Fully Private Revocable Predicate Encryption*

Juan Manuel González Nieto[1] and Mark Manulis[2] and Dongdong Sun[1]

[1]Queensland University of Technology, Brisbane QLD 4001, Australia
j.gonzaleznieto@qut.edu.au, dd.sun@student.qut.edu.au
[2]University of Surrey, Guildford, United Kingdom
mark@manulis.eu

Abstract. We introduce the concept of *Revocable Predicate Encryption (RPE)*, which extends the previous PE setting with revocation support: private keys can be used to decrypt an RPE ciphertext only if they match the decryption policy (defined via attributes encoded into the ciphertext and predicates associated with private keys) and were not revoked by the time the ciphertext was created.

The first challenge in RPE schemes is to preserve privacy for RPE ciphertexts, namely to ensure the *attribute-hiding* property, which is inherent to traditional PE constructions, and which implies the more basic property of payload hiding, used in the context of Attribute-Based Encryption (ABE). We formalize the notion of attribute hiding in the presence of revocation and propose our first RPE construction, called AH-RPE, which is attribute-hiding under the Decision Linear assumption in the standard model. In the AH-RPE scheme we deploy the revocation system of Lewko, Sahai, and Waters (IEEE S&P 2010), introduced for a simpler setting of broadcast encryption, which we modify for integration with the payload-hiding ABE scheme of Okamoto and Takashima (CRYPTO 2010), after making the latter attribute-hiding by borrowing additional techniques from Lewko, Okamoto, Sahai, Takashima, and Waters (Eurocrypt 2010).

As a second major step we show that RPE schemes may admit more stringent privacy requirements in comparison to PE schemes, especially when it comes to the revocation of private keys. In addition to attribute-hiding, RPE ciphertexts should ideally not leak any information about the revoked keys and by this about the revoked users. We formalize this stronger privacy notion, termed *full hiding*, and propose another RPE scheme, called FH-RPE, which achieves this notion in the setting of "sender-local revocation" of Attrapadung and Imai (Cryptography and Coding 2009), under the same assumptions as our AH-RPE construction. Our FH-RPE scheme is also based on the attribute-hiding variant of Okamoto and Takashima's ABE scheme, yet with a different revocation method, in which we integrate the Subset-Cover Framework of Naor, Naor, and Lotspiech (CRYPTO 2001) for better efficiency.

1 Introduction

Functional Encryption In recent years, asymmetric encryption has experienced a paradigm shift from encryption of secret messages for particular recipients (Public Key Encryption or Identity-Based Encryption) towards more flexible encryption mechanisms, which offer manifold forms of access control to encrypted data. These mechanisms rely on arbitrary *functional* relationships between policies and attributes encoded in ciphertexts and recipients' decryption keys. Functional Encryption has emerged from Identity-Based Encryption techniques [7,24], and encompasses novel concepts such as Attribute-Based Encryption [4,10,12,23], Hidden-Vector Encryption [9], and Predicate-Based Encryption [13, 14, 19, 20, 25–27]. At a high level these schemes implement the idea of creating ciphertexts without prior knowledge of potential recipients. The success of message recovery depends usually on some relation, which is implicitly evaluated through the decryption procedure, on input the information encoded in the ciphertext and information contributed by the recipient's private key.

* This is full version of the paper that appeared in Proceedings of the 17th Australasian Conference on Information Security and Privacy (ACISP 2012), LNCS 7372, pp. 350–363, Springer.

Predicate Encryption In this work we focus on the notion of Predicate Encryption (PE), formalized by Katz, Sahai, and Waters [13], and further studied in [14, 19, 20, 25, 27]. In PE schemes the private keys of users are associated with *predicates* f and ciphertexts are bound to *attributes* a. The decryption procedure is successful if and only if $f(a) = 1$. If this relation is not satisfied then no information about the plaintext is leaked. In contrast to Attribute-Based Encryption, which also states this requirement on the security of the decryption procedure, PE schemes offer privacy of attributes that legitimate recipients of PE ciphertexts must possess, that is PE ciphertexts ensure *attribute hiding* in that they do not leak any information about a for which the condition $f(a) = 1$ would be satisfied. Concrete constructions of PE schemes typically focus on the realization of certain types of predicates f. In their seminal work, Katz, Sahai, and Waters [13] introduced PE schemes supporting Inner-Product Encryption (IPE), i.e. vector $\overrightarrow{y}$ represents attributes and vector $\overrightarrow{x}$ determines the predicate $f_{\overrightarrow{x}}$ such that $f_{\overrightarrow{x}}(\overrightarrow{y}) = 1$ iff $\overrightarrow{x} \cdot \overrightarrow{y} = 0$ ($\overrightarrow{x} \cdot \overrightarrow{y}$ denotes the inner product of vectors $\overrightarrow{x}$ and $\overrightarrow{y}$ over a field or ring). It has been shown that IPE can be leveraged to evaluate a wide class of predicates such as conjunctions or disjunctions of equality tests, conjunctions of comparison or subset tests, and more generally, arbitrary CNF or DNF formulae. The original scheme of Katz, Sahai, and Waters [13] has been proven selectively secure under less standard assumptions (in the generic group model), while more recent schemes by Okamoto and Takashima [20] achieve the stronger notion of adaptive security using standard assumptions.

Revocation in Functional Encryption The revocation challenge in FE schemes turns out to be more subtle than in previous encryption paradigms, e.g. in comparison to CRL-based revocation mechanisms for traditional PKE schemes (within public key infrastructures) [1,18], to time-based revocation mechanisms [7] for IBE schemes, where the identities of receivers are linked to time periods and unrevoked users must be in possession of up-to-date private keys, obtained from the Private Key Generator (PKG), or to revocation mechanisms for certificateless encryption [11] that try to achieve the best of PKE and IBE worlds. The revocation problem in FE is apparent in that FE ciphertexts are encrypted for predicates f that can possibly be satisfied by multiple recipients, all in possession of suitable attributes a. Using time-based revocation for users' attributes is inappropriate here for several reasons: First, a user may be in possession of several attributes and if time periods for all attributes in the system are not synchronized then unrevoked users would have to update their private keys whenever any of their attributes expires. Note that due to the necessary collusion-resistance property of FE schemes a user's private key must depend on all attributes of that user. Second, even if time periods are synchronized then the problem with scalability still remains. Indeed, the PKG would have to be regularly contacted by all unrevoked users in the system to obtain updates for their private keys. This would require online presence of the PKG, establishment of secure channels between the PKG and each user for the transmission of updated private keys, and authentication of users towards the PKG to prove eligibility with regard to the update procedure. The amount of work performed by the PKG is then linear in the number of (unrevoked) users and attributes available in the system. A more efficient approach for handling revocation in IBE systems was suggested by Boldyreva, Goyal, and Kumar (BGK) [5], where the PKG on each time period publishes some update information that is then used by unrevoked users to update their private keys locally. The amount of work performed by PKG is logarithmic and, more importantly, no online communication between the PKG and unrevoked users is required. The approach from [5] could also be applied to ABE systems, in which case, however, it would result in a significant limitation — while in IBE systems revoking user identities is sufficient, revoking attributes in ABE systems would implicitly revoke private keys of all users with those attributes. That is revocation of users (which is possible with the time-based approach of Boneh and Franklin [7] when applied to ABE systems) would no longer possible with the BGK approach. Another limitation of the BGK approach is that unrevoked users still have to update their private keys for each time period.

To alleviate this limitation, Attrapadung and Imai [3] suggested another way for revocation in ABE schemes: Instead of enforcing revocation via an authority, the revocation is carried out by the senders directly, i.e., the senders encrypt a message under a normal attribute set, as well as a revocation list. Each user's private key has an associated policy and some unique identifier. A private key can be used to decrypt the ciphertext if the attributes in the ciphertext satisfy the policy associated with the key *and* the identifier of

the key is not contained in the revocation list encoded into the ciphertext. This method solves the mentioned problem behind the BGK approach, namely each user's private key can now be issued by PKG once and need not be updated thereafter. Later on, Attrapadung and Imai [2] proposed another system by combining techniques from [5] with their previous work from [3], which inherits the advantages of both approaches.

Revocation in PE Schemes and Privacy The different ABE revocation techniques mentioned above, aside from their scalability issues, are only partially applicable to PE schemes due to the distinguished attribute-hiding property of the latter. In particular, care should be taken to ensure that by introducing revocation to a PE system this privacy property is preserved. To the best of our knowledge, revocation in PE schemes has not been investigated so far and it is not clear whether revocation introduces further privacy challenges, in addition to the challenge of preserving their basic attribute-hiding property. We observe that additional privacy problems may arise in scenarios, where revocation is performed for individual private keys. For example, in the revocable ABE scheme of Attrapadung and Imai [3], each sender builds a revocation list on-the-fly, using unique identifiers of users' private keys, and encodes this list into the ciphertext. However, a close inspection of the scheme shows that ciphertexts reveal information about the encoded key identifiers and by this leak information about the revoked users. In this work we explore the concept of privacy-preserving revocation in PE schemes. Our contributions are detailed in the following.

1.1 Our Contributions

We formalize the concept of **Revocable Predicate Encryption (RPE)** and propose two RPE schemes allowing for efficient revocation of individual private keys. The underlying revocation mechanisms do not require any private key update procedures on the recipient's side and more importantly preserve and even strengthen the expected privacy properties of PE schemes. Both mechanisms use revocation information on the sender's side only to perform the encryption operation. Only holders of unrevoked private keys are able to decrypt the ciphertext, still provided that their keys also match the decryption policy. In the following we discuss the two notions of privacy behind our schemes, their usage, and underlying techniques in more detail.

Attribute-Hiding RPE Scheme Our first scheme, termed AH-RPE, offers attribute-hiding, which is the standard PE property (and further implies payload-hiding used in the context of ABE). The revocation concept behind AH-RPE uses revocation lists (RL) and is mostly suitable for applications where revocation management is handled centrally by the PKG. It is assumed that senders obtain up-to-date RL published by the PKG prior to encryption. The attribute-hiding property of our AH-RPE scheme is proven against adaptive adversaries in the standard model under the established DLIN assumption. The AH-RPE scheme has *constant-size* private and public keys while the length of its ciphertexts remains linear in the number of *revoked* keys.

Full-Hiding RPE Scheme Our second scheme, termed FH-RPE, offers even stronger privacy guarantees. Since the RPE concept assumes that revocation lists are used by senders, the standard requirement of attribute-hiding may not be sufficient for applications, where in addition to privacy-protection for attributes (and plaintexts) one is interested in preserving the privacy of users, whose decryption keys were revoked. Hence, we introduce a stronger notion of full-hiding, which we formalize as part of the RPE security model. This property ensures that no information about revoked users is leaked from a given ciphertext and is a natural extension in the context of PE that cares about privacy. Our FH-RPE scheme can be used in applications where senders may freely decide to exclude certain key holders from running a successful decryption operation, even if private keys of those holder match the ciphertext policy. Such **sender-local revocation (SLR)**, as considered in [2] for ABE schemes, allows for more flexible forms of access control to PE plaintexts and the requirement of full-hiding keeps revoked recipients undisclosed.

The full-hiding property of FH-RPE also relies on the DLIN assumption; yet this stronger privacy property comes with additional performance overhead in comparison to our AH-RPE scheme in that the length of keys and ciphertexts becomes *logarithmic* in the number of decryption keys.

Techniques Our RPE schemes are based on the Dual System Encryption of Waters [28] and the Dual Pairing Vector Spaces (DPVS) of Okamoto and Takashima [19]. Our AH-RPE scheme deploys the revocation system of Lewko, Sahai and Waters [15], introduced originally for public-key broadcast encryption, and modified here for an integration with the (payload-hiding) FE scheme of Okamoto and Takashima [20] in a way that achieves attribute-hiding by further using some techniques underlying the PE scheme by Lewko et al. [14]. Our FH-RPE scheme is obtained from Okamoto and Takashima [20] and Lewko et al. [14] in a more direct way: each private key corresponds to an index, which is defined at derivation time. Indices of revoked keys are used to build the revocation list. The revocation mechanism extends individual private keys with additional index-dependent components that provide decryption capabilities as long as the index remains unrevoked — revoked indices are encoded by the sender into the ciphertext in a privacy-preserving way. We note that to be able to create a ciphertext, the sender needs to know not only an attribute but also the indexes of the revoked keys. Using indexes are essential in our scheme. If a sender wishes to revoke some particular key from decrypting a ciphertext then some identifier for that key must be used in the encryption procedure; otherwise revoking certain keys becomes impossible since the keys would not be distinguishable from other keys with the same predicates. We first discuss that applying Okamoto and Takashima [20] and Lewko et al. [14] directly would result in a linear complexity for the lengths of main parameters. The (better) logarithmic complexity of our FH-RPE scheme is due to the use of the **complete-subtree** technique by Naor et al. [18], whose integration preserving the full-hiding property was a challenge. We note that it is possible to obtain a FH-RPE with the combinations of a PE [14] and an anonymous broadcasting scheme [17]. However, in the standard model, the size of the ciphertext in the resulting system is linear in the number of keys. In order to prove security of our RPE schemes we utilize the modular approach from Okamoto and Takashima [20] that breaks the proof down into several higher-level (artificially looking) assumptions and proves them to be secure under the DLIN assumption. The technical challenge in our proofs is to actually adopt the proving techniques from Okamoto and Takashima [20] towards the RPE requirements of attribute-hiding (for AH-RPE) and full-hiding (for FH-RPE), whose definitions have more sophisticated "win conditions".

2 Dual Pairing Vector Spaces and Assumptions

Let $\mathcal{G}_{\mathsf{bpg}}$ be an algorithm that takes as input a security parameter 1^λ and outputs a description of the symmetric bilinear group setting $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ where q is a prime, $\mathbb{G}$ and $\mathbb{G}_T$ are two cyclic groups of order q, G is the generator of $\mathbb{G}$, e is a non-degenerate bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, i.e., $e(sG, tG) = e(G, G)^{st}$ and $e(G, G) \neq 1$.

Vector Spaces. Let $\mathbb{V} = \overbrace{\mathbb{G} \times \cdots \times \mathbb{G}}^{N}$ be a *vector space* and each element in $\mathbb{V}$ be expressed by an N-*dimensional vector* $\boldsymbol{x} = (x_1 G, \ldots, x_N G)$ ($x_i \in \mathbb{F}_q$ for $i = 1, \ldots, N$). The *canonical base* $\mathbb{A}$ of $\mathbb{V}$ is $\mathbb{A} = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N)$, where $\boldsymbol{a}_1 = (G, 0, \ldots, 0)$, $\boldsymbol{a}_2 = (0, G, 0, \ldots, 0), \ldots, \boldsymbol{a}_N = (0, \ldots, 0, G)$. Given two vectors $\boldsymbol{x} = (x_1 G, \ldots, x_N G) = x_1 \boldsymbol{a}_1 + \cdots + x_N \boldsymbol{a}_N \in \mathbb{V}$ and $\boldsymbol{y} = (y_1 G, \ldots, y_N G) = y_1 \boldsymbol{a}_1 + \cdots + y_N \boldsymbol{a}_N \in \mathbb{V}$, where $\overrightarrow{x} = (x_1, \ldots, x_N)$ and $\overrightarrow{y} = (y_1, \ldots, y_N)$, the pairing operation is defined as $e(\boldsymbol{x}, \boldsymbol{y}) = \prod_{i=1}^{N} e(x_i G, y_i G) = e(G, G)^{\sum_{i=1}^{N} x_i y_i} = g_T^{\overrightarrow{x}\,\overrightarrow{y}} \in \mathbb{G}_T$.

Definition 1 (Dual Pairing Vector Space (DPVS) [19]). *Let $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ be a symmetric pairing group. A* Dual Pairing Vector Space *$(q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e)$ is a tuple of a prime q, N-dimensional vector space $\mathbb{V}$ over $\mathbb{F}_q$, a cyclic group $\mathbb{G}_T$ of order q, canonical base $\mathbb{A} = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N)$ of $\mathbb{V}$, and pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ that satisfy the following conditions:*

1. Non-degenerate bilinear pairing: *There exists a polynomial-time computable non-degenerate bilinear pairing $e(\boldsymbol{x}, \boldsymbol{y}) = \prod_{i=1}^{N} e(G_i, H_i)$ where $\boldsymbol{x} = (G_1, \ldots, G_N) \in \mathbb{V}$ and $\boldsymbol{y} = (H_1, \ldots, H_N) \in \mathbb{V}$. This is non-degenerate bilinear i.e., $e(s\boldsymbol{x}, t\boldsymbol{y}) = e(\boldsymbol{x}, \boldsymbol{y})^{st}$ and if $e(\boldsymbol{x}, \boldsymbol{y}) = 1$ for all $\boldsymbol{y} \in \mathbb{V}$, then $\boldsymbol{x} = \boldsymbol{0}$.*
2. Dual orthonormal bases: *$\mathbb{A}$ and e satisfy that $e(\boldsymbol{a_i}, \boldsymbol{a_j}) = g_T^{\delta_{i,j}}$ for all i and j, where $\delta_{i,j} = 1$ if $i = j$, and 0 otherwise, and $g_T \neq 1 \in \mathbb{G}_T$.*

3. DISTORTION MAPS: *Linear transformations $\phi_{i,j}$ on $\mathbb{V}$ s.t. $\phi_{i,j}(\boldsymbol{a}_j) = \boldsymbol{a}_i$ and $\phi_{i,j}(\boldsymbol{a}_k) = \mathbf{0}$ if $k \neq j$ are polynomial-time computable. We call $\phi_{i,j}$ "distortion maps".*

ORTHONORMAL BASES. Let $\mathbb{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_N)$ be the basis of $\mathbb{V}$ which is obtained from the canonical basis $\mathbb{A}$ using a uniformly chosen linear transformation, $\Lambda = (\lambda_{i,j}) \xleftarrow{U} GL(N, \mathbb{F}_q)$ (note that $GL(N, \mathbb{F}_q)$ creates a matrix of size $N \times N$ in which each element is uniformly selected from $\mathbb{F}_q$), such that $\boldsymbol{b}_i = \sum_{j=1}^{N} \lambda_{i,j} \boldsymbol{a}_j$, for $i = 1, \ldots, N$. Similarly, $\mathbb{B}^* = (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_N^*)$ of $\mathbb{V}$ is also obtained from $\mathbb{A}$, such that $\mu_{i,j} = (\Lambda^T)^{-1}$, $\boldsymbol{b}_i^* = \sum_{j=1}^{N} \mu_{i,j} \boldsymbol{a}_j$, for $i = 1, \ldots, N$. It can be shown that $e(\boldsymbol{b}_i, \boldsymbol{b}_j^*) = g_T^{\delta_{i,j}}$, where $\delta_{i,j} = 1$ if $i = j$, and $\delta_{i,j} = 0$ if $i \neq j$. $\mathbb{B}$ and $\mathbb{B}^*$ are thus dual orthonormal bases of $\mathbb{V}$.

In our schemes, we will use the following probabilistic generator $\mathcal{G}_{\mathsf{ob}}$ for dual orthonormal bases:

$$
\begin{aligned}
\mathcal{G}_{\mathsf{ob}}(1^\lambda, \overrightarrow{n} = (d; n_1, \ldots, n_d)) : \mathsf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e) &\xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda), \\
\psi &\xleftarrow{\mathsf{U}} \mathbb{F}_q^\times, N_0 = 5, N_l = 3n_l + 1 \text{ for } l = 1, \ldots, d; \\
&\text{For } l = 0, \ldots, d: \\
&\quad \mathsf{param}_{\mathbb{V}_t} = (q, \mathbb{V}_l, \mathbb{G}_T, \mathbb{A}_l, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N_l, \mathsf{param}_{\mathbb{G}}), \\
&\quad \Lambda^{(l)} = (\lambda_{i,j}^{(l)}) \xleftarrow{\mathsf{U}} GL(N_l, \mathbb{F}_q), \ (\mu_{i,j}^{(l)}) = \psi \cdot (\Lambda^{(l)^T})^{-1}, \\
&\quad \boldsymbol{b}_i{}^{(l)} = \sum_{j=1}^{N_l} \lambda_{i,j}^{(l)} \boldsymbol{a}_j^{(l)} \text{ for } i = 1, \ldots, N_l, \mathbb{B}^{(l)} = (\boldsymbol{b}_1^{(l)}, \ldots, \boldsymbol{b}_{N_l}^{(l)}), \\
&\quad \boldsymbol{b}_i^{*(l)} = \sum_{j=1}^{N_l} \mu_{i,j}^{(l)} \boldsymbol{a}_j^{(l)} \text{ for } i = 1, \ldots, N_l, \mathbb{B}^{*(l)} = (\boldsymbol{b}_1^{*(l)}, \ldots, \boldsymbol{b}_{N_l}^{*(l)}), \\
g_T = e(G, G)^\psi, \mathsf{param}_{\overrightarrow{n}} &= (\{\mathsf{param}_{\mathbb{V}_l}\}_{l=0,\ldots,d}, g_T), \\
\text{Output } (\mathsf{param}_{\overrightarrow{n}}, &\{\mathbb{B}^{(l)}, \mathbb{B}^{*(l)}\}_{l=0,\ldots,d}).
\end{aligned}
$$

Note that $g_T = e(\boldsymbol{b}_i{}^{(l)}, \boldsymbol{b}_i^{*(l)})$ for $l = 0, \ldots, d$; $i = 1, \ldots, N_l$.

Definition 2 (Decisional Linear Assumption (DLIN) [6]). *The DLIN problem is to find bit $\beta \in \{0, 1\}$, given the output $(\mathsf{param}_{\mathbb{G}}, G, aG, bG, acG, bdG, Y_\beta)$ of the probabilistic algorithm*

$$
\begin{aligned}
\mathcal{G}_\beta^{\mathrm{DLIN}}(1^\lambda) : \mathsf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e) &\xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda), a, b, c, d \xleftarrow{\mathsf{U}} \mathbb{F}_q, \\
Y_0 = (c + d)G, \ Y_1 &\xleftarrow{\mathsf{U}} \mathbb{G}, \ \beta \xleftarrow{\mathsf{U}} \{0, 1\}; \\
\text{Output } (\mathsf{param}_{\mathbb{G}}, &G, aG, bG, acG, bdG, Y_\beta).
\end{aligned}
$$

The advantage of a probabilistic polynomial-time DLIN solver $\mathcal{D}$ is defined as

$$
\mathsf{Adv}_{\mathcal{D}}^{\mathrm{DLIN}}(\lambda) = \Big| \ \mathsf{Pr}\Big[\mathcal{D}(1^\lambda, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathrm{DLIN}}(1^\lambda)\Big] \ - \ \mathsf{Pr}\Big[\mathcal{D}(1^\lambda, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathrm{DLIN}}(1^\lambda)\Big] \ \Big|.
$$

The DLIN assumption states that for any such $\mathcal{D}$ this advantage is negligible in λ.

3 Revocable Predicate Encryption: Model and Privacy Definitions

In predicate encryption for inner-product relations, an attribute is expressed as a vector $\overrightarrow{y} \in \mathbb{F}_q^n \setminus \{\overrightarrow{0}\}$ and a predicate $f_{\overrightarrow{x}}$ is associated with a vector $\overrightarrow{x} \in \mathbb{F}_q^n \setminus \{\overrightarrow{0}\}$, where $f_{\overrightarrow{x}}(\overrightarrow{y}) = 1$, iff $\overrightarrow{y} \cdot \overrightarrow{x} = 0$. Let $\mathbb{A} = \mathbb{F}_q^n \setminus \{\overrightarrow{0}\}$ be the attribute space, and $\mathbb{P} = \{f_{\overrightarrow{x}} | \overrightarrow{x} \in \mathbb{F}_q^n \setminus \{\overrightarrow{0}\}\}$ be the predicate space. We assume that indexes are

in the set $\Gamma = \{1, \ldots, N\}$, where N is the number of keys in the system. In our definitions and schemes, we assume that attribute vector, $\overrightarrow{y} = (y_1, \ldots, y_{n_1})$, is normalized such that $y_1 = 1$ (If $\overrightarrow{y}$ is not normalized, change it to a normalized one by $(1/y_1) \cdot \overrightarrow{y}$, assuming that y_1 is non-zero). $\overrightarrow{e}_i^{(k)}$ is the canonical basis vector $(\overbrace{0, \ldots, 0}^{i-1}, 1, \overbrace{0, \ldots, 0}^{n_k - i}) \in \mathbb{F}_q^{n_k}$ for $k = 1, 2$ and $i = 1, \ldots, n_k$.

3.1 Syntax

Definition 3 (Revocable Predicate Encryption). *A revocable predicate encryption scheme* (RPE) *is a tuple of four algorithms* (Setup, GenKey, Encrypt, Decrypt) *and has an associated attribute space* $\mathbb{A}$*, a predicate space* $\mathbb{P}$ *and an index space* Γ*.*

- $\mathsf{Setup}(1^\lambda, \Delta)$ *The* Setup *algorithm takes as input a security parameter* 1^λ *and format* Δ *of attribute and index. It outputs a public key* PK*, a master secret key* MSK*, and a state information* S*.*
- $\mathsf{GenKey}(MSK, S, \overrightarrow{x})$ *The* GenKey *algorithm takes as input a master secret key* MSK*, a state information* S*, and a predicate vector* $\overrightarrow{x}$*. It outputs an updated state* S *and a secret key* $\boldsymbol{k}^*_{\overrightarrow{x},I}$*, where* $I \in \Gamma$ *denotes the associated index of the key and is included in the key.*
- $\mathsf{Encrypt}(PK, L, \overrightarrow{y}, M)$ *The* Encrypt *algorithm takes as input a public key* PK*, a revocation list* $L \subseteq \Gamma$*, an attribute vector* $\overrightarrow{y}$*, and a message* M *in some associated message space. It outputs a ciphertext* C*.*
- $\mathsf{Decrypt}(C, \boldsymbol{k}^*_{\overrightarrow{x},I})$ *The* Decrypt *algorithm takes as input a ciphertext* C *and a secret key* $\boldsymbol{k}^*_{\overrightarrow{x},I}$*. It outputs either a message* M *or the distinguished symbol* $\perp$*.*

Correctness. *The correctness property of the schemes says that for all* PK *and* MSK *output by* Setup *algorithm, all predicate* $f_{\overrightarrow{x}} \in \mathbb{P}$*, all message* M*, all attribute* $\overrightarrow{y} \in \mathbb{A}$*, and all possible valid state information* S *output by* Setup *or* GenKey *algorithm, if the key* $\boldsymbol{k}^*_{\overrightarrow{x},I}$ *was not revoked, i.e.,* $I \notin L$*, then for correctly generated* $\boldsymbol{k}^*_{\overrightarrow{x},I} \xleftarrow{R} \mathsf{GenKey}(MSK, S, \overrightarrow{x})$ *and* $C \xleftarrow{R} \mathsf{Encrypt}(PK, L, \overrightarrow{y}, M)$*:*

- *If* $f_{\overrightarrow{x}}(\overrightarrow{y}) = 1$ *then* $\mathsf{Decrypt}(C, \boldsymbol{k}^*_{\overrightarrow{x},I}) = M$*.*
- *If* $f_{\overrightarrow{x}}(\overrightarrow{y}) = 0$ *then* $\mathsf{Decrypt}(C, \boldsymbol{k}^*_{\overrightarrow{x},I}) = \perp$ *with all but negligible probability.*

3.2 Definitions of Attribute-Hiding and Full-Hiding in RPE

Our first security definition for RPE schemes is the standard property of attribute hiding, which we extend to address revocation. We allow the adversary to specify the revocation list used to create the challenge ciphertext but we do not require ciphertexts to hide information about revoked key indices. This definition suits applications where revocation lists are managed and published by the master authority.

Definition 4 (Attribute-Hiding RPE). *An* RPE *scheme is adaptively attribute hiding against chosen plaintext attacks if for all PPT adversaries* $\mathcal{A}$*, the advantage* $\mathsf{Adv}^{\mathsf{AH}}_{\mathcal{A},\mathsf{RPE}}(\lambda)$ *of* $\mathcal{A}$ *in the following game is negligible in the security parameter* λ*:*

Setup. *A challenger* $\mathcal{C}$ *runs the* Setup *algorithm to generate a public key* PK*, a master secret key* MSK*, and* S*.* PK *is given to* $\mathcal{A}$*.*

Query phase 1. $\mathcal{A}$ *adaptively makes a polynomial number of* GenKey *queries:* $\mathcal{A}$ *produces a predicate* $\overrightarrow{x}$*,* $\mathcal{C}$ *computes the key* $\boldsymbol{k}^*_{\overrightarrow{x},I} \xleftarrow{R} \mathsf{GenKey}(MSK, S, \overrightarrow{x})$ *associated with an index* I*, and gives it to* $\mathcal{A}$*.*

Challenge. $\mathcal{A}$ *outputs challenge attribute vectors* $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$*, challenge plaintexts* $(M^{(0)}, M^{(1)})$*, and a revocation list* L*, subject to one of the following restrictions for each queried key* $\boldsymbol{k}^*_{\overrightarrow{x},I}$*:*

1. $I \in L$
2. $I \notin L$ *and* $f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 0$*.*

$\mathcal{C}$ *flips a random coin* b*. If* $b = 0$ *then* $\mathcal{A}$ *is given* $C = \mathsf{Encrypt}(PK, L, \overrightarrow{y}^{(0)}, M^{(0)})$*. If* $b = 1$ *then* $\mathcal{A}$ *is given* $C = \mathsf{Encrypt}(PK, L, \overrightarrow{y}^{(1)}, M^{(1)})$*.*

Query phase 2. *Repeat the* **Query phase 1** *subject to the restrictions as in the challenge phase.*

Guess. $\mathcal{A}$ *outputs a guess* b' *of* b*, and succeeds if* $b' = b$.

The advantage of $\mathcal{A}$ *is defined to be* $\mathsf{Adv}^{\mathsf{AH}}_{\mathcal{A},\mathsf{RPE}}(\lambda) = |\mathsf{Pr}[b = b'] - 1/2|$.

Our second security notion, called *full hiding*, offers stronger privacy guarantees. In addition to attribute hiding, it ensures that ciphertexts do not leak any information about revoked indexes. This privacy goal becomes especially relevant when key indexes can be linked to users and whenever senders wish to exclude certain users from decryption — the latter concept of sender-local revocation (SLR) [2] allows senders to define revocation lists (per ciphertext) during the encryption process and by this flexibly refine access control to encrypted data. The SLR property is particularly useful for broadcast systems, e.g. in Pay-TV, where the sender distributes the content and also manages revocation lists and keys (e.g. so-called target broadcast system from [12]). The sender could locally revoke certain customers (e.g. those in delay with payment) for a number of transmissions. The full hiding property is also relevant for applications where revocation lists are sensitive. Consider an illustrative example, where some intelligence agency may want to broadcast confidential information to all agents with certain attributes and yet still exclude James from accessing the information, i.e. irrespective of whether James' key satisfies the policy of the ciphertext. The full hiding property would effectively hide the fact that James' decrypting rights were revoked for that particular ciphertext, even from James himself, who wouldn't know whether decryption failed because of revocation or due to a policy mismatch between his key and the ciphertext.

Definition 5 (Full-Hiding RPE). *An* RPE *scheme is adaptively full hiding against chosen plaintext attacks if for all PPT adversaries* $\mathcal{A}$*, the advantage* $\mathsf{Adv}^{\mathsf{FH}}_{\mathcal{A},\mathsf{RPE}}(\lambda)$ *of* $\mathcal{A}$ *in the following game is negligible in the security parameter* λ*:*

Setup. *A challenger* $\mathcal{C}$ *runs the* Setup *algorithm to generate a public key* PK*, a master secret key* MSK*, and* S*.* PK *is given to* $\mathcal{A}$.

Query phase 1. $\mathcal{A}$ *adaptively makes a polynomial number of* GenKey *queries:* $\mathcal{A}$ *produces a predicate* $\overrightarrow{x}$*,* $\mathcal{C}$ *computes the key* $\boldsymbol{k}^*_{\overrightarrow{x},I} \xleftarrow{R} \mathsf{GenKey}(MSK, S, \overrightarrow{x})$ *associated with an index* I*, and gives it to* $\mathcal{A}$.

Challenge. $\mathcal{A}$ *outputs challenge attribute vectors* $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$*, challenge revocation lists* $(L^{(0)}, L^{(1)})$*, and challenge plaintexts* $(M^{(0)}, M^{(1)})$*, subject to one of the following restrictions for each queried key* $\boldsymbol{k}^*_{\overrightarrow{x},I}$*:*

1. $f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 0$
2. $f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 1$ *and* $(I \in L^{(0)} \wedge I \in L^{(1)})$
3. $(f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = 1 \wedge f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 0)$ *and* $I \in L^{(0)}$
4. $(f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = 0 \wedge f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 1)$ *and* $I \in L^{(1)}$.

$\mathcal{C}$ *flips a random coin* b*. If* $b = 0$ *then* $\mathcal{A}$ *is given* $C = \mathsf{Encrypt}(PK, L^{(0)}, \overrightarrow{y}^{(0)}, M^{(0)})$*. If* $b = 1$ *then* $\mathcal{A}$ *is given* $C = \mathsf{Encrypt}(PK, L^{(1)}, \overrightarrow{y}^{(1)}, M^{(1)})$.

Query phase 2. *Repeat the* **Query phase 1** *subject to the restrictions as in the challenge phase.*

Guess. $\mathcal{A}$ *outputs a guess* b' *of* b*, and succeeds if* $b' = b$.

The advantage of $\mathcal{A}$ *is defined to be* $\mathsf{Adv}^{\mathsf{FH}}_{\mathcal{A},\mathsf{RPE}}(\lambda) = |\mathsf{Pr}[b = b'] - 1/2|$.

Remark 1. In Definition 5, adversary $\mathcal{A}$ is not allowed to ask a key query for an index I and a predicate $\overrightarrow{x}$ such that $I \notin L^{(b)}$ and $f_{\overrightarrow{x}}(\overrightarrow{y}^{(b)}) = 1$ for some $b \in \{0, 1\}$, i.e., the queried key is not allowed to decrypt the challenge ciphertext. Recently, Okamoto and Takashima [22] proposed a (Hierarchical) Predicate Encryption scheme which allows such key query, provided that $M^{(0)} = M^{(1)}$. We observe that their techniques can be applied in our RPE schemes to handle such queries too.

