsymbol{x}_0 + \mathbf{U}_1\mathbf{c}^* \\ M_0 \cdot \boldsymbol{x}_0 + \mathbf{U}_2\mathbf{c}^* \end{bmatrix}_1$
 4. $j \in \{4,5,6,7\}$,    $[\mathbf{ct}^*]_1 := \begin{bmatrix} \mathbf{c}^* \\ \boldsymbol{x}_1 + \mathbf{U}_1\mathbf{c}^* \\ M_1 \cdot \boldsymbol{x}_1 + \mathbf{U}_2\mathbf{c}^* \end{bmatrix}_1$
 5. return $\mathsf{CT} := ([\mathbf{ct}^*]_1, \pi^*)$

$\mathcal{O}_{\mathsf{KG}}(\boldsymbol{y})$:
 1. return $\mathsf{sk}_{\boldsymbol{y}} := \begin{pmatrix} -\mathbf{U}_1^\top \boldsymbol{y} \\ -\mathbf{U}_2^\top \boldsymbol{y} \\ \boldsymbol{y} \end{pmatrix}$

$\mathcal{O}_{\mathsf{Dec}}(\tau, \mathsf{CT}, \boldsymbol{y})$:
 1. $j \in [7]$,    if $\tau = \tau^*$, return $\perp$
 2. $j \in [7]$,    $[\mathbf{ct}]_1 = \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix}_1$, $[\boldsymbol{v}_1]_1 := \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_1 \end{bmatrix}_1$, $[\boldsymbol{v}_2]_1 := \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_2 \end{bmatrix}_1$
 3. $j \in [7] \setminus \{3,4\}$, if $e(\pi, [\boldsymbol{\alpha}]_2) \neq e([\mathbf{c}]_1, [\boldsymbol{\beta}_1 + \tau\boldsymbol{\beta}_2]_2)$, return $\perp$
    $j \in \{3,4\}$,    if $(e(\pi, [\boldsymbol{\alpha}]_2) \neq e([\mathbf{c}]_1, [\boldsymbol{\beta}_1 + \tau\boldsymbol{\beta}_2]_2) \wedge [\mathbf{c}^\top \mathbf{a}^\perp]_1 \neq [0]_1)$,
                                          return $\perp$
 4. $j \in [7]$,    $\mathsf{sk}_{\boldsymbol{y}} := \begin{pmatrix} -\mathbf{U}_1^\top \boldsymbol{y} \\ -\mathbf{U}_2^\top \boldsymbol{y} \\ \boldsymbol{y} \end{pmatrix} = \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \boldsymbol{y} \end{pmatrix}$, $\varsigma_1 := \begin{pmatrix} \mathbf{s}_1 \\ \boldsymbol{y} \end{pmatrix}$, $\varsigma_2 := \begin{pmatrix} \mathbf{s}_2 \\ \boldsymbol{y} \end{pmatrix}$
 5. $j \in [7]$,    $\mu := [\langle \boldsymbol{v}_1, \varsigma_1 \rangle]_1$, $\mu' := [\langle \boldsymbol{v}_2, \varsigma_2 \rangle]_1$
 6. $j \in [7]$,    if $\mu = [0]_1$, return $\perp$
 7. return $\log_{g_1}(\mu' \cdot \mu^{-1})$
```

Figure 5: Sequence of Games used in the proof of Th. 2

of a challenger in game 1, and if $\mathbf{c}^*$ is picked uniformly at random from $\mathbb{Z}_p^2$ then $\mathcal{B}_1$ simulates game 2. By DDH assumption we get $|\Pr[\mathsf{G}_1] - \Pr[\mathsf{G}_2]| \le \mathsf{Adv}^{\mathsf{DDH}}_{\mathcal{B}_1,\mathsf{GGen}_1}(\lambda)$.

In this game, we observe that the probability of $\mathbf{c}^*$ belonging to $\mathrm{Span}(\mathbf{a})$ is negligible, precisely $(1 - \frac{1}{p})$. Hence, there exits a vector $\mathbf{a}^\top \in \mathbb{Z}_p$ such that $\mathbf{a}^\top \mathbf{a}^\perp = 0$ and $\mathbf{c}^{*\top}\mathbf{a}^\perp = 1$ whenever $\mathbf{c}^* \notin \mathrm{Span}(\mathbf{a})$.

Game 3: It is identical to game 2, except that in the decryption oracle we perform an additional check on the queried ciphertext $\mathsf{CT} = ([\mathbf{ct}]_1, \pi)$. With the usual verification of $([\mathbf{c}]_1, \pi)$, the oracle also returns $\perp$ if $[\mathbf{c}^\top \mathbf{a}^\perp]_1 \neq [0]_1$ where $[\mathbf{c}]$ is the first component of $[\mathbf{ct}]_1$.