Remark 2. Definitions 4 and 5 can be easily extended to capture chosen-ciphertext attacks (CCA) by allowing decryption queries (for all but the challenge ciphertext). The advantage of $\mathcal{A}$ in such CCA game is defined to be $\mathsf{Adv}^{\mathsf{X\text{-}CCA}}_{\mathcal{A},\mathsf{RPE}}(\lambda) = |\mathsf{Pr}[b = b'] - 1/2|$, where $X \in \{\mathsf{AH}, \mathsf{FH}\}$. We can also define relaxed selective security, where the adversary is required to specify the challenge attributes and the revocation list in advance (before obtaining public key PK).

4 An RPE Scheme with Attribute Hiding (AH-RPE)

In this section we present our first RPE scheme, which achieves the property of attribute hiding. We construct a system in which the sizes of public and private keys are small and constant. The size of the ciphertext is linear in the number of revoked keys, which is small relative to the total number of users. The efficiency of our scheme is comparable to the existing schemes of Lewko *et al.* [15] and Attrapadung *et al.* [3]. The broadcast encryption scheme proposed by Boneh *et al.* [8] produces ciphertexts and private keys of constant size, however the size of the public keys is linear in the number of users in the system.

4.1 Intuition behind AH-RPE

Our construction uses the "two equation" revocation technique of the public broadcast encryption system of Lewko, Sahai and Waters (LSW) [15]. In the LSW scheme the secret s that allows decryption of the ciphertext is broken into as many shares as revoked indexes. Using the"two equation" technique, a key whose index is not revoked in the ciphetext can be used to compute all the shares of the secret s. We combine LSW's "two equation" concept [15] and Okamoto and Takashima's FE scheme [20] to construct our first scheme. Informally, we compute a key for a predicate $\overrightarrow{x}$ and an associated index I, the ciphertext is encrypted with a message M, an attribute $\overrightarrow{y}$ and the set of indexes $\{I_1, \ldots, I_r\}$ of the revoked keys, where r is the number of the revoked keys. The ciphertext can be decrypted with the key if $I \neq I_i$ for all $i \in [1, r]$. In our scheme, we realize the "two equation" technique by employing non zero inner product evaluations. If the inner product of two vectors is non zero (signifying that the two indexes are not equal), then we can recover the share s_i, otherwise the decryption fails.

4.2 Specification of AH-RPE

We now give detailed specification of our AH-RPE scheme:

$\mathsf{Setup}\big(1^\lambda, \Delta = (\overrightarrow{n} = (2; n_1, n_2 = 2), N)\big)$: Perform the following computations:

$(\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{R} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \overrightarrow{n})$,

$\widetilde{\mathbb{B}}^{(0)} = (\boldsymbol{b}_1^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_5^{(0)})$, $\widetilde{\mathbb{B}}^{(1)} = (\boldsymbol{b}_1^{(1)}, \ldots, \boldsymbol{b}_{n_1}^{(1)}, \boldsymbol{b}_{3n_1+1}^{(1)})$, $\widetilde{\mathbb{B}}^{(2)} = (\boldsymbol{b}_1^{(2)}, \boldsymbol{b}_2^{(2)}, \boldsymbol{b}_7^{(2)})$,

$\widetilde{\mathbb{B}}^{*(0)} = (\boldsymbol{b}_1^{*(0)}, \boldsymbol{b}_3^{*(0)}, \boldsymbol{b}_4^{*(0)})$, $\widetilde{\mathbb{B}}^{*(1)} = (\boldsymbol{b}_1^{*(1)}, \ldots, \boldsymbol{b}_{n_1}^{*(1)}, \boldsymbol{b}_{2n_1+1}^{*(1)}, \ldots, \boldsymbol{b}_{3n_1}^{*(1)})$, $\widetilde{\mathbb{B}}^{*(2)} = (\boldsymbol{b}_1^{*(2)}, \boldsymbol{b}_2^{*(2)}, \boldsymbol{b}_5^{*(2)}, \boldsymbol{b}_6^{*(2)})$.

Let S denote the (initially empty) state information on the so far assigned indices I. The output of the algorithm is given by the public key $PK = \big(1^\lambda, N, \mathsf{param}_{\overrightarrow{n}}, \{\widetilde{\mathbb{B}}^{(k)}\}_{k\in\{0,1,2\}}\big)$, the master secret key $MSK = \big(\{\widetilde{\mathbb{B}}^{*(k)}\}_{k\in\{0,1,2\}}\big)$, and the state information S.

$\mathsf{GenKey}(MSK, S, \overrightarrow{x} = (x_1, \ldots, x_{n_1}) \in \mathbb{F}_q^{n_1} \setminus \{\overrightarrow{0}\})$: Choose $s, \eta, \beta, \eta_1, \ldots, \eta_{n_1}, \rho_1, \rho_2 \xleftarrow{U} \mathbb{F}_q$, choose $s_1, s_2 \xleftarrow{U} \mathbb{F}_q$ such that $s = s_1 + s_2$, choose index $I \xleftarrow{U} \Gamma$ such that $I \notin S$, then set $S = S \cup \{I\}$ and compute:

$$\begin{aligned}
\boldsymbol{k}_0 &= (-s, 0, 1, \eta, 0)_{\mathbb{B}^{*(0)}},\\
\boldsymbol{k}_1 &= (\overbrace{s_1 \overrightarrow{e}_1^{(1)} + \beta \overrightarrow{x}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\eta_1, \ldots, \eta_{n_1}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}},\\
\boldsymbol{k}_2 &= (\overbrace{s_2(1, I)}^{2}, \overbrace{0^2}^{2}, \overbrace{\rho_1, \rho_2}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}}.
\end{aligned}$$

Output the updated state information S and the secret key $\boldsymbol{k}^*_{\overrightarrow{x},I} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \boldsymbol{k}_2)$.

$\mathsf{Encrypt}(PK, L, \overrightarrow{y} = (y_1, \ldots, y_{n_1}) \in \mathbb{F}_q^{n_1} \setminus \{\overrightarrow{0}\}, M \in \mathbb{G}_T)$: If L is empty, set $L = \{N+1\}$, where $N+1$ is a dummy index. Choose $\delta, \zeta, \varphi, \varphi' \xleftarrow{\mathsf{U}} \mathbb{F}_q$, also choose $\varphi_r, \delta_r \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for all $r \in L$ such that $\delta = \sum_{r \in L} \delta_r$, and compute:

$$
\begin{aligned}
& \boldsymbol{c}_0 = (\delta, 0, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}}, \\
& \boldsymbol{c}_1 = (\overbrace{\delta \overrightarrow{y}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi'}^{1})_{\mathbb{B}^{(1)}}, \\
\forall r \in L: \quad & \boldsymbol{c}_r = (\overbrace{\delta_r(-r, 1)}^{2}, \overbrace{0^2}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_r}^{1})_{\mathbb{B}^{(2)}}, \\
& \boldsymbol{c}_M = g_T^{\zeta} M.
\end{aligned}
$$

Output the ciphertext $C = (L, \boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_r\}_{r \in L}, \boldsymbol{c}_M)$.

$\mathsf{Decrypt}(C, \boldsymbol{k}^*_{\overrightarrow{x}, I})$: Given a ciphertext $C = (L, \boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_r\}_{r \in L}, \boldsymbol{c}_M)$ and a secret key $\boldsymbol{k}^*_{\overrightarrow{x}, I} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \boldsymbol{k}_2)$, if $I \in L$, output $\perp$; otherwise compute and output message M:

$$
M = \frac{\boldsymbol{c}_M}{e(\boldsymbol{c}_0, \boldsymbol{k}_0) e(\boldsymbol{c}_1, \boldsymbol{k}_1) \prod_{r \in L} e(\boldsymbol{c}_r, \boldsymbol{k}_2)^{\frac{1}{I-r}}}.
$$

4.3 Correctness of AH-RPE

The correctness of our scheme holds due to the following observation. Let C and $\boldsymbol{k}^*_{\overrightarrow{x}, I}$ be as above. If $\overrightarrow{x} \cdot \overrightarrow{y} = 0$ and $I \notin L$ then M can be recovered by as specified in the decryption procedure due to the equality

$$
e(\boldsymbol{c}_0, \boldsymbol{k}_0) e(\boldsymbol{c}_1, \boldsymbol{k}_1) \prod_{r \in L} e(\boldsymbol{c}_r, \boldsymbol{k}_2)^{\frac{1}{I-r}} = g_T^{-s\delta+\zeta} g_T^{s_1 \delta + \beta \delta \overrightarrow{x} \cdot \overrightarrow{y}} g_T^{s_2 \sum_{r \in L} \delta_r} = g_T^{-s\delta+\zeta} g_T^{s\delta} = g_T^{\zeta}.
$$

Remark 3. In the $\mathsf{Encrypt}$ algorithm, if the revocation list L is empty, i.e., no key is revoked, a dummy index $N+1$ is placed into the revocation list. Since $N+1$ is not in the index space Γ, the ciphertext computed from $L = \{N+1\}$ and an attribute $\overrightarrow{y}$ can be decrypted by any key $\boldsymbol{k}^*_{\overrightarrow{x}, I}$ provided $\overrightarrow{x} \cdot \overrightarrow{y} = 0$.

In our scheme, the size of private key is small and constant, i.e., $(13 + 3n_1)|\mathbb{G}|$, and size of the ciphertext is linear in the number of revoked keys, i.e., $(6 + 3n_1 + 7r)|\mathbb{G}| + |\mathbb{G}_T|$, where $r, |\mathbb{G}|$ and $|\mathbb{G}_T|$ denotes the number of revoked keys, size of group element in $\mathbb{G}$ and size of group element in $\mathbb{G}_T$ respectively.

4.4 Security Analysis of AH-RPE

The attribute hiding property of our AH-RPE scheme is established through the following theorem.

Theorem 1. *Our AH-RPE is adaptively attribute hiding against chosen plaintext attacks under the DLIN assumption. For any adversary $\mathcal{A}$, there exists a probabilistic polynomial time machine $\mathcal{D}$ such that for any security parameter λ,*

$$
\mathsf{Adv}^{\mathsf{AH}}_{\mathcal{A}, \mathsf{AH\text{-}RPE}}(\lambda) \leq (2\nu + 1) \mathsf{Adv}^{\mathrm{DLIN}}_{\mathcal{D}}(\lambda) + \psi
$$

where ν is the maximum number of $\mathcal{A}$'s key queries and $\psi = (2\nu|L| + 18\nu + 10)/q$ ($|L|$ denotes the number of revoked keys).

The proof of Theorem 1 is presented in Appendix A.

5 An RPE Scheme with Full Hiding (FH-RPE)

In this section, we present our second RPE scheme, which achieves the property of full hiding. The scheme is based on Okamoto and Takashima's FE [20] and the Subset-Cover Framework due to Naor *et al.* [18].

5.1 Intuition

The intuition behind the new construction is as follows. In addition to having vectors representing predicates and attributes, we have index vectors and revocation vectors. The scheme can be seen as the composition of two encryption steps, one using the attributes and predicate vectors, and the other one using the index and revocation vectors. Attributes and predicates vectors use a different basis from index and revocation list vectors. By using separate bases, we avoid the quadratic length growth that would otherwise occur if we concatenated the vectors using the same basis. We also use a secret sharing scheme over bilinear pairing groups to combine the components in the secret key, so that it is hard to modify the secret key to make valid a key that has otherwise been revoked.

Each key has an associated index vector $\overrightarrow{x}_I$, which encodes the key index I by assigning a random value to the vector component at the corresponding index position. To revoke a set of keys, the encryptor sets random values on positions in the revocation vector $\overrightarrow{y}_L$ that correspond to the indexes of revoked keys. We see that if $\overrightarrow{y}_L$ has a random value in the I^{th} component, then $\overrightarrow{x}_I \cdot \overrightarrow{y}_L \neq 0$ and results in a random group element on decryption. This indicates that the key with index I is revoked. We assume that both index vector and revocation vector are initially set to $\overrightarrow{0}$. If we denote the number of keys in the system as N, then the size of both index vectors and revocation vectors is $O(N)$. If the predicate/attribute vector is of size n, then the size of ciphertexts and keys is $O(n+N)$.

The major drawback of such approach is, however, the space cost. To alleviate this limitation, our scheme takes advantage of the **complete-subtree** data structure from [18]. Informally, in a binary tree with N leaves, the index I of a key will be associated with a leaf node. Each node in the tree will be assigned a unique identity. To compute a key with an index, we compute on identities of all the nodes on the path from the leaf node associated with I to the root node. To encrypt, the sender first finds a minimal set of nodes which contains an ancestor (or, the node itself) of all the non-revoked indexes. It then computes ciphertext on the attribute and the identities of all the nodes in that set. To retain the full-hiding property we apply the binary structure in an anonymous setting. Decryption works if there exists one common node (identity) between the key and the ciphertext, which is given for unrevoked keys only. By adopting this approach the length of ciphertexts in our scheme ranges between $O(1)$ (when no key is revoked) and $O(\frac{N}{2})$ (in the worst case when every second key is revoked) while the length of private keys is $O(\log N)$. All in all, by adopting the complete-subtree structure we achieve a remarkable space gain in comparison to the above case.

Example. Consider the following illustrative example: in Figure 1 (left), indexes 2 and 6 are associated with different keys, say $\boldsymbol{k}^*_{\overrightarrow{x_2},2}$ and $\boldsymbol{k}^*_{\overrightarrow{x_6},6}$ respectively. The key $\boldsymbol{k}^*_{\overrightarrow{x_2},2}$ is computed on the predicate vector $\overrightarrow{x_2}$ and tree nodes $\{ID_1, ID_2, ID_3, ID_4\}$, whereas $\boldsymbol{k}^*_{\overrightarrow{x_6},6}$ is computed on predicate vector $\overrightarrow{x_6}$ and nodes $\{ID_1, ID_5, ID_6, ID_7\}$. Assume in Figure 1 (right) that $\boldsymbol{k}^*_{\overrightarrow{x_2},2}$ is revoked. The minimal subset of nodes covering all other indexes is $\{ID_5, ID_8, ID_9\}$. The ciphertext $\boldsymbol{c}_x$ will thus be computed on nodes $\{ID_5, ID_8, ID_9\}$. Since $\boldsymbol{c}_x$ doesn't have any common node with revoked key $\boldsymbol{k}^*_{\overrightarrow{x_2},2}$, decryption with this key will fail but it will succeed with $\boldsymbol{k}^*_{\overrightarrow{x_6},6}$ due to the common node ID_5.

5.2 Specification of FH-RPE

We proceed with a detailed specification of our FH-RPE scheme:

$\mathsf{Setup}\big(1^\lambda, \Delta = (\overrightarrow{n} = (2; n_1, n_2 = 2), N)\big)$: Perform the following computations:
$(\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{R} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \overrightarrow{n})$,

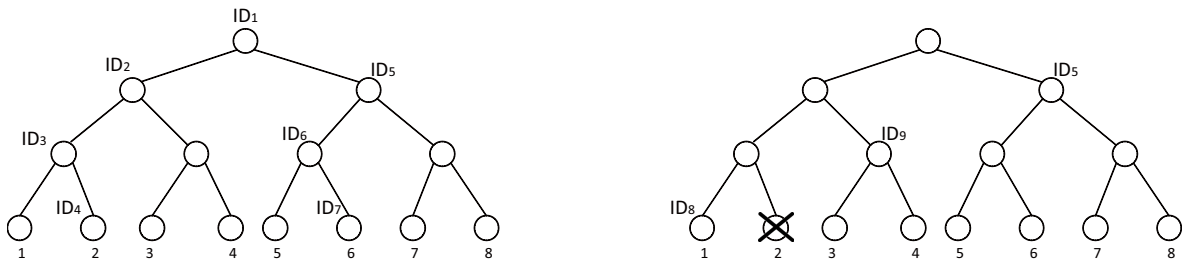

Fig. 1. Complete Subtree Technique [18] in RPE (intuitive idea)

$\widetilde{\mathbb{B}}^{(0)} = (\boldsymbol{b}_1^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_5^{(0)})$, $\widetilde{\mathbb{B}}^{(1)} = (\boldsymbol{b}_1^{(1)}, \ldots, \boldsymbol{b}_{n_1}^{(1)}, \boldsymbol{b}_{3n_1+1}^{(1)})$, $\widetilde{\mathbb{B}}^{(2)} = (\boldsymbol{b}_1^{(2)}, \boldsymbol{b}_2^{(2)}, \boldsymbol{b}_7^{(2)})$,

$\widetilde{\mathbb{B}}^{*(0)} = (\boldsymbol{b}_1^{*(0)}, \boldsymbol{b}_3^{*(0)}, \boldsymbol{b}_4^{*(0)})$, $\widetilde{\mathbb{B}}^{*(1)} = (\boldsymbol{b}_1^{*(1)}, \ldots, \boldsymbol{b}_{n_1}^{*(1)}, \boldsymbol{b}_{2n_1+1}^{*(1)}, \ldots, \boldsymbol{b}_{3n_1}^{*(1)})$, $\widetilde{\mathbb{B}}^{*(2)} = (\boldsymbol{b}_1^{*(2)}, \boldsymbol{b}_2^{*(2)}, \boldsymbol{b}_5^{*(2)}, \boldsymbol{b}_6^{*(2)})$.
Let *Tree* be a complete binary tree structure with at least N leaf nodes, which corresponds to the number of keys in the system. Each node x in *Tree* has unique identity ID_x. Let state information S, which records the assigned indexes I so far, be an initially empty set.
The output of the algorithm is given by the public key $PK = \big(1^\lambda, \mathsf{param}_{\overrightarrow{n}}, \{\widetilde{\mathbb{B}}^{(k)}\}_{k=0,1,2}, \mathit{Tree}\big)$, the master secret key $\mathit{MSK} = \big(\{\widetilde{\mathbb{B}}^{*(k)}\}_{k=0,1,2}\big)$, and the state information S.

$\mathsf{GenKey}(\mathit{MSK}, S, \overrightarrow{x} = (x_1, \ldots, x_{n_1}) \in \mathbb{F}_q^{n_1} \backslash \{\overrightarrow{0}\})$: Choose $\alpha, \eta, \eta_1^{(1)}, \ldots, \eta_{n_1}^{(1)}, \beta^{(1)} \xleftarrow{\mathsf{U}} \mathbb{F}_q$, and choose $\alpha^{(1)}, \alpha^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q$ such that $\alpha = \alpha^{(1)} + \alpha^{(2)}$; choose index $I \xleftarrow{\mathsf{U}} \Gamma$ such that $I \notin S$, and set $S = S \cup \{I\}$. Then, compute

$$\begin{aligned}
\boldsymbol{k}_0 &= (-\alpha, 0, 1, \eta, 0)_{\mathbb{B}^{*(0)}}, \\
\boldsymbol{k}_1 &= (\overbrace{\alpha^{(1)}\overrightarrow{e}_1^{(1)} + \beta^{(1)}\overrightarrow{x}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\eta_1^{(1)}, \ldots, \eta_{n_1}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}}, \\
\forall x \in \mathbb{P}(I): \quad \boldsymbol{k}_x &= (\overbrace{\alpha^{(2)} + \beta_x^{(2)} ID_x, \beta_x^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\eta_{1,x}^{(2)}, \eta_{2,x}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}}, \text{ with } \beta_x^{(2)}, \eta_{1,x}^{(2)}, \eta_{2,x}^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q.
\end{aligned}$$

The output is given by S and the secret key $\boldsymbol{k}^*_{\overrightarrow{x},I} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \{\boldsymbol{k}_x\}_{x \in \mathbb{P}(I)})$.
(Note that I is associated with the I_{th} leaf node in the binary tree. $\mathbb{P}(I)$ denotes all the nodes on the path from the leaf node I up to the root node (leaf and root nodes inclusive). The secret key $\boldsymbol{k}^*_{\overrightarrow{x},I}$ thus contains secrets for all nodes ID_x on the mentioned path from I to the root.)

$\mathsf{Encrypt}(PK, L, \overrightarrow{y} = (y_1, \ldots, y_{n_1}) \in \mathbb{F}_q^{n_1} \setminus \{\overrightarrow{0}\}, M \in \mathbb{G}_T)$: Choose $\delta, \zeta, \varphi, \varphi^{(1)} \xleftarrow{\mathsf{U}} \mathbb{F}_q$ and compute:

$$\begin{aligned}
\boldsymbol{c}_0 &= (\delta, 0, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}}, \\
\boldsymbol{c}_1 &= (\overbrace{\delta\overrightarrow{y}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\
\forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L): \quad \boldsymbol{c}_x &= (\overbrace{\delta, \delta(-ID_x)}^{2}, \overbrace{0^2}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_x^{(2)}}^{1})_{\mathbb{B}^{(2)}} \text{ where } \varphi_x^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q, \\
\boldsymbol{c}_M &= g_T^\zeta M.
\end{aligned}$$

Output the ciphertext $C = (\boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_x\}_{x \in \mathsf{RevokeNodes}(\mathit{Tree}, L)}, \boldsymbol{c}_M)$.

Remark 4. Note that $\mathsf{RevokeNodes}(\mathit{Tree}, L)$ outputs a minimal set of nodes which contains an ancestor (or, the node itself) of all the non-revoked indexes. Each ciphertext component $\boldsymbol{c}_x$ is then computed on all the identities of the nodes in the set.

$\mathsf{Decrypt}(C, \boldsymbol{k}^*_{\overrightarrow{x},I})$: Given a ciphertext $C = (\boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_x\}_{x\in\mathsf{RevokeNodes}(\mathit{Tree},L)}, \boldsymbol{c}_M)$ and a secret key $\boldsymbol{k}^*_{\overrightarrow{x},I} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \{\boldsymbol{k}_{x'}\}_{x'\in\mathbb{P}(I)})$ compute

$$\forall x, x' : \quad M_{x,x'} = \frac{\boldsymbol{c}_M}{e(\boldsymbol{c}_0, \boldsymbol{k}_0)e(\boldsymbol{c}_1, \boldsymbol{k}_1)e(\boldsymbol{c}_x, \boldsymbol{k}_{x'})}.$$

If there exists a pair (x, x') corresponding to the same node in *Tree* and $\overrightarrow{x} \cdot \overrightarrow{y} = 0$, the decrypted message is $M = M_{x,x'}$. Otherwise, obtained messages are random with all but negligible probability. Note that it is not necessary to test all pairs of (x, x'), i.e., it is possible to mark the level of each node encrypted in the ciphertext (without revealing the node's identifier) and compute using pairs (x, x') that are located on the same level in the tree. In this way decryption costs can be decreased from $O((\log N)^2)$ to $O(\log N)$.

5.3 Correctness of FH-RPE

To see why the FH-RPE scheme is correct, let C and $\boldsymbol{k}^*_{\overrightarrow{x},I}$ be as above. If $\overrightarrow{x} \cdot \overrightarrow{y} = 0$ and $I \notin L$ then M can be recovered as specified in the decryption procedure due to the following equality

$$e(\boldsymbol{c}_0, \boldsymbol{k}_0)e(\boldsymbol{c}_1, \boldsymbol{k}_1)e(\boldsymbol{c}_x, \boldsymbol{k}_{x'}) = g_T^{-\alpha\delta+\zeta} g_T^{\alpha^{(1)}\delta+\beta^{(1)}\delta\overrightarrow{x}\cdot\overrightarrow{y}} g_T^{\alpha^{(2)}\delta+\beta^{(2)}_{x'}\delta(ID_{x'}-ID_x)} = g_T^{-\alpha\delta+\zeta} g_T^{\alpha\delta} = g_T^{\zeta}.$$

5.4 Security Analysis of FH-RPE

The full hiding property of our FH-RPE scheme is established through the following theorem.

Theorem 2. *FH-RPE is adaptively full hiding against chosen plaintext attacks under the DLIN assumption (provided the restriction in Remark 5 holds). For any adversary $\mathcal{A}$, there exists a probabilistic polynomial time machine $\mathcal{D}$ such that for any security parameter λ,*

$$\mathsf{Adv}^{\mathsf{FH}}_{\mathcal{A},\mathsf{FH\text{-}RPE}}(\lambda) \leq (2\nu + 1)\mathsf{Adv}^{\mathrm{DLIN}}_{\mathcal{D}}(\lambda) + \psi$$

where ν is the maximum number of $\mathcal{A}$'s key queries and $\psi = (2\log N\nu + 18\nu + \log N + 10)/q$.

The proof of Theorem 2 is presented in Appendix B.

Remark 5. Proof of Theorem 2 assumes that $|X^{(0)}| = |X^{(1)}|$, where $X^{(0)} = \{x | x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(0)})\}$ and $X^{(1)} = \{x' | x' \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(1)})\}$. The revocation lists $L^{(0)}$ and $L^{(1)}$ are defined in the challenge phase of Definition 5. This restriction is necessary to prevent the adversary from trivially winning the game based on the length of the challenge ciphertext.

Remark 6. The PE scheme from [14], on which our RPE construction is based, is proven adaptive attribute-hiding in the standard model. Its security proof cannot be applied to Theorem 2 directly — in [14] the only restriction on the challenge attribute vector $\overrightarrow{y}^{(b)}$ and predicate vector $\overrightarrow{x}$ was $\overrightarrow{x} \cdot \overrightarrow{y}^{(b)} \neq 0$, where $b \in \{0, 1\}$. The more sophisticated challenge phase restrictions in our Definition 5 of full hiding make the proof of Theorem 2 slightly more complicated than in [14].

6 Conclusion

We formalized Revocable Predicate Encryption (RPE) and proposed two RPE schemes. Our AH-RPE scheme is attribute hiding whereas our FH-RPE scheme offers stronger full hiding. Both schemes are proven secure in the standard model under the DLIN assumption. Recently, Okamoto and Takashima [21] proposed a PE scheme with short private keys. We observe that private keys in our RPE constructions can be further reduced in size by adopting their techniques.

Acknowledgements

This research work is part of the bilateral research project between Germany and Australia, funded jointly by the German Academic Exchange Service (DAAD) through grant Nr. 53361649 and by Australia's Department of Innovation, Industry, Science and Research (DIISR). Mark Manulis was also supported by the German Research Foundation (DFG) through grant MA 4096. He wishes further to acknowledge support from the Center of Advanced Security Research Darmstadt (CASED) and the European Center for Security and Privacy by Design (EC SPRIDE).

References

1. William Aiello, Sachin Lodha, and Rafail Ostrovsky. Fast digital identity revocation (extended abstract). In *CRYPTO '98*, volume 1462 of *LNCS*, pages 137–152. Springer, 1998.
2. Nuttapong Attrapadung and Hideki Imai. Attribute-based encryption supporting direct/indirect revocation modes. In *Cryptography and Coding 2009*, volume 5921 of *LNCS*, pages 278–300. Springer, 2009.
3. Nuttapong Attrapadung and Hideki Imai. Conjunctive broadcast and attribute-based encryption. In *Pairing 2009*, volume 5671 of *LNCS*, pages 248–265. Springer, 2009.
4. John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. In *IEEE Symposium on Security and Privacy*, pages 321–334. IEEE, 2007.
5. Alexandra Boldyreva, Vipul Goyal, and Virendra Kumar. Identity-based encryption with efficient revocation. In *ACM CCS 2008*, pages 417 – 426. ACM, 2008.
6. Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In *CRYPTO 2004*, volume 3152 of *LNCS*, pages 41–55. Springer, 2004.
7. Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil Pairing. In *CRYPTO 2001*, volume 2139 of *LNCS*, pages 213–229. Springer, 2001.
8. Dan Boneh, Craig Gentry, and Brent Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In *CRYPTO 2005*, volume 3621 of *LNCS*, pages 258–275. Springer, 2005.
9. Dan Boneh and Brent Waters. Conjunctive, subset, and range queries on encrypted data. In *TCC 2007*, volume 4392 of *LNCS*, pages 535–554. Springer, 2007.
10. Melissa Chase. Multi-authority attribute based encryption. In *TCC 2007*, volume 4392 of *LNCS*, pages 515–534. Springer, 2007.
11. Craig Gentry. Certificate-based encryption and the certificate revocation problem. In *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 272–293. Springer, 2003.
12. Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *ACM CCS 2006*, pages 89–98. ACM, 2006.
13. Jonathan Katz, Amit Sahai, and Brent Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 146–162. Springer, 2008.
14. Allison B. Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, and Brent Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 62–91. Springer, 2010.
15. Allison B. Lewko, Amit Sahai, and Brent Waters. Revocation systems with very small private keys. In *IEEE S&P*, pages 273 – 285. IEEE, 2010.
16. Allison B. Lewko and Brent Waters. New techniques for dual system encryption and fully secure hibe with short ciphertexts. In *TCC 2010*, volume 5978 of *LNCS*, pages 455–479. Springer, 2010.
17. Benoit Libert, Kenneth G. Paterson, and Elizabeth A. Quaglia. Anonymous broadcast encryption: Adaptive security and efficient constructions in the standard model. Cryptology ePrint Archive, Report 2011/476, 2011. `http://eprint.iacr.org/`.
18. Dalit Naor, Moni Naor, and Jeffery Lotspiech. Revocation and tracing schemes for stateless receivers. In *CRYPTO 2001*, volume 2139 of *LNCS*, pages 41–62. Springer, 2001.
19. Tatsuaki Okamoto and Katsuyuki Takashima. Hierarchical predicate encryption for inner-products. In *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 214–231. Springer, 2009.
20. Tatsuaki Okamoto and Katsuyuki Takashima. Fully secure functional encryption with general relations from the decisional linear assumption. In *CRYPTO 2010*, volume 6223 of *LNCS*, pages 191–208. Springer, 2010.
21. Tatsuaki Okamoto and Katsuyuki Takashima. Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption. In *CANS*, volume 7092 of *LNCS*, pages 138–159. Springer, 2011.

22. Tatsuaki Okamoto and Katsuyuki Takashima. Adaptively attribute-hiding (hierarchical) inner product encryption. In *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 591–608. Springer, 2012.
23. Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 457–473. Springer, 2005.
24. Adi Shamir. Identity based cryptosystems and signature schemes. In *CRYPTO'84*, volume 0196 of *LNCS*, pages 47–53. Springer, 1984.
25. Emily Shen, Elaine Shi, and Brent Waters. Predicate privacy in encryption systems. In *TCC 2009*, volume 5444 of *LNCS*, pages 457–473. Springer, 2009.
26. Elaine Shi, John Bethencourt, Hubert T.-H. Chan, Dawn Xiaodong Song, and Adrian Perrig. Multi-Dimensional Range Query over Encrypted Data. In *IEEE S&P*, pages 350–364. IEEE, 2007.
27. Elaine Shi and Brent Waters. Delegating Capabilities in Predicate Encryption Systems. In *ICALP 2008 (2)*, volume 5126 of *LNCS*, pages 560–578. Springer, 2008.
28. Brent Waters. Dual system encryption: Realizing fully secure ibe and hibe under simple assumptions. In *CRYPTO 2009*, volume 5677 of *LNCS*, pages 619–636. Springer, 2009.

A Security Proof of Theorem 1

We start by defining two high-level computational problems, Problem 1 and 2, and show that each of these problems is hard under the classical DLIN assumption. Our proof uses a sequence of games. In our games we define semi-functional keys expressed by Eq.(7). Semi-functional ciphertexts are defined in Eqs.(1)-(4). Semi-functional keys can decrypt all normal ciphertexts, but not semi-functional ciphertexts. Semi-functional ciphertexts can be decrypted only by normal keys. We also introduce nominal semi-functional ciphertexts and keys Eq.(6) and Eq.(5), similar to [16].

Normal ciphertexts and keys are used in Game 0 (the original game of Definition 4), while their nominal semi-functional or semi-functional counterparts are used in subsequent games only. In Game 1, the challenge ciphertext is changed to a semi-functional one. We then consider 2ν game hops from Game 1 (Game 2-0), Game 2-0′, Game 2-1, Game 2-1′, ..., to Game 2-$(\nu-1)'$ and Game 2-ν. In Game 2-m ($m = 0, \ldots, \nu-1$), the first m keys are semi-functional and the rest of the keys are normal, and challenge ciphertext is semi-functional. In Game 2-m' ($m = 0, \ldots, \nu-1$), the first m keys are semi-functional and the $(m+1)$-th key is nominal semi-functional while the remaining keys are normal, and challenge ciphertext is nominal semi-functional. In game 3, all queried keys are semi-functional Eq.(7). In the last game, Game 3, all keys and challenge ciphertext are semi-functional, hence the adversary has 0 advantage.

We then show that the difference in the adversary's advantage between Games 0 and 1 is bounded by the advantage of any adversary against Problem 1. The advantage difference between Games 2-m' and 2-m is equivalent to the advantage of Problem 2 (i.e., advantage of the DLIN assumption). Here, we introduce special forms of nominal semi-functional keys $\boldsymbol{k}^{*\,\mathsf{spec\text{-}nom\text{-}semi}}_{\overrightarrow{x},I}$ and ciphertext $C^{\mathsf{sepc\text{-}nom\text{-}semi}}$, respectively. They equal their counterparts in semi-functional forms except that $\epsilon w = \gamma = \gamma^{(1)} + \gamma^{(2)}$ and $\epsilon \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (note that $\epsilon, w \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $\boldsymbol{k}^{*\,\mathsf{nom\text{-}semi}}_{\overrightarrow{x},I}$ and $C^{\mathsf{nom\text{-}semi}}$). $\boldsymbol{k}^{*\,\mathsf{spec\text{-}nom\text{-}semi}}_{\overrightarrow{x},I}$ and $C^{\mathsf{sepc\text{-}nom\text{-}semi}}$ are simulated using Problem 2 instance when $\beta = 1$. Due to the algebraic structure, $\boldsymbol{k}^{*\,\mathsf{spec\text{-}nom\text{-}semi}}_{\overrightarrow{x},I}$ can always decrypt $C^{\mathsf{sepc\text{-}nom\text{-}semi}}$ when the predicate holds and the key is not revoked. Therefore, it is hard for the simulator to identify if $(\boldsymbol{k}^{*\,\mathsf{spec\text{-}nom\text{-}semi}}_{\overrightarrow{x},I}, C^{\mathsf{sepc\text{-}nom\text{-}semi}})$ for Game 2-m' or $(\boldsymbol{k}^{*}_{\overrightarrow{x},I}, C^{\mathsf{semi}})$ for Game 2-m under Problem 2. On the other hand, γ is independently distributed from the other variables when either the predicate does not hold or the key is revoked. That is, the joint distribution of $(\boldsymbol{k}^{*\,\mathsf{spec\text{-}nom\text{-}semi}}_{\overrightarrow{x},I}, C^{\mathsf{sepc\text{-}nom\text{-}semi}})$ is equivalent to that of $(\boldsymbol{k}^{*\,\mathsf{nom\text{-}semi}}_{\overrightarrow{x},I}, C^{\mathsf{nom\text{-}semi}})$ when either condition holds. Hence, both of them appear identical from the adversary's view, since from the security definition the adversary's queries should satisfy at least one of the conditions (predicate does not holds and key is revoked). With the similar argument, we show that the advantage difference between Games 2-m' and 2-$(m+1)$ is equivalent to the advantage of Problem 2 (i.e., advantage of the DLIN assumption). We also show that 2-ν can be conceptually changed to Game 3 whose advantage is 0.

Definition 6 (Problem 1). *Problem 1 is to find bit $\beta \in \{0,1\}$, given* $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{t}_{\beta}^{(0)}, \boldsymbol{t}_{\beta,1}^{(1)}, \boldsymbol{t}_{\beta,1}^{(2)}, \{\boldsymbol{t}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{t}_2^{(2)}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\beta}^{\mathrm{P1}}(1^{\lambda}, \overrightarrow{n} = (2; n_1, n_2 = 2))$*, where*

$$
\begin{aligned}
&\mathcal{G}_{\beta}^{\mathrm{P1}}(1^{\lambda}, \overrightarrow{n} = (2; n_1, n_2 = 2)): \ (\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^{\lambda}, \overrightarrow{n}),\\
&\quad \widehat{\mathbb{B}}^{*(0)} = (\boldsymbol{b}_1^{*(0)}, \boldsymbol{b}_3^{*(0)}, \boldsymbol{b}_4^{*(0)}, \boldsymbol{b}_5^{*(0)}), \qquad \widehat{\mathbb{B}}^{*(1)} = (\boldsymbol{b}_1^{*(1)}, \ldots, \boldsymbol{b}_{n_1}^{*(1)}, \boldsymbol{b}_{2n_1+1}^{*(1)}, \ldots, \boldsymbol{b}_{3n_1+1}^{*(1)}),\\
&\quad \widehat{\mathbb{B}}^{*(2)} = (\boldsymbol{b}_1^{*(2)}, \boldsymbol{b}_2^{*(2)}, \boldsymbol{b}_5^{*(2)}, \boldsymbol{b}_6^{*(2)}, \boldsymbol{b}_7^{*(2)}),\\
&\quad \delta, u, \rho \xleftarrow{\mathsf{U}} \mathbb{F}_q, \ \ \boldsymbol{t}_0^{(0)} = (\delta, 0, 0, 0, \rho)_{\mathbb{B}^{(0)}}, \ \ \boldsymbol{t}_1^{(0)} = (\delta, u, 0, 0, \rho)_{\mathbb{B}^{(0)}},\\
&\quad \rho^{(1)} \xleftarrow{\mathsf{U}} \mathbb{F}_q, \ \overrightarrow{u}^{(1)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_1},\\
&\quad \boldsymbol{t}_{0,1}^{(1)} = (\overbrace{\delta \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\rho^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \ \ \boldsymbol{t}_{1,1}^{(1)} = (\overbrace{\delta \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{\overrightarrow{u}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\rho^{(1)}}^{1})_{\mathbb{B}^{(1)}},\\
&\quad \text{For } i = 2, \ldots, n_1 : \boldsymbol{t}_i^{(1)} = \delta \boldsymbol{b}_i^{(1)};\\
&\quad \rho^{(2)}, u_1^{(2)}, u_2^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q,\\
&\quad \boldsymbol{t}_{0,1}^{(2)} = (\overbrace{\delta, 0,}^{2} \overbrace{0^2,}^{2} \overbrace{0^2,}^{2} \overbrace{\rho^{(2)}}^{1})_{\mathbb{B}^{(2)}}, \ \ \boldsymbol{t}_{1,1}^{(2)} = (\overbrace{\delta, 0,}^{2} \overbrace{u_1^{(2)}, u_2^{(2)}}^{2} \overbrace{0^2,}^{2} \overbrace{\rho^{(2)}}^{1})_{\mathbb{B}^{(2)}},\\
&\quad \boldsymbol{t}_2^{(2)} = \delta \boldsymbol{b}_2^{(2)},\\
&\text{Output} \ \ (\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{t}_{\beta}^{(0)}, \boldsymbol{t}_{\beta,1}^{(1)}, \boldsymbol{t}_{\beta,1}^{(2)}, \{\boldsymbol{t}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{t}_2^{(2)}).
\end{aligned}
$$

Let $\mathcal{B}$ be a probabilistic machine, we define the advantage of $\mathcal{B}$ for Problem 1 as follows:

$$
\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P1}}(\lambda) = \Big| \ \mathsf{Pr}\Big[\mathcal{B}(1^{\lambda}, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathrm{P1}}(1^{\lambda}, \overrightarrow{n}, l)\Big] \ - \ \mathsf{Pr}\Big[\mathcal{B}(1^{\lambda}, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathrm{P1}}(1^{\lambda}, \overrightarrow{n}, l)\Big] \ \Big|.
$$

Lemma 1. *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{D}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P1}}(\lambda) \leq \mathsf{Adv}_{\mathcal{D}}^{\mathrm{DLIN}}(\lambda) + 8/q$.*

Definition 7 (Problem 2). *Problem 2 is to find bit $\beta \in \{0,1\}$, given* $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{h}_{\beta}^{*(0)}, \boldsymbol{t}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{h}_{\beta,i}^{*(k)}, \boldsymbol{t}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2}) \xleftarrow{R} \mathcal{G}_{\beta}^{\mathrm{P2}}(1^{\lambda}, \overrightarrow{n} = (2; n_1, n_2 = 2))$*, where*

$$
\begin{aligned}
&\mathcal{G}_{\beta}^{\mathrm{P2}}(1^{\lambda}, \overrightarrow{n} = (2; n_1, n_2 = 2)): \ (\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{R} \mathcal{G}_{\mathsf{ob}}(1^{\lambda}, \overrightarrow{n}),\\
&\quad \widehat{\mathbb{B}}^{(0)} = (\boldsymbol{b}_1^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_4^{(0)}, \boldsymbol{b}_5^{(0)}),\\
&\quad \widehat{\mathbb{B}}^{(1)} = (\boldsymbol{b}_1^{(1)}, \ldots, \boldsymbol{b}_{n_1}^{(1)}, \boldsymbol{b}_{2n_1+1}^{(1)}, \ldots, \boldsymbol{b}_{3n_1+1}^{(1)}),\\
&\quad \widehat{\mathbb{B}}^{(2)} = (\boldsymbol{b}_1^{(2)}, \boldsymbol{b}_2^{(2)}, \boldsymbol{b}_5^{(2)}, \boldsymbol{b}_6^{(2)}, \boldsymbol{b}_7^{(2)}),\\
&\quad \omega, \xi, \delta \xleftarrow{\mathsf{U}} \mathbb{F}_q, \ \ z, \pi \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}, \ \ u = z^{-1},\\
&\quad \boldsymbol{h}_0^{*(0)} = (\omega, 0, 0, \xi, 0)_{\mathbb{B}^{*(0)}}, \ \ \boldsymbol{h}_1^{*(0)} = (\omega, z, 0, \xi, 0)_{\mathbb{B}^{*(0)}}, \ \ \boldsymbol{t}^{(0)} = (\delta, \pi u, 0, 0, 0)_{\mathbb{B}^{(0)}},\\
&\quad \text{For } k = 1, 2 :\\
&\qquad \text{For } i = 1, \ldots, n_k \text{ and } j = 1, \ldots, n_k :\\
&\qquad\quad (u_{i,j}^{(k)}) \xleftarrow{\mathsf{U}} GL(\mathbb{F}_q, n_k), \ (z_{i,j}^{(k)}) = \big((u_{i,j}^{(k)})^{-1}\big)^T;\\
&\qquad \text{For } i = 1, \ldots, n_k :\\
&\qquad\quad \overrightarrow{\omega}_i^{(k)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_k},\\
&\qquad\quad \boldsymbol{h}_{0,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{\overrightarrow{\omega}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}},
\end{aligned}
$$

$$\boldsymbol{h}_{1,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{z_{i,1}^{(k)}, \ldots, z_{i,n_k}^{(k)}}^{n_k}, \overbrace{\overrightarrow{\omega}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}},$$
$$\boldsymbol{t}_i^{(k)} = (\overbrace{\delta \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{\pi u_{i,1}^{(k)}, \ldots, \pi u_{i,n_k}^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{(k)}},$$
$$\text{Output} \quad (\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{h}_\beta^{*(0)}, \boldsymbol{t}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{h}_{\beta,i}^{*(k)}, \boldsymbol{t}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2}).$$

Let $\mathcal{B}$ be a probabilistic machine, we define the advantage of $\mathcal{B}$ for Problem 2 as

$$\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P2}}(\lambda) = \Big| \ \mathsf{Pr}\Big[\mathcal{B}(1^\lambda, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{R} \mathcal{G}_0^{\mathrm{P2}}(1^\lambda, \overrightarrow{n})\Big] \ - \ \mathsf{Pr}\Big[\mathcal{B}(1^\lambda, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{R} \mathcal{G}_1^{\mathrm{P2}}(1^\lambda, \overrightarrow{n})\Big] \ \Big|.$$

Lemma 2. *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{D}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P2}}(\lambda) \leq \mathsf{Adv}_{\mathcal{D}}^{\mathrm{DLIN}}(\lambda) + 5/q$.*

Lemma 3. *For $p \in \mathbb{F}_q$, let $C_p = \{(\overrightarrow{x}, \overrightarrow{v}) | \overrightarrow{x} \cdot \overrightarrow{v} = p\} \subset V \times V^*$ where V is n-dimensional vector space $\mathbb{F}_q^n$, and V^* its dual. For all $(\overrightarrow{x}, \overrightarrow{v}) \in C_p$, for all $(\overrightarrow{r}, \overrightarrow{w}) \in C_p$, $\Pr[\overrightarrow{x}U = \overrightarrow{r} \wedge \overrightarrow{v}Z = \overrightarrow{w}] = \Pr[\overrightarrow{x}Z = \overrightarrow{r} \wedge \overrightarrow{v}U = \overrightarrow{w}] = 1/\sharp C_p$, where $Z \xleftarrow{\mathsf{U}} GL(n, \mathbb{F}_q)$, $U = (Z^{-1})^T$, and $\sharp C_p$ denotes the number of elements in C_p.*

We then consider the following games:

Game 0. Let Game 0 denote the real security game defined in **Definition 4**.

Game 1. Game 1 is almost identical to Game 0, except that the target ciphertext $C = (L, \boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_r\}_{r \in L}, \boldsymbol{c}_M)$ for challenge attribute vectors $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$, challenge plaintexts $(M^{(0)}, M^{(1)})$ and a revocation list L is

$$\boldsymbol{c}_0 = (\delta, w, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}}, \tag{1}$$
$$\boldsymbol{c}_1 = (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{w_1^{(1)}, \ldots, w_{n_1}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \tag{2}$$
$$\forall r \in L: \quad \boldsymbol{c}_r = (\overbrace{\delta_r(-r, 1)}^{2}, \overbrace{w_{1,r}^{(2)}, w_{2,r}^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_r}^{1})_{\mathbb{B}^{(2)}}, \tag{3}$$
$$\boldsymbol{c}_M = g_T^\zeta M^{(b)}, \tag{4}$$

where $\delta, w, \zeta, \varphi, \varphi^{(1)}, \varphi_r, w_{1,r}^{(2)}, w_{2,r}^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $b \xleftarrow{\mathsf{U}} \{0,1\}$, $\overrightarrow{y}^{(b)} = (y_1^{(b)}, \ldots, y_{n_1}^{(b)})$, and $(w_1^{(1)}, \ldots, w_{n_1}^{(1)}) \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_1} \setminus \{\overrightarrow{0}\}$.

Game 2-m' $(m = 0, \ldots, \nu - 1)$. Game 2-0 is Game 1. Game 2-m' is almost identical to Game 2-m, except the reply to the $(m+1)$-th GenKey query for $\overrightarrow{x} = (x_1, \ldots, x_{n_1})$, and challenge ciphertext have the following form

$$\left.\begin{aligned}
\boldsymbol{k}_0 &= (-s, \epsilon, 1, \eta, 0)_{\mathbb{B}^{*(0)}}, \\
\boldsymbol{k}_1 &= (\overbrace{s_1 \overrightarrow{e}_1^{(1)} + \beta^{(1)} \overrightarrow{x}}^{n_1}, \overbrace{(\gamma^{(1)} \overrightarrow{e}_1^{(1)} + \sigma^{(1)} \overrightarrow{x}) \cdot Z^{(1)}}^{n_1}, \overbrace{\eta_1^{(1)}, \ldots, \eta_{n_1}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}}, \\
\boldsymbol{k}_2 &= (\overbrace{s_2(1, I)}^{2}, \overbrace{\gamma^{(2)}(1, I) \cdot Z^{(2)}}^{2}, \overbrace{\eta_1^{(2)}, \eta_2^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}}.
\end{aligned}\right\} \tag{5}$$

$$\left.\begin{aligned} \boldsymbol{c}_0 =& (\delta, w, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}}, \\ \boldsymbol{c}_1 =& (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{\overrightarrow{y}^{(b)} \cdot U^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\ \forall r \in L: \quad \boldsymbol{c}_r =& (\overbrace{\delta_r(-r,1)}^{2}, \overbrace{\delta'_r(-r,1) \cdot U^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_r}^{1})_{\mathbb{B}^{(2)}}, \\ \boldsymbol{c}_M =& g_T^{\zeta} M^{(b)}, \end{aligned}\right\} \tag{6}$$

where $\epsilon, \gamma^{(1)}, \gamma^{(2)}, \sigma^{(1)} \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $I \xleftarrow{\mathsf{U}} \Gamma$, $Z^{(k)} \xleftarrow{\mathsf{U}} GL(\mathbb{F}_q, n_k)$, $U^{(k)} = \left(Z^{(k)^{-1}}\right)^T, k = 1, 2$; $\delta'_r \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $r \in L$ such that $\sum_{r \in L} \delta'_r = 1$ and all the other variables are generated as in Game 2-m.

Game 2-$(m+1)$ $(m = 0, \ldots, \nu - 1)$. Game 2-$(m+1)$ is almost identical to Game 2-m', except the reply to the $(m+1)$-th GenKey query for $\overrightarrow{x} = (x_1, \ldots, x_{n_1})$ is:

$$\left.\begin{aligned} \boldsymbol{k}_0 =& (-s, \epsilon, 1, \eta, 0)_{\mathbb{B}^{*(0)}}, \\ \boldsymbol{k}_1 =& (\overbrace{s_1 \overrightarrow{e}_1^{(1)} + \beta^{(1)} \overrightarrow{x}}^{n_1}, \overbrace{v_1^{(1)}, \ldots, v_{n_1}^{(1)}}^{n_1}, \overbrace{\eta_1^{(1)}, \ldots, \eta_{n_1}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}}, \\ \boldsymbol{k}_2 =& (\overbrace{s_2(1, I)}^{2}, \overbrace{v_1^{(2)}, v_2^{(2)}}^{2}, \overbrace{\eta_1^{(2)}, \eta_2^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}}. \end{aligned}\right\} \tag{7}$$

the challenge ciphertext is the same as Eqs.(1)-(4), where $(v_1^{(k)}, \ldots, v_{n_k}^{(k)}) \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_k} \setminus \{\overrightarrow{0}\}$, $k = 1, 2$, and all the other variables are generated as in Game 2-m'.

Game 3. Game 3 is almost identical to Game 2-ν, except that the target ciphertext C for challenge attribute vectors $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$, challenge plaintexts $(M^{(0)}, M^{(1)})$, and a revocation list L is:

$$\left.\begin{aligned} \boldsymbol{c}_0 =& (\delta, w, \zeta', 0, \varphi)_{\mathbb{B}^{(0)}}, \\ \boldsymbol{c}_1 =& (\overbrace{\delta \overrightarrow{y}'}^{n_1}, \overbrace{w_1^{(1)}, \ldots, w_{n_1}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\ \forall r \in L: \quad \boldsymbol{c}_r =& (\overbrace{\delta_r(-r,1)}^{2}, \overbrace{w_{1,r}^{(2)}, w_{2,r}^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_r}^{1})_{\mathbb{B}^{(2)}}, \\ \boldsymbol{c}_M =& g_T^{\zeta} M^{(b)}. \end{aligned}\right\} \tag{8}$$

where $\zeta' \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\overrightarrow{y}' = (y'_1, \ldots, y'_{n_1}) \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_1}$. We note that ζ' and $(y'_1, \ldots, y'_{n_1})$ are chosen uniformly and independently from ζ, $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$ respectively.

Let $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$ be $\mathsf{Adv}_{\mathcal{A},\mathsf{AH\text{-}RPE}}^{\mathsf{AH}}(\lambda)$ in Game 0, and $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 1, 2-m, 2-m' and 3 respectively. We first prove Lemmas 4 - 8 that evaluate consecutive probability gaps between $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}(m+1))}(\lambda)$ for $m = 0, \ldots, \nu-1$, and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$, respectively.

Based on Lemmas 4 - 8 and Lemmas 1 and 2, we then obtain:

$$\begin{aligned} \mathsf{Adv}_{\mathcal{A},\mathsf{AH\text{-}RPE}}^{\mathsf{AH}}(\lambda) &= \mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) \\ &\leq |\, \mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda) \,| + \sum_{m=0}^{\nu-1} |\, \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda) \,| + \\ &\quad \sum_{m=0}^{\nu-1} |\, \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}(m+1))}(\lambda) \,| + |\, \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) \,| + \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) \end{aligned}$$

$$\leq \mathsf{Adv}^{\mathrm{P1}}_{\mathcal{B}_1}(\lambda) + \sum_{m=0}^{\nu-1} \mathsf{Adv}^{\mathrm{P2}}_{\mathcal{B}'_{2m}}(\lambda) + \sum_{m=0}^{\nu-1} \mathsf{Adv}^{\mathrm{P2}}_{\mathcal{B}_{2(m+1)}}(\lambda) + (2\nu|L| + 8\nu + 2)/q$$
$$\leq (2\nu+1)\mathsf{Adv}^{\mathrm{DLIN}}_{\mathcal{D}}(\lambda) + (2\nu|L| + 18\nu + 10)/q.$$

This completes the proof of Theorem 1. □

Lemma 4. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_1$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\ \mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)\ | = \mathsf{Adv}^{\mathrm{P1}}_{\mathcal{B}_1}(\lambda)$.*

Proof. Suppose a polynomial time adversary $\mathcal{A}$ can successfully distinguish between Game 0 and Game 1. We construct a simulator $\mathcal{B}_1$ that uses $\mathcal{A}$ as a black box to solve Problem 1. The reduction proceeds as follows:

1. $\mathcal{B}_1$ is given an instance of Problem 1, i.e. $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{t}^{(0)}_{\beta}, \boldsymbol{t}^{(1)}_{\beta,1}, \boldsymbol{t}^{(2)}_{\beta,1}, \{\boldsymbol{t}^{(1)}_i\}_{i=2,\ldots,n_1}, \boldsymbol{t}^{(2)}_2)$, and plays the role of the challenger in the security game against adversary $\mathcal{A}$.
2. At the beginning of the game, $\mathcal{B}_1$ gives $\mathcal{A}$ the public key $PK = \big(1^\lambda, \mathsf{param}_{\overrightarrow{n}}, (\boldsymbol{b}^{(0)}_1, \boldsymbol{b}^{(0)}_3, \boldsymbol{b}^{(0)}_5, \boldsymbol{b}^{(1)}_1, \ldots, \boldsymbol{b}^{(1)}_{n_1}, \boldsymbol{b}^{(1)}_{3n_1+1}, \boldsymbol{b}^{(2)}_1, \boldsymbol{b}^{(2)}_2, \boldsymbol{b}^{(2)}_7)$, which is obtained from the Problem 1 instance.
3. When a GenKey query is issued, $\mathcal{B}_1$ computes a normal secret key using $(\widehat{\mathbb{B}}^{*(0)}, \widehat{\mathbb{B}}^{*(1)}, \widehat{\mathbb{B}}^{*(2)})$, which is obtained from the Problem 1 instance.
4. When $\mathcal{B}_1$ receives challenge attribute vectors $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$, challenge plaintexts $(M^{(0)}, M^{(1)})$ and a revocation list L from $\mathcal{A}$, $\mathcal{B}_1$ computes and returns $C = (L, \boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_r\}_{r\in L}, \boldsymbol{c}_M)$ s.t. $\boldsymbol{c}_0 = \boldsymbol{t}^{(0)}_{\beta} + \zeta \boldsymbol{b}^{(0)}_3$, $\boldsymbol{c}_1 = y^{(b)}_1 \boldsymbol{t}^{(1)}_{\beta,1} + \sum_{i=2}^{n_1} y^{(b)}_i \boldsymbol{t}^{(1)}_i$, $\forall r \in L : \boldsymbol{c}_r = (p_r/p)(-r)\boldsymbol{t}^{(2)}_{\beta,1} + (p_r/p)\boldsymbol{t}^{(2)}_2$ and $\boldsymbol{c}_M = g_T^{\zeta} M^{(b)}$, using $(\boldsymbol{t}^{(0)}_{\beta}, \boldsymbol{t}^{(1)}_{\beta,1}, \boldsymbol{t}^{(2)}_{\beta,1}, \{\boldsymbol{t}^{(1)}_i\}_{i=2,\ldots,n_1}, \boldsymbol{t}^{(2)}_2, \boldsymbol{b}^{(0)}_3)$ from the instance of Problem 1, $\overrightarrow{y}^{(b)}$, $M^{(b)}$ and L, where $p, p_r \xleftarrow{\cup} \mathbb{F}_q$ for all $r \in L$ such that $p = \sum_{r\in L} p_r$, $\zeta \xleftarrow{\cup} \mathbb{F}_q$, $b \xleftarrow{\cup} \{0,1\}$.
5. After the challenge phase, GenKey oracle simulation for a key query is executed in the same manner as step 3.
6. $\mathcal{A}$ outputs a bit b'. If $b = b'$, $\mathcal{B}_1$ outputs 1. Otherwise, $\mathcal{B}_1$ outputs 0.

Claim. For $\beta = 0$ the challenge ciphertext $C = (L, \boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_r\}_{r\in L}, \boldsymbol{c}_M)$ generated in step 4 is distributed exactly as in Game 0, whereas if $\beta = 1$, the challenge ciphertext $C = (L, \boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_r\}_{r\in L}, \boldsymbol{c}_M)$ generated in step 4 is distributed exactly as in Game 1.

Proof. First recall that $y^{(b)}_1 = 1$. If $\beta = 0$, $\boldsymbol{c}_0 = (\delta, 0, \zeta, 0, \rho)_{\mathbb{B}^{(0)}}$, $\boldsymbol{c}_1 = (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\rho^{(1)}}^{1})_{\mathbb{B}^{(1)}}$, $\forall r \in L : \boldsymbol{c}_r = (\overbrace{\delta_r(-r,1)}^{2}, \overbrace{0^2}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_r}^{1})_{\mathbb{B}^{(2)}}$, and $\boldsymbol{c}_M = g_T^{\zeta} M^{(b)}$ where $\delta_r = (p_r\delta)/p$ and $\varphi_r = ((-r)p_r\varphi^{(2)})/p$. It is the challenge ciphertext in Game 0. If $\beta = 1$, $\boldsymbol{c}_0 = (\delta, u, \zeta, 0, \rho)_{\mathbb{B}^{(0)}}$, $\boldsymbol{c}_1 = (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{\overrightarrow{u}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\rho^{(1)}}^{1})_{\mathbb{B}^{(1)}}$, $\forall r \in L : \boldsymbol{c}_r = (\overbrace{\delta_r(-r,1)}^{2}, \overbrace{u^{(2)}_{1,r}, u^{(2)}_{2,r},}^{2} \overbrace{0^2}^{2}, \overbrace{\varphi_r}^{1})_{\mathbb{B}^{(2)}}$, where $\overrightarrow{u}^{(1)} = (u^{(1)}_1, \ldots, u^{(1)}_{n_1})$, $\delta_r = (p_r\delta)/p$, $\varphi_r = ((-r)p_r\varphi^{(2)})/p$, $u^{(2)}_{i,r} = \frac{(-r)p_r\varphi^{(2)}}{p} u^{(2)}_i$ for $i = 1, 2$. Since $\overrightarrow{u}^{(1)} \in \mathbb{F}_q^{n_1}$, $u^{(2)}_{1,r}, u^{(2)}_{2,r}, \delta_r, \delta, \rho_r, \rho^{(1)} \in \mathbb{F}_q$ are independently uniform, it is the challenge ciphertext in Game 1.

From the above claim, if $\beta = 0$, the distribution of simulated values in the above simulation is exactly as in Game 0, whereas if $\beta = 1$, this simulation results in a an identical distribution to Game 1. Therefore, $|\ \mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)\ | = \Big|\ \mathsf{Pr}\Big[\mathcal{B}_1(1^\lambda, \varpi) \to 1 \ \Big|\ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathrm{P1}}_0(1^\lambda, \overrightarrow{n})\Big] - \mathsf{Pr}\Big[\mathcal{B}_1(1^\lambda, \varpi) \to 1 \ \Big|\ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathrm{P1}}_1(1^\lambda, \overrightarrow{n})\Big]\ \Big| = \mathsf{Adv}^{\mathrm{P1}}_{\mathcal{B}_1}(\lambda)$, which completes our proof of **Lemma 4**. □

Lemma 5. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}'_{2m}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\ \mathsf{Adv}^{(2\text{-}m)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2\text{-}m')}_{\mathcal{A}}(\lambda)\ | \leq \mathsf{Adv}^{\mathrm{P2}}_{\mathcal{B}'_{2m}}(\lambda) + (|L|+4)/q$.*

Proof. Suppose a polynomial time adversary $\mathcal{A}$ can successfully distinguish between Game 2-m and Game 2-m'. We construct a simulator $\mathcal{B}'_{2m}$ that uses $\mathcal{A}$ as a black box to solve Problem 2. The reduction proceeds as follows:

1. $\mathcal{B}'_{2m}$ is given an instance of Problem 2, that is a tuple $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{h}^{*(0)}_{\beta}, \boldsymbol{t}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{h}^{*(k)}_{\beta,i}, \boldsymbol{t}^{(k)}_{i}\}_{i=1,\ldots,n_k}\}_{k=1,2})$. $\mathcal{B}'_{2m}$ plays the role of the challenger in the security game against adversary $\mathcal{A}$.
2. At the beginning of the game, $\mathcal{B}'_{2m}$ gives $\mathcal{A}$ the public key $PK = \big(1^{\lambda}, \mathsf{param}_{\overrightarrow{n}}, (\boldsymbol{b}^{(0)}_1, \boldsymbol{b}^{(0)}_3, \boldsymbol{b}^{(0)}_5, \boldsymbol{b}^{(1)}_1, \ldots, \boldsymbol{b}^{(1)}_{n_1}, \boldsymbol{b}^{(1)}_{3n_1+1}, \boldsymbol{b}^{(2)}_1, \boldsymbol{b}^{(2)}_2, \boldsymbol{b}^{(2)}_7)$, which is obtained from the Problem 2 instance.
3. When the s-th GenKey query is issued for a predicate $\overrightarrow{x} = (x_1, \ldots, x_{n_1})$, $\mathcal{B}'_{2m}$ answers as follows:
   a) For $1 \leq s \leq m$, $\mathcal{B}'_{2m}$ computes a semi-functional key using $\{\mathbb{B}^{*(k)}\}_{k=0,1,2}$ of the Problem 2 instance.
   b) For $s = m+1$, $\mathcal{B}'_{2m}$ computes $\boldsymbol{k}^{*}_{\overrightarrow{x},I}$, where $\boldsymbol{k}^{*}_{\overrightarrow{x},I} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \boldsymbol{k}_2)$ using $\{\boldsymbol{h}^{*(0)}_{\beta}, \boldsymbol{b}^{*(0)}_1, \boldsymbol{b}^{*(0)}_3, \{\boldsymbol{h}^{*(i)}_{\beta,j}, \boldsymbol{b}^{*(i)}_{j}\}_{i=1,2;j=1,\ldots,n_i}\}$ from the Problem 2 instance as follows:

$$
\begin{aligned}
&\text{For } i = 1,2 : \varrho_i, v_i, v'_i, \theta_i, \xleftarrow{\cup} \mathbb{F}_q;\\
&\boldsymbol{s}^{(0)}_{\beta} = \sum_{i=1}^{2} (\varrho_i \boldsymbol{h}^{*(0)}_{\beta} + v_i \boldsymbol{b}^{*(0)}_1), \quad \boldsymbol{k}_0 = -\boldsymbol{s}^{(0)}_{\beta} + \boldsymbol{b}^{*(0)}_3,\\
&\text{For } i = 1,2 \text{ and } j = 1, \ldots, n_i :\\
&\qquad \boldsymbol{s}^{(i)}_{\beta,j} = \theta_i \boldsymbol{h}^{*(i)}_{\beta,j} + v'_i \boldsymbol{b}^{*(i)}_{j}, \quad \widehat{\boldsymbol{s}}^{(i)}_{\beta,j} = \varrho_i \boldsymbol{h}^{*(i)}_{\beta,j} + v_i \boldsymbol{b}^{*(i)}_{j},\\
&\boldsymbol{k}_1 = \sum_{j=1}^{n_1} x_j \boldsymbol{s}^{(1)}_{\beta,j} + \widehat{\boldsymbol{s}}^{(1)}_{\beta,1},\\
&\boldsymbol{k}_2 = \widehat{\boldsymbol{s}}^{(2)}_{\beta,1} + I \cdot \widehat{\boldsymbol{s}}^{(2)}_{\beta,2}.
\end{aligned}
$$

   c) For $s \geq m+2$, $\mathcal{B}'_{2m}$ computes a normal key using $\{\mathbb{B}^{*(k)}\}_{k=0,1,2}$ of the Problem 2 instance.
4. When $\mathcal{B}'_{2m}$ receives challenge attribute vectors $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$, challenge plaintexts $(M^{(0)}, M^{(1)})$ and a revocation list L from $\mathcal{A}$, $\mathcal{B}'_{2m}$ computes and returns $C = (L, \boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_r\}_{r\in L}, \boldsymbol{c}_M)$ s.t. $\boldsymbol{c}_0 = \boldsymbol{t}^{(0)} + \zeta \boldsymbol{b}^{(0)}_3 + \varphi \boldsymbol{b}^{(0)}_5$, $\boldsymbol{c}_1 = \sum_{j=1}^{n_1} y^{(b)}_j \boldsymbol{t}^{(1)}_j + \varphi^{(1)} \boldsymbol{b}^{(1)}_{3n_1+1}$, $\forall r \in L :\ \boldsymbol{c}_r = \frac{(-r)p_r}{p} \boldsymbol{t}^{(2)}_1 + \frac{p_r}{p} \boldsymbol{t}^{(2)}_2 + \varphi_r \boldsymbol{b}^{(2)}_7$, and $\boldsymbol{c}_M = g_T^{\zeta} M^{(b)}$, using $(\boldsymbol{t}^{(0)}, \{\boldsymbol{t}^{(1)}_i\}_{i=1,\ldots,n_1}, \boldsymbol{t}^{(2)}_1, \boldsymbol{t}^{(2)}_2, \boldsymbol{b}^{(0)}_3, \boldsymbol{b}^{(0)}_5, \boldsymbol{b}^{(1)}_{3n_1+1}, \boldsymbol{b}^{(2)}_7)$ in the Problem 2 instance, $\overrightarrow{y}^{(b)}$, $M^{(b)}$ and L, where $p, p_r \xleftarrow{\cup} \mathbb{F}_q$ for all $r \in L$ such that $p = \sum_{r\in L} p_r$, $\zeta, \varphi, \varphi^{(1)}, \varphi_r \xleftarrow{\cup} \mathbb{F}_q$, $b \xleftarrow{\cup} \{0,1\}$.
5. After the challenge phase, GenKey oracle simulation for a key query is executed in the same manner as step 3.
6. $\mathcal{A}$ outputs a bit b'. If $b = b'$, $\mathcal{B}'_{2m}$ outputs 1. Otherwise, $\mathcal{B}'_{2m}$ outputs 0.

Claim. The distribution of the view of adversary $\mathcal{A}$ in the above-mentioned game simulated by $\mathcal{B}'_{2m}$ given a Problem 2 instance is the same as that in Game (2-m) (resp. Game (2-m')) if $\beta = 0$ (resp. $\beta = 1$) except with probability $(3 + |L|)/q$ (resp. $1/q$).

Proof. It is clear that $\mathcal{B}'_{2m}$'s simulation of the public key generation (step 2) and the answers to the i-th GenKey query where $i \neq m+1$ (case (a) and (c) of steps (3) and (5)) are exactly the same as the Setup and the GenKey oracles in Game 2-m and Game 2-m'.