Suppose the additional check fails, but the tuple $(\tau, [\mathbf{c}]_1, \pi)$ passes the verification $e(\pi, [\boldsymbol{\alpha}]_2) = e([\mathbf{c}]_1, [\boldsymbol{\beta}_1 + \tau\boldsymbol{\beta}_2]_2)$, then we construct a PPT adversary $\mathcal{B}_2$ against KerMDH assumption in group $\mathbb{G}_2$ (Def. 2). On receiving a challenge vector $[\boldsymbol{\alpha}]_2$ from it's challenger, $\mathcal{B}_2$ picks $\mathbf{a} = (1, a) \leftarrow \mathbb{Z}_p^2$, $(\mathbf{U}_1, \mathbf{U}_2) \leftarrow (\mathbb{Z}_p^{\ell \times 2})$, $\mathbf{K}_1, \mathbf{K_1} \leftarrow \mathbb{Z}_p^{2\times 2}$ and simulates the game for $\mathcal{A}$ as defined in Fig. 5. Note that, $\mathcal{A}$ already gets a simulated proof as $\pi^* = [(\mathbf{K}_1 + \tau^*\mathbf{K}_2)\mathbf{c}]_1$ included in the challenge ciphertext. If $\mathcal{A}$ submits a decryption query containing a tuple $(\tau \neq \tau^*, [\mathbf{c}]_1, \pi = [\mathbf{z}]_1)$ such that $[\mathbf{c}^\top \mathbf{a}^\perp]_1 \neq [0]_1$ and $e([\mathbf{z}]_1, [\boldsymbol{\alpha}]_2) = e([\mathbf{c}]_1, [\boldsymbol{\beta}_1 + \tau\boldsymbol{\beta}_2]_2)$ then $\mathbf{c} \notin \mathrm{Span}(\mathbf{a})$ and $\mathbf{z}^\top\boldsymbol{\alpha} = \mathbf{c}^\top(\boldsymbol{\beta}_1 + \tau\boldsymbol{\beta}_2) = \mathbf{c}^\top(\mathbf{K}_1 + \tau\mathbf{K}_2)\boldsymbol{\alpha}$. Let $[\boldsymbol{\alpha}^\perp]_1 = [\mathbf{z} - (\mathbf{K}_1 + \tau\mathbf{K}_2)^\top\mathbf{c}]_1$. From Lemma 1, with $n = 2, t = k = 1$, we have $\Pr[\mathbf{z} - (\mathbf{K}_1 + \tau\mathbf{K}_2)^\top\mathbf{c} = \mathbf{0}] \le \frac{1}{p}$. Therefore, $\mathcal{B}_2$ is able to find a (non-zero) vector $[\boldsymbol{\alpha}^\perp]_1 \in \mathbb{G}_1^2$ such that $\boldsymbol{\alpha}^\top\boldsymbol{\alpha}^\perp = 0$. Thus, $\mathcal{B}_2$ violates the KerMDH assumption in group $\mathbb{G}_2$, if $\mathcal{A}$ is able to find such a decryption query. If Q_{Dec} is the the total number of decryption queries of $\mathcal{A}$, then we have

$$|\Pr[\mathsf{G}_2] - \Pr[\mathsf{G}_3]| \le Q_{\mathsf{Dec}} \cdot \mathsf{Adv}^{\mathsf{KerMDH}}_{\mathcal{B}_2,\mathsf{GGen}_2}(\lambda) + \mathsf{negl}(\lambda).$$

Game 4: In this game, we replace the pair $(\boldsymbol{x}_0, M_0)$ in the challenge ciphertext with the pair $(\boldsymbol{x}_1, M_1)$. In particular, last two components of $\mathbf{ct}_1^*$ become $\boldsymbol{x}_1 + \mathbf{U}_1\mathbf{c}^*$ and $M_1 \cdot \boldsymbol{x}_1 + \mathbf{U}_2\mathbf{c}^*$. We claim that the two games 3 and 4 are identical in $\mathcal{A}$'s view. In other words, we show that $\Pr[\mathsf{G}_3] = \Pr[\mathsf{G}_4]$.

First, we assume that $\mathcal{A}$ chooses the challenge pair $((\boldsymbol{x}_0, M_0), (\boldsymbol{x}_1, M_1))$ independent of MPK and the corresponding advantages of $\mathcal{A}$ in game 3 and 4 are $\Pr[\mathsf{G}_3^s]$ and $\Pr[\mathsf{G}_4^s]$ respectively. Then guessing the challenge pair in the adaptive game will incur an exponential security loss, i.e. $\Pr[\mathsf{G}_j] = p^{2\ell}|\mathcal{M}| \Pr[\mathsf{G}_j^s]$ for $j = 3, 4$. If we can show that $\Pr[\mathsf{G}_3^s] = \Pr[\mathsf{G}_4^s]$ (in selective experiment) then this automatically leads to $\Pr[\mathsf{G}_3] = \Pr[\mathsf{G}_4]$.

Finally, we assume that the challenge pair $((\boldsymbol{x}_0, M_0), (\boldsymbol{x}_1, M_1))$ independent of MPK. Since $(\mathbf{U}_1, \mathbf{U}_2)$ are chosen uniformly at random from $(\mathbb{Z}_p^{\ell\times 2})^2$, the following distributions are statistically close over $(\mathbb{Z}_p^{\ell\times 2})^2$:

$$(\mathbf{U}_1, \mathbf{U}_2) \text{ and } (\mathbf{U}_1 + (\boldsymbol{x}_1 - \boldsymbol{x}_0)(\mathbf{a}^\perp)^\top, \mathbf{U}_2 + (M_1 \cdot \boldsymbol{x}_1 - M_0 \cdot \boldsymbol{x}_0)(\mathbf{a}^\perp)^\top)$$