Next we analyze the distribution of the i-th GenKey query where $i = m+1$ (case (b) of steps (3) and (5)). Values $\boldsymbol{s}_{\beta}^{(0)}, \boldsymbol{s}_{\beta,j}^{(i)}, \widehat{\boldsymbol{s}}_{\beta,j}^{(i)}$ for $i = 1,2$ and $j = 1,\ldots,n_i$ in this case can be expressed as follows. Let $\beta^{(i)} = \theta_i\omega + v_i'$, $\alpha^{(i)} = \varrho_i\omega + v_i$, $\alpha = \alpha^{(1)} + \alpha^{(2)}$, $\gamma = \varrho_1 + \varrho_2$, and $\epsilon = \gamma z$. Then,

$$\boldsymbol{s}_0^{(0)} = (\alpha, 0, 0, \gamma\xi, 0)_{\mathbb{B}^{*(0)}}, \qquad \boldsymbol{s}_1^{(0)} = (\alpha, \epsilon, 0, \gamma\xi, 0)_{\mathbb{B}^{*(0)}},$$

$$\boldsymbol{s}_{0,j}^{(i)} = (\overbrace{\beta^{(i)}\overrightarrow{e}_j^{(i)}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\theta_i\overrightarrow{\omega}_j^{(i)}}^{n_i}, \overbrace{0}^{1})_{\mathbb{B}^{*(i)}}, \quad \boldsymbol{s}_{1,j}^{(i)} = (\overbrace{\beta^{(i)}\overrightarrow{e}_j^{(i)}}^{n_i}, \overbrace{\theta_i\overrightarrow{z}_j^{(i)}}^{n_i}, \overbrace{\theta_i\overrightarrow{\omega}_j^{(i)}}^{n_i}, \overbrace{0}^{1})_{\mathbb{B}^{*(i)}},$$

$$\widehat{\boldsymbol{s}}_{0,j}^{(i)} = (\overbrace{\alpha^{(i)}\overrightarrow{e}_j^{(i)}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\varrho_i\overrightarrow{\omega}_j^{(i)}}^{n_i}, \overbrace{0}^{1})_{\mathbb{B}^{*(i)}}, \quad \widehat{\boldsymbol{s}}_{1,j}^{(i)} = (\overbrace{\alpha^{(i)}\overrightarrow{e}_j^{(i)}}^{n_i}, \overbrace{\varrho_i\overrightarrow{z}_j^{(i)}}^{n_i}, \overbrace{\varrho_i\overrightarrow{\omega}_j^{(i)}}^{n_i}, \overbrace{0}^{1})_{\mathbb{B}^{*(i)}},$$

where $\overrightarrow{z}_j^{(i)} = z_{j,1}^{(i)}, \ldots, z_{j,n_i}^{(i)}, \omega, z, \xi, \{\overrightarrow{\omega}_j^{(i)}, \overrightarrow{z}_j^{(i)}\}_{i=1,2;j=1,\ldots,n_i}$ are defined in Problem 2. When $\beta = 1$ in Problem 2 instance, $\boldsymbol{k}_{\overrightarrow{x},I}^{*} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \boldsymbol{k}_2)$ has the same distribution as Eq. 5, except that $\epsilon w = \gamma$, where γ and $w = u \xleftarrow{\cup} \mathbb{F}_q$ of $\boldsymbol{c}_0$ in Eq. 6.

Next, we show that the joint distribution of the answer to $(m+1)$-th GenKey query and the challenge ciphertext produced by $\mathcal{B}'_{2m}$'s simulation on input an instance of Problem 2 is equivalent to the one in Game 2-m (resp. Game 2-m') if $\beta = 0$ (resp. $\beta = 1$).

If $\beta = 0$ then it is straightforward to show that distributions are equivalent unless one of following conditions holds: (1) ω defined in Problem 2 is zero, (2) $w = 0$, (3) $(w_1^{(1)}, \ldots, w_{n_1}^{(1)}) = \overrightarrow{0}$, (4) $(w_{1,r}^{(2)}, w_{2,r}^{(2)}) = \overrightarrow{0}$, where $r \in L$, and w, $(w_1^{(1)}, \ldots, w_{n_1}^{(1)})$ and $(w_{1,r}^{(2)}, w_{2,r}^{(2)})$ are defined in Eqs. 1, 2 and 3 respectively. Those events occur with probability $(3 + |L|)/q$.

If $\beta = 1$ then the key simulated by $\mathcal{B}'_{2m}$ is identical to that in Eq.5 and $\mathcal{B}'_{2m}$'s simulation for the challenge ciphertext is the same as in Eq.6, except that $\epsilon w = \gamma$, where $\gamma = \varrho_1 + \varrho_2$ and $w \xleftarrow{\cup} \mathbb{F}_q$ of $\boldsymbol{c}_0$ in Eq. 6.

Therefore, we will show that γ is uniformly and independently distributed from the other variables in $\mathcal{B}'_{2m}$'s simulation. Since γ is related to $\overrightarrow{A}_1, \overrightarrow{A}_2, \overrightarrow{B}_1$, and $\overrightarrow{B}_{2,r}$, where $\overrightarrow{A}_1 = (\varrho_1\overrightarrow{e}_1^{(1)} + \theta_1\overrightarrow{x}) \cdot Z^{(1)}$, $\overrightarrow{A}_2 = (\varrho_2(1, I)) \cdot Z^{(2)}$, and $\overrightarrow{B}_1 = \overrightarrow{y}^{(b)} \cdot U^{(1)}$, $\overrightarrow{B}_{2,r} = \delta_r'(-r, 1) \cdot U^{(2)}$ for all $r \in L$, where $b \in \{0,1\}$. We will consider this joint distribution for the two cases from **Definition 4**.

1. If $I \in L$ then by Lemma 3, the pair $(\overrightarrow{A}_2, \overrightarrow{B}_{2,r})$ is uniformly and independently distributed over $C_0 = \{(\overrightarrow{w}, \overrightarrow{r}) | \overrightarrow{w} \cdot \overrightarrow{r} = 0\}$ (over $Z^{(2)} \xleftarrow{\cup} GL(\mathbb{F}_q, 2)$) for $r = I$. When $r \neq I$, the pair $(\overrightarrow{A}_2, \overrightarrow{B}_{2,r})$ is uniformly and independently distributed over $C_{\varrho_2\delta_r'(I-r)} = \{(\overrightarrow{w}, \overrightarrow{r}) | \overrightarrow{w} \cdot \overrightarrow{r} = \varrho_2\delta_r'(I-r)\}$ (over $Z^{(2)} \xleftarrow{\cup} GL(\mathbb{F}_q, 2)$). The pair $(\overrightarrow{A}_2, \overrightarrow{B}_{2,r})$ (for $r \in L$) is uniformly and independently distributed over $\mathbb{F}_q^4$.
2. If $I \notin L$ and $f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 0$ then by Lemma 3, the pair $(\overrightarrow{A}_1, \overrightarrow{B}_1)$ is uniformly and independently distributed on $C_{\theta_1 \cdot (\overrightarrow{x} \cdot \overrightarrow{y}^{(b)}) + \varrho_1}$ (over $Z^{(1)} \xleftarrow{\cup} GL(\mathbb{F}_q, n_1)$). Since $\theta_1 \xleftarrow{\cup} \mathbb{F}_q$, the pair $(\overrightarrow{A}_1, \overrightarrow{B}_1)$ is uniformly and independently distributed over $\mathbb{F}_q^{2n_1}$. The pair $(\overrightarrow{A}_2, \overrightarrow{B}_{2,r})$ (for $r \in L$) is uniformly and independently distributed over $C_{\varrho_2\delta_r(I-r)}$.

Due to the restriction of adversary $\mathcal{A}$'s key queries defined in **Definition 4**, in each of the two cases above, at least one of $(\overrightarrow{A}_1, \overrightarrow{B}_1)$ and $(\overrightarrow{A}_2, \overrightarrow{B}_{2,r})$ is uniformly and independently distributed over $\mathbb{F}_q^{2n_k}$ for $k = 1, 2$. Case 1 is obviously independent from γ. Therefore, the distribution of $\gamma = \varrho_1 + \varrho_2$ is independent from the distribution of ϱ_2, which can be given by $(\overrightarrow{A}_2, \overrightarrow{B}_{2,r})$. Thus, γ is uniformly distributed and is independent from other variables used in the simulation performed by $\mathcal{B}'_{2m}$.

Therefore, the view of the adversary $\mathcal{A}$ in the game simulated by $\mathcal{B}'_{2m}$ for an instance of Problem 2 with $\beta = 1$ is the same as that in Game 2-m' unless $\omega = 0$ occurs. This event occurs with probability $1/q$.

This completes the proof of **Lemma 5**. □

Lemma 6. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_{2(m+1)}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\ \mathsf{Adv}^{(2\text{-}m')}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2\text{-}(m+1))}_{\mathcal{A}}(\lambda)\ | \leq \mathsf{Adv}^{\mathrm{P2}}_{\mathcal{B}_{2(m+1)}}(\lambda) + (|L|+4)/q$.*

Proof. Suppose a polynomial time adversary $\mathcal{A}$ can successfully distinguish between Game 2-m' and Game 2-$(m+1)$. We construct a simulator $\mathcal{B}_{2(m+1)}$ that leverages $\mathcal{A}$ as a black box to solve Problem 2. The procedure is same as that in the proof of **Lemma 5** except that in case (b) of step 3, $\boldsymbol{k}^{*}_{\overrightarrow{x},I} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \boldsymbol{k}_2)$ is computed as follows:

$$\boldsymbol{k}_0 = -\boldsymbol{s}^{(0)}_{\beta} + \epsilon' \boldsymbol{b}^{*(0)}_2 + \boldsymbol{b}^{*(0)}_3,$$
$$\boldsymbol{k}_1 = \sum_{j=1}^{n_1} x_j \boldsymbol{s}^{(1)}_{\beta,j} + \widehat{\boldsymbol{s}}^{(1)}_{\beta,1} + \sum_{j=1}^{n_1} v^{(1)}_j \boldsymbol{b}^{*(1)}_{n_1+j},$$
$$\boldsymbol{k}_2 = \widehat{\boldsymbol{s}}^{(2)}_{\beta,1} + I \cdot \widehat{\boldsymbol{s}}^{(2)}_{\beta,2} + \sum_{j=1}^{2} v^{(2)}_j \boldsymbol{b}^{*(2)}_{2+j},$$

where $\epsilon' \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $v^{(2)}_1, v^{(2)}_2 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $(v^{(1)}_1, \ldots, v^{(1)}_{n_1}) \xleftarrow{\mathsf{U}} \mathbb{F}^{n_1}_q \setminus \{\overrightarrow{0}\}$. In the last step, if $b = b'$, $\mathcal{B}_{2(m+1)}$ outputs 0. Otherwise, $\mathcal{B}_{2(m+1)}$ outputs 1.

The view of $\mathcal{A}$ in the game simulated by $\mathcal{B}_{2(m+1)}$ given a Problem 2 instance with $\beta = 0$ is the same as that in Game 2-$(m+1)$, unless one of the following events occur: (1) $\omega = 0$ in Problem 2 instance, or (2) $w = 0$, or (3) $(w^{(1)}_1, \ldots, w^{(1)}_{n_1}) = \overrightarrow{0}$, or (4) $(w^{(2)}_{1,r}, w^{(2)}_{2,r}) = \overrightarrow{0}$, where $r \in L$, and w, $(w^{(1)}_1, \ldots, w^{(1)}_{n_1})$, and $(w^{(2)}_{1,r}, w^{(2)}_{2,r})$ are defined in Eqs. 1, 2 and 3 respectively. Those events occur with probability $(3 + |L|)/q$. If $\beta = 1$ then by a similar argument as in the proof of **Lemma 5**, namely that each variable is uniformly and independently distributed from the other variables in $\mathcal{B}_{2(m+1)}$'s simulation, the view of $\mathcal{A}$ is the same to the one in Game 2-m' unless $\omega = 0$ in the instance of Problem 2. This event occurs with probability $1/q$. □

Lemma 7. *For any adversary $\mathcal{A}$, $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{(2\text{-}\nu)}_{\mathcal{A}}(\lambda) + 2/q$.*

Proof. First, we show the distribution $(\mathsf{param}_{\overrightarrow{n}}, \{\widehat{\mathbb{B}}^{(k)}\}_{k=0,1,2}, \{\boldsymbol{k}^{*(j)}_{\overrightarrow{x},I}\}_{j=1,\ldots,\nu}, C)$ of Game 3 is same as that of Game 2-ν, where $\boldsymbol{k}^{*(j)}_{\overrightarrow{x},I}$ is the answer to the j-th key query, and C is the challenge ciphertext. We will define new bases $\mathbb{D}^{(k)}$ of $\mathbb{V}_k$ and $\mathbb{D}^{*(k)}$ of $\mathbb{V}_k$, $k = 0, 1$.

For $k = 0$, we set $\boldsymbol{d}^{(0)}_2 = \boldsymbol{b}^{(0)}_2 - \lambda \boldsymbol{b}^{(0)}_3$ and $\boldsymbol{d}^{*(0)}_3 = \boldsymbol{b}^{*(0)}_3 + \lambda \boldsymbol{b}^{*(0)}_2$, where $\lambda \xleftarrow{\mathsf{U}} \mathbb{F}_q$. The new bases are $\mathbb{D}^{(0)} = (\boldsymbol{b}^{(0)}_1, \boldsymbol{d}^{(0)}_2, \boldsymbol{b}^{(0)}_3, \boldsymbol{b}^{(0)}_4, \boldsymbol{b}^{(0)}_5)$ and $\mathbb{D}^{*(0)} = (\boldsymbol{b}^{*(0)}_1, \boldsymbol{b}^{*(0)}_2, \boldsymbol{d}^{*(0)}_3, \boldsymbol{b}^{*(0)}_4, \boldsymbol{b}^{*(0)}_5)$. We can easily verify that $\mathbb{D}^{(0)}$ and $\mathbb{D}^{*(0)}$ are dual orthonormal, and are distributed the same as the original bases $\mathbb{B}^{(0)}$ and $\mathbb{B}^{*(0)}$ respectively.

For $i = 1, \ldots, n_1$ and $j = 1, \ldots, n_1$, choose $Q^{(1)} = (\mu^{(1)}_{i,j}) \xleftarrow{\mathsf{U}} \mathbb{F}^{n_1 \times n_1}_q$, and compute $\boldsymbol{d}^{(1)}_{n_1+i} = \boldsymbol{b}^{(1)}_{n_1+i} + \sum_{j=1}^{n_1} \mu^{(1)}_{i,j} \boldsymbol{b}^{(1)}_j$, $\boldsymbol{d}^{*(1)}_i = \boldsymbol{b}^{*(1)}_i - \sum_{j=1}^{n_1} \mu^{(1)}_{j,i} \boldsymbol{b}^{*(1)}_{n_1+j}$, which are equivalent to the following matrix computations:

$$\begin{pmatrix} \overrightarrow{B}^{(1)}_1 \\ \overrightarrow{D}^{(1)}_2 \end{pmatrix} = \begin{pmatrix} I_{n_1} & 0_{n_1} \\ Q^{(1)} & I_{n_1} \end{pmatrix} \begin{pmatrix} \overrightarrow{B}^{(1)}_1 \\ \overrightarrow{B}^{(1)}_2 \end{pmatrix}, \qquad \begin{pmatrix} \overrightarrow{D}^{*(1)}_1 \\ \overrightarrow{B}^{*(1)}_2 \end{pmatrix} = \begin{pmatrix} I_{n_1} & -Q^{\mathrm{T}(1)} \\ 0_{n_1} & I_{n_1} \end{pmatrix} \begin{pmatrix} \overrightarrow{B}^{*(1)}_1 \\ \overrightarrow{B}^{*(1)}_2 \end{pmatrix}.$$

where $\overrightarrow{B}^{(1)}_1 = (\boldsymbol{b}^{(1)}_1, \ldots, \boldsymbol{b}^{(1)}_{n_1})^{\mathrm{T}}$, $\overrightarrow{B}^{(1)}_2 = (\boldsymbol{b}^{(1)}_{n_1+1}, \ldots, \boldsymbol{b}^{(1)}_{2n_1})^{\mathrm{T}}$, $\overrightarrow{B}^{*(1)}_1 = (\boldsymbol{b}^{*(1)}_1, \ldots, \boldsymbol{b}^{*(1)}_{n_1})^{\mathrm{T}}$, $\overrightarrow{B}^{*(1)}_2 = (\boldsymbol{b}^{*(1)}_{n_1+1}, \ldots, \boldsymbol{b}^{*(1)}_{2n_1})^{\mathrm{T}}$, $\overrightarrow{D}^{(1)}_2 = (\boldsymbol{d}^{(1)}_{n_1+1}, \ldots, \boldsymbol{d}^{(1)}_{2n_1})^{\mathrm{T}}$, $\overrightarrow{D}^{*(1)}_1 = (\boldsymbol{d}^{*(1)}_1, \ldots, \boldsymbol{d}^{*(1)}_{n_1})^{\mathrm{T}}$.

The new bases are $\mathbb{D}^{(1)} = (\boldsymbol{b}^{(1)}_1, \ldots, \boldsymbol{b}^{(1)}_{n_1}, \boldsymbol{d}^{(1)}_{n_1+1}, \ldots, \boldsymbol{d}^{(1)}_{2n_1}, \boldsymbol{b}^{(1)}_{2n_1+1}, \ldots, \boldsymbol{b}^{(1)}_{3n_1+1})$ and $\mathbb{D}^{*(1)} = (\boldsymbol{d}^{*(1)}_1, \ldots, \boldsymbol{d}^{*(1)}_{n_1}, \boldsymbol{b}^{*(1)}_{n_1+1}, \ldots, \boldsymbol{b}^{*(1)}_{2n_1}, \boldsymbol{b}^{*(1)}_{2n_1+1}, \ldots, \boldsymbol{b}^{*(1)}_{3n_1+1})$. It is clear that $\mathbb{D}^{(1)}$ and $\mathbb{D}^{*(1)}$ are dual orthonormal, and have same distribution as the original bases $\mathbb{B}^{(1)}$ and $\mathbb{B}^{*(1)}$, respectively. We also set $\mathbb{D}^{(2)} = \mathbb{B}^{(2)}$ and $\mathbb{D}^{*(2)} = \mathbb{B}^{*(2)}$.

The secret keys and the challenge ciphertext, i.e. $(\{\boldsymbol{k}^{*(j)}_{\overrightarrow{x},I}\}_{j=1,\ldots,\nu}, C)$, in Game 2-$\nu$ can be expressed over bases $\mathbb{B}^{(k)}$ and $\mathbb{B}^{*(k)}$, $k = 0, 1, 2$ as follows:

$$\boldsymbol{k}_{0,j} = (-\alpha_j, \epsilon_j, 1, \eta_j, 0)_{\mathbb{B}^{*(0)}},$$

$$
\begin{aligned}
\boldsymbol{k}_{1,j} &= (\overbrace{s_{1,j}\overrightarrow{e}_1^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j}^{n_1}, \overbrace{v_{1,j}^{(1)}, \ldots, v_{n_1,j}^{(1)}}^{n_1}, \overbrace{\eta_{1,j}^{(1)}, \ldots, \eta_{n_1,j}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}},\\
\boldsymbol{k}_{2,j} &= (\overbrace{s_{2,j}(1, I)}^{2}, \overbrace{v_{1,j}^{(2)}, v_{2,j}^{(2)}}^{2}, \overbrace{\eta_{1,j}^{(2)}, \eta_{2,j}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}},
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{c}_0 &= (\delta, w, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}},\\
\boldsymbol{c}_1 &= (\overbrace{\delta\overrightarrow{y}^{(b)}}^{n_1}, \overbrace{w_1^{(1)}, \ldots, w_{n_1}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}},\\
\forall r \in L: \quad \boldsymbol{c}_r &= (\overbrace{\delta_r(-r, 1)}^{2}, \overbrace{w_{1,r}^{(2)}, w_{2,r}^{(2)},}^{2} \overbrace{0^2,}^{2} \overbrace{\varphi_r}^{1})_{\mathbb{B}^{(2)}},\\
\boldsymbol{c}_M &= g_T^{\zeta} M^{(b)}.
\end{aligned}
$$

They can also be expressed over bases $\mathbb{D}^{(k)}$ and $\mathbb{D}^{*(k)}$ for $k = 0, 1, 2$ as follows. The first component of each key can be represented as $\boldsymbol{k}_{0,j} = (-\alpha_j, \epsilon_j, 1, \eta_j, 0)_{\mathbb{B}^{*(0)}} = (-\alpha_j, \theta_j, 1, \eta_j, 0)_{\mathbb{D}^{*(0)}}$, where $\theta_j = \epsilon_j - \lambda$ are uniform and independent due to $\epsilon_j \xleftarrow{\cup} \mathbb{F}_q$. The remaining key components can be expressed in a similar way, i.e.

$$
\begin{aligned}
\boldsymbol{k}_{1,j} &= (\overbrace{s_{1,j}\overrightarrow{e}_1^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j}^{n_1}, \overbrace{v_{1,j}^{(1)}, \ldots, v_{n_1,j}^{(1)}}^{n_1}, \overbrace{\eta_{1,j}^{(1)}, \ldots, \eta_{n_1,j}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}}\\
&= (\overbrace{s_{1,j}\overrightarrow{e}_1^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j}^{n_1}, \overbrace{\mu_{1,1}^{(1)}s_{1,j} + \beta_j^{(1)}\overrightarrow{x}_j \cdot \overrightarrow{\mu}_1^{(1)} + v_{1,j}^{(1)}, \ldots, \mu_{n_1,1}^{(1)}\alpha_j^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j \cdot \overrightarrow{\mu}_{n_1}^{(1)} + v_{n_1,j}^{(1)}}^{n_1},\\
&\quad \overbrace{\eta_{1,j}^{(1)}, \ldots, \eta_{n_1,j}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{D}^{*(1)}}\\
&= (\overbrace{\alpha_j^{(1)}\overrightarrow{e}_1^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j}^{n_1}, \overbrace{\theta_{1,j}^{(1)}, \ldots, \theta_{n_1,j}^{(1)}}^{n_1}, \overbrace{\eta_{1,j}^{(1)}, \ldots, \eta_{n_1,j}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{D}^{*(1)}}, \text{and}\\
\boldsymbol{k}_{2,j} &= (\overbrace{s_{2,j}(1, I)}^{2}, \overbrace{v_{1,j}^{(2)}, v_{2,j}^{(2)}}^{2}, \overbrace{\eta_{1,j}^{(2)}, \eta_{2,j}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}}\\
&= (\overbrace{s_{2,j}(1, I)}^{2}, \overbrace{v_{1,j}^{(2)}, v_{2,j}^{(2)}}^{2}, \overbrace{\eta_{1,j}^{(2)}, \eta_{2,j}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{D}^{*(2)}},
\end{aligned}
$$

where $\theta_{i,j}^{(1)} = \mu_{i,1}^{(1)}s_{1,j} + \beta_j^{(1)}\overrightarrow{x}_j \cdot \overrightarrow{\mu}_i^{(1)} + v_{i,j}^{(1)}$, $i = 1, \ldots, n_1$, $j = 1, \ldots, \nu$ are uniformly distributed since $v_{i,j}^{(1)} \xleftarrow{\cup} \mathbb{F}_q$.

The first component of the ciphertext can be expressed as $\boldsymbol{c}_0 = (\delta, w, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}} = (\delta, w, \zeta', 0, \varphi)_{\mathbb{D}^{(0)}}$, where $\zeta' = \zeta + \lambda w$ is uniformly distributed since $w, \zeta \xleftarrow{\cup} \mathbb{F}_q$. Similarly, other components of the ciphertext can be represented as

$$
\begin{aligned}
\boldsymbol{c}_1 &= (\overbrace{\delta\overrightarrow{y}^{(b)}}^{n_1}, \overbrace{w_1^{(1)}, \ldots, w_{n_1}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}\\
&= (\overbrace{\overrightarrow{y}'}^{n_1}, \overbrace{w_1^{(1)}, \ldots, w_{n_1}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{D}^{(1)}},\\
\forall r \in L: \quad \boldsymbol{c}_r &= (\overbrace{\delta_r(-r, 1)}^{2}, \overbrace{w_{1,r}^{(2)}, w_{2,r}^{(2)},}^{2} \overbrace{0^2,}^{2} \overbrace{\varphi_r}^{1})_{\mathbb{B}^{(2)}}
\end{aligned}
$$

$$= (\overbrace{\delta_r(-r,1),}^{2} \overbrace{w^{(2)}_{1,r}, w^{(2)}_{2,r},}^{2} \overbrace{0^2,}^{2} \overbrace{\varphi_r}^{1})_{\mathbb{D}^{(2)}},$$

where $\overrightarrow{y}' = (y'_1, \ldots, y'_{n_1})$, $y'_i = \delta y^{(b)}_i - \sum_{j=1}^{n_1} w^{(1)}_j \mu^{(1)}_{j,i}$, $i = 1, \ldots, n_1$, are uniform and independent due to $(w^{(1)}_1, \ldots, w^{(1)}_{n_1}) \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_1}$.
In the light of the adversary's view, both $(\mathbb{B}^{(k)}, \mathbb{B}^{*(k)})$ and $(\mathbb{D}^{(k)}, \mathbb{D}^{*(k)})$ for $k = 0, 1, 2$ are consistent with public key $(1^\lambda, \mathsf{param}_{\overrightarrow{n}}, \{\widehat{\mathbb{B}}^{(k)}\}_{k=0,1,2})$. Therefore, $\{\boldsymbol{k}^{*(j)}_{\overrightarrow{x},I}\}_{j=1,\ldots,\nu}$ and C can be expressed in two ways: in Game 2-ν over bases $(\mathbb{B}^{(k)}, \mathbb{B}^{*(k)})$ and in Game 3 over bases $(\mathbb{D}^{(k)}, \mathbb{D}^{*(k)})$. Thus, Game 2-$\nu$ can be conceptually changed to Game 3 if $w \neq 0$ and $(w^{(1)}_1, \ldots, w^{(1)}_{n_1}) \neq \overrightarrow{0}$, i.e., except with probability $2/q$. □

Lemma 8. *For any adversary $\mathcal{A}$, $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$.*

Proof. The value of b is independent from the adversary's view in Game 3. Therefore, $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$. □

A.1 Proof of Lemmas 1, 2 and 3

In order to reduce the DLIN problem from Definition 2 to Problems 1 and 2 from Definitions 6 and 7, respectively, we further introduce three "basic problems" that will serve in intermediate steps of the reduction:

- Basic Problem 0 in Definition 8.
- Basic Problem 1 in Definition 9.
- Basic Problem 2 in Definition 10.

In order to prove **Lemmas** 1 and 2 we use two additional **Lemmas** 9 and 10 which are common lemmas used in the proofs of **Lemmas** 1 and 2.

Lemma 9. *Let $(q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e)$ be dual pairing vector spaces by direct product of symmetric pairing groups. Using $\{\phi_{i,j}\}$, we can efficiently sample a random linear transformation $W = \sum_{i=1,j=1}^{N,N} r_{i,j}\phi_{i,j}$ of $\mathbb{V}$ with random coefficients $\{r_{i,j}\}_{i,j\in\{1,\ldots,N\}} \xleftarrow{\mathsf{U}} GL(N, \mathbb{F}_q)$. The matrix $(r^*_{i,j}) = (\{r_{i,j}\}^{-1})^T$ defines the adjoint action on $\mathbb{V}$ for pairing e, i.e., $e(W(\boldsymbol{x}), (W^{-1})^T(\boldsymbol{y})) = e(\boldsymbol{x}, \boldsymbol{y})$ for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{V}$, where $(W^{-1})^T = \sum_{i=1,j=1}^{N,N} r^*_{i,j}\phi_{i,j}$.*

The proof of Lemma 9 is provided in [20].

Definition 8 (Basic Problem 0). *Basic Problem 0 is to decide bit β, given $(\mathsf{param}_{\mathrm{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}^*_\beta, \boldsymbol{f}, bG, aG, acG) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathrm{BP0}}_\beta(1^\lambda)$ for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$ with probability non-negligibly better than by a random guess, where*

$$\begin{aligned}
&\mathcal{G}^{\mathrm{BP0}}_\beta\ (1^\lambda):\\
&\quad \mathsf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda),\\
&\quad \mathsf{param}_{\mathbb{V}} = (q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 3, \mathsf{param}_{\mathbb{G}}),\\
&\quad \Lambda = (\lambda_{i,j}) \xleftarrow{\mathsf{U}} GL(3, \mathbb{F}_q),\ (\mu_{i,j}) = (\Lambda^T)^{-1},\quad b, a \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\\
&\quad \boldsymbol{b}_i = b\sum_{j=1}^{3} \lambda_{i,j}\boldsymbol{a}_j,\ i = 1, 3,\ \widehat{\mathbb{B}} = (\boldsymbol{b}_1, \boldsymbol{b}_3),\\
&\quad \boldsymbol{b}^*_i = a\sum_{j=1}^{3} \mu_{i,j}\boldsymbol{a}_j,\ i = 1, 2, 3,\ \mathbb{B}^* = (\boldsymbol{b}^*_1, \boldsymbol{b}^*_2, \boldsymbol{b}^*_3),\\
&\quad g_T = e(G, G)^{ab},\quad \mathsf{param}_{\mathrm{BP0}} = (\mathsf{param}_{\mathbb{V}}, g_T),\\
&\quad \delta, \sigma, \omega \xleftarrow{\mathsf{U}} \mathbb{F}_q,\quad \rho, \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,
\end{aligned}$$

$$\boldsymbol{y}_0^* = (\delta, 0, \sigma)_{\mathbb{B}^*}, \quad \boldsymbol{y}_1^* = (\delta, \rho, \sigma)_{\mathbb{B}^*}, \quad \boldsymbol{f} = (\omega, \tau, 0)_{\mathbb{B}},$$
$$\text{Output } (\mathsf{param}_{\mathrm{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}_\beta^*, \boldsymbol{f}, bG, aG, acG).$$

Let $\mathsf{Adv}_{\mathcal{F}}^{\mathrm{BP0}}(\lambda)$ *denote the corresponding advantage of a PPT algorithm* $\mathcal{F}$ *for the Basic Problem* 0.

Lemma 10. *For any adversary* $\mathcal{F}$*, there exists a probabilistic machine* $\mathcal{D}$*, whose running time is essentially the same as that of* $\mathcal{D}$*, such that for any security parameter* λ*,* $\mathsf{Adv}_{\mathcal{F}}^{\mathrm{BP0}}(\lambda) \leq \mathsf{Adv}_{\mathcal{D}}^{\mathrm{DLIN}}(\lambda) + 5/q$*.*

The proof of **Lemma** 10 can be found in [20].

Proof of Lemma 1: Combining **Lemma** 9, 10, 11 and 12, we obtain **Lemma** 1.