The corresponding changes in MPK, $\mathcal{O}_{\mathsf{KG}}(\cdot)$, $\mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)$ and $\mathcal{O}_{\mathsf{Enc}}(\cdot,\cdot)$ are as follows:
<u>MPK</u>: $(\mathbf{U}_1 + (\boldsymbol{x}_1 - \boldsymbol{x}_0)(\mathbf{a}^\perp)^\top)\mathbf{a} = \mathbf{U}_1\mathbf{a}$, $(\mathbf{U}_2 + (M_1 \cdot \boldsymbol{x}_1 - M_0 \cdot \boldsymbol{x}_0)(\mathbf{a}^\perp)^\top)\mathbf{a} = \mathbf{U}_2\mathbf{a}$

<u>$\mathcal{O}_{\mathsf{KG}}(\boldsymbol{y})$</u>: $\mathsf{sk}_{\boldsymbol{y}} := \begin{pmatrix} -\mathbf{U}_1^\top\boldsymbol{y} + \mathbf{a}^\perp(\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top\boldsymbol{y} \\ -\mathbf{U}_2^\top\boldsymbol{y} + \mathbf{a}^\perp(M_1 \cdot \boldsymbol{x}_1 - M_0 \cdot \boldsymbol{x}_0)^\top\boldsymbol{y} \\ \boldsymbol{y} \end{pmatrix} = \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \boldsymbol{y} \end{pmatrix}$

<u>$\mathcal{O}_{\mathsf{Dec}}(\tau, \mathsf{CT} = ([\mathbf{ct}]_1, \pi), \boldsymbol{y})$</u>: Let $[\mathbf{ct}]_1 = \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix}_1$, $[\boldsymbol{v}_1]_1 := \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_1 \end{bmatrix}_1$, $[\boldsymbol{v}_2]_1 := \begin{bmatrix} \mathbf{c} \\ \mathbf{c}_2 \end{bmatrix}_1$. If $[\mathbf{c}^\top\mathbf{a}^\perp]_1 \neq [0]_1$, then the oracle returns $\perp$. The oracle computes a secret-key $\mathsf{sk}_{\boldsymbol{y}}$ as above and set $\varsigma_1 :=$

$\begin{pmatrix}\mathbf{s}_1\\ \boldsymbol{y}\end{pmatrix}, \varsigma_2 := \begin{pmatrix}\mathbf{s}_2\\ \boldsymbol{y}\end{pmatrix}$. We observe that

$$\begin{aligned}
\mu := [\langle \boldsymbol{v}_1, \varsigma_1\rangle]_1 &= \begin{bmatrix}\langle -(\mathbf{U}_1 + (\boldsymbol{x}_1 - \boldsymbol{x}_0)(\mathbf{a}^\perp)^\top)^\top \boldsymbol{y}, \mathbf{c}\rangle \\ \langle \boldsymbol{y}, \mathbf{c}_1\rangle\end{bmatrix}_1 \\
&= \begin{bmatrix}\langle -\mathbf{U}_1^\top \boldsymbol{y}, \mathbf{c}\rangle \\ \langle \boldsymbol{y}, \mathbf{c}_1\rangle\end{bmatrix}_1 && (\text{when } [\mathbf{c}^\top \mathbf{a}^\perp]_1 = [0]_1) \\
&= \left[\begin{pmatrix}-\mathbf{U}_1^\top \boldsymbol{y}\\ \boldsymbol{y}\end{pmatrix}^\top \cdot \begin{pmatrix}\mathbf{c}\\ \mathbf{c}_1\end{pmatrix}\right]_1
\end{aligned}$$

and similarly $\mu' := [\langle \boldsymbol{v}_2, \varsigma_2\rangle]_1 = \left[\begin{pmatrix}-\mathbf{U}_2^\top \boldsymbol{y}\\ \boldsymbol{y}\end{pmatrix}^\top \cdot \begin{pmatrix}\mathbf{c}\\ \mathbf{c}_2\end{pmatrix}\right]_1$. Therefore, decryption performs correctly.

$\underline{\mathcal{O}_{\mathsf{Enc}}(\tau^*, \{\boldsymbol{x}_b, M_b\}_{b\in\{0,1\}})}$: Finally, the challenge ciphertext component is distributed as

$$\begin{aligned}
[\mathbf{ct}^*]_1 := &\begin{bmatrix}\mathbf{c}\\ \boldsymbol{x}_0 + \mathbf{U}_1\mathbf{c}\\ M_0\cdot \boldsymbol{x}_0 + \mathbf{U}_2\mathbf{c}\end{bmatrix}_1 && (\text{in Game 3}) \\
\approx &\begin{bmatrix}\mathbf{c}\\ \boldsymbol{x}_0 + (\mathbf{U}_1 + (\boldsymbol{x}_1 - \boldsymbol{x}_0)(\mathbf{a}^\perp)^\top)\mathbf{c}\\ M_0\cdot \boldsymbol{x}_0 + (\mathbf{U}_2 + (M_1\cdot \boldsymbol{x}_1 - M_0\cdot \boldsymbol{x}_0)(\mathbf{a}^\perp)^\top)\mathbf{c}\end{bmatrix}_1 && (\text{statistically close}) \\
= &\begin{bmatrix}\mathbf{c}\\ \boldsymbol{x}_0 + \mathbf{U}_1\mathbf{c} + (\boldsymbol{x}_1 - \boldsymbol{x}_0)\\ M_1\cdot \boldsymbol{x}_1 + \mathbf{U}_2\mathbf{c} + (M_1\cdot \boldsymbol{x}_1 - M_0\cdot \boldsymbol{x}_0)\end{bmatrix}_1 && (\text{as } \mathbf{c}^\top\mathbf{a}^\perp = 1) \\
= &\begin{bmatrix}\mathbf{c}\\ \boldsymbol{x}_1 + \mathbf{U}_1\mathbf{c}\\ M_1\cdot \boldsymbol{x}_1 + \mathbf{U}_2\mathbf{c}\end{bmatrix}_1 && (\text{in Game 4})
\end{aligned}$$

Hence, we have $\Pr[\mathsf{G}_3^s] = \Pr[\mathsf{G}_4^s]$ which directly implies $\Pr[\mathsf{G}_3] = \Pr[\mathsf{G}_4]$.

Game 5: It is identical to game 4, except we omit the additional check in decryption oracle. For a query $(\tau, \mathsf{CT}, \boldsymbol{y})$, the decryption oracle only verifies $e(\pi, [\boldsymbol{\alpha}]_2) = e([\mathbf{c}]_1, [\boldsymbol{\beta}_1 + \tau\boldsymbol{\beta}_2]_2)$ to proceed further. Following the same argument as in game 3, we get

$$|\Pr[\mathsf{G}_4] - \Pr[\mathsf{G}_5]| \leq Q_{\mathsf{Dec}} \cdot \mathsf{Adv}^{\mathsf{KerMDH}}_{\mathcal{B}_2,\mathsf{GGen}_2}(\lambda).$$

Game 6: In this game instead of picking $\mathbf{c}^*$ uniformly from $\mathbb{Z}_p^2$, we set $\mathbf{c}^* := \mathbf{a}r$ for $r \leftarrow \mathbb{Z}_p$. Relying on DDH assumption in group $\mathbb{G}_1$, as in game 2, we get

$$|\Pr[\mathsf{G}_5] - \Pr[\mathsf{G}_6]| \leq \mathsf{Adv}^{\mathsf{DDH}}_{\mathcal{B}_1,\mathsf{GGen}_1}(\lambda).$$

Game 7: Finally, we use the witness r to set the proof $\pi^* := [(\boldsymbol{\vartheta}_1 + \tau^*\boldsymbol{\vartheta}_2)r]_1$. From equation 3, we have $\Pr[\mathsf{G}_6] = \Pr[\mathsf{G}_7]$. Note that, game 7 is the standard AHNIPE experiment $\mathsf{Expt}^{\mathsf{AHNIPE}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, 1)$. Combining all the probabilities, we conclude the proof. □

5 Applications of AHNIPE

5.1 Anonymous Identity-Based Revocation

In this section, we present one particular application of our AHNIPE in identity-based revocation (IBR) scheme [29]. Attrapadung and Libert showed in [5] that an NIPE can be used to build an IBR with constant size ciphertext. As their NIPE is only payload-hiding, the resulting IBR system fails to provide users anonymity. We strengthen the security of an IBR system using our AHNIPE following the technique of [5]. Recall that, in an IBR system messages are encrypted with respect to a revoked set R and a secret-key $\mathsf{sk}_{\mathsf{id}}$ corresponding to an identity id can recover the message only if $id \notin R$. Given all the secret-keys associated to the identities in R, an adversary remains oblivious about the message. In the IBR of [5] based on NIPE, the ciphertexts trivially contains the list of all revoked users which often becomes unacceptable in many applications where identities include sensitive users credentials [9, 30, 39, 28].

Definition 8. (Identity-based revocation)*[29] A tag-based identity-based revocation (IBR) scheme for an identity space $\mathcal{ID}$, a tag-space $\mathcal{T}$ and a message space $\mathcal{M}$ consists of four PPT algorithms IBR = (Setup, Enc, KeyGen, Dec) and works as follows:*

- *(MSK, MPK) $\leftarrow$ Setup$(1^\lambda, 1^r)$: The setup algorithm takes as input a security parameter λ and a bound on the number of revoked users r, and generates a master public-key MPK and a master secret-key MSK.*
- *CT $\leftarrow$ Enc(MPK, τ, R, M): A data owner encrypts a message $M \in \mathcal{M}$ with a tag $\tau \in \mathcal{T}$ and a revoked list $R \subset \mathcal{ID}$ containing at most r identities using the master public-key MPK, and publishes a ciphertext CT. Note that CT does not include the list R, but may contain the tag τ.*
- *$\mathsf{sk}_{\mathsf{id}} \leftarrow$ KeyGen(MSK, id): A trusted authority generates a secret-key $\mathsf{sk}_{\mathsf{id}}$ for an identity id $\in \mathcal{ID}$ using the master secret-key MSK. The identity may contain user's sensitive information.*
- *$\perp$ or $M \leftarrow$ Dec(MPK, τ, $\mathsf{sk}_{\mathsf{id}}$, CT): An user decrypts a ciphertext CT associated with a tag τ using the master public-key MPK and its own secret-key $\mathsf{sk}_{\mathsf{id}}$ to either recover a message $M \in \mathcal{M}$ or face a failure.*

Correctness: For any $\lambda, r \in \mathbb{N}$, $\mathsf{id} \in \mathcal{ID}$, $\tau \in \mathcal{T}$, $M \in \mathcal{M}$, $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, 1^r)$, $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyGen}(\mathsf{MSK}, \mathsf{id})$, $\mathsf{CT} \leftarrow \mathsf{Enc}(\mathsf{MPK}, \tau, R, M)$ we have

$$\Pr[M = \mathsf{Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\mathsf{id}}, \mathsf{CT})] = 1 - \mathsf{negl}(\lambda)$$

```
$(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, 1^r)$
$(\tau^*, (R_0, M_0), (R_1, M_1)) \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{KG}}(\cdot), \mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)}(1^\lambda)$
$\mathsf{CT}^* \leftarrow \mathsf{Enc}(\mathsf{MPK}, \tau^*, R_b, M_b)$
$b' \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{KG}}(\cdot), \mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)}(\mathsf{CT}^*)$
return $b'$

$\underline{O_{\mathsf{KG}}(\cdot)}$:
    input: $\mathsf{id} \in \mathcal{ID}$
    return $\mathsf{KeyGen}(\mathsf{MSK}, \mathsf{id})$
$\underline{O_{\mathsf{Dec}}(\cdot, \cdot, \cdot)}$:
    input: $\tau \in \mathcal{T}, \mathsf{CT}, \mathsf{id} \in \mathcal{ID}$
    $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, \mathsf{id})$
    return $\mathsf{Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\mathsf{id}}, \mathsf{CT})$
```

Figure 6: $\mathsf{Expt}^{\mathsf{ANON\text{-}IBR}}_{\mathcal{A},\mathsf{CCA}}(1^\lambda, b)$