Definition 9 (Basic Problem 1). *Basic Problem* 1 *is to find bit* β*, given* $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{f}_\beta^{(0)}, \boldsymbol{f}_{\beta,1}^{(1)}, \boldsymbol{f}_{\beta,1}^{(2)}, \{\boldsymbol{f}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{f}_2^{(2)}) \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathrm{BP1}}(1^\lambda, \overrightarrow{n} = (2; n_1, n_2 = 2))$ *for* $\beta \xleftarrow{\mathsf{U}} \{0,1\}$*, with probability non-negligibly better than by a random guess, where*

$$\begin{aligned}
&\mathcal{G}_\beta^{\mathrm{BP1}} \ (1^\lambda, \overrightarrow{n} = (2; n_1, n_2 = 2)) : \\
&\quad (\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \overrightarrow{n}), \\
&\quad \widehat{\mathbb{B}}^{*(0)} = (\boldsymbol{b}_1^{*(0)}, \boldsymbol{b}_3^{*(0)}, \boldsymbol{b}_4^{*(0)}, \boldsymbol{b}_5^{*(0)}), \\
&\quad \widehat{\mathbb{B}}^{*(1)} = (\boldsymbol{b}_1^{*(1)}, \ldots, \boldsymbol{b}_{n_1}^{*(1)}, \boldsymbol{b}_{n_1+2}^{*(1)}, \ldots, \boldsymbol{b}_{3n_1+1}^{*(1)}), \\
&\quad \widehat{\mathbb{B}}^{*(2)} = (\boldsymbol{b}_1^{*(2)}, \boldsymbol{b}_2^{*(2)}, \boldsymbol{b}_4^{*(2)}, \boldsymbol{b}_5^{*(2)}, \boldsymbol{b}_6^{*(2)}, \boldsymbol{b}_7^{*(2)}), \\
&\quad \omega, \gamma \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times, \quad \boldsymbol{f}_0^{(0)} = (\omega, 0, 0, 0, \gamma)_{\mathbb{B}^{(0)}}, \quad \boldsymbol{f}_1^{(0)} = (\omega, \tau, 0, 0, \gamma)_{\mathbb{B}^{(0)}}, \\
&\quad \rho^{(1)}, \rho^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q, \\
&\quad \text{For } k = 1, 2 : \overrightarrow{u}^{(k)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_k}; \\
&\quad \boldsymbol{f}_{0,1}^{(1)} = (\overbrace{\omega \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\gamma}^{1})_{\mathbb{B}^{(1)}}, \\
&\quad \boldsymbol{f}_{1,1}^{(1)} = (\overbrace{\omega \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{\tau \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\gamma}^{1})_{\mathbb{B}^{(1)}}, \\
&\quad \text{For } i = 2, \ldots, n_1 : \boldsymbol{f}_i^{(1)} = \omega \boldsymbol{b}_i^{(1)}; \\
&\quad \boldsymbol{f}_{0,1}^{(2)} = (\overbrace{\omega, 0,}^{2} \overbrace{0^2,}^{2} \overbrace{0^2,}^{2} \overbrace{\gamma}^{1})_{\mathbb{B}^{(2)}}, \\
&\quad \boldsymbol{f}_{1,1}^{(2)} = (\overbrace{\omega, 0,}^{2} \overbrace{\tau, 0,}^{2} \overbrace{0^2,}^{2} \overbrace{\gamma}^{1})_{\mathbb{B}^{(2)}}, \\
&\quad \boldsymbol{f}_2^{(2)} = \omega \boldsymbol{b}_2^{(2)}, \\
&\quad \text{Output } \ (\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{f}_\beta^{(0)}, \boldsymbol{f}_{\beta,1}^{(1)}, \boldsymbol{f}_{\beta,1}^{(2)}, \{\boldsymbol{f}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{f}_2^{(2)}).
\end{aligned}$$

Let $\mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP1}}(\lambda)$ *denote the advantage of a PPT algorithm* $\mathcal{C}$ *for the Basic Problem* 1*.*

Lemma 11. *For any adversary* $\mathcal{C}$*, there exists a probabilistic machine* $\mathcal{F}$*, whose running time is essentially the same as that of* $\mathcal{C}$*, such that for any security parameter* λ*,* $\mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP1}}(\lambda) \leq \mathsf{Adv}_{\mathcal{F}}^{\mathrm{BP0}}(\lambda)$ *for* $\overrightarrow{n} = (2; n_1, n_2 = 2)$*.*

Proof. $\mathcal{F}$ is given a Basic Problem 0 instance $(\mathsf{param}_{\mathrm{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}_\beta^*, \boldsymbol{f}, bG, aG, acG)$. Using $\mathsf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e)$ contained in $\mathsf{param}_{\mathrm{BP0}}$, $\mathcal{F}$ computes:

$$\mathsf{param}_{\mathbb{V}_0} = (q, \mathbb{V}_0, \mathbb{G}_T, \mathbb{A}_0, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 5, \mathsf{param}_{\mathbb{G}}),$$

$$\mathsf{param}_{\mathbb{V}_l} = (q, \mathbb{V}_l, \mathbb{G}_T, \mathbb{A}_l, e) \xleftarrow{R} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 3n_l+1, \mathsf{param}_{\mathbb{G}}),\ l=1,2,$$
$$\mathsf{param}_{\overrightarrow{n}} = (\{\mathsf{param}_{\mathbb{V}_l}\}_{l=0,1,2}, g_T),$$

where g_T is contained in $\mathsf{param}_{\mathrm{BP0}}$. $\mathcal{F}$ generates random linear transformation W_l on $\mathbb{V}_l (l=0,1,2)$ given in Lemma 9, then sets

$$\begin{aligned}
&\boldsymbol{d}_l^{(0)} = W_0(\boldsymbol{b}_l^*, 0, 0), \quad l=1,2; \qquad \boldsymbol{d}_3^{(0)} = W_0(0,0,0,0,aG),\\
&\boldsymbol{d}_4^{(0)} = W_0(0,0,0,aG,0), \qquad \boldsymbol{d}_5^{(0)} = W_0(\boldsymbol{b}_3^*, 0, 0),\\
&\boldsymbol{d}_l^{*(0)} = (W_0^{-1})^T(\boldsymbol{b}_l, 0, 0), \quad l=1,2; \qquad \boldsymbol{d}_3^{*(0)} = (W_0^{-1})^T(0,0,0,0,bG),\\
&\boldsymbol{d}_4^{*(0)} = (W_0^{-1})^T(0,0,0,bG,0), \qquad \boldsymbol{d}_5^{*(0)} = (W_0^{-1})^T(\boldsymbol{b}_3, 0, 0),\\
&\boldsymbol{g}_\beta^{(0)} = W_0(\boldsymbol{y}_\beta^*, 0, 0),
\end{aligned}$$

$$\begin{aligned}
&\boldsymbol{d}_1^{(1)} = W_1(\boldsymbol{b}_1^*, 0^{N_1-3}), \qquad \boldsymbol{d}_{n_1+1}^{(1)} = W_1(\boldsymbol{b}_2^*, 0^{N_1-3}), \qquad \boldsymbol{d}_{N_1}^{(1)} = W_1(\boldsymbol{b}_3^*, 0^{N_1-3}),\\
&\boldsymbol{d}_l^{(1)} = W_1(0^m, aG, 0^{N_1-m-1}) \text{ where } \begin{cases} m = l+1 \text{ if } l \in \{2,\dots,n_1\},\\ m = l \text{ if } l \in \{n_1+2,\dots,N_1-1\}, \end{cases}\\
&\boldsymbol{d}_1^{*(1)} = (W_1^{-1})^T(\boldsymbol{b}_1, 0^{N_1-3}), \qquad \boldsymbol{d}_{n_1+1}^{*(1)} = (W_1^{-1})^T(\boldsymbol{b}_2, 0^{N_1-3}), \qquad \boldsymbol{d}_{N_1}^{*(1)} = (W_1^{-1})^T(\boldsymbol{b}_3, 0^{N_1-3}),\\
&\boldsymbol{d}_l^{*(1)} = (W_1^{-1})^T(0^m, bG, 0^{N_1-m-1}) \text{ where } \begin{cases} m = l+1 \text{ if } l \in \{2,\dots,n_1\},\\ m = l \text{ if } l \in \{n_1+2,\dots,N_1-1\}, \end{cases}\\
&\boldsymbol{g}_{\beta,1}^{(1)} = W_1(\boldsymbol{y}_\beta^*, 0^{N_1-3}),\\
&\boldsymbol{g}_l^{(1)} = W_1(0^{l+1}, acG, 0^{N_1-l-2}), \quad l=2,\dots,n_1;
\end{aligned}$$

$$\begin{aligned}
&\boldsymbol{d}_1^{(2)} = W_2(\boldsymbol{b}_1^*, 0^4), \qquad \boldsymbol{d}_3^{(2)} = W_2(\boldsymbol{b}_2^*, 0^4), \qquad \boldsymbol{d}_7^{(2)} = W_2(\boldsymbol{b}_3^*, 0^4),\\
&\boldsymbol{d}_l^{(2)} = W_2(0^m, aG, 0^{7-m-1}) \text{ where } \begin{cases} m = 3 \text{ if } l = 2,\\ m = l \text{ if } l \in \{4,\dots,6\}, \end{cases}\\
&\boldsymbol{d}_1^{*(2)} = (W_2^{-1})^T(\boldsymbol{b}_1, 0^4), \qquad \boldsymbol{d}_3^{*(2)} = (W_2^{-1})^T(\boldsymbol{b}_2, 0^4), \qquad \boldsymbol{d}_7^{*(2)} = (W_2^{-1})^T(\boldsymbol{b}_3, 0^4),\\
&\boldsymbol{d}_l^{*(2)} = (W_2^{-1})^T(0^m, bG, 0^{7-m-1}) \text{ where } \begin{cases} m = 3 \text{ if } l = 2,\\ m = l \text{ if } l \in \{4,\dots,6\}, \end{cases}\\
&\boldsymbol{g}_{\beta,1}^{(2)} = W_2(\boldsymbol{x}_\beta^*, 0^4),\\
&\boldsymbol{g}_2^{(2)} = W_2(0^3, acG, 0^3),
\end{aligned}$$

where $(\boldsymbol{v}, 0^{N_l-3}) = (G', G'', G''', 0^{N_l-3})$ for any $\boldsymbol{v} = (G', G'', G''') \in \mathbb{V} = \mathbb{G}^3$. In this way bases $\mathbb{D}^{(0)} = (\boldsymbol{d}_l^{(0)})_{l=1,\dots,5}$ and $\mathbb{D}^{*(0)} = (\boldsymbol{d}_l^{*(0)})_{l=1,\dots,5}$, $\mathbb{D}^{(j)} = (\boldsymbol{d}_l^{(j)})_{l=1,\dots,3n_j+1}$ and $\mathbb{D}^{*(j)} = (\boldsymbol{d}_l^{*(j)})_{l=1,\dots,3n_j+1}$, $j=1,2$ are dual orthonormal bases.

Therefore, from $\widehat{\mathbb{B}} = (\boldsymbol{b}_1, \boldsymbol{b}_3)$, $\mathbb{B}^*$, bG, and aG the algorithm $\mathcal{F}$ can compute $\mathbb{D}^{(j)}$, $j=0,1,2$; $\widehat{\mathbb{D}}^{*(0)} = (\boldsymbol{d}_1^{*(0)}, \boldsymbol{d}_3^{*(0)}, \boldsymbol{d}_4^{*(0)}, \boldsymbol{d}_5^{*(0)})$, and $\widehat{\mathbb{D}}^{*(j)} = (\boldsymbol{d}_1^{*(j)}, \dots, \boldsymbol{d}_{n_j}^{*(j)}, \boldsymbol{d}_{n_j+2}^{*(j)}, \dots, \boldsymbol{d}_{3n_j+1}^{*(j)})$, $j=1,2$.

Finally, $\mathcal{F}$ hands $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{D}^{(k)}, \widehat{\mathbb{D}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{g}_\beta^{(0)}, \boldsymbol{g}_{\beta,1}^{(1)}, \boldsymbol{g}_{\beta,1}^{(2)}, \{\boldsymbol{g}_i^{(1)}\}_{i=2,\dots,n_1}, \boldsymbol{g}_2^{(2)})$ over to $\mathcal{C}$ and, if $\mathcal{C}$ outputs its bit β' then $\mathcal{F}$ forwards this bit as its own output.

We observe that:

$$\begin{aligned}
&\boldsymbol{g}_0^{(0)} = (\omega', 0, 0, 0, \gamma')_{\mathbb{D}^{(0)}}, && \boldsymbol{g}_1^{(0)} = (\omega', \tau', 0, 0, \gamma')_{\mathbb{D}^{(0)}},\\
&\boldsymbol{g}_{0,1}^{(1)} = (\overbrace{\omega' \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\gamma'}^{1})_{\mathbb{D}^{(1)}}, && \boldsymbol{g}_{0,1}^{(2)} = (\overbrace{\omega', 0,}^{2} \overbrace{0^2,}^{2} \overbrace{0^2,}^{2} \overbrace{\gamma'}^{1})_{\mathbb{D}^{(2)}},
\end{aligned}$$

$$\boldsymbol{g}_{1,1}^{(1)} = (\overbrace{\omega'\overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{\tau'\overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\gamma'}^{1})_{\mathbb{D}^{(1)}}, \quad \boldsymbol{g}_{1,1}^{(2)} = (\overbrace{\omega', 0,}^{2} \overbrace{\tau', 0,}^{2} \overbrace{0^2,}^{2} \overbrace{\gamma'}^{1})_{\mathbb{D}^{(2)}},$$
$$\boldsymbol{g}_i^{(1)} = \omega'\boldsymbol{b}_i^{(1)}, \quad i = 2, \ldots, n_1; \qquad \boldsymbol{g}_2^{(2)} = \omega'\boldsymbol{b}_2^{(2)},$$

where $\omega' = \delta, \tau' = \rho, \gamma' = \sigma$ are distributed uniformly in $\mathbb{F}_q$, Therefore, the distribution of $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{D}^{(k)}, \widehat{\mathbb{D}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{g}_\beta^{(0)}, \boldsymbol{g}_{\beta,1}^{(1)}, \boldsymbol{g}_{\beta,1}^{(2)}, \{\boldsymbol{g}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{g}_2^{(2)})$ is exactly the same as in the instance of Basic Problem 1. □

Lemma 12. *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P1}}(\lambda) \leq \mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP1}}(\lambda) + 3/q$ for $(\overrightarrow{n} = (2; n_1, n_2 = 2))$.*

Proof. $\mathcal{C}$ is given an instance of the Basic Problem 1, i.e., $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{f}_\beta^{(0)}, \boldsymbol{f}_{\beta,1}^{(1)}, \boldsymbol{f}_{\beta,1}^{(2)}, \{\boldsymbol{f}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{f}_2^{(2)})$, and computes $\boldsymbol{r} \xleftarrow{\mathsf{U}} \mathsf{span}\langle \boldsymbol{b}_{3n_1+1}^{(1)} \rangle$, $\boldsymbol{r}' \xleftarrow{\mathsf{U}} \mathsf{span}\langle \boldsymbol{b}_7^{(2)} \rangle$, and sets $\boldsymbol{t}_{\beta,1}^{(1)} = \boldsymbol{f}_{\beta,1}^{(1)} + \boldsymbol{r}$ and $\boldsymbol{t}_{\beta,1}^{(2)} = \boldsymbol{f}_{\beta,1}^{(2)} + \boldsymbol{r}'$.

Then, $\mathcal{C}$ chooses $u_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times$, $(u_{i,j}^{(k)}) \xleftarrow{\mathsf{U}} GL(\mathbb{F}_q, n_k)$, $(z_{i,j}^{(k)}) = \left((u_{i,j}^{(k)})^{-1}\right)^T$ for $i = 1, \ldots, n_k, j = 1, \ldots, n_k$, and $k = 1, 2$, and computes:

$$\boldsymbol{d}_2^{(0)} = (0, u_0, 0, 0, 0)_{\mathbb{B}^{(0)}},$$
$$\boldsymbol{d}_{n_k+i}^{(k)} = (\overbrace{0^{n_k},}^{n_k} \overbrace{u_{i,1}^{(k)}, \ldots, u_{i,n_k}^{(k)},}^{n_k} \overbrace{0^{n_k},}^{n_k} \overbrace{0}^{1})_{\mathbb{B}^{(k)}}, \ i = 1, \ldots, n_k, \ k = 1, 2.$$

$\mathcal{C}$ then sets dual orthonormal basis vectors

$$\boldsymbol{d}_2^{*(0)} = (0, u_0^{-1}, 0, 0, 0)_{\mathbb{B}^{*(0)}},$$
$$\boldsymbol{d}_{n_k+i}^{*(k)} = (\overbrace{0^{n_k},}^{n_k} \overbrace{z_{i,1}^{(k)}, \ldots, z_{i,n_k}^{(k)},}^{n_k} \overbrace{0^{n_k},}^{n_k} \overbrace{0}^{1})_{\mathbb{B}^{*(k)}}, \ i = 1, \ldots, n_k, \ k = 1, 2.$$

Note that $\mathcal{C}$ cannot compute $\boldsymbol{d}_2^{*(0)}$ and $\boldsymbol{d}_{n_k+i}^{*(k)}$, $i = 1, \ldots, n_k$, $k = 1, 2$ due to the lack of $\boldsymbol{b}_2^{*(0)}$ and $\boldsymbol{b}_{n_k+1}^{*(k)}$.

Then, $\mathcal{C}$ sets bases $\mathbb{D}^{(0)} = (\boldsymbol{b}_1^{(0)}, \boldsymbol{d}_2^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_4^{(0)}, \boldsymbol{b}_5^{(0)})$, $\widehat{\mathbb{D}}^{*(0)} = (\boldsymbol{b}_1^{*(0)}, \boldsymbol{b}_3^{*(0)}, \boldsymbol{b}_4^{*(0)}, \boldsymbol{b}_5^{*(0)})$, $\mathbb{D}^{(k)} = (\boldsymbol{b}_1^{(k)}, \ldots, \boldsymbol{b}_{n_k}^{(k)}, \boldsymbol{d}_{n_k+1}^{(k)}, \ldots, \boldsymbol{d}_{2n_k}^{(k)}, \boldsymbol{b}_{2n_k+1}^{(k)}, \ldots, \boldsymbol{b}_{3n_k+1}^{(k)})$, $\widehat{\mathbb{D}}^{*(k)} = (\boldsymbol{b}_1^{*(k)}, \ldots, \boldsymbol{b}_{n_k}^{*(k)}, \boldsymbol{b}_{2n_k+1}^{*(k)}, \ldots, \boldsymbol{b}_{3n_k+1}^{*(k)})$, $k = 1, 2$.

Finally, $\mathcal{C}$ hands $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{D}^{(k)}, \widehat{\mathbb{D}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{f}_\beta^{(0)}, \boldsymbol{t}_{\beta,1}^{(1)}, \boldsymbol{t}_{\beta,1}^{(2)}, \{\boldsymbol{f}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{f}_2^{(2)})$ over to $\mathcal{B}$ and, if $\mathcal{B}$ outputs its bit β' then $\mathcal{C}$ forwards this bit as its own output. Note that with respect to $\mathbb{D}^{(k)}, \mathbb{D}^{*(k)}$, $k = 0, 1, 2$, the above input to $\mathcal{B}$ has the same distribution as the instance of the Problem 1 unless following events occur: $u = 0$, $\overrightarrow{u}^{(1)} = \overrightarrow{0}$, or $\overrightarrow{u}^{(2)} = \overrightarrow{0}$. Those events occur with probability $3/q$ when $\beta = 1$. □

Proof of Lemma 2: Combining **Lemmas** 9, 10, 13 and 14, we obtain **Lemma** 2.

Definition 10 (Basic Problem 2). *Basic Problem* 2 *is to find bit β, given $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{y}_\beta^{*(0)}, \boldsymbol{f}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{y}_{\beta,i}^{*(k)}, \boldsymbol{f}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2}) \xleftarrow{R} \mathcal{G}_\beta^{\mathrm{BP2}}(1^\lambda, \overrightarrow{n} = (2; n_1, n_2 = 2))$ for $\beta \xleftarrow{\mathsf{U}} \{0, 1\}$ with probability non-negligibly better than by a random guess, where*

$$\mathcal{G}_\beta^{\mathrm{BP2}} \ (1^\lambda, \overrightarrow{n} = (2; n_1, n_2 = 2)) :$$
$$(\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{R} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \overrightarrow{n}),$$
$$\widehat{\mathbb{B}}^{(0)} = (\boldsymbol{b}_1^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_4^{(0)}, \boldsymbol{b}_5^{(0)}),$$

$\widehat{\mathbb{B}}^{(1)} = (\boldsymbol{b}_1^{(1)}, \ldots, \boldsymbol{b}_{n_1}^{(1)}, \boldsymbol{b}_{2n_1+1}^{(1)}, \ldots, \boldsymbol{b}_{3n_1+1}^{(1)})$,
$\widehat{\mathbb{B}}^{(2)} = (\boldsymbol{b}_1^{(2)}, \boldsymbol{b}_2^{(2)}, \boldsymbol{b}_5^{(2)}, \boldsymbol{b}_6^{(2)}, \boldsymbol{b}_7^{(2)})$,
$\omega, \xi, \delta \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad z, \pi \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}$,
$\boldsymbol{y}_0^{*(0)} = (\omega, 0, 0, \xi, 0)_{\mathbb{B}^{*(0)}}, \quad \boldsymbol{y}_1^{*(0)} = (\omega, z, 0, \xi, 0)_{\mathbb{B}^{*(0)}}, \quad \boldsymbol{f}^{(0)} = (\delta, \pi, 0, 0, 0)_{\mathbb{B}^{(0)}}$,
For $k = 1, 2$ and $i = 1, \ldots, n_k$:
$\boldsymbol{y}_{0,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{\xi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}}$,
$\boldsymbol{y}_{1,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{z \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{\xi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}}$,
$\boldsymbol{f}_i^{(k)} = (\overbrace{\delta \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{\pi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{(k)}}$,
Output $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{y}_{\beta}^{*(0)}, \boldsymbol{f}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{y}_{\beta,i}^{*(k)}, \boldsymbol{f}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$.

Let $\mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP2}}(\lambda)$ *denote the corresponding advantage of a PPT algorithm* $\mathcal{C}$ *for the Basic Problem* 2.

Lemma 13. *For any adversary* $\mathcal{C}$*, there exists a probabilistic machine* $\mathcal{F}$*, whose running time is essentially the same as that of* $\mathcal{C}$*, such that for any security parameter* λ*,* $\mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP2}}(\lambda) = \mathsf{Adv}_{\mathcal{F}}^{\mathrm{BP0}}(\lambda)$ *for* $\overrightarrow{n} = (2; n_1, n_2 = 2)$.

Proof. $\mathcal{F}$ is given an instance of the Basic Problem 0, i.e. $(\mathsf{param}_{\mathrm{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}_{\beta}^*, \boldsymbol{f}, bG, aG, acG)$. Using $\mathsf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e)$ contained in $\mathsf{param}_{\mathrm{BP0}}$ it computes

$$\begin{aligned}
&\mathsf{param}_{\mathbb{V}_0} = (q, \mathbb{V}_0, \mathbb{G}_T, \mathbb{A}_0, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^{\lambda}, 5, \mathsf{param}_{\mathbb{G}}),\\
&\mathsf{param}_{\mathbb{V}_l} = (q, \mathbb{V}_l, \mathbb{G}_T, \mathbb{A}_l, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^{\lambda}, 3n_l + 1, \mathsf{param}_{\mathbb{G}}),\ l = 1, 2,\\
&\mathsf{param}_{\overrightarrow{n}} = (\{\mathsf{param}_{\mathbb{V}_l}\}_{l=0,1,2}, g_T),
\end{aligned}$$

where g_T is contained in $\mathsf{param}_{\mathrm{BP0}}$. Then, $\mathcal{F}$ generates random linear transformation W_l on $\mathbb{V}_l (l = 0, 1, 2)$ given in Lemma 9 and sets

$$\begin{aligned}
&\boldsymbol{d}_l^{(0)} = W_0(\boldsymbol{b}_l, 0, 0),\ l = 1, 2; \quad \boldsymbol{d}_3^{(0)} = W_0(0, 0, 0, 0, bG),\\
&\boldsymbol{d}_4^{(0)} = W_0(\boldsymbol{b}_3, 0, 0), \quad \boldsymbol{d}_5^{(0)} = W_0(0, 0, 0, bG, 0),\\
&\boldsymbol{d}_l^{*(0)} = (W_0^{-1})^T(\boldsymbol{b}_l^*, 0, 0),\ l = 1, 2; \quad \boldsymbol{d}_3^{*(0)} = (W_0^{-1})^T(0, 0, 0, 0, aG),\\
&\boldsymbol{d}_4^{*(0)} = (W_0^{-1})^T(\boldsymbol{b}_3^*, 0, 0), \quad \boldsymbol{d}_5^{*(0)} = (W_0^{-1})^T(0, 0, 0, aG, 0),\\
&\boldsymbol{p}_{\beta}^{*(0)} = (W_0^{-1})^T(\boldsymbol{y}_{\beta}^*, 0, 0), \quad \boldsymbol{g}^{(0)} = W_0(\boldsymbol{f}, 0, 0),\\
&\text{For } k = 1, 2 :\\
&\quad \text{For } l = 1, 2, 3 \text{ and } i = 1, \ldots, n_k :\\
&\quad\quad \boldsymbol{d}_{(l-1)n_k+i}^{(k)} = W_k(0^{3(i-1)}, \boldsymbol{b}_l, 0^{3(n_k-i)}, 0),\\
&\quad \boldsymbol{d}_{3n_k+1}^{(k)} = W_k(0^{3n_k}, bG),\\
&\quad \text{For } l = 1, 2, 3 \text{ and } i = 1, \ldots, n_k :\\
&\quad\quad \boldsymbol{d}_{(l-1)n_k+i}^{*(k)} = (W_k^{-1})^T(0^{3(i-1)}, \boldsymbol{b}_l^*, 0^{3(n_k-i)}, 0),\\
&\quad \boldsymbol{d}_{3n_k+1}^{*(k)} = (W_k^{-1})^T(0^{3n_k}, aG),\\
&\quad \text{For } i = 1, \ldots, n_k :\\
&\quad\quad \boldsymbol{p}_{\beta,i}^{*(k)} = (W_k^{-1})^T(0^{3(i-1)}, \boldsymbol{y}_{\beta}^*, 0^{3(n_k-i)}, 0),
\end{aligned}$$

$$\boldsymbol{g}_i^{(k)} = W_1(0^{3(i-1)}, \boldsymbol{f}, 0^{3(n_k-i)}, 0).$$

Observe that bases $\mathbb{D}^{(0)} = (\boldsymbol{d}_l^{(0)})_{l=1,\ldots,5}$ and $\mathbb{D}^{*(0)} = (\boldsymbol{d}_l^{*(0)})_{l=1,\ldots,5}$, $\mathbb{D}^{(j)} = (\boldsymbol{d}_l^{(j)})_{l=1,\ldots,3n_j+1}$ and $\mathbb{D}^{*(j)} = (\boldsymbol{d}_l^{*(j)})_{l=1,\ldots,3n_j+1}$, $j = 1, 2$ are dual orthonormal bases.
Therefore, $\mathcal{F}$ can use $\widehat{\mathbb{B}} = (\boldsymbol{b}_1, \boldsymbol{b}_3)$, $\mathbb{B}^*$, bG, and aG to compute bases $\mathbb{D}^{*(j)}$, $j = 0, 1, 2$; $\widehat{\mathbb{D}}^{(0)} = (\boldsymbol{d}_1^{(0)}, \boldsymbol{d}_3^{(0)}, \boldsymbol{d}_4^{(0)}, \boldsymbol{d}_5^{(0)})$, and $\widehat{\mathbb{D}}^{(j)} = (\boldsymbol{d}_l^{(j)}, \ldots, \boldsymbol{d}_{n_j}^{(j)}, \boldsymbol{d}_{2n_j+1}^{(j)}, \ldots, \boldsymbol{d}_{3n_j+1}^{(j)})$, $j = 1, 2$.
Finally, $\mathcal{F}$ hands $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{D}}^{(0)}, \mathbb{D}^{*(0)}, \boldsymbol{p}_\beta^{*(0)}, \boldsymbol{g}^{(0)}, \{\widehat{\mathbb{D}}^{(k)}, \mathbb{D}^{*(k)}, \{\boldsymbol{p}_{\beta,i}^{*(k)}, \boldsymbol{g}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$ over to $\mathcal{C}$ and, if $\mathcal{C}$ outputs a bit β', forwards this bit as its own output.
We observe that:

$$\boldsymbol{p}_0^{*(0)} = (\omega, 0, 0, \xi, 0)_{\mathbb{D}^{*(0)}}, \quad \boldsymbol{p}_1^{*(0)} = (\omega, z, 0, \xi, 0)_{\mathbb{D}^{*(0)}}, \quad \boldsymbol{g}^{(0)} = (\delta, \pi, 0, 0, 0)_{\mathbb{D}^{(0)}},$$

For $k = 1, 2$ and $i = 1, \ldots, n_k$:

$$\boldsymbol{p}_{0,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{\xi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{D}^{*(k)}},$$

$$\boldsymbol{p}_{1,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{z \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{\xi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{D}^{*(k)}},$$

$$\boldsymbol{g}_i^{(k)} = (\overbrace{\delta \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{\pi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{D}^{(k)}}.$$

Therefore, the distribution of $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{D}}^{(0)}, \mathbb{D}^{*(0)}, \boldsymbol{p}_\beta^{*(0)}, \boldsymbol{g}^{(0)}, \{\widehat{\mathbb{D}}^{(k)}, \mathbb{D}^{*(k)}, \{\boldsymbol{p}_{\beta,i}^{*(k)}, \boldsymbol{g}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$ is exactly the same as in the instance of the Basic Problem 2.

Lemma 14. *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P2}}(\lambda) = \mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP2}}(\lambda)$.*

Proof. Given an instance of the Basic Problem 2, i.e., $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{y}_\beta^{*(0)}, \boldsymbol{f}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{y}_{\beta,i}^{*(k)}, \boldsymbol{f}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$ the algorithm $\mathcal{C}$ computes $\boldsymbol{r}_i^{*(k)} \xleftarrow{\mathsf{U}} \mathsf{span} < \boldsymbol{b}_{2n_k+1}^{*(k)}, \ldots, \boldsymbol{b}_{3n_k}^{*(k)} >$ and sets $\boldsymbol{h}_{\beta,i}^{*(k)} = \boldsymbol{y}_{\beta,i}^{*(k)} + \boldsymbol{r}_i^{*(k)}$, $k = 1, 2$.
Then, $\mathcal{C}$ chooses $z_0' \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times$, $(z_{i,j}'^{(k)}) \xleftarrow{\mathsf{U}} GL(\mathbb{F}_q, n_k), i = 1, \ldots, n_k, j = 1, \ldots, n_k, k = 1, 2$, and computes:

$$\boldsymbol{d}_2^{*(0)} = (0, z_0', 0, 0, 0)_{\mathbb{B}^{*(0)}},$$

$$\boldsymbol{d}_{n_k+i}^{*(k)} = (\overbrace{0^{n_k}}^{n_k}, \overbrace{z_{i,1}'^{(k)}, \ldots, z_{i,n_k}'^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}},\ i = 1, \ldots, n_k,\ k = 1, 2,$$

Then, $\mathcal{C}$ sets $z_0 = z^{-1} z_0', u_0 = z_0^{-1}, (z_{i,j}^{(k)}) = z^{-1}(z_{i,j}'^{(k)})$, and $(u_{i,j}^{(k)}) = \left((z_{i,j}^{(k)})^{-1}\right)^T$, where z is defined as in the Basic Problem 2. This leads to

$$\boldsymbol{d}_2^{*(0)} = (0, z z_0, 0, 0, 0)_{\mathbb{B}^{*(0)}},$$

$$\boldsymbol{d}_{n_k+i}^{*(k)} = (\overbrace{0^{n_k}}^{n_k}, \overbrace{z z_{i,1}^{(k)}, \ldots, z z_{i,n_k}^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}},\ i = 1, \ldots, n_k,\ k = 1, 2,$$

$$\boldsymbol{d}_2^{(0)} = (0, z^{-1} u_0, 0, 0, 0)_{\mathbb{B}^{(0)}},$$

$$\boldsymbol{d}_{n_k+i}^{(k)} = (\overbrace{0^{n_k}}^{n_k}, \overbrace{z^{-1} u_{i,1}^{(k)}, \ldots, z^{-1} u_{i,n_k}^{(k)}}^{n_k}, \overbrace{0,}^{1} \overbrace{0}^{1})_{\mathbb{B}^{(k)}},\ i = 1, \ldots, n_k,\ k = 1, 2.$$

$\mathcal{C}$ then computes $\mathbb{D}^{*(0)} = (\boldsymbol{b}_1^{*(0)}, \boldsymbol{d}_2^{*(0)}, \boldsymbol{b}_3^{*(0)}, \boldsymbol{b}_4^{*(0)}, \boldsymbol{b}_5^{*(0)})$, $\widehat{\mathbb{D}}^{(0)} = (\boldsymbol{b}_1^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_4^{(0)}, \boldsymbol{b}_5^{(0)})$, $\mathbb{D}^{*(k)} = (\boldsymbol{b}_1^{*(k)}, \ldots, \boldsymbol{b}_{n_k}^{*(k)}, \boldsymbol{d}_{n_k+1}^{*(k)}, \ldots, \boldsymbol{d}_{2n_k}^{*(k)}, \boldsymbol{b}_{2n_k+1}^{*(k)}, \ldots, \boldsymbol{b}_{3n_k+1}^{*(k)})$, $\widehat{\mathbb{D}}^{(k)} = (\boldsymbol{b}_1^{(k)}, \ldots, \boldsymbol{b}_{n_k}^{(k)}, \boldsymbol{b}_{2n_k+1}^{(k)}, \ldots, \boldsymbol{b}_{3n_k+1}^{(k)})$, $k = 1, 2$.

Finally, $\mathcal{C}$ hands $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{D}}^{(0)}, \mathbb{D}^{*(0)}, \boldsymbol{y}_{\beta}^{*(0)}, \boldsymbol{f}^{(0)}, \{\widehat{\mathbb{D}}^{(k)}, \mathbb{D}^{*(k)}, \{\boldsymbol{y}_{\beta,i}^{*(k)}, \boldsymbol{f}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$ over to $\mathcal{B}$ and outputs $\beta' \in \{0,1\}$ if $\mathcal{B}$ outputs β'.
For π in Basic Problem 2, let $\pi' = z\pi$. Then, with respect to $\pi', \mathbb{D}^{(k)}, \mathbb{D}^{*(k)}$, $k = 0,1,2$, the above answer to $\mathcal{B}$ has the same distribution as in the instance of Problem 2. □

Proof of Lemma 3: The proof of Lemma 3 was given in [20].

B Security Proof of Theorem 2

In the proof, the concepts of normal, semi-functional and nominal semi-functional forms are defined similar as in the proof of Theorem 1, A semi-functional secret key $\boldsymbol{k}_{\overrightarrow{x},I}^{*\mathsf{semi}}$ and a semi-functional ciphertext C^{semi} are expressed by Eq. (15) and Eqs.(9)-(12) respectively. Meanwhile, a nominal semi-functional secret key $\boldsymbol{k}_{\overrightarrow{x},I}^{*\mathsf{nom\text{-}semi}}$ and a nominal semi-functional ciphertext $C^{\mathsf{nom\text{-}semi}}$ are expressed by Eq.(13) and Eq.(14) respectively. The theorem is then proven through a sequence from Game 0 (original game) to Game 3 using similar techniques as for Theorem 1. First, we prove that the probability difference between Games 0 and 1 is equivalent to the advantage of Problem 3. The difference between Games 2-m' and 2-m is equivalent to the advantage of Problem 4 (i.e., advantage of the DLIN assumption). We also show that the difference between Games 2-m' and 2-$(m+1)$ is equivalent to the advantage of Problem 4 (i.e., advantage of the DLIN assumption). In the final step, we show that Game 2-ν can be conceptually changed to Game 3 where the adversary has 0 advantage.