Definition 9. (Adaptively anonymous CCA security for IBR) *A (tag-based) identity-based revocation scheme IBR = (Setup, Enc, KeyGen, Dec) for an identity space $\mathcal{ID}$, a tag-space $\mathcal{T}$ and a message space $\mathcal{M}$ is said to be adaptively anonymously secure under chosen-ciphertext attacks (ANON-IBR) if, for any PPT adversary $\mathcal{A}$, for any $\lambda \in \mathbb{N}$, the advantage*

$$\mathsf{Adv}^{\mathsf{ANON\text{-}IBR}}_{\mathcal{A},CCA}(\lambda) = \left| Pr[\mathsf{Expt}^{\mathsf{ANON\text{-}IBR}}_{\mathcal{A},CCA}(1^\lambda, 0) = 1] - Pr[\mathsf{Expt}^{\mathsf{ANON\text{-}IBR}}_{\mathcal{A},CCA}(1^\lambda, 1) = 1] \right|$$

is negligible in λ, where $\mathsf{Expt}^{\mathsf{ANON\text{-}IBR}}_{\mathcal{A},CCA}(1^\lambda, b)$ is defined in Fig. 6 with the following restriction on $\mathcal{A}$'s queries:

– *All secret-key queries {id} to the key generation oracle $\mathcal{O}_{KG}(\cdot)$ should satisfy that id $\in R_0 \cap R_1$.*

- *All decryption queries $\{(\tau, CT, id)\}$ to the decryption oracle $\mathcal{O}_{\mathsf{Dec}}(\cdot,\cdot,\cdot)$ should satisfy that $\tau \neq \tau^*$.*

Construction. Let us consider an AHNIPE = (Setup, Enc, KeyGen, Dec) for $\mathcal{P} = \mathsf{Q} = \mathbb{Z}_p^{r+1}$, $\mathcal{T} = \mathcal{I} = \mathbb{Z}_p$ and $\mathcal{M} \subset \mathbb{Z}_p$. We build an ANON-IBR scheme for $\mathcal{ID} = \mathbb{Z}_p$ with the same message and tag spaces:

- $\underline{\mathsf{Setup}(1^\lambda, 1^r)}$: It compute $(\mathsf{MSK}, \mathsf{MPK}) \leftarrow \mathsf{AHNIPE.Setup}(1^\lambda, 1^{r+1})$ and outputs $(\mathsf{MSK}, \mathsf{MPK})$.
- $\underline{\mathsf{Enc}(\mathsf{MPK}, \tau, R, M)}$: Let $R = \{\mathsf{id}_1, \ldots, \mathsf{id}_r\} \subset \mathbb{Z}_p$ be the set of revoked identities (without loss of generality we take $|R| = r$). Then it computes a polynomial $P(X) = (X - \mathsf{id}_1)\cdots(X - \mathsf{id}_r) = x_0 + x_1 X + \cdots + x_r X^r \in \mathbb{Z}_p[X]$ and set $\boldsymbol{x}_R = (x_0, \ldots, x_r) \in \mathbb{Z}_p^{r+1}$. It returns $\mathsf{CT} \leftarrow \mathsf{AHNIPE.Enc}(\mathsf{MPK}, \tau, \boldsymbol{x}_R, M)$.
- $\underline{\mathsf{KeyGen}(\mathsf{MSK}, \mathsf{id})}$: For an identity $\mathsf{id} \in \mathbb{Z}_p$, it sets $\boldsymbol{y}_{\mathsf{id}} = (1, \mathsf{id}, \ldots, \mathsf{id}^r) \in \mathbb{Z}_p^{r+1}$. Then it returns $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{AHNIPE.KeyGen}(\mathsf{MSK}, \boldsymbol{y}_{\mathsf{id}})$.
- $\underline{\mathsf{Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\mathsf{id}}, \mathsf{CT})} = \mathsf{AHNIPE.Dec}(\mathsf{MPK}, \tau, \mathsf{sk}_{\mathsf{id}}, \mathsf{CT})$

We note that $\langle \boldsymbol{x}_R, \boldsymbol{y}_{\mathsf{id}} \rangle = P(\mathsf{id}) = 0$ if and only if $\mathsf{id} \in R$. Therefore, correctness of the above IBR follows directly from the AHNIPE system. For security, we assume that $\mathcal{A}$ adaptively submits a challenge tuple $(\tau^*, (R_0, M_0), (R_1, M_1))$. Then, $\langle \boldsymbol{x}_{R_0}, \boldsymbol{y}_{\mathsf{id}} \rangle = \langle \boldsymbol{x}_{R_0}, \boldsymbol{y}_{\mathsf{id}} \rangle = 0$ for all id queried by $\mathcal{A}$ to the key generation oracle. Moreover, $\mathcal{A}$ can not query a tuple $(\tau^*, \mathsf{CT}, \mathsf{id})$ for decryption. Therefore, adaptively attribute-hiding CCA security of AHNIPE ensures that the challenge ciphertext $\mathsf{CT}^* \leftarrow \mathsf{AHNIPE.Enc}(\mathsf{MPK}, \tau, \boldsymbol{x}_{R_b}, M_b)$ hides b from $\mathcal{A}$'s view. We state the security of the IBR in the following theorem.

Theorem 3. *Assuming the AHNIPE is a tag-based adaptively attribute-hiding CCA secure non-zero inner product encryption, the ANON-IBR described above is a adaptively anonymous CCA secure identity-based revocation scheme.*

Remark 2. *Using the generic AHNIPEs of Sec. 3, we achieve CCA secure ANON-IBR schemes from various assumptions such as DDH, LWE, DCR, DDH-f and HSM along with a QANIZK proof system. We also instantiate the ANON-IBR scheme using our CCA secure AHNIPE from Sec. 4 based on plain DDH and KerMDH assumptions. A secret-key $\mathsf{sk}_{\mathsf{id}}$ consists of only 4 elements of $\mathbb{Z}$ and a ciphertext associated to a revoked list of size r contains $2r + 6$ group elements. We formally state the security in the following theorem.*

Theorem 4. *Assuming the DDH assumption holds in the group $\mathbb{G}_1$ and the KerMDH assumption holds in the group $\mathbb{G}_2$, there exists an ANON-IBR scheme which is adaptively anonymously secure under chosen-ciphertext attacks. More specifically, for any PPT adversary $\mathcal{A}$, there exist PPT adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ such that:*

$$\mathsf{Adv}_{\mathcal{A},\mathsf{CCA}}^{\mathsf{ANON\text{-}IBR}}(\lambda) \leq 2 \cdot \mathsf{Adv}_{\mathcal{B}_1,\mathsf{GGen}_1}^{\mathsf{DDH}}(\lambda) + 2Q_{\mathsf{Dec}} \cdot \mathsf{Adv}_{\mathcal{B}_2,\mathsf{GGen}_2}^{\mathsf{KerMDH}}(\lambda) + \mathsf{negl}(\lambda)$$

where Q_{Dec} denotes the total number of decryption queries made by $\mathcal{A}$.