Definition 11 (Problem 3). *Problem* 3 *is to find bit* β *given a tuple* $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{t}_{\beta}^{(0)}, \boldsymbol{t}_{\beta,1}^{(1)}, \{\boldsymbol{t}_{\beta,1,j}^{(2)}\}_{j=1,\ldots,d}, \{\boldsymbol{t}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{t}_2^{(2)}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\beta}^{\mathrm{P3}}(1^{\lambda}, \overrightarrow{n} = (2; n_1, n_2 = 2), d)$ *for* $\beta \xleftarrow{\mathsf{U}} \{0,1\}$ *with probability non-negligibly better than by a random guess, where*

$$
\begin{aligned}
&\mathcal{G}_{\beta}^{\mathrm{P3}}(1^{\lambda}, \overrightarrow{n} = (2; n_1, n_2 = 2), d): \ (\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^{\lambda}, \overrightarrow{n}),\\
&\quad \widehat{\mathbb{B}}^{*(0)} = (\boldsymbol{b}_1^{*(0)}, \boldsymbol{b}_3^{*(0)}, \boldsymbol{b}_4^{*(0)}, \boldsymbol{b}_5^{*(0)}),\\
&\quad \widehat{\mathbb{B}}^{*(1)} = (\boldsymbol{b}_1^{*(1)}, \ldots, \boldsymbol{b}_{n_1}^{*(1)}, \boldsymbol{b}_{2n_1+1}^{*(1)}, \ldots, \boldsymbol{b}_{3n_1+1}^{*(1)}),\\
&\quad \widehat{\mathbb{B}}^{*(2)} = (\boldsymbol{b}_1^{*(2)}, \boldsymbol{b}_2^{*(2)}, \boldsymbol{b}_5^{*(2)}, \boldsymbol{b}_6^{*(2)}, \boldsymbol{b}_7^{*(2)}),\\
&\quad \delta, u, \rho \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad \boldsymbol{t}_0^{(0)} = (\delta, 0, 0, 0, \rho)_{\mathbb{B}^{(0)}}, \quad \boldsymbol{t}_1^{(0)} = (\delta, u, 0, 0, \rho)_{\mathbb{B}^{(0)}},\\
&\quad \rho^{(1)} \xleftarrow{\mathsf{U}} \mathbb{F}_q, \ \overrightarrow{u}^{(1)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_1},\\
&\quad \boldsymbol{t}_{0,1}^{(1)} = (\overbrace{\delta \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\rho^{(1)}}^{1})_{\mathbb{B}^{(1)}},\\
&\quad \boldsymbol{t}_{1,1}^{(1)} = (\overbrace{\delta \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{\overrightarrow{u}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\rho^{(1)}}^{1})_{\mathbb{B}^{(1)}},\\
&\quad \text{For } i = 2, \ldots, n_1: \ \boldsymbol{t}_i^{(1)} = \delta \boldsymbol{b}_i^{(1)};\\
&\quad \text{For } j = 1, \ldots, d:\\
&\qquad \rho_j^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q, \overrightarrow{u}_j^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^2,\\
&\qquad \boldsymbol{t}_{0,1,j}^{(2)} = (\overbrace{\delta, 0,}^{2} \overbrace{0^2,}^{2} \overbrace{0^2,}^{2} \overbrace{\rho_j^{(2)}}^{1})_{\mathbb{B}^{(2)}},\\
&\qquad \boldsymbol{t}_{1,1,j}^{(2)} = (\overbrace{\delta, 0,}^{2} \overbrace{\overrightarrow{u}_j^{(2)},}^{2} \overbrace{0^2,}^{2} \overbrace{\rho_j^{(2)}}^{1})_{\mathbb{B}^{(2)}},\\
&\quad \boldsymbol{t}_2^{(2)} = \delta \boldsymbol{b}_2^{(2)},
\end{aligned}
$$

$$\text{return} \quad (\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{t}_{\beta}^{(0)}, \boldsymbol{t}_{\beta,1}^{(1)}, \{\boldsymbol{t}_{\beta,1,j}^{(2)}\}_{j=1,\ldots,d}, \{\boldsymbol{t}_{i}^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{t}_{2}^{(2)}).$$

The corresponding advantage of PPT algorithm $\mathcal{B}$ in solving Problem 3 is defined as follows:

$$\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P3}}(\lambda) = \Big| \ \mathsf{Pr}\Big[\mathcal{B}(1^{\lambda}, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathrm{P3}}(1^{\lambda}, \overrightarrow{n})\Big] \ - \ \mathsf{Pr}\Big[\mathcal{B}(1^{\lambda}, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathrm{P3}}(1^{\lambda}, \overrightarrow{n})\Big] \ \Big|.$$

Lemma 15. *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{D}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P3}}(\lambda) \leq \mathsf{Adv}_{\mathcal{D}}^{\mathrm{DLIN}}(\lambda) + 8/q$.*

Definition 12 (Problem 4). *Problem 4 is to find bit β given $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{h}_{\beta}^{*(0)}, \boldsymbol{t}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{h}_{\beta,i}^{*(k)}, \boldsymbol{t}_{i}^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2}) \xleftarrow{R} \mathcal{G}_{\beta}^{\mathrm{P4}}(1^{\lambda}, \overrightarrow{n} = (2; n_1, n_2 = 2))$ for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$ with probability non-negligibly better than by a random guess, where*

$$\begin{aligned}
&\mathcal{G}_{\beta}^{\mathrm{P4}}(1^{\lambda}, \overrightarrow{n} = (2; n_1, n_2 = 2)) : \ (\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{R} \mathcal{G}_{\mathsf{ob}}(1^{\lambda}, \overrightarrow{n}), \\
&\quad \widehat{\mathbb{B}}^{(0)} = (\boldsymbol{b}_1^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_4^{(0)}, \boldsymbol{b}_5^{(0)}), \\
&\quad \widehat{\mathbb{B}}^{(1)} = (\boldsymbol{b}_1^{(1)}, \ldots, \boldsymbol{b}_{n_1}^{(1)}, \boldsymbol{b}_{2n_1+1}^{(1)}, \ldots, \boldsymbol{b}_{3n_1+1}^{(1)}), \\
&\quad \widehat{\mathbb{B}}^{(2)} = (\boldsymbol{b}_1^{(2)}, \boldsymbol{b}_2^{(2)}, \boldsymbol{b}_5^{(2)}, \boldsymbol{b}_6^{(2)}, \boldsymbol{b}_7^{(2)}), \\
&\quad \omega, \xi, \delta \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad z, \pi \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}, \quad u = z^{-1}, \\
&\quad \boldsymbol{h}_0^{*(0)} = (\omega, 0, 0, \xi, 0)_{\mathbb{B}^{*(0)}}, \quad \boldsymbol{h}_1^{*(0)} = (\omega, z, 0, \xi, 0)_{\mathbb{B}^{*(0)}}, \quad \boldsymbol{t}^{(0)} = (\delta, \pi u, 0, 0, 0)_{\mathbb{B}^{(0)}}, \\
&\quad \text{For } k = 1, 2 : \\
&\qquad \text{For } i = 1, \ldots, n_k \text{ and } j = 1, \ldots, n_k : \\
&\qquad\quad (u_{i,j}^{(k)}) \xleftarrow{\mathsf{U}} GL(\mathbb{F}_q, n_k), \ (z_{i,j}^{(k)}) = \big((u_{i,j}^{(k)})^{-1}\big)^{T}, \\
&\qquad \text{For } i = 1, \ldots, n_k : \\
&\qquad\quad \overrightarrow{\omega}_i^{(k)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_k}, \\
&\qquad\quad \boldsymbol{h}_{0,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{\overrightarrow{\omega}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}}, \\
&\qquad\quad \boldsymbol{h}_{1,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{z_{i,1}^{(k)}, \ldots, z_{i,n_k}^{(k)}}^{n_k}, \overbrace{\overrightarrow{\omega}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}}, \\
&\qquad\quad \boldsymbol{t}_i^{(k)} = (\overbrace{\delta \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{\pi u_{i,1}^{(k)}, \ldots, \pi u_{i,n_k}^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{(k)}}, \\
&\text{return} \quad (\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{h}_{\beta}^{*(0)}, \boldsymbol{t}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{h}_{\beta,i}^{*(k)}, \boldsymbol{t}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2}).
\end{aligned}$$

Let $\mathcal{B}$ be a probabilistic machine, we define the advantage of $\mathcal{B}$ for Problem 4 as follows:

$$\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P4}}(\lambda) = \Big| \ \mathsf{Pr}\Big[\mathcal{B}(1^{\lambda}, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{R} \mathcal{G}_0^{\mathrm{P4}}(1^{\lambda}, \overrightarrow{n})\Big] \ - \ \mathsf{Pr}\Big[\mathcal{B}(1^{\lambda}, \varpi) \to 1 \ \Big| \ \varpi \xleftarrow{R} \mathcal{G}_1^{\mathrm{P4}}(1^{\lambda}, \overrightarrow{n})\Big] \ \Big|.$$

Lemma 16. *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{D}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P4}}(\lambda) \leq \mathsf{Adv}_{\mathcal{D}}^{\mathrm{DLIN}}(\lambda) + 5/q$.*

Lemma 17. *For $p \in \mathbb{F}_q$, let $C_p = \{(\overrightarrow{x}, \overrightarrow{v}) | \overrightarrow{x} \cdot \overrightarrow{v} = p\} \subset V \times V^*$ where V is n-dimensional vector space $\mathbb{F}_q^n$, and V^* its dual. For all $(\overrightarrow{x}, \overrightarrow{v}) \in C_p$, for all $(\overrightarrow{r}, \overrightarrow{w}) \in C_p$, $\Pr[\overrightarrow{x}U = \overrightarrow{r} \wedge \overrightarrow{v}Z = \overrightarrow{w}] = \Pr[\overrightarrow{x}Z = \overrightarrow{r} \wedge \overrightarrow{v}U = \overrightarrow{w}] = 1/\sharp C_p$, where $Z \xleftarrow{\mathsf{U}} GL(n, \mathbb{F}_q)$, $U = (Z^{-1})^{T}$, and $\sharp C_p$ denotes the number of elements in C_p.*

We then consider the following games:

Game 0. This is the real security game from **Definition 5**.

Game 1. Game 1 is almost identical to Game 0, except that the ciphertext for challenge attribute vectors $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$, challenge revocation lists $(L^{(0)}, L^{(1)})$, and challenge plaintexts $(M^{(0)}, M^{(1)})$ is

$$\boldsymbol{c}_0 = (\delta, w, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}}, \tag{9}$$

$$\boldsymbol{c}_1 = (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{w_1^{(1)}, \ldots, w_{n_1}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \tag{10}$$

$$\forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)}): \quad \boldsymbol{c}_x = (\overbrace{\delta, \delta(-ID_x)}^{2}, \overbrace{w_{1,x}^{(2)}, w_{2,x}^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_x^{(2)}}^{1})_{\mathbb{B}^{(2)}}, \tag{11}$$

$$\boldsymbol{c}_M = g_T^{\zeta} M^{(b)}, \tag{12}$$

where $\delta, w, \zeta, \varphi, \varphi^{(1)}, \varphi_x^{(2)}, w_{1,x}^{(2)}, w_{2,x}^{(2)} \xleftarrow{\cup} \mathbb{F}_q$, $b \xleftarrow{\cup} \{0,1\}$, $\overrightarrow{y}^{(b)} = (y_1^{(b)}, \ldots, y_{n_1}^{(b)})$, and $(w_1^{(1)}, \ldots, w_{n_1}^{(1)}) \xleftarrow{\cup} \mathbb{F}_q^{n_1} \setminus \{\overrightarrow{0}\}$.

Game 2-m' $(m = 0, \ldots, \nu - 1)$. Game 2-0 is Game 1. Game 2-m' is almost identical to Game 2-m, except the reply to the $(m+1)$-th GenKey query for $\overrightarrow{x} = (x_1, \ldots, x_{n_1})$, and the challenge ciphertext are computed as follows:

$$\left.\begin{aligned} \boldsymbol{k}_0 &= (-\alpha, -\epsilon, 1, \eta, 0)_{\mathbb{B}^{*(0)}}, \\ \boldsymbol{k}_1 &= (\overbrace{\alpha^{(1)} \overrightarrow{e}_1^{(1)} + \beta^{(1)} \overrightarrow{x}}^{n_1}, \overbrace{(\gamma^{(1)} \overrightarrow{e}_1^{(1)} + \sigma^{(1)} \overrightarrow{x}) \cdot Z^{(1)}}^{n_1}, \overbrace{\eta_1^{(1)}, \ldots, \eta_{n_1}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}}, \\ \forall x \in \mathbb{P}(I): & \\ \boldsymbol{k}_x &= (\overbrace{\alpha^{(2)} + \beta_x^{(2)} ID_x, \beta_x^{(2)}}^{2}, \overbrace{(\gamma^{(2)} + \sigma_x^{(2)} ID_x, \sigma_x^{(2)}) \cdot Z^{(2)}}^{2}, \overbrace{\eta_{1,x}^{(2)}, \eta_{2,x}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}}, \end{aligned}\right\} \tag{13}$$

$$\left.\begin{aligned} \boldsymbol{c}_0 &= (\delta, w, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}}, \\ \boldsymbol{c}_1 &= (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{\overrightarrow{y}^{(b)} \cdot U^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\ \forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)}): & \\ \boldsymbol{c}_x &= (\overbrace{\delta(1, -ID_x)}^{2}, \overbrace{(1, -ID_x) \cdot U^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_x^{(2)}}^{1})_{\mathbb{B}^{(2)}}, \\ \boldsymbol{c}_M &= g_T^{\zeta} M^{(b)}, \end{aligned}\right\} \tag{14}$$

where $\epsilon, \gamma^{(1)}, \gamma^{(2)}, \sigma^{(1)}, \sigma_x^{(2)} \xleftarrow{\cup} \mathbb{F}_q$, $I \xleftarrow{\cup} \Gamma$, $Z^{(k)} \xleftarrow{\cup} GL(\mathbb{F}_q, n_k)$, $U^{(k)} = \left(Z^{(k)^{-1}}\right)^T, k = 1, 2$, and all the other variables are generated as in Game 2-m.

Game 2-$(m+1)$ $(m = 0, \ldots, \nu - 1)$. Game 2-$(m+1)$ is almost identical to Game 2-m', except the reply to the $(m+1)$-th GenKey query for $\overrightarrow{x} = (x_1, \ldots, x_{n_1})$ is

$$\left.\begin{aligned} \boldsymbol{k}_0 &= (-\alpha, \epsilon, 1, \eta, 0)_{\mathbb{B}^{*(0)}}, \\ \boldsymbol{k}_1 &= (\overbrace{\alpha^{(1)} \overrightarrow{e}_1^{(1)} + \beta^{(1)} \overrightarrow{x}}^{n_1}, \overbrace{v_1^{(1)}, \ldots, v_{n_1}^{(1)}}^{n_1}, \overbrace{\eta_1^{(1)}, \ldots, \eta_{n_1}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}}, \\ \forall x \in \mathbb{P}(I): \quad \boldsymbol{k}_x &= (\overbrace{\alpha^{(2)} + \beta_x^{(2)} ID_x, \beta_x^{(2)}}^{2}, \overbrace{v_{1,x}^{(2)}, v_{2,x}^{(2)}}^{2}, \overbrace{\eta_{1,x}^{(2)}, \eta_{2,x}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}}, \end{aligned}\right\} \tag{15}$$

the challenge cipertext is the same as Eqs.(9)-(12), where $v_{1,x}^{(2)}, v_{2,x}^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $(v_1^{(1)}, \ldots, v_{n_1}^{(1)}) \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_1} \setminus \{\overrightarrow{0}\}$, and all the other variables are generated as in Game 2-m'.

Game 3. Game 3 is almost identical to Game 2-ν, except that the target ciphertext C for challenge attribute vectors $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$, challenge revocation lists $(L^{(0)}, L^{(1)})$, and challenge plaintexts $(M^{(0)}, M^{(1)})$ is computed as follows:

$$\left.\begin{aligned} \boldsymbol{c}_0 =& (\delta, w, \zeta', 0, \varphi)_{\mathbb{B}^{(0)}}, \\ \boldsymbol{c}_1 =& (\overbrace{\overrightarrow{y}'}^{n_1}, \overbrace{w_1^{(1)}, \ldots, w_{n_1}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\ \forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)}) : \boldsymbol{c}_x =& (\overbrace{ID'_{x,1}, ID'_{x,2}}^{2}, \overbrace{w_{1,x}^{(2)}, w_{2,x}^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_x^{(2)}}^{1})_{\mathbb{B}^{(2)}}, \\ \boldsymbol{c}_M =& g_T^{\zeta} M^{(b)}. \end{aligned}\right\} \tag{16}$$

where $\zeta' \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\overrightarrow{y}' = (y'_1, \ldots, y'_{n_1}) \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_1}$, and $ID'_{x,1}, ID'_{x,2} \xleftarrow{\mathsf{U}} \mathbb{F}_q$. We note that ζ', $(y'_1, \ldots, y'_{n_1})$ and $ID'_{x,1}, ID'_{x,2}$ are chosen uniformly and independently from ζ, $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$ and ID_x, respectively.

Let $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$ be $\mathsf{Adv}_{\mathcal{A},\mathsf{FH\text{-}RPE}}^{\mathsf{FH}}(\lambda)$ in Game 0, and $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 1, 2-m, 2-m' and 3 respectively. We can show Lemmas 18 - 22 which evaluate the gaps between pairs of $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}(m+1))}(\lambda)$ for $m = 0, \ldots, \nu-1$, and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$. From these lemmas and Lemma 15 and 16 we obtain:

$$\begin{aligned} \mathsf{Adv}_{\mathcal{A},\mathsf{FH\text{-}RPE}}^{\mathsf{FH}}(\lambda) &= \mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) \\ &\leq |\, \mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda) \,| + \sum_{m=0}^{\nu-1} |\, \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda) \,| + \\ &\quad \sum_{m=0}^{\nu-1} |\, \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}(m+1))}(\lambda) \,| + |\, \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) \,| + \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) \\ &\leq \mathsf{Adv}_{\mathcal{B}_1}^{\mathrm{P3}}(\lambda) + \sum_{m=0}^{\nu-1} \mathsf{Adv}_{\mathcal{B}'_{2m}}^{\mathrm{P4}}(\lambda) + \sum_{m=0}^{\nu-1} \mathsf{Adv}_{\mathcal{B}_{2(m+1)}}^{\mathrm{P4}}(\lambda) + (2\log N\nu + 8\nu + \log N + 2)/q \\ &\leq (2\nu+1)\mathsf{Adv}_{\mathcal{D}}^{\mathrm{DLIN}}(\lambda) + (2\log N\nu + 18\nu + \log N + 10)/q. \end{aligned}$$

This completes the proof of Theorem 2. □

Lemma 18. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_1$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\, \mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda) \,| = \mathsf{Adv}_{\mathcal{B}_1}^{\mathrm{P3}}(\lambda)$.*

Proof. Suppose a polynomial time adversary $\mathcal{A}$ can successfully distinguish between Game 0 and Game 1. We construct a simulator $\mathcal{B}_1$ that uses $\mathcal{A}$ as a black box to solve Problem 3. The reduction proceeds as follows:

1. $\mathcal{B}_1$ is given an instance of Problem 3, i.e. a tuple $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{t}_{\beta}^{(0)}, \boldsymbol{t}_{\beta,1}^{(1)}, \{\boldsymbol{t}_{\beta,1,j}^{(2)}\}_{j=1,\ldots,d}, \{\boldsymbol{t}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{t}_2^{(2)})$, and plays the role of the challenger in the security game against adversary $\mathcal{A}$. Let d denote the total number of the nodes in the binary tree such that each node is associated with one element in $\{\boldsymbol{t}_{\beta,1,j}^{(2)}\}_{j=1,\ldots,d}$.
2. At the beginning of the game, $\mathcal{B}_1$ gives $\mathcal{A}$ the public key $PK = \big(1^{\lambda}, \mathsf{param}_{\overrightarrow{n}}, (\boldsymbol{b}_1^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_5^{(0)}, \boldsymbol{b}_1^{(1)}, \ldots, \boldsymbol{b}_{n_1}^{(1)}, \boldsymbol{b}_{3n_1+1}^{(1)}, \boldsymbol{b}_1^{(2)}, \boldsymbol{b}_2^{(2)}, \boldsymbol{b}_7^{(2)})$, which is obtained from the Problem 3 instance.

3. When a GenKey query is issued, $\mathcal{B}_1$ computes a normal secret key using $(\widehat{\mathbb{B}}^{*(0)}, \widehat{\mathbb{B}}^{*(1)}, \widehat{\mathbb{B}}^{*(2)})$, which is obtained from the Problem 3 instance.
4. When $\mathcal{B}_1$ receives challenge attribute vectors $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$, challenge revocation lists $(L^{(0)}, L^{(1)})$, and challenge plaintexts $(M^{(0)}, M^{(1)})$ from $\mathcal{A}$, $\mathcal{B}_1$ computes and returns

$$C = (\boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_x\}_{x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)})}, \boldsymbol{c}_M),$$

where $\boldsymbol{c}_0 = \boldsymbol{t}_\beta^{(0)} + \zeta \boldsymbol{b}_3^{(0)}$, $\boldsymbol{c}_1 = y_1^{(b)} \boldsymbol{t}_{\beta,1}^{(1)} + \sum_{i=2}^{n_1} y_i^{(b)} \boldsymbol{t}_i^{(1)}$, $\forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)}) : \boldsymbol{c}_x = \boldsymbol{t}_{\beta,1,x}^{(2)} + (-ID_x) \cdot \boldsymbol{t}_2^{(2)}$, and $\boldsymbol{c}_M = g_T^\zeta M^{(b)}$ using $(\boldsymbol{t}_\beta^{(0)}, \boldsymbol{t}_{\beta,1}^{(1)}, \{\boldsymbol{t}_{\beta,1,j}^{(2)}\}_{j=1,\ldots,d}, \{\boldsymbol{t}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{t}_2^{(2)}, \boldsymbol{b}_3^{(0)})$ from the instance of Problem 3 and $\overrightarrow{y}^{(b)}$, $L^{(b)}$, $M^{(b)}$ where $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $b \xleftarrow{\mathsf{U}} \{0,1\}$.
5. After the challenge phase, GenKey oracle simulation for a key query is executed in the same manner as step 3.
6. $\mathcal{A}$ outputs a bit b'. If $b = b'$, $\mathcal{B}_1$ outputs 1. Otherwise, $\mathcal{B}_1$ outputs 0.

Claim. If $\beta = 0$ then the challenge ciphertext $C = (\boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_x\}_{x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)})}, \boldsymbol{c}_M)$ generated in step 4 is distributed exactly as in Game 0. If $\beta = 1$ then the challenge ciphertext $C = (\boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_x\}_{x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)})}, \boldsymbol{c}_M)$ generated in step 4 is identically distributed to Game 1.

Proof. First recall that $y_1^{(b)} = 1$. If $\beta = 0$ then the ciphertext given by

$$\begin{aligned}
\boldsymbol{c}_0 &= (\delta, 0, \zeta, 0, \rho)_{\mathbb{B}^{(0)}}, \\
\boldsymbol{c}_1 &= (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\rho^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\
\forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)}) : \boldsymbol{c}_x &= (\overbrace{\delta, \delta(-ID_x)}^{2}, \overbrace{0^2}^{2}, \overbrace{0^2}^{2}, \overbrace{\rho_x^{(2)}}^{1})_{\mathbb{B}^{(2)}}, \\
\boldsymbol{c}_M &= g_T^\zeta M^{(b)},
\end{aligned}$$

is the challenge ciphertext from Game 0. In contrast, if $\beta = 1$ then the following components of the ciphertext have a different form

$$\begin{aligned}
\boldsymbol{c}_0 &= (\delta, u, \zeta, 0, \rho)_{\mathbb{B}^{(0)}}, \\
\boldsymbol{c}_1 &= (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{\overrightarrow{u}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\rho^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\
\forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)}) : \boldsymbol{c}_x &= (\overbrace{\delta, \delta(-ID_x)}^{2}, \overbrace{\overrightarrow{u}_x^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\rho_x^{(2)}}^{1})_{\mathbb{B}^{(2)}},
\end{aligned}$$

where $\overrightarrow{u}^{(1)} = (u_1^{(1)}, \ldots, u_{n_1}^{(1)})$, $\overrightarrow{u}_x^{(2)} = (u_{1,x}^{(2)}, u_{2,x}^{(2)})$. Since $\overrightarrow{u}^{(1)} \in \mathbb{F}_q^{n_1}$, $\overrightarrow{u}_x^{(2)} \in \mathbb{F}_q^2$, $\rho^{(1)}, \rho_x^{(2)} \in \mathbb{F}_q$ are independently uniform this ciphertext corresponds to the challenge ciphertext from Game 1.

From the above claim, if $\beta = 0$ then simulated ciphertexts are distributed exactly as in Game 0, whereas for $\beta = 1$ their distribution is identical to Game 1. Therefore, $|\ \mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)\ | = \Big|\ \mathsf{Pr}\Big[\mathcal{B}_1(1^\lambda, \varpi) \to 1 \ \Big|\ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathrm{P3}}(1^\lambda, \overrightarrow{n})\Big] - \mathsf{Pr}\Big[\mathcal{B}_1(1^\lambda, \varpi) \to 1 \ \Big|\ \varpi \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathrm{P3}}(1^\lambda, \overrightarrow{n})\Big]\ \Big| = \mathsf{Adv}_{\mathcal{B}_1}^{\mathrm{P3}}(\lambda)$. This completes the proof of **Lemma 18**. □

Lemma 19. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}'_{2m}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\ \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda)\ | \leq \mathsf{Adv}_{\mathcal{B}'_{2m}}^{\mathrm{P4}}(\lambda) + (4 + \log N)/q$.*

Proof. Suppose a polynomial time adversary $\mathcal{A}$ can successfully distinguish between Game 2-m and Game 2-m'. We construct a simulator $\mathcal{B}'_{2m}$ that uses $\mathcal{A}$ as a black box to solve Problem 4. The reduction proceeds as follows:

1. $\mathcal{B}'_{2m}$ is given an instance of Problem 4, i.e. a tuple $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{h}^{*(0)}_{\beta}, \boldsymbol{t}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{h}^{*(k)}_{\beta,i}, \boldsymbol{t}^{(k)}_i\}_{i=1,\ldots,n_k}\}_{k=1,2})$, and acts as a challenger in the security game against adversary $\mathcal{A}$.
2. At the beginning of the game, $\mathcal{B}'_{2m}$ gives $\mathcal{A}$ the public key $PK = \big(1^{\lambda}, \mathsf{param}_{\overrightarrow{n}}, (\boldsymbol{b}^{(0)}_1, \boldsymbol{b}^{(0)}_3, \boldsymbol{b}^{(0)}_5, \boldsymbol{b}^{(1)}_1, \ldots, \boldsymbol{b}^{(1)}_{n_1}, \boldsymbol{b}^{(1)}_{3n_1+1}, \boldsymbol{b}^{(2)}_1, \boldsymbol{b}^{(2)}_2, \boldsymbol{b}^{(2)}_7)$, which is obtained from the Problem 4 instance.
3. When the s-th GenKey query is issued for a predicate $\overrightarrow{x} = (x_1, \ldots, x_{n_1})$, $\mathcal{B}'_{2m}$ answers as follows:
   a) For $1 \le s \le m$ the algorithm $\mathcal{B}'_{2m}$ computes a semi-functional key using $\{\mathbb{B}^{*(k)}\}_{k=0,1,2}$ of the Problem 4 instance.
   b) For $s = m+1$ it computes the key $\boldsymbol{k}^{*}_{\overrightarrow{x},I} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \{\boldsymbol{k}_x\}_{x\in\mathbb{P}(I)})$ using $\{\boldsymbol{h}^{*(0)}_{\beta}, \boldsymbol{b}^{*(0)}_1, \boldsymbol{b}^{*(0)}_3, \{\boldsymbol{h}^{*(i)}_{\beta,j}, \boldsymbol{b}^{*(i)}_j\}_{i=1,2;j=1,\ldots,n_i}\}$ of the Problem 4 instance as follows:

$$
\begin{aligned}
&\text{For } i = 1, 2: \ \varrho_i, v_i, v'_i, \theta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q;\\
&\boldsymbol{s}^{(0)}_{\beta} = \sum_{i=1}^{2} (\varrho_i \boldsymbol{h}^{*(0)}_{\beta} + v_i \boldsymbol{b}^{*(0)}_1), \quad \boldsymbol{k}_0 = -\boldsymbol{s}^{(0)}_{\beta} + \boldsymbol{b}^{*(0)}_3,\\
&\text{For } i = 1, 2 \text{ and } j = 1, \ldots, n_i:\\
&\quad \boldsymbol{s}^{(i)}_{\beta,j} = \theta_i \boldsymbol{h}^{*(i)}_{\beta,j} + v'_i \boldsymbol{b}^{*(i)}_j, \quad \widehat{\boldsymbol{s}}^{(i)}_{\beta,j} = \varrho_i \boldsymbol{h}^{*(i)}_{\beta,j} + v_i \boldsymbol{b}^{*(i)}_j,\\
&\boldsymbol{k}_1 = \sum_{j=1}^{n_1} x_j \boldsymbol{s}^{(1)}_{\beta,j} + \widehat{\boldsymbol{s}}^{(1)}_{\beta,1};\\
&\forall x \in \mathbb{P}(I): \quad \boldsymbol{k}_x = ID_x \boldsymbol{s}^{(2)}_{\beta,1} + \boldsymbol{s}^{(2)}_{\beta,2} + \widehat{\boldsymbol{s}}^{(2)}_{\beta,1},
\end{aligned}
$$

   c) For $s \ge m+2$ the algorithm $\mathcal{B}'_{2m}$ computes a normal key using $\{\mathbb{B}^{*(k)}\}_{k=0,1,2}$ from the instance of Problem 4.
4. Once $\mathcal{B}'_{2m}$ receives challenge attribute vectors $(\overrightarrow{y}^{(0)}, \overrightarrow{y}^{(1)})$, revocation lists $(L^{(0)}, L^{(1)})$, and plaintexts $(M^{(0)}, M^{(1)})$ from $\mathcal{A}$ it returns the challenge ciphertext $C = (\boldsymbol{c}_0, \boldsymbol{c}_1, \{\boldsymbol{c}_x\}_{x\in\mathsf{RevokeNodes}(Tree, L^{(b)})}, \boldsymbol{c}_M)$ where $\boldsymbol{c}_0 = \boldsymbol{t}^{(0)} + \zeta \boldsymbol{b}^{(0)}_3 + \varphi \boldsymbol{b}^{(0)}_5$, $\boldsymbol{c}_1 = \sum_{j=1}^{n_1} y^{(b)}_j \boldsymbol{t}^{(1)}_j + \varphi^{(1)} \boldsymbol{b}^{(1)}_{3n_1+1}$, $\forall x \in \mathsf{RevokeNodes}(Tree, L^{(b)}) : \boldsymbol{c}_x = \boldsymbol{t}^{(2)}_1 + (-ID_x)\boldsymbol{t}^{(2)}_2 + \varphi^{(2)}_x \boldsymbol{b}^{(2)}_7$ and $\boldsymbol{c}_M = g_T^{\zeta} M^{(b)}$, using $(\boldsymbol{t}^{(0)}, \{\boldsymbol{t}^{(1)}_i\}_{i=1,\ldots,n_1}, \boldsymbol{t}^{(2)}_1, \boldsymbol{t}^{(2)}_2, \boldsymbol{b}^{(0)}_3, \boldsymbol{b}^{(0)}_5, \boldsymbol{b}^{(1)}_{3n_1+1}, \boldsymbol{b}^{(2)}_7)$ from the instance of Problem 4 and $\overrightarrow{y}^{(b)}$, $L^{(b)}$, $M^{(b)}$, where $\zeta, \varphi, \varphi^{(1)}, \varphi^{(2)}_x \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $b \xleftarrow{\mathsf{U}} \{0,1\}$.
5. After the challenge phase, GenKey oracle simulation for a key query is executed in the same manner as step 3.
6. $\mathcal{A}$ outputs a bit b'. If $b = b'$, $\mathcal{B}'_{2m}$ outputs 1. Otherwise, $\mathcal{B}'_{2m}$ outputs 0.

Claim. The distribution of the view of adversary $\mathcal{A}$ in the above-mentioned game simulated by $\mathcal{B}'_{2m}$ given a Problem 4 instance is the same as that in Game (2-m) (resp. Game (2-m')) if $\beta = 0$ (resp. $\beta = 1$) except with probability $(3 + \log N)/q$ (resp. $1/q$).

Proof. It is clear that $\mathcal{B}'_{2m}$'s simulation of the public key generation (step 2) and the answers to the i-th GenKey query where $i \neq m+1$ (case (a) and (c) of steps (3) and (5)) are exactly the same as the Setup and the GenKey oracles in Game 2-m and Game 2-m'.

Next we analyze the distribution of the i-th GenKey query where $i = m+1$ (case (b) of steps (3) and (5)). In this case values $\boldsymbol{s}^{(0)}_{\beta}, \boldsymbol{s}^{(i)}_{\beta,j}, \widehat{\boldsymbol{s}}^{(i)}_{\beta,j}$, $i = 1, 2$, $j = 1, \ldots, n_i$ can be expressed as follows. Let $\beta^{(i)} = \theta_i \omega + v'_i$, $\alpha^{(i)} = \varrho_i \omega + v_i$, $\alpha = \alpha^{(1)} + \alpha^{(2)}$, $\gamma = \varrho_1 + \varrho_2$, $\epsilon = \gamma z$. Then,

$$
\boldsymbol{s}^{(0)}_0 = (\alpha, 0, 0, \gamma\xi, 0)_{\mathbb{B}^{*(0)}}, \qquad \boldsymbol{s}^{(0)}_1 = (\alpha, \epsilon, 0, \gamma\xi, 0)_{\mathbb{B}^{*(0)}},
$$

$$\boldsymbol{s}_{0,j}^{(i)} = (\overbrace{\beta^{(i)}\overrightarrow{e}_j^{(i)}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\theta_i\overrightarrow{\omega}_j^{(i)}}^{n_i}, \overbrace{0}^{1})_{\mathbb{B}^{*(i)}}, \quad \boldsymbol{s}_{1,j}^{(i)} = (\overbrace{\beta^{(i)}\overrightarrow{e}_j^{(i)}}^{n_i}, \overbrace{\theta_i\overrightarrow{z}_j^{(i)}}^{n_i}, \overbrace{\theta_i\overrightarrow{\omega}_j^{(i)}}^{n_i}, \overbrace{0}^{1})_{\mathbb{B}^{*(i)}},$$
$$\widehat{\boldsymbol{s}}_{0,j}^{(i)} = (\overbrace{\alpha^{(i)}\overrightarrow{e}_j^{(i)}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\varrho_i\overrightarrow{\omega}_j^{(i)}}^{n_i}, \overbrace{0}^{1})_{\mathbb{B}^{*(i)}}, \quad \widehat{\boldsymbol{s}}_{1,j}^{(i)} = (\overbrace{\alpha^{(i)}\overrightarrow{e}_j^{(i)}}^{n_i}, \overbrace{\varrho_i\overrightarrow{z}_j^{(i)}}^{n_i}, \overbrace{\varrho_i\overrightarrow{\omega}_j^{(i)}}^{n_i}, \overbrace{0}^{1})_{\mathbb{B}^{*(i)}},$$

where $\overrightarrow{z}_j^{(i)} = z_{j,1}^{(i)}, \ldots, z_{j,n_i}^{(i)}$, $\omega, z, \xi, \{\overrightarrow{\omega}_j^{(i)}, \overrightarrow{z}_j^{(i)}\}_{i=1,2;j=1,\ldots,n_i}$ are defined as in Problem 4. If $\beta = 1$ in the instance of Problem 4 then the secret key $\boldsymbol{k}_{\overrightarrow{x},I}^{*} = (I, \boldsymbol{k}_0, \boldsymbol{k}_1, \{\boldsymbol{k}_x\}_{x\in\mathbb{P}(I)})$ has the same distribution as in Eq. 13, except that $\epsilon w = \gamma$, where $\gamma = \varrho_1 + \varrho_2$ and $w = u \xleftarrow{\mathsf{U}} \mathbb{F}_q$ of $\boldsymbol{c}_0$ in Eq. 14.