Going through the state of art, the ANON-IBR improves the security assumption where existing CPA secure IBR schemes either hide only messages based on DDH like assumptions in both groups $\mathbb{G}_1, \mathbb{G}_2$ (i.e. similar to SXDH assumption) [36, 17] or provide anonymity from pairing-based DH assumptions [39]. The only CCA secure ANON-IBR of [22] is proven secure relying on BDDH assumption in the random oracle model whereas we provide anonymity based on plain DDH assumption and CCA security based on a simple computational KerMDH (weaker than the DDH [26]) assumption in the standard model.

```
Trace(pd, $R, S, \mathcal{O}^{\mathcal{D}}$):
Find $M, M' \in \mathcal{M}$ such that the following quantity is non-negligible:
    $\left| \Pr_{\mathsf{CT}_R \leftarrow \mathsf{Enc}(\mathsf{MPK}, R, M)} [\mathcal{O}^{\mathcal{D}}(\mathsf{CT}_R, M) = 1] - \Pr_{\mathsf{CT}'_R \leftarrow \mathsf{Enc}(\mathsf{MPK}, R, M')} [\mathcal{O}^{\mathcal{D}}(\mathsf{CT}'_R, M) = 1] \right|$
$S_1 := \{\mathsf{id}_1, \ldots, \mathsf{id}_k\} = S \setminus R$
compute $\boldsymbol{x}_R \in \mathbb{Z}_p^{r+1} \setminus \{\mathbf{0}\}$ such that $\langle \boldsymbol{x}_R, \boldsymbol{y}_{\mathsf{id}} \rangle = 0 \ \forall \mathsf{id} \in R$
for $i$ runs from 1 to $k$
  if $i = 1$,
    $\boldsymbol{x}_{S_i} := \mathbf{0}$
  if $S_i = \emptyset$
    $\boldsymbol{x}_{S_i} := (M' - M)\boldsymbol{x}_R$
  else compute $\boldsymbol{x}_{S_i}$ such that $(\langle \boldsymbol{y}_{\mathsf{id}}, \boldsymbol{x}_{S_i} \rangle = 0 \ \forall \mathsf{id} \in S_i \cup R) \wedge$
                           $(\langle \boldsymbol{y}_{\mathsf{id}}, \boldsymbol{x}_{S_i} \rangle = (M' - M)\langle \boldsymbol{y}_{\mathsf{id}}, \boldsymbol{x}_R \rangle \ \forall \mathsf{id} \in S_1 \setminus S_i)$
  Repeat the following steps sufficiently many times (as dictated by Hoeffding's inequality) to compute an approximation of the probability $p_i$ that the output of $\mathcal{O}^{\mathcal{D}}$ is $b_i = 1$:
    $\boldsymbol{x} = \boldsymbol{x}_{S_i} + M\boldsymbol{x}_R$
    $\mathsf{CT}_{S_i} \leftarrow \mathsf{AHNIPE.Enc}'(\mathsf{MPK}, \boldsymbol{x}_R, \boldsymbol{x})$ where we define
    $\mathsf{AHNIPE.Enc}'(\mathsf{MPK}, \boldsymbol{x}_R, \boldsymbol{x}) := (\mathsf{IPFE.Enc}(\mathsf{mpk}, \boldsymbol{x}_R), \mathsf{IPFE.Enc}(\mathsf{mpk}, \boldsymbol{x}))$
    $b_i \leftarrow \mathcal{O}^{\mathcal{D}}(\mathsf{CT}_{S_i})$
if $(i > 1) \wedge (|p_i - p_{i-1}|$ is non-negligible)
  return $\mathsf{id}_{i-1}$ and abort
$S_i = \emptyset$
  return $\perp$ and abort
else $S_{i+1} := S_i \setminus \{\mathsf{id}_i\}$
```

Figure 7: Tracing in Anonymous IBTR scheme

5.2 Anonymous Identity-Based Trace and Revoke

Agrawal et al. [2] gave a generic transformation of an identity-based trace and revoke (IBTR) scheme from any IPFE. An IBTR scheme works in the same way as an IBR system except that it has an additional trace algorithm. The purpose of tracing is to identify malicious users who build pirate decoders. We extend our $\mathsf{ANON\text{-}IBR}$ to achieve anonymous IBTRs ($\mathsf{ANON\text{-}IBTR}$) by modifying the tracing algorithm of Agrawal et al.'s scheme [2]. Note that, the ciphertexts of the IBTR of [2] do not hide the revoked list whereas our $\mathsf{ANON\text{-}IBTR}$ achieves anonymity of users identities. Therefore, the generic AHNIPE of Sec. 3 without the QANIZK (see Remark **??**) leads us to CPA secure $\mathsf{ANON\text{-}IBTR}$ schemes based on DDH, LWE and DCR assumptions.

To identify malicious users (traitors) whose keys are compromised in building a pirate decoder, tracing is necessary to revoke those users from the system. We take the tag-free version of our AHNIPE omitting the QANIZK proof system. We describe the black box tracing given by Agrawal et al. [2] where it is assumed that the tracing algorithm has the access to an oracle $\mathcal{O}^{\mathcal{D}}$ that on input a ciphertext-message pair (CT_R, M) outputs 1 if $\mathcal{D}(\mathsf{CT}_R) = M$, otherwise outputs 0. Let CT_R be the output of $\mathsf{Enc}(\mathsf{MPK}, R, M)$ where R is a revoked list and M is a message belonging to the message space $\mathcal{M}$. If a decoder cannot decrypt the ciphertext CT_R to the correct message, then it is of no use. Therefore, we must define the behaviour of a good decoder as follows:

$$\Pr_{\substack{M \leftarrow \mathcal{M} \\ \mathsf{CT}_R \leftarrow \mathsf{Enc}(\mathsf{MPK}, R, M)}} \left[\mathcal{O}^{\mathcal{D}}(\mathsf{CT}_R, M) = 1 \right] \geq \frac{1}{|\mathcal{M}|} + \frac{1}{\lambda^c} \qquad (4)$$

for some constant $c > 0$. This probability can be estimated by repeated queries to $\mathcal{O}^{\mathcal{D}}$ on arbitrary ciphertext-message pairs, using Hoeffding's inequality.

In public traceability, the adversary $\mathcal{A}$ adaptively asks at most t traitor keys $\mathsf{sk}_{\mathsf{id}}$ and the challenger keeps a track on all these traitors in the list T. $\mathcal{A}$ submits a revoked list R containing at most r users and receives all the secret-keys of the users in R. Finally, $\mathcal{A}$ outputs a decoder $\mathcal{D}$ and a suspected set S of cardinality less or equal to t that contains T. The challenger runs a trace algorithm $\mathsf{Trace}(\mathsf{pd}, R, S, \mathcal{O}^{\mathcal{D}})$ where pd denotes the public directory of users' identities. The adversary wins if $\mathcal{D}$ satisfies equation 4 and Trace either outputs $\perp$ or an identity $\mathsf{id} \notin T$ with high probability.

We describe the tracing algorithm in Fig. 7. The main observation is that if $S_i \neq \emptyset$ and $\mathsf{id} \in S_i \cup R$ then $\langle \boldsymbol{y}_{\mathsf{id}}, \boldsymbol{x} \rangle = M \langle \boldsymbol{y}_{\mathsf{id}}, \boldsymbol{x}_R \rangle$ which means $\mathsf{AHNIPE.Enc}'(\mathsf{MPK}, \boldsymbol{x}_R, \boldsymbol{x})$ is distributed as $\mathsf{AHNIPE.Enc}(\mathsf{MPK}, \boldsymbol{x}_R, M)$. On the other hand, if $S_i = \emptyset$ or $\mathsf{id} \in S_1 \setminus S_i$ then $\langle \boldsymbol{y}_{\mathsf{id}}, \boldsymbol{x} \rangle = M' \langle \boldsymbol{y}_{\mathsf{id}}, \boldsymbol{x}_R \rangle$ which means $\mathsf{AHNIPE.Enc}'(\mathsf{MPK}, \boldsymbol{x}_R, \boldsymbol{x})$ is distributed as $\mathsf{AHNIPE.Enc}(\mathsf{MPK}, \boldsymbol{x}_R, M')$. If $\mathcal{O}^{\mathcal{D}}$ satisfies equation 4 then it must behave differently for the dual nature of $\mathsf{AHNIPE.Enc}'$. Therefore, there exists i such that $|p_i - p_{i-1}|$ is non-negligible and Trace successfully identifies a traitor from T. Finally, we note that step 1 of the Trace algorithm can be implemented efficiently as $\mathcal{O}^{\mathcal{D}}$ satisfies equation 4 (Lemma 14 of [2]).

6 Conclusion

We investigate the way of achieving CCA security for NIPE schemes with the capability of hiding attributes based on standard assumptions. Firstly, we have described a generic transformation for establishing CCA secure AHNIPE from any existing CPA secure IPFE schemes and a QANIZK proof system. In our concrete construction of AHNIPE relying on plain DDH assumption, we employ the shared randomness technique in Naor-Yung paradigm to reduce public-key and ciphertext size. Furthermore, it makes the decryption much faster than our generic construction. We show that AHNIPE directly implies $\mathsf{ANON\text{-}IBR}$ scheme which has significant applications in the area of broadcast encryption. Our concrete AHNIPE leads us to the first CCA secure $\mathsf{ANON\text{-}IBR}$ scheme based on the plain DDH assumption in the standard model. Moreover, we extend our $\mathsf{ANON\text{-}IBR}$ to a set of CPA secure $\mathsf{ANON\text{-}IBTR}$ scheme by adding a tracing algorithm utilizing the work of [2]. Future work includes finding an efficient CCA secure AHNIPE based on LWE assumption. Also, exploring CCA secure constructions for full-hiding NIPE [34] with an efficient decryption procedure.

References

[1] M. Abdalla, F. Benhamouda, and D. Pointcheval. Disjunctions for hash proof systems: New constructions and applications. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 69–100. Springer, 2015.

[2] S. Agrawal, S. Bhattacherjee, D. H. Phan, D. Stehlé, and S. Yamada. Efficient public trace and revoke from standard assumptions. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 2277–2293, 2017.

[3] S. Agrawal, B. Libert, and D. Stehlé. Fully secure functional encryption for inner products, from standard assumptions. In *Annual International Cryptology Conference*, pages 333–362. Springer, 2016.

[4] M. Ambrona, G. Barthe, and B. Schmidt. Generic transformations of predicate encodings: Constructions and applications. In *Annual International Cryptology Conference*, pages 36–66. Springer, 2017.

[5] N. Attrapadung and B. Libert. Functional encryption for inner product: Achieving constant-size ciphertexts with adaptive security or support for negation. In *International Workshop on Public Key Cryptography*, pages 384–402. Springer, 2010.

[6] N. Attrapadung, B. Libert, and E. De Panafieu. Expressive key-policy attribute-based encryption with constant-size ciphertexts. In *International Workshop on Public Key Cryptography*, pages 90–108. Springer, 2011.