Next, we show that the joint distribution of the response to $(m+1)$-th GenKey query and of the challenge ciphertext in the simulation by $\mathcal{B}'_{2m}$ for the given instance of Problem 4 is equivalent to the distribution in Game 2-m if $\beta = 0$ and to the distribution in Game 2-m' if $\beta = 1$.

If $\beta = 0$ then this equivalence follows easily, unless one of the following conditions holds: (1) ω defined in Problem 4 is zero, (2) $w = 0$, (3) $(w_1^{(1)}, \ldots, w_{n_1}^{(1)}) = \overrightarrow{0}$, (4) $(w_{1,x}^{(2)}, w_{2,x}^{(2)}) = \overrightarrow{0}$, where $x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)})$, and w, $(w_1^{(1)}, \ldots, w_{n_1}^{(1)})$ and $(w_{1,x}^{(2)}, w_{2,x}^{(2)})$ are defined in Eqs. 9, 10 and 11 respectively. However, those events occur with probability $(3 + \log N)/q$.

If $\beta = 1$, then $\mathcal{B}'_{2m}$'s simulation for the key is the same as that in Eq.13 and $\mathcal{B}'_{2m}$'s simulation for the challenge ciphertext is the same as that in Eq.14, except that $\epsilon w = \gamma$, where $\gamma = \varrho_1 + \varrho_2$, and $w \xleftarrow{\mathsf{U}} \mathbb{F}_q$ of $\boldsymbol{c}_0$ in Eq. 14.

Therefore, we will show that γ is uniformly distributed and is independent from the other variables used in the simulation by $\mathcal{B}'_{2m}$. Since γ is related to $\overrightarrow{A}_1, \overrightarrow{A}_{2,x}, \overrightarrow{B}_1^{(b)}$, and $\overrightarrow{B}_{2,x}^{(b)}$, where $\overrightarrow{A}_1 = (\varrho_1\overrightarrow{e}_1^{(1)} + \theta_1\overrightarrow{x}) \cdot Z^{(1)}$, $\overrightarrow{A}_{2,x} = (\varrho_2\overrightarrow{e}_1^{(2)} + \theta_2(ID_x, 1)) \cdot Z^{(2)}$ $x \in \mathbb{P}(I)$, and $\overrightarrow{B}_1^{(b)} = \overrightarrow{y}^{(b)} \cdot U^{(1)}$, $\overrightarrow{B}_{2,x}^{(b)} = (1, -ID_x^{(b)}) \cdot U^{(2)}$ $x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)})$, where $b \in \{0,1\}$. We analyze joint distribution of these variables for the four distinct cases that appear in **Definition 5**.

1. When $f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 0$, due to Lemma 17, The pair $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(b)})$ $(b \in \{0,1\})$ is uniformly and independently distributed over $C_{\theta_1\cdot(\overrightarrow{x}\cdot\overrightarrow{y}^{(b)})+\varrho_1}$ $(b \in \{0,1\})$ (over $Z^{(1)} \xleftarrow{\mathsf{U}} GL(\mathbb{F}_q, n_1)$). Since $\theta_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, the pair $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(b)})$ $(b \in \{0,1\})$ is thus uniformly and independently distributed over $\mathbb{F}_q^{2n_1}$.
2. When $f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 1$ and $(I \in L^{(0)} \wedge I \in L^{(1)})$, the pair $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(b)})$ $(b \in \{0,1\})$ is uniformly and independently distributed over C_{ϱ_1} (over $Z^{(1)} \xleftarrow{\mathsf{U}} GL(\mathbb{F}_q, n_1)$). The pair $(\overrightarrow{A}_{2,x}, \overrightarrow{B}_{2,x}^{(b)})$ $(b \in \{0,1\})$ is uniformly and independently distributed over $\mathbb{F}_q^4$.
3. When $(f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = 1 \wedge f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 0)$ and $I \in L^{(0)}$, the pair $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(0)})$ (resp. $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(1)})$) is uniformly and independently distributed over C_{ϱ_1} (resp. $\mathbb{F}_q^{2n_1}$). The pair $(\overrightarrow{A}_{2,x}, \overrightarrow{B}_{2,x}^{(0)})$ is uniformly and independently distributed over $\mathbb{F}_q^4$.
4. When $(f_{\overrightarrow{x}}(\overrightarrow{y}^{(0)}) = 0 \wedge f_{\overrightarrow{x}}(\overrightarrow{y}^{(1)}) = 1)$ and $I \in L^{(1)}$, the pair $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(0)})$ (resp. $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(1)})$) is uniformly and independently distributed over $\mathbb{F}_q^{2n_1}$ (resp. C_{ϱ_1}). The pair $(\overrightarrow{A}_{2,x}, \overrightarrow{B}_{2,x}^{(1)})$ is uniformly and independently distributed over $\mathbb{F}_q^4$.

Considering the adversary $\mathcal{A}$'s restriction on key queries from **Definition 5**, in each of the above four cases at least one of $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(b)})$ and $(\overrightarrow{A}_{2,x}, \overrightarrow{B}_{2,x}^{(b)})$ is uniformly and independently distributed over $\mathbb{F}_q^{2n_k}$ for $k = 1, 2$. Therefore, $\gamma = \varrho_1 + \varrho_2$ is independent from the distribution of ϱ_1(resp. ϱ_2), which can be given by $(\overrightarrow{A}_1, \overrightarrow{B}_1^{(b)})$ (resp. $(\overrightarrow{A}_{2,x}, \overrightarrow{B}_{2,x}^{(b)})$). Thus, γ is uniformly and independently distributed from the other variables in the simulation of $\mathcal{B}'_{2m}$.

Therefore, the view of $\mathcal{A}$ in the game simulated by $\mathcal{B}'_{2m}$ on input an instance of Problem 4 with $\beta = 1$ is the same as in Game 2-m' unless $\omega = 0$ occurs. This event happens with probability $1/q$.

This completes the proof of **Lemma 19**. □

Lemma 20. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_{2(m+1)}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\ \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}m')}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}(m+1))}(\lambda)\ | \leq \mathsf{Adv}_{\mathcal{B}_{2(m+1)}}^{\mathrm{P4}}(\lambda) + (4+\log N)/q$.*

Proof. Suppose a polynomial time adversary $\mathcal{A}$ can successfully distinguish between Game 2-m' and Game 2-$(m+1)$. We construct a simulator $\mathcal{B}_{2(m+1)}$ that leverages $\mathcal{A}$ as a black box to solve Problem 4. The procedure is same as that in the proof of **Lemma 19** except that in case (b) of step 3 $\boldsymbol{k}^*_{\overrightarrow{x},I}$ is computed as follows:

$$\boldsymbol{k}_0 = -\boldsymbol{s}^{(0)}_{\beta} + \epsilon' \boldsymbol{b}^{*(0)}_2 + \boldsymbol{b}^{*(0)}_3,$$
$$\boldsymbol{k}_1 = \sum_{j=1}^{n_1} x_j \boldsymbol{s}^{(1)}_{\beta,j} + \widehat{\boldsymbol{s}}^{(1)}_{\beta,1} + \sum_{j=1}^{n_1} v^{(1)}_j \boldsymbol{b}^{*(1)}_{n_1+j},$$
$$\forall x \in \mathbb{P}(I) : \boldsymbol{k}_x = ID_x \boldsymbol{s}^{(2)}_{\beta,1} + \boldsymbol{s}^{(2)}_{\beta,2} + \widehat{\boldsymbol{s}}^{(2)}_{\beta,1} + \sum_{j=1}^{2} v^{(2)}_{j,x} \boldsymbol{b}^{*(2)}_{2+j},$$

where $\epsilon' \xleftarrow{\cup} \mathbb{F}_q$, $v^{(2)}_{1,x}, v^{(2)}_{2,x} \xleftarrow{\cup} \mathbb{F}_q$, $(v^{(1)}_1, \ldots, v^{(1)}_{n_1}) \xleftarrow{\cup} \mathbb{F}^{n_1}_q \setminus \{\overrightarrow{0}\}$. In the last step, if $b = b'$, $\mathcal{B}_{2(m+1)}$ outputs 0. Otherwise, $\mathcal{B}_{2(m+1)}$ outputs 1.

The view of the adversary $\mathcal{A}$ in the game simulated by $\mathcal{B}_{2(m+1)}$ given an instance of Problem 4 with $\beta = 0$ is the same as in Game 2-$(m+1)$ unless one of following events occur: (1) $\omega = 0$ in Problem 4 instance, (2) $w = 0$, (3) $(w^{(1)}_1, \ldots, w^{(1)}_{n_1}) = \overrightarrow{0}$, (4) $(w^{(2)}_{1,x}, w^{(2)}_{2,x}) = \overrightarrow{0}$, where $x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)})$, and w, $(w^{(1)}_1, \ldots, w^{(1)}_{n_1})$ and $(w^{(2)}_{1,x}, w^{(2)}_{2,x})$ are defined in Eqs. 9, 10 and 11 respectively. Those events occur with probability $(3+\log N)/q$. In case $\beta = 1$ the argument is similar to that in the proof of **Lemma 19**, i.e., each variable has uniform distribution and is independent from other variables occurring in the simulation by $\mathcal{B}_{2(m+1)}$. The view of the adversary $\mathcal{A}$ is the same as its view in Game 2-m' unless $\omega = 0$ occurs in the instance of Problem 4. The event $\omega = 0$ occurs with probability $1/q$.

Lemma 21. *For any adversary $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) \leq \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu)}(\lambda) + (2+\log N)/q$.*

Proof. First we show the distribution $(\mathsf{param}_{\overrightarrow{n}}, \{\widehat{\mathbb{B}}^{(k)}\}_{k=0,1,2}, \{\boldsymbol{k}^{*(j)}_{\overrightarrow{x},I}\}_{j=1,\ldots,\nu}, C)$ of Game 3 is same as that of Game 2-ν, where $\boldsymbol{k}^{*(j)}_{\overrightarrow{x},I}$ is the answer to the j-th key query, and C is the challenge ciphertext. We will define new bases $\mathbb{D}^{(k)}$ of $\mathbb{V}_k$ and $\mathbb{D}^{*(k)}$ of $\mathbb{V}_k$, $k = 0, 1, 2$.

For $k = 0$, we set $\boldsymbol{d}^{(0)}_2 = \boldsymbol{b}^{(0)}_2 - \lambda \boldsymbol{b}^{(0)}_3$ and $\boldsymbol{d}^{*(0)}_3 = \boldsymbol{b}^{*(0)}_3 + \lambda \boldsymbol{b}^{*(0)}_2$, where $\lambda \xleftarrow{\cup} \mathbb{F}_q$. The new bases are $\mathbb{D}^{(0)} = (\boldsymbol{b}^{(0)}_1, \boldsymbol{d}^{(0)}_2, \boldsymbol{b}^{(0)}_3, \boldsymbol{b}^{(0)}_4, \boldsymbol{b}^{(0)}_5)$ and $\mathbb{D}^{*(0)} = (\boldsymbol{b}^{*(0)}_1, \boldsymbol{b}^{*(0)}_2, \boldsymbol{d}^{*(0)}_3, \boldsymbol{b}^{*(0)}_4, \boldsymbol{b}^{*(0)}_5)$. We can easily verify that $\mathbb{D}^{(0)}$ and $\mathbb{D}^{*(0)}$ are dual orthonormal, and are distributed the same as the original bases $\mathbb{B}^{(0)}$ and $\mathbb{B}^{*(0)}$ respectively.

For $i = 1, \ldots, n_k$, $j = 1, \ldots, n_k$, $k = 1, 2$, choose $Q^{(k)} = (\mu^{(k)}_{i,j}) \xleftarrow{\cup} \mathbb{F}^{n_k \times n_k}_q$, and compute $\boldsymbol{d}^{(k)}_{n_k+i} = \boldsymbol{b}^{(k)}_{n_k+i} + \sum_{j=1}^{n_k} \mu^{(k)}_{i,j} \boldsymbol{b}^{(k)}_j$, $\boldsymbol{d}^{*(k)}_i = \boldsymbol{b}^{*(k)}_i - \sum_{j=1}^{n_k} \mu^{(k)}_{j,i} \boldsymbol{b}^{*(k)}_{n_k+j}$, which are equivalent to the following matrix computations:

$$\begin{pmatrix} \overrightarrow{B}^{(k)}_1 \\ \overrightarrow{D}^{(k)}_2 \end{pmatrix} = \begin{pmatrix} I_{n_k} & 0_{n_k} \\ Q^{(k)} & I_{n_k} \end{pmatrix} \begin{pmatrix} \overrightarrow{B}^{(k)}_1 \\ \overrightarrow{B}^{(k)}_2 \end{pmatrix}, \qquad \begin{pmatrix} \overrightarrow{D}^{*(k)}_1 \\ \overrightarrow{B}^{*(k)}_2 \end{pmatrix} = \begin{pmatrix} I_{n_k} & -Q^{\mathrm{T}(k)} \\ 0_{n_k} & I_{n_k} \end{pmatrix} \begin{pmatrix} \overrightarrow{B}^{*(k)}_1 \\ \overrightarrow{B}^{*(k)}_2 \end{pmatrix},$$

where $\overrightarrow{B}^{(k)}_1 = (\boldsymbol{b}^{(k)}_1, \ldots, \boldsymbol{b}^{(k)}_{n_k})^{\mathrm{T}}$, $\overrightarrow{B}^{(k)}_2 = (\boldsymbol{b}^{(k)}_{n_k+1}, \ldots, \boldsymbol{b}^{(k)}_{2n_k})^{\mathrm{T}}$, $\overrightarrow{B}^{*(k)}_1 = (\boldsymbol{b}^{*(k)}_1, \ldots, \boldsymbol{b}^{*(k)}_{n_k})^{\mathrm{T}}$, $\overrightarrow{B}^{*(k)}_2 = (\boldsymbol{b}^{*(k)}_{n_k+1}, \ldots, \boldsymbol{b}^{*(k)}_{2n_k})^{\mathrm{T}}$, $\overrightarrow{D}^{(k)}_2 = (\boldsymbol{d}^{(k)}_{n_k+1}, \ldots, \boldsymbol{d}^{(k)}_{2n_k})^{\mathrm{T}}$, $\overrightarrow{D}^{*(k)}_1 = (\boldsymbol{d}^{*(k)}_1, \ldots, \boldsymbol{d}^{*(k)}_{n_k})^{\mathrm{T}}$.

For $k = 1, 2$, the new bases are $\mathbb{D}^{(k)} = (\boldsymbol{b}^{(k)}_1, \ldots, \boldsymbol{b}^{(k)}_{n_k}, \boldsymbol{d}^{(k)}_{n_k+1}, \ldots, \boldsymbol{d}^{(k)}_{2n_k}, \boldsymbol{b}^{(k)}_{2n_k+1}, \ldots, \boldsymbol{b}^{(k)}_{3n_k+1})$ and $\mathbb{D}^{*(k)} = (\boldsymbol{d}^{*(k)}_1, \ldots, \boldsymbol{d}^{*(k)}_{n_k}, \boldsymbol{b}^{*(k)}_{n_k+1}, \ldots, \boldsymbol{b}^{*(k)}_{2n_k}, \boldsymbol{b}^{*(k)}_{2n_k+1}, \ldots, \boldsymbol{b}^{*(k)}_{3n_k+1})$. It is clear that $\mathbb{D}^{(k)}$ and $\mathbb{D}^{*(k)}$ are dual orthonormal, and are distributed the same as the original bases $\mathbb{B}^{(k)}$ and $\mathbb{B}^{*(k)}$ respectively.

The secret keys and challenge ciphertext $(\{\boldsymbol{k}^{*(j)}_{\overrightarrow{x},I}\}_{j=1,\ldots,\nu}, C)$ in Game 2-ν are expressed over the bases $\mathbb{B}^{(k)}$ and $\mathbb{B}^{*(k)}$, $k = 0, 1, 2$ as follows:

$$
\begin{aligned}
\boldsymbol{k}_{0,j} &= (-\alpha_j, \epsilon_j, 1, \eta_j, 0)_{\mathbb{B}^{*(0)}}, \\
\boldsymbol{k}_{1,j} &= (\overbrace{\alpha_j^{(1)}\overrightarrow{e}_1^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j}^{n_1}, \overbrace{\gamma_{1,j}^{(1)}, \ldots, \gamma_{n_1,j}^{(1)}}^{n_1}, \overbrace{\eta_{1,j}^{(1)}, \ldots, \eta_{n_1,j}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}}, \\
\forall x \in \mathbb{P}(I): \quad \boldsymbol{k}_{x,j} &= (\overbrace{\alpha_j^{(2)} + \beta_{x,j}^{(2)} ID_{x,j}, \beta_{x,j}^{(2)}}^{2}, \overbrace{\gamma_{1,x,j}^{(2)}, \gamma_{2,x,j}^{(2)}}^{2}, \overbrace{\eta_{1,x,j}^{(2)}, \eta_{2,x,j}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}}, \\
\\
\boldsymbol{c}_0 &= (\delta, w, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}}, \\
\boldsymbol{c}_1 &= (\overbrace{\delta\overrightarrow{y}^{(b)}}^{n_1}, \overbrace{w_1^{(1)}, \ldots, w_{n_1}^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\
\forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)}): \boldsymbol{c}_x &= (\overbrace{\delta(1, -ID_x)}^{2}, \overbrace{w_{1,x}^{(2)}, w_{2,x}^{(2)}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi_x^{(2)}}^{1})_{\mathbb{B}^{(2)}}, \\
\boldsymbol{c}_M &= g_T^{\zeta} M^{(b)}.
\end{aligned}
$$

The above keys and challenge ciphertext can also be expressed over bases $\mathbb{D}^{(k)}$ and $\mathbb{D}^{*(k)}$, $k = 0, 1, 2$ as specified in the following. The first components of secret keys can be expressed as $\boldsymbol{k}_{0,j} = (-\alpha_j, \epsilon_j, 1, \eta_j, 0)_{\mathbb{B}^{*(0)}} = (-\alpha_j, \theta_j, 1, \eta_j, 0)_{\mathbb{D}^{*(0)}}$, where $\theta_j = \epsilon_j - \lambda$ are uniform and independent since $\epsilon_j \xleftarrow{\mathsf{U}} \mathbb{F}_q$. Similarly, other key components can be represented as

$$
\begin{aligned}
\boldsymbol{k}_{1,j} &= (\overbrace{\alpha_j^{(1)}\overrightarrow{e}_1^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j}^{n_1}, \overbrace{\gamma_{1,j}^{(1)}, \ldots, \gamma_{n_1,j}^{(1)}}^{n_1}, \overbrace{\eta_{1,j}^{(1)}, \ldots, \eta_{n_1,j}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{B}^{*(1)}} \\
&= (\overbrace{\alpha_j^{(1)}\overrightarrow{e}_1^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j}^{n_1}, \overbrace{\mu_{1,1}^{(1)}\alpha_j^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j \cdot \overrightarrow{\mu}_1^{(1)} + \gamma_{1,j}^{(1)}, \ldots, \mu_{n_1,1}^{(1)}\alpha_j^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j \cdot \overrightarrow{\mu}_{n_1}^{(1)} + \gamma_{n_1,j}^{(1)}}^{n_1}, \\
&\quad \overbrace{\eta_{1,j}^{(1)}, \ldots, \eta_{n_1,j}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{D}^{*(1)}} \\
&= (\overbrace{\alpha_j^{(1)}\overrightarrow{e}_1^{(1)} + \beta_j^{(1)}\overrightarrow{x}_j}^{n_1}, \overbrace{\theta_{1,j}^{(1)}, \ldots, \theta_{n_1,j}^{(1)}}^{n_1}, \overbrace{\eta_{1,j}^{(1)}, \ldots, \eta_{n_1,j}^{(1)}}^{n_1}, \overbrace{0}^{1})_{\mathbb{D}^{*(1)}},
\end{aligned}
$$

$$
\begin{aligned}
\forall x \in \mathbb{P}(I) \quad : \\
\boldsymbol{k}_{x,j} &= (\overbrace{\alpha_j^{(2)} + \beta_{x,j}^{(2)} ID_{x,j}, \beta_{x,j}^{(2)}}^{2}, \overbrace{\gamma_{1,x,j}^{(2)}, \gamma_{2,x,j}^{(2)}}^{2}, \overbrace{\eta_{1,x,j}^{(2)}, \eta_{2,x,j}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^{*(2)}} \\
&= (\overbrace{\alpha_j^{(2)} + \beta_{x,j}^{(2)} ID_{x,j}, \beta_{x,j}^{(2)}}^{2}, \\
&\quad \overbrace{\mu_{1,1}^{(2)}\alpha_j^{(2)} + \beta_j^{(2)}(ID_{x,j}\mu_{1,1}^{(2)} + \mu_{1,2}^{(2)}) + \gamma_{1,x,j}^{(2)}, \mu_{2,1}^{(2)}\alpha_j^{(2)} + \beta_j^{(2)}(ID_{x,j}\mu_{2,1}^{(2)} + \mu_{2,2}^{(2)}) + \gamma_{2,x,j}^{(2)}}^{2}, \\
&\quad \overbrace{\eta_{1,x,j}^{(2)}, \eta_{2,x,j}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{D}^{*(2)}} \\
&= (\overbrace{\alpha_j^{(2)} + \beta_{x,j}^{(2)} ID_{x,j}, \beta_{x,j}^{(2)}}^{2}, \overbrace{\theta_{1,x,j}^{(2)}, \theta_{2,x,j}^{(2)}}^{2}, \overbrace{\eta_{1,x,j}^{(2)}, \eta_{2,x,j}^{(2)}}^{2}, \overbrace{0}^{1})_{\mathbb{D}^{*(2)}},
\end{aligned}
$$

where $\theta^{(1)}_{i,j} = \mu^{(1)}_{i,1}\alpha^{(1)}_j + \beta^{(1)}_j \overrightarrow{x}_j \cdot \overrightarrow{\mu}^{(1)}_i + \gamma^{(1)}_{i,j}$ for $i = 1, \ldots, n_1$ and $\theta^{(2)}_{i,x,j} = \mu^{(2)}_{i,1}\alpha^{(2)}_j + \beta^{(2)}_j (ID_{x,j}\mu^{(2)}_{i,1} + \mu^{(2)}_{i,2}) + \gamma^{(2)}_{i,x,j}$ for $i = 1, 2$ and $j = 1, \ldots, \nu$ are uniform and independent since $\gamma^{(k)}_{i,x,j} \xleftarrow{\mathsf{U}} \mathbb{F}_q$.

The first component of the ciphertext can be expressed as $\boldsymbol{c}_0 = (\delta, w, \zeta, 0, \varphi)_{\mathbb{B}^{(0)}} = (\delta, w, \zeta', 0, \varphi)_{\mathbb{D}^{(0)}}$, where $\zeta' = \zeta + \lambda w$ is uniform and independent due to $w, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$. Similarly, the remaining components of the ciphertext can be expressed as

$$
\begin{aligned}
\boldsymbol{c}_1 &= (\overbrace{\delta \overrightarrow{y}^{(b)}}^{n_1}, \overbrace{w^{(1)}_1, \ldots, w^{(1)}_{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{B}^{(1)}}, \\
&= (\overbrace{\overrightarrow{y}'}^{n_1}, \overbrace{w^{(1)}_1, \ldots, w^{(1)}_{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\varphi^{(1)}}^{1})_{\mathbb{D}^{(1)}}, \\
\forall x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)}) : \boldsymbol{c}_x &= (\overbrace{\delta(1, -ID_x)}^{2}, \overbrace{w^{(2)}_{1,x}, w^{(2)}_{2,x}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi^{(2)}_x}^{1})_{\mathbb{B}^{(2)}} \\
&= (\overbrace{\delta - \sum_{j=1}^{2} w^{(2)}_{j,x}\mu^{(2)}_{j,1}, -\delta ID_x - \sum_{j=1}^{2} w^{(2)}_{j,x}\mu^{(2)}_{j,2}}^{2}, \\
&\quad \overbrace{w^{(2)}_{1,x}, w^{(2)}_{2,x}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi^{(2)}_x}^{1})_{\mathbb{D}^{(2)}}, \\
&= (\overbrace{ID'_{x,1}, ID'_{x,2}}^{2}, \overbrace{w^{(2)}_{1,x}, w^{(2)}_{2,x}}^{2}, \overbrace{0^2}^{2}, \overbrace{\varphi^{(2)}_x}^{1})_{\mathbb{D}^{(2)}},
\end{aligned}
$$

where $\overrightarrow{y}' = (y'_1, \ldots, y'_{n_1})$, $y'_i = \delta y^{(b)}_i - \sum_{j=1}^{n_1} w^{(1)}_j \mu^{(1)}_{j,i}$, $i = 1, \ldots, n_1$, $ID'_{x,1} = \delta - \sum_{j=1}^{2} w^{(2)}_{j,x}\mu^{(2)}_{j,1}$, $ID'_{x,2} = -\delta ID_x - \sum_{j=1}^{2} w^{(2)}_{j,x}\mu^{(2)}_{j,2}$ which are uniformly and independently distributed since $w^{(1)}_1, \ldots, w^{(1)}_{n_1} \xleftarrow{\mathsf{U}} \mathbb{F}_q$ and $w^{(2)}_{1,x}, w^{(2)}_{2,x} \xleftarrow{\mathsf{U}} \mathbb{F}_q$.

In the light of the adversary's view, both $(\mathbb{B}^{(k)}, \mathbb{B}^{*(k)})$ and $(\mathbb{D}^{(k)}, \mathbb{D}^{*(k)})$ for $k = 0, 1, 2$ are consistent with public key $(1^\lambda, \mathsf{param}_{\overrightarrow{n}}, \{\widehat{\mathbb{B}}^{(k)}\}_{k=0,1,2})$. Therefore, $\{\boldsymbol{k}^{*(j)}_{\overrightarrow{x},I}\}_{j=1,\ldots,\nu}$ and C can be expressed as keys and ciphertext in two ways, in Game 2-ν over bases $(\mathbb{B}^{(k)}, \mathbb{B}^{*(k)})$ and in Game 3 over bases $(\mathbb{D}^{(k)}, \mathbb{D}^{*(k)})$. Thus, Game 2-$\nu$ can be conceptually changed to Game 3 if $w \neq 0$ and $(w^{(1)}_1, \ldots, w^{(1)}_{n_1}) \neq \overrightarrow{0}$ and $(w^{(2)}_{1,x}, w^{(2)}_{2,x}) \neq \overrightarrow{0}$ $x \in \mathsf{RevokeNodes}(\mathit{Tree}, L^{(b)})$, i.e., except with probability $(2 + \log N)/q$. □

Lemma 22. *For any adversary $\mathcal{A}$, $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$.*

Proof. The value of b is independent from the adversary's view in Game 3. Therefore, $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$. □

B.1 Proof of Lemma 15, 16 and 17

In order to reduce the DLIN problem from Definition 2 to Problems 3 and 4 from Definitions 11 and 12, respectively, we further introduce three "basic problems" that will serve in intermediate steps of the reduction:

- Basic Problem 0 in Definition 13.
- Basic Problem 3 in Definition 14.
- Basic Problem 4 in Definition 15.

In order to prove **Lemmas** 15 and 16 we use two intermediate **Lemmas** 23 and 24 which are two common lemmas in the proofs of **Lemmas** 15 and 16.

Lemma 23. *Let $(q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e)$ be dual pairing vector spaces by direct product of symmetric pairing groups. Using $\{\phi_{i,j}\}$, we can efficiently sample a random linear transformation $W = \sum_{i=1,j=1}^{N,N} r_{i,j}\phi_{i,j}$ of $\mathbb{V}$ with random coefficients $\{r_{i,j}\}_{i,j\in\{1,\ldots,N\}} \xleftarrow{\mathsf{U}} GL(N, \mathbb{F}_q)$. The matrix $(r^*_{i,j}) = (\{r_{i,j}\}^{-1})^T$ defines the adjoint action on $\mathbb{V}$ for pairing e, i.e., $e(W(\boldsymbol{x}), (W^{-1})^T(\boldsymbol{y})) = e(\boldsymbol{x}, \boldsymbol{y})$ for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{V}$, where $(W^{-1})^T = \sum_{i=1,j=1}^{N,N} r^*_{i,j}\phi_{i,j}$.*

Lemma 23 is proved in [20].

Definition 13 (Basic Problem 0). *Basic Problem 0 is to find bit β given $(\mathsf{param}_{\mathrm{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}^*_\beta, \boldsymbol{f}, bG, aG, acG) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathrm{BP0}}_\beta(1^\lambda)$ for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$ with probability non-negligibly greater than by a random guess, where*

$$
\begin{aligned}
&\mathcal{G}^{\mathrm{BP0}}_\beta\ (1^\lambda):\\
&\quad \mathsf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda),\\
&\quad \mathsf{param}_{\mathbb{V}} = (q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 3, \mathsf{param}_{\mathbb{G}}),\\
&\quad \Lambda = (\lambda_{i,j}) \xleftarrow{\mathsf{U}} GL(3, \mathbb{F}_q),\ (\mu_{i,j}) = (\Lambda^T)^{-1},\quad b, a \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\\
&\quad \boldsymbol{b}_i = b\sum_{j=1}^{3} \lambda_{i,j}\boldsymbol{a}_j, i = 1, 3,\ \widehat{\mathbb{B}} = (\boldsymbol{b}_1, \boldsymbol{b}_3),\\
&\quad \boldsymbol{b}^*_i = a\sum_{j=1}^{3} \mu_{i,j}\boldsymbol{a}_j, i = 1, 2, 3,\ \mathbb{B}^* = (\boldsymbol{b}^*_1, \boldsymbol{b}^*_2, \boldsymbol{b}^*_3),\\
&\quad g_T = e(G, G)^{ab},\quad \mathsf{param}_{\mathrm{BP0}} = (\mathsf{param}_{\mathbb{V}}, g_T),\\
&\quad \delta, \sigma, \omega \xleftarrow{\mathsf{U}} \mathbb{F}_q,\quad \rho, \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\\
&\quad \boldsymbol{y}^*_0 = (\delta, 0, \sigma)_{\mathbb{B}^*},\quad \boldsymbol{y}^*_1 = (\delta, \rho, \sigma)_{\mathbb{B}^*},\quad \boldsymbol{f} = (\omega, \tau, 0)_{\mathbb{B}},\\
&\quad \text{Output } (\mathsf{param}_{\mathrm{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}^*_\beta, \boldsymbol{f}, bG, aG, acG).
\end{aligned}
$$

Let $\mathsf{Adv}^{\mathrm{BP0}}_{\mathcal{F}}(\lambda)$ denote the advantage of a PPT algorithm $\mathcal{F}$ for the Basic Problem 0.

Lemma 24. *For any adversary $\mathcal{F}$, there exists a probabilistic machine $\mathcal{D}$, whose running time is essentially the same as that of $\mathcal{D}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathrm{BP0}}_{\mathcal{F}}(\lambda) \le \mathsf{Adv}^{\mathrm{DLIN}}_{\mathcal{D}}(\lambda) + 5/q$.*

The proof of **Lemma** 24 was given in [20].

Proof of Lemma 15: Combining **Lemmas** 23, 24, 25 and 26 we obtain **Lemma** 15.