[7] F. Benhamouda, F. Bourse, and H. Lipmaa. Cca-secure inner-product functional encryption from projective hash functions. In *IACR International Workshop on Public Key Cryptography*, pages 36–66. Springer, 2017.

[8] S. Biagioni, D. Masny, and D. Venturi. Naor–yung paradigm with shared randomness and applications. *Theoretical Computer Science*, 692:90–113, 2017.

[9] D. Boneh and M. Hamburg. Generalized identity based and broadcast encryption schemes. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 455–470. Springer, 2008.

[10] D. Boneh, A. Sahai, and B. Waters. Functional encryption: Definitions and challenges. In *Theory of Cryptography Conference*, pages 253–273. Springer, 2011.

[11] D. Boneh and B. Waters. Conjunctive, subset, and range queries on encrypted data. In *Theory of Cryptography Conference*, pages 535–554. Springer, 2007.

[12] G. Castagnos, F. Laguillaumie, and I. Tucker. Practical fully secure unrestricted inner product functional encryption modulo p. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 733–764. Springer, 2018.

[13] J. Chen, B. Libert, and S. C. Ramanna. Non-zero inner product encryption with short ciphertexts and private keys. In *International Conference on Security and Cryptography for Networks*, pages 23–41. Springer, 2016.

[14] J. Chen and H. Wee. Doubly spatial encryption from dbdh. *Theoretical Computer Science*, 543:79–89, 2014.

[15] A. Fiat and M. Naor. Broadcast encryption. In *Annual International Cryptology Conference*, pages 480–491. Springer, 1993.

[16] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. *SIAM Journal on Computing*, 45(3):882–929, 2016.

[17] A. Ge and P. Wei. Identity-based broadcast encryption with efficient revocation. In *IACR International Workshop on Public Key Cryptography*, pages 405–435. Springer, 2019.

[18] C. Gentry, A. Sahai, and B. Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In *Annual Cryptology Conference*, pages 75–92. Springer, 2013.

[19] S. Gorbunov, V. Vaikuntanathan, and H. Wee. Attribute-based encryption for circuits. *Journal of the ACM (JACM)*, 62(6):45, 2015.

[20] S. Gorbunov, V. Vaikuntanathan, and H. Wee. Predicate encryption for circuits from lwe. In *Annual Cryptology Conference*, pages 503–523. Springer, 2015.

[21] J. Groth, R. Ostrovsky, and A. Sahai. New techniques for noninteractive zero-knowledge. *Journal of the ACM (JACM)*, 59(3):1–35, 2012.

[22] K. He, J. Weng, J.-N. Liu, J. K. Liu, W. Liu, and R. H. Deng. Anonymous identity-based broadcast encryption with chosen-ciphertext security. In *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, pages 247–255, 2016.

[23] S. Katsumata and S. Yamada. Non-zero inner product encryption schemes from various assumptions: Lwe, ddh and dcr. PKC, 2019.

[24] J. Katz, A. Sahai, and B. Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In *annual international conference on the theory and applications of cryptographic techniques*, pages 146–162. Springer, 2008.

[25] Y. Kawai and K. Takashima. Fully-anonymous functional proxy-re-encryption. Cryptology ePrint Archive, Report 2013/318, 2013.

[26] E. Kiltz and H. Wee. Quasi-adaptive nizk for linear subspaces revisited. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 101–128. Springer, 2015.

[27] V. Koppula and B. Waters. Realizing chosen ciphertext security generically in attribute-based encryption and predicate encryption. In *Annual International Cryptology Conference*, pages 671–700. Springer, 2019.

[28] J. Lai, Y. Mu, F. Guo, W. Susilo, and R. Chen. Anonymous identity-based broadcast encryption with revocation for file sharing. In *Australasian Conference on Information Security and Privacy*, pages 223–239. Springer, 2016.

[29] A. Lewko, A. Sahai, and B. Waters. Revocation systems with very small private keys. In *2010 IEEE Symposium on Security and Privacy*, pages 273–285. IEEE, 2010.

[30] B. Libert, K. G. Paterson, and E. A. Quaglia. Anonymous broadcast encryption: Adaptive security and efficient constructions in the standard model. In *International Workshop on Public Key Cryptography*, pages 206–224. Springer, 2012.

[31] M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 427–437, 1990.

[32] T. Okamoto and K. Takashima. Fully secure functional encryption with general relations from the decisional linear assumption. In *Annual cryptology conference*, pages 191–208. Springer, 2010.

[33] T. Okamoto and K. Takashima. Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption. *Designs, Codes and Cryptography*, 77(2-3):725–771, 2015.

[34] S. Patranabis, D. Mukhopadhyay, and S. C. Ramanna. Function private predicate encryption for low min-entropy predicates. In *IACR International Workshop on Public Key Cryptography*, pages 189–219. Springer, 2019.

[35] C. Peikert and S. Shiehian. Noninteractive zero knowledge for np from (plain) learning with errors. In *Annual International Cryptology Conference*, pages 89–114. Springer, 2019.

[36] S. C. Ramanna and P. Sarkar. Efficient adaptively secure ibbe from the sxdh assumption. *IEEE Transactions on Information Theory*, 62(10):5709–5726, 2016.

[37] A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pages 543–553. IEEE, 1999.

[38] S. Yamada, N. Attrapadung, G. Hanaoka, and N. Kunihiro. A framework and compact constructions for non-monotonic attribute-based encryption. In *International Workshop on Public Key Cryptography*, pages 275–292. Springer, 2014.

[39] L. Zhang, Q. Wu, and Y. Mu. Anonymous identity-based broadcast encryption with adaptive security. In *Cyberspace Safety and Security*, pages 258–271. Springer, 2013.