Definition 14 (Basic Problem 3). *Basic Problem 3 is to find bit β given $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{f}^{(0)}_\beta, \boldsymbol{f}^{(1)}_{\beta,1}, \boldsymbol{f}^{(2)}_{\beta,1}, \{\boldsymbol{f}^{(1)}_i\}_{i=2,\ldots,n_1}, \boldsymbol{f}^{(2)}_2) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathrm{BP3}}_\beta(1^\lambda, \overrightarrow{n} = (2; n_1, n_2 = 2))$ for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$ with probability non-negligibly greater than by a random guess, where*

$$
\begin{aligned}
&\mathcal{G}^{\mathrm{BP3}}_\beta\ (1^\lambda, \overrightarrow{n} = (2; n_1, n_2 = 2)):\\
&\quad (\mathsf{param}_{\overrightarrow{n}}, \mathbb{B}^{(0)}, \mathbb{B}^{*(0)}, \mathbb{B}^{(1)}, \mathbb{B}^{*(1)}, \mathbb{B}^{(2)}, \mathbb{B}^{*(2)}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \overrightarrow{n}),\\
&\quad \widehat{\mathbb{B}}^{*(0)} = (\boldsymbol{b}^{*(0)}_1, \boldsymbol{b}^{*(0)}_3, \boldsymbol{b}^{*(0)}_4, \boldsymbol{b}^{*(0)}_5),\\
&\quad \widehat{\mathbb{B}}^{*(1)} = (\boldsymbol{b}^{*(1)}_1, \ldots, \boldsymbol{b}^{*(1)}_{n_1}, \boldsymbol{b}^{*(1)}_{n_1+2}, \ldots, \boldsymbol{b}^{*(1)}_{3n_1+1}),\\
&\quad \widehat{\mathbb{B}}^{*(2)} = (\boldsymbol{b}^{*(2)}_1, \boldsymbol{b}^{*(2)}_2, \boldsymbol{b}^{*(2)}_4, \boldsymbol{b}^{*(2)}_5, \boldsymbol{b}^{*(2)}_6, \boldsymbol{b}^{*(2)}_7),\\
&\quad \omega, \gamma \xleftarrow{\mathsf{U}} \mathbb{F}_q,\quad \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\quad \boldsymbol{f}^{(0)}_0 = (\omega, 0, 0, 0, \gamma)_{\mathbb{B}^{(0)}},\quad \boldsymbol{f}^{(0)}_1 = (\omega, \tau, 0, 0, \gamma)_{\mathbb{B}^{(0)}},\\
&\quad \rho^{(1)}, \rho^{(2)} \xleftarrow{\mathsf{U}} \mathbb{F}_q,
\end{aligned}
$$

For $k=1,2:\overrightarrow{u}^{(k)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_k}$;

$$\boldsymbol{f}_{0,1}^{(1)} = (\overbrace{\omega\overrightarrow{e}_1^{(1)},}^{n_1}\ \overbrace{0^{n_1},}^{n_1}\ \overbrace{0^{n_1},}^{n_1}\ \overbrace{\gamma}^{1})_{\mathbb{B}^{(1)}},$$

$$\boldsymbol{f}_{1,1}^{(1)} = (\overbrace{\omega\overrightarrow{e}_1^{(1)},}^{n_1}\ \overbrace{\tau\overrightarrow{e}_1^{(1)},}^{n_1}\ \overbrace{0^{n_1},}^{n_1}\ \overbrace{\gamma}^{1})_{\mathbb{B}^{(1)}},$$

For $i=2,\ldots,n_1:\boldsymbol{f}_i^{(1)}=\omega\boldsymbol{b}_i^{(1)}$;

$$\boldsymbol{f}_{0,1}^{(2)} = (\overbrace{\omega,0,}^{2}\ \overbrace{0^2,}^{2}\ \overbrace{0^2,}^{2}\ \overbrace{\gamma}^{1})_{\mathbb{B}^{(2)}},$$

$$\boldsymbol{f}_{1,1}^{(2)} = (\overbrace{\omega,0,}^{2}\ \overbrace{\tau,0,}^{2}\ \overbrace{0^2,}^{2}\ \overbrace{\gamma}^{1})_{\mathbb{B}^{(2)}},$$

$$\boldsymbol{f}_2^{(2)} = \omega\boldsymbol{b}_2^{(2)},$$

Output $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{f}_\beta^{(0)}, \boldsymbol{f}_{\beta,1}^{(1)}, \boldsymbol{f}_{\beta,1}^{(2)}, \{\boldsymbol{f}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{f}_2^{(2)})$.

Let $\mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP3}}(\lambda)$ *denote the advantage of a PPT algorithm* $\mathcal{C}$ *for the Basic Problem* 3.

Lemma 25. *For any adversary* $\mathcal{C}$*, there exists a probabilistic machine* $\mathcal{F}$*, whose running time is essentially the same as that of* $\mathcal{C}$*, such that for any security parameter* λ*,* $\mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP3}}(\lambda) \le \mathsf{Adv}_{\mathcal{F}}^{\mathrm{BP0}}(\lambda)$ *for* $\overrightarrow{n}=(2;n_1,n_2=2)$.

Proof. $\mathcal{F}$ is given a Basic Problem 0 instance $(\mathsf{param}_{\mathrm{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}_\beta^*, \boldsymbol{f}, bG, aG, acG)$.

With $\mathsf{param}_{\mathbb{G}}=(q,\mathbb{G},\mathbb{G}_T,G,e)$ contained in $\mathsf{param}_{\mathrm{BP0}}$, $\mathcal{F}$ computes

$$\mathsf{param}_{\mathbb{V}_0} = (q,\mathbb{V}_0,\mathbb{G}_T,\mathbb{A}_0,e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda,5,\mathsf{param}_{\mathbb{G}}),$$
$$\mathsf{param}_{\mathbb{V}_l} = (q,\mathbb{V}_l,\mathbb{G}_T,\mathbb{A}_l,e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda,3n_l+1,\mathsf{param}_{\mathbb{G}}),\ l=1,2,$$
$$\mathsf{param}_{\overrightarrow{n}} = (\{\mathsf{param}_{\mathbb{V}_l}\}_{l=0,1,2}, g_T),$$

where g_T is contained in $\mathsf{param}_{\mathrm{BP0}}$. $\mathcal{F}$ generates random linear transformation W_l on $\mathbb{V}_l (l=0,1,2)$ given in **Lemma** 23 and sets

$$\boldsymbol{d}_l^{(0)} = W_0(\boldsymbol{b}_l^*,0,0),\ \ l=1,2; \qquad \boldsymbol{d}_3^{(0)} = W_0(0,0,0,0,aG),$$
$$\boldsymbol{d}_4^{(0)} = W_0(0,0,0,aG,0), \qquad \boldsymbol{d}_5^{(0)} = W_0(\boldsymbol{b}_3^*,0,0),$$
$$\boldsymbol{d}_l^{*(0)} = (W_0^{-1})^T(\boldsymbol{b}_l,0,0),\ \ l=1,2; \qquad \boldsymbol{d}_3^{*(0)} = (W_0^{-1})^T(0,0,0,0,bG),$$
$$\boldsymbol{d}_4^{*(0)} = (W_0^{-1})^T(0,0,0,bG,0), \qquad \boldsymbol{d}_5^{*(0)} = (W_0^{-1})^T(\boldsymbol{b}_3,0,0),$$
$$\boldsymbol{g}_\beta^{(0)} = W_0(\boldsymbol{y}_\beta^*,0,0),$$

$$\boldsymbol{d}_1^{(1)} = W_1(\boldsymbol{b}_1^*,0^{N_1-3}), \qquad \boldsymbol{d}_{n_1+1}^{(1)} = W_1(\boldsymbol{b}_2^*,0^{N_1-3}), \qquad \boldsymbol{d}_{N_1}^{(1)} = W_1(\boldsymbol{b}_3^*,0^{N_1-3}),$$
$$\boldsymbol{d}_l^{(1)} = W_1(0^m,aG,0^{N_1-m-1}) \text{ where } \begin{cases} m=l+1 \text{ if } l\in\{2,\ldots,n_1\}, \\ m=l \text{ if } l\in\{n_1+2,\ldots,N_1-1\}, \end{cases}$$
$$\boldsymbol{d}_1^{*(1)} = (W_1^{-1})^T(\boldsymbol{b}_1,0^{N_1-3}), \qquad \boldsymbol{d}_{n_1+1}^{*(1)} = (W_1^{-1})^T(\boldsymbol{b}_2,0^{N_1-3}), \qquad \boldsymbol{d}_{N_1}^{*(1)} = (W_1^{-1})^T(\boldsymbol{b}_3,0^{N_1-3}),$$
$$\boldsymbol{d}_l^{*(1)} = (W_1^{-1})^T(0^m,bG,0^{N_1-m-1}) \text{ where } \begin{cases} m=l+1 \text{ if } l\in\{2,\ldots,n_1\}, \\ m=l \text{ if } l\in\{n_1+2,\ldots,N_1-1\}, \end{cases}$$
$$\boldsymbol{g}_{\beta,1}^{(1)} = W_1(\boldsymbol{y}_\beta^*,0^{N_1-3}),$$
$$\boldsymbol{g}_l^{(1)} = W_1(0^{l+1},acG,0^{N_1-l-2}),\ \ l=2,\ldots,n_1;$$

$$\boldsymbol{d}_1^{(2)} = W_2(\boldsymbol{b}_1^*, 0^4), \quad \boldsymbol{d}_3^{(2)} = W_2(\boldsymbol{b}_2^*, 0^4), \quad \boldsymbol{d}_7^{(2)} = W_2(\boldsymbol{b}_3^*, 0^4),$$
$$\boldsymbol{d}_l^{(2)} = W_2(0^m, aG, 0^{7-m-1}) \text{ where } \begin{cases} m = 3 \text{ if } l = 2, \\ m = l \text{ if } l \in \{4, \ldots, 6\}, \end{cases}$$
$$\boldsymbol{d}_1^{*(2)} = (W_2^{-1})^T(\boldsymbol{b}_1, 0^4), \quad \boldsymbol{d}_3^{*(2)} = (W_2^{-1})^T(\boldsymbol{b}_2, 0^4), \quad \boldsymbol{d}_7^{*(2)} = (W_2^{-1})^T(\boldsymbol{b}_3, 0^4),$$
$$\text{otherwise, } \boldsymbol{d}_l^{*(2)} = (W_2^{-1})^T(0^m, bG, 0^{7-m-1}) \text{ where } \begin{cases} m = 3 \text{ if } l = 2, \\ m = l \text{ if } l \in \{4, \ldots, 6\}, \end{cases}$$
$$\boldsymbol{g}_{\beta,1}^{(2)} = W_2(\boldsymbol{x}_\beta^*, 0^4),$$
$$\boldsymbol{g}_2^{(2)} = W_2(0^3, acG, 0^3),$$

where $(\boldsymbol{v}, 0^{N_l-3}) = (G', G'', G''', 0^{N_l-3})$ for any $\boldsymbol{v} = (G', G'', G''') \in \mathbb{V} = \mathbb{G}^3$. This implies that $\mathbb{D}^{(0)} = (\boldsymbol{d}_l^{(0)})_{l=1,\ldots,5}$ and $\mathbb{D}^{*(0)} = (\boldsymbol{d}_l^{*(0)})_{l=1,\ldots,5}$, $\mathbb{D}^{(j)} = (\boldsymbol{d}_l^{(j)})_{l=1,\ldots,3n_j+1}$ and $\mathbb{D}^{*(j)} = (\boldsymbol{d}_l^{*(j)})_{l=1,\ldots,3n_j+1}$, $j = 1, 2$ are dual orthonormal bases. $\mathcal{F}$ can compute $\mathbb{D}^{(j)}$, $j = 0, 1, 2$; $\widehat{\mathbb{D}}^{*(0)} = (\boldsymbol{d}_1^{*(0)}, \boldsymbol{d}_3^{*(0)}, \boldsymbol{d}_4^{*(0)}, \boldsymbol{d}_5^{*(0)})$, $\widehat{\mathbb{D}}^{*(j)} = (\boldsymbol{d}_1^{*(j)}, \ldots, \boldsymbol{d}_{n_j}^{*(j)}, \boldsymbol{d}_{n_j+2}^{*(j)}, \ldots, \boldsymbol{d}_{3n_j+1}^{*(j)})$, $j = 1, 2$ using $\widehat{\mathbb{B}} = (\boldsymbol{b}_1, \boldsymbol{b}_3)$, $\mathbb{B}^*$, bG, and aG. $\mathcal{F}$ then gives $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{D}^{(k)}, \widehat{\mathbb{D}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{g}_\beta^{(0)}, \boldsymbol{g}_{\beta,1}^{(1)}, \boldsymbol{g}_{\beta,1}^{(2)}, \{\boldsymbol{g}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{g}_2^{(2)})$ to $\mathcal{C}$, and outputs bit β' if $\mathcal{C}$ outputs β'.
We observe that

$$\boldsymbol{g}_0^{(0)} = (\omega', 0, 0, 0, \gamma')_{\mathbb{D}^{(0)}}, \qquad \boldsymbol{g}_1^{(0)} = (\omega', \tau', 0, 0, \gamma')_{\mathbb{D}^{(0)}},$$
$$\boldsymbol{g}_{0,1}^{(1)} = (\overbrace{\omega' \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\gamma'}^{1})_{\mathbb{D}^{(1)}}, \qquad \boldsymbol{g}_{0,1}^{(2)} = (\overbrace{\omega', 0,}^{2} \overbrace{0^2,}^{2} \overbrace{0^2,}^{2} \overbrace{\gamma'}^{1})_{\mathbb{D}^{(2)}},$$
$$\boldsymbol{g}_{1,1}^{(1)} = (\overbrace{\omega' \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{\tau' \overrightarrow{e}_1^{(1)}}^{n_1}, \overbrace{0^{n_1}}^{n_1}, \overbrace{\gamma'}^{1})_{\mathbb{D}^{(1)}}, \qquad \boldsymbol{g}_{1,1}^{(2)} = (\overbrace{\omega', 0,}^{2} \overbrace{\tau', 0,}^{2} \overbrace{0^2,}^{2} \overbrace{\gamma'}^{1})_{\mathbb{D}^{(2)}},$$
$$\boldsymbol{g}_i^{(1)} = \omega' \boldsymbol{b}_i^{(1)} \quad i = 2, \ldots, n_1; \qquad \boldsymbol{g}_2^{(2)} = \omega' \boldsymbol{b}_2^{(2)},$$

where $\omega' = \delta, \tau' = \rho, \gamma' = \sigma$ are distributed uniformly in $\mathbb{F}_q$, Therefore, the distribution of $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{D}^{(k)}, \widehat{\mathbb{D}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{g}_\beta^{(0)}, \boldsymbol{g}_{\beta,1}^{(1)}, \boldsymbol{g}_{\beta,1}^{(2)}, \{\boldsymbol{g}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{g}_2^{(2)})$ is exactly the same as in the instance of the Basic Problem 3.

Lemma 26. *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P3}}(\lambda) \leq \mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP3}}(\lambda) + 3/q$ for $(\overrightarrow{n} = (2; n_1, n_2 = 2), d)$.*

Proof. $\mathcal{C}$ is given an instance of the Basic Problem 3, i.e. a tuple $(\mathsf{param}_{\overrightarrow{n}}, \{\mathbb{B}^{(k)}, \widehat{\mathbb{B}}^{*(k)}\}_{k=0,1,2}, \boldsymbol{f}_\beta^{(0)}, \boldsymbol{f}_{\beta,1}^{(1)}, \boldsymbol{f}_{\beta,1}^{(2)}, \{\boldsymbol{f}_i^{(1)}\}_{i=2,\ldots,n_1}, \boldsymbol{f}_2^{(2)})$. It computes $\boldsymbol{r} \xleftarrow{\cup} \mathsf{span}\langle \boldsymbol{b}_{3n_1+1}^{(1)} \rangle$, $\boldsymbol{r_j} \xleftarrow{\cup} \mathsf{span}\langle \boldsymbol{b}_7^{(2)} \rangle$, $j = 1, \ldots, d$ and sets $\boldsymbol{t}_{\beta,1}^{(1)} = \boldsymbol{f}_{\beta,1}^{(1)} + \boldsymbol{r}, \boldsymbol{t}_{\beta,1,j}^{(2)} = \boldsymbol{f}_{\beta,1}^{(2)} + \boldsymbol{r}_j$, $j = 1, \ldots, d$.

Then, $\mathcal{C}$ chooses $u_0 \xleftarrow{\cup} \mathbb{F}_q^\times$, $(u_{i,j}^{(k)}) \xleftarrow{\cup} GL(\mathbb{F}_q, n_k)$, $(z_{i,j}^{(k)}) = \left((u_{i,j}^{(k)})^{-1}\right)^T, i = 1, \ldots, n_k, j = 1, \ldots, n_k, k = 1, 2.$ and computes:

$$\boldsymbol{d}_2^{(0)} = (0, u_0, 0, 0, 0)_{\mathbb{B}^{(0)}},$$
$$\boldsymbol{d}_{n_k+i}^{(k)} = (\overbrace{0^{n_k},}^{n_k} \overbrace{u_{i,1}^{(k)}, \ldots, u_{i,n_k}^{(k)},}^{n_k} \overbrace{0^{n_k},}^{n_k} \overbrace{0}^{1})_{\mathbb{B}^{(k)}}, \ i = 1, \ldots, n_k, \ k = 1, 2;$$

$\mathcal{C}$ then sets dual orthonormal basis vectors

$$\boldsymbol{d}_2^{*(0)} = (0, u_0^{-1}, 0, 0, 0)_{\mathbb{B}^{*(0)}},$$
$$\boldsymbol{d}_{n_k+i}^{*(k)} = (\overbrace{0^{n_k},}^{n_k} \overbrace{z_{i,1}^{(k)}, \ldots, z_{i,n_k}^{(k)},}^{n_k} \overbrace{0^{n_k},}^{n_k} \overbrace{0}^{1})_{\mathbb{B}^{*(k)}}, \ i = 1, \ldots, n_k, \ k = 1, 2.$$

Note that $\mathcal{C}$ cannot compute $\boldsymbol{d}_2^{*(0)}$ and $\boldsymbol{d}_{n_k+i}^{*(k)}$, $i=1,\ldots,n_k$, $k=1,2$ due to the lack of $\boldsymbol{b}_2^{*(0)}$ and $\boldsymbol{b}_{n_k+1}^{*(k)}$. Then, $\mathcal{C}$ computes $\mathbb{D}^{(0)}=(\boldsymbol{b}_1^{(0)},\boldsymbol{d}_2^{(0)},\boldsymbol{b}_3^{(0)},\boldsymbol{b}_4^{(0)},\boldsymbol{b}_5^{(0)})$, $\widehat{\mathbb{D}}^{*(0)}=(\boldsymbol{b}_1^{*(0)},\boldsymbol{b}_3^{*(0)},\boldsymbol{b}_4^{*(0)},\boldsymbol{b}_5^{*(0)})$, $\mathbb{D}^{(k)}=(\boldsymbol{b}_1^{(k)},\ldots,\boldsymbol{b}_{n_k}^{(k)},\boldsymbol{d}_{n_k+1}^{(k)},\ldots,\boldsymbol{d}_{2n_k}^{(k)},\boldsymbol{b}_{2n_k+1}^{(k)},\ldots,\boldsymbol{b}_{3n_k+1}^{(k)})$, $\widehat{\mathbb{D}}^{*(k)}=(\boldsymbol{b}_1^{*(k)},\ldots,\boldsymbol{b}_{n_k}^{*(k)},\boldsymbol{b}_{2n_k+1}^{*(k)},\ldots,\boldsymbol{b}_{3n_k+1}^{*(k)})$, $k=1,2$.
Finally, $\mathcal{C}$ hands $(\mathsf{param}_{\overrightarrow{n}},\{\mathbb{D}^{(k)},\widehat{\mathbb{D}}^{*(k)}\}_{k=0,1,2},\boldsymbol{f}_\beta^{(0)},\boldsymbol{t}_{\beta,1}^{(1)},\{\boldsymbol{t}_{\beta,1,j}^{(2)}\}_{j=1,\ldots,d},\{\boldsymbol{f}_i^{(1)}\}_{i=2,\ldots,n_1},\boldsymbol{f}_2^{(2)})$ over to $\mathcal{B}$ and outputs $\beta'\in\{0,1\}$ if $\mathcal{B}$ outputs β'.
Observe that with respect to $\mathbb{D}^{(k)},\mathbb{D}^{*(k)}$, $k=0,1,2$, the input to $\mathcal{B}$ has the same distribution as the instance of Problem 3 unless the following events occur: $u=0$, $\overrightarrow{u}^{(1)}=\overrightarrow{0}$, or $\overrightarrow{u}_j^{(2)}=\overrightarrow{0}$. Those events occur with probability $3/q$ when $\beta=1$.

Proof of Lemma 16: Combining **Lemmas** 23, 24, 27, and 28 we obtain **Lemma** 16.

Definition 15 (Basic Problem 4). *Basic Problem* 4 *is to find bit* β *given* $(\mathsf{param}_{\overrightarrow{n}},\widehat{\mathbb{B}}^{(0)},\mathbb{B}^{*(0)},\boldsymbol{y}_\beta^{*(0)},\boldsymbol{f}^{(0)},\{\widehat{\mathbb{B}}^{(k)},\mathbb{B}^{*(k)},\{\boldsymbol{y}_{\beta,i}^{*(k)},\boldsymbol{f}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})\xleftarrow{R}\mathcal{G}_\beta^{\mathrm{BP4}}(1^\lambda,\overrightarrow{n}=(2;n_1,n_2=2))$ *for* $\beta\xleftarrow{\cup}\{0,1\}$ *with probability non-negligibly greater than by a random guess, where*

$$
\begin{aligned}
&\mathcal{G}_\beta^{\mathrm{BP4}}\ (1^\lambda,\overrightarrow{n}=(2;n_1,n_2=2)):\\
&\quad(\mathsf{param}_{\overrightarrow{n}},\mathbb{B}^{(0)},\mathbb{B}^{*(0)},\mathbb{B}^{(1)},\mathbb{B}^{*(1)},\mathbb{B}^{(2)},\mathbb{B}^{*(2)})\xleftarrow{R}\mathcal{G}_{\mathsf{ob}}(1^\lambda,\overrightarrow{n}),\\
&\quad\widehat{\mathbb{B}}^{(0)}=(\boldsymbol{b}_1^{(0)},\boldsymbol{b}_3^{(0)},\boldsymbol{b}_4^{(0)},\boldsymbol{b}_5^{(0)}),\\
&\quad\widehat{\mathbb{B}}^{(1)}=(\boldsymbol{b}_1^{(1)},\ldots,\boldsymbol{b}_{n_1}^{(1)},\boldsymbol{b}_{2n_1+1}^{(1)},\ldots,\boldsymbol{b}_{3n_1+1}^{(1)}),\\
&\quad\widehat{\mathbb{B}}^{(2)}=(\boldsymbol{b}_1^{(2)},\boldsymbol{b}_2^{(2)},\boldsymbol{b}_5^{(2)},\boldsymbol{b}_6^{(2)},\boldsymbol{b}_7^{(2)}),\\
&\quad\omega,\xi,\delta\xleftarrow{\cup}\mathbb{F}_q,\quad z,\pi\xleftarrow{\cup}\mathbb{F}_q^\times,\\
&\quad\boldsymbol{y}_0^{*(0)}=(\omega,0,0,\xi,0)_{\mathbb{B}^{*(0)}},\quad\boldsymbol{y}_1^{*(0)}=(\omega,z,0,\xi,0)_{\mathbb{B}^{*(0)}},\quad\boldsymbol{f}^{(0)}=(\delta,\pi,0,0,0)_{\mathbb{B}^{(0)}},\\
&\quad\text{For } k=1,2 \text{ and } i=1,\ldots,n_k:\\
&\qquad\boldsymbol{y}_{0,i}^{*(k)}=(\overbrace{\omega\overrightarrow{e}_i^{(k)}}^{n_k},\overbrace{0^{n_k}}^{n_k},\overbrace{\xi\overrightarrow{e}_i^{(k)}}^{n_k},\overbrace{0}^{1})_{\mathbb{B}^{*(k)}},\\
&\qquad\boldsymbol{y}_{1,i}^{*(k)}=(\overbrace{\omega\overrightarrow{e}_i^{(k)}}^{n_k},\overbrace{z\overrightarrow{e}_i^{(k)}}^{n_k},\overbrace{\xi\overrightarrow{e}_i^{(k)}}^{n_k},\overbrace{0}^{1})_{\mathbb{B}^{*(k)}},\\
&\qquad\boldsymbol{f}_i^{(k)}=(\overbrace{\delta\overrightarrow{e}_i^{(k)}}^{n_k},\overbrace{\pi\overrightarrow{e}_i^{(k)}}^{n_k},\overbrace{0^{n_k}}^{n_k},\overbrace{0}^{1})_{\mathbb{B}^{(k)}},\\
&\quad\text{Output}\ \ (\mathsf{param}_{\overrightarrow{n}},\widehat{\mathbb{B}}^{(0)},\mathbb{B}^{*(0)},\boldsymbol{y}_\beta^{*(0)},\boldsymbol{f}^{(0)},\{\widehat{\mathbb{B}}^{(k)},\mathbb{B}^{*(k)},\{\boldsymbol{y}_{\beta,i}^{*(k)},\boldsymbol{f}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2}).
\end{aligned}
$$

Let $\mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP4}}(\lambda)$ *denote the advantage of a PPT algorithm* $\mathcal{C}$ *for the Basic Problem* 4.

Lemma 27. *For any adversary* $\mathcal{C}$*, there exists a probabilistic machine* $\mathcal{F}$*, whose running time is essentially the same as that of* $\mathcal{C}$*, such that for any security parameter* λ*,* $\mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP4}}(\lambda)=\mathsf{Adv}_{\mathcal{F}}^{\mathrm{BP0}}(\lambda)$ *for* $\overrightarrow{n}=(2;n_1,n_2=2)$.

Proof. $\mathcal{F}$ is given a Basic Problem 0 instance $(\mathsf{param}_{\mathrm{BP0}},\widehat{\mathbb{B}},\mathbb{B}^*,\boldsymbol{y}_\beta^*,\boldsymbol{f},bG,aG,acG)$.

With $\mathsf{param}_{\mathbb{G}}=(q,\mathbb{G},\mathbb{G}_T,G,e)$ contained in $\mathsf{param}_{\mathrm{BP0}}$, $\mathcal{C}$ computes

$$
\begin{aligned}
&\mathsf{param}_{\mathbb{V}_0}=(q,\mathbb{V}_0,\mathbb{G}_T,\mathbb{A}_0,e)\xleftarrow{\mathsf{R}}\mathcal{G}_{\mathsf{dpvs}}(1^\lambda,5,\mathsf{param}_{\mathbb{G}}),\\
&\mathsf{param}_{\mathbb{V}_l}=(q,\mathbb{V}_l,\mathbb{G}_T,\mathbb{A}_l,e)\xleftarrow{\mathsf{R}}\mathcal{G}_{\mathsf{dpvs}}(1^\lambda,3n_l+1,\mathsf{param}_{\mathbb{G}}),\ l=1,2,\\
&\mathsf{param}_{\overrightarrow{n}}=(\{\mathsf{param}_{\mathbb{V}_l}\}_{l=0,1,2},g_T),
\end{aligned}
$$

where g_T is contained in $\mathsf{param}_{\mathrm{BP0}}$. $\mathcal{F}$ generates random linear transformation W_l on $\mathbb{V}_l (l = 0, 1, 2)$ given in **Lemma** 23, then sets

$$\begin{aligned}
&\boldsymbol{d}_l^{(0)} = W_0(\boldsymbol{b}_l, 0, 0),\ l = 1, 2, \quad \boldsymbol{d}_3^{(0)} = W_0(0, 0, 0, 0, bG),\\
&\boldsymbol{d}_4^{(0)} = W_0(\boldsymbol{b}_3, 0, 0), \quad \boldsymbol{d}_5^{(0)} = W_0(0, 0, 0, bG, 0),\\
&\boldsymbol{d}_l^{*(0)} = (W_0^{-1})^T(\boldsymbol{b}_l^*, 0, 0),\ l = 1, 2, \quad \boldsymbol{d}_3^{*(0)} = (W_0^{-1})^T(0, 0, 0, 0, aG),\\
&\boldsymbol{d}_4^{*(0)} = (W_0^{-1})^T(\boldsymbol{b}_3^*, 0, 0), \quad \boldsymbol{d}_5^{*(0)} = (W_0^{-1})^T(0, 0, 0, aG, 0),\\
&\boldsymbol{p}_\beta^{*(0)} = (W_0^{-1})^T(\boldsymbol{y}_\beta^*, 0, 0), \quad \boldsymbol{g}^{(0)} = W_0(\boldsymbol{f}, 0, 0),\\
&\text{For } k = 1, 2:\\
&\quad \text{For } l = 1, 2, 3 \text{ and } i = 1, \ldots, n_k:\\
&\quad\quad \boldsymbol{d}_{(l-1)n_k+i}^{(k)} = W_k(0^{3(i-1)}, \boldsymbol{b}_l, 0^{3(n_k-i)}, 0);\\
&\quad \boldsymbol{d}_{3n_k+1}^{(k)} = W_k(0^{3n_k}, bG),\\
&\quad \text{For } l = 1, 2, 3 \text{ and } i = 1, \ldots, n_k:\\
&\quad\quad \boldsymbol{d}_{(l-1)n_k+i}^{*(k)} = (W_k^{-1})^T(0^{3(i-1)}, \boldsymbol{b}_l^*, 0^{3(n_k-i)}, 0);\\
&\quad \boldsymbol{d}_{3n_k+1}^{*(k)} = (W_k^{-1})^T(0^{3n_k}, aG),\\
&\quad \text{For } i = 1, \ldots, n_k:\\
&\quad\quad \boldsymbol{p}_{\beta,i}^{*(k)} = (W_k^{-1})^T(0^{3(i-1)}, \boldsymbol{y}_\beta^*, 0^{3(n_k-i)}, 0),\\
&\quad\quad \boldsymbol{g}_i^{(k)} = W_1(0^{3(i-1)}, \boldsymbol{f}, 0^{3(n_k-i)}, 0).
\end{aligned}$$

Hence, we have that $\mathbb{D}^{(0)} = (\boldsymbol{d}_l^{(0)})_{l=1,\ldots,5}$ and $\mathbb{D}^{*(0)} = (\boldsymbol{d}_l^{*(0)})_{l=1,\ldots,5}$ as well as $\mathbb{D}^{(j)} = (\boldsymbol{d}_l^{(j)})_{l=1,\ldots,3n_j+1}$ and $\mathbb{D}^{*(j)} = (\boldsymbol{d}_l^{*(j)})_{l=1,\ldots,3n_j+1}$, $j = 1, 2$ are dual orthonormal bases. $\mathcal{F}$ can compute $\mathbb{D}^{*(j)}$, $j = 0, 1, 2$; $\widehat{\mathbb{D}}^{(0)} = (\boldsymbol{d}_1^{(0)}, \boldsymbol{d}_3^{(0)}, \boldsymbol{d}_4^{(0)}, \boldsymbol{d}_5^{(0)})$, $\widehat{\mathbb{D}}^{(j)} = (\boldsymbol{d}_l^{(j)}, \ldots, \boldsymbol{d}_{n_j}^{(j)}, \boldsymbol{d}_{2n_j+1}^{(j)}, \ldots, \boldsymbol{d}_{3n_j+1}^{(j)})$, $j = 1, 2$, using $\widehat{\mathbb{B}} = (\boldsymbol{b}_1, \boldsymbol{b}_3)$, $\mathbb{B}^*$, bG, and aG.
Then, $\mathcal{F}$ hands $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{D}}^{(0)}, \mathbb{D}^{*(0)}, \boldsymbol{p}_\beta^{*(0)}, \boldsymbol{g}^{(0)}, \{\widehat{\mathbb{D}}^{(k)}, \mathbb{D}^{*(k)}, \{\boldsymbol{p}_{\beta,i}^{*(k)}, \boldsymbol{g}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$ over to $\mathcal{C}$ and outputs bit β' if $\mathcal{C}$ outputs β'.
We observe that

$$\begin{aligned}
&\boldsymbol{p}_0^{*(0)} = (\omega, 0, 0, \xi, 0)_{\mathbb{D}^{*(0)}}, \quad \boldsymbol{p}_1^{*(0)} = (\omega, z, 0, \xi, 0)_{\mathbb{D}^{*(0)}}, \quad \boldsymbol{g}^{(0)} = (\delta, \pi, 0, 0, 0)_{\mathbb{D}^{(0)}},\\
&\text{For } k = 1, 2 \text{ and } i = 1, \ldots, n_k:\\
&\boldsymbol{p}_{0,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{\xi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{D}^{*(k)}},\\
&\boldsymbol{p}_{1,i}^{*(k)} = (\overbrace{\omega \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{z \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{\xi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0}^{1})_{\mathbb{D}^{*(k)}},\\
&\boldsymbol{g}_i^{(k)} = (\overbrace{\delta \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{\pi \overrightarrow{e}_i^{(k)}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{D}^{(k)}}.
\end{aligned}$$

Therefore, the distribution of $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{D}}^{(0)}, \mathbb{D}^{*(0)}, \boldsymbol{p}_\beta^{*(0)}, \boldsymbol{g}^{(0)}, \{\widehat{\mathbb{D}}^{(k)}, \mathbb{D}^{*(k)}, \{\boldsymbol{p}_{\beta,i}^{*(k)}, \boldsymbol{g}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$ is exactly the same as in the instance of the Basic Problem 4. □

Lemma 28. *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathrm{P4}}(\lambda) = \mathsf{Adv}_{\mathcal{C}}^{\mathrm{BP4}}(\lambda)$.*

Proof. Given an instance of the Basic Problem 4, i.e. a tuple $(\mathsf{param}_{\overrightarrow{n}}, \widehat{\mathbb{B}}^{(0)}, \mathbb{B}^{*(0)}, \boldsymbol{y}_\beta^{*(0)}, \boldsymbol{f}^{(0)}, \{\widehat{\mathbb{B}}^{(k)}, \mathbb{B}^{*(k)}, \{\boldsymbol{y}_{\beta,i}^{*(k)}, \boldsymbol{f}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$ the algorithm $\mathcal{C}$ computes $\boldsymbol{r}_i^{*(k)} \xleftarrow{\mathsf{U}} \mathsf{span}\langle \boldsymbol{b}_{2n_k+1}^{*(k)}, \ldots, \boldsymbol{b}_{3n_k}^{*(k)} \rangle$ and sets $\boldsymbol{h}_{\beta,i}^{*(k)} = \boldsymbol{y}_{\beta,i}^{*(k)} + \boldsymbol{r}_i^{*(k)}$, $i = 1, \ldots, n_k$, $k = 1, 2$.

Then, $\mathcal{C}$ chooses $z'_0 \xleftarrow{\cup} \mathbb{F}_q^{\times}$, $(z'^{(k)}_{i,j}) \xleftarrow{\cup} GL(\mathbb{F}_q, n_k), i = 1, \ldots, n_k, j = 1, \ldots, n_k, k = 1, 2$, and computes:

$$\begin{aligned}
\boldsymbol{d}_2^{*(0)} &= (0, z'_0, 0, 0, 0)_{\mathbb{B}^{*(0)}}, \\
\boldsymbol{d}_{n_k+i}^{*(k)} &= (\overbrace{0^{n_k}}^{n_k}, \overbrace{z'^{(k)}_{i,1}, \ldots, z'^{(k)}_{i,n_k}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}},\ i = 1, \ldots, n_k,\ k = 1, 2.
\end{aligned}$$

Then, $\mathcal{C}$ sets $z_0 = z^{-1} z'_0, u_0 = z_0^{-1}, (z^{(k)}_{i,j}) = z^{-1}(z'^{(k)}_{i,j})$, and $(u^{(k)}_{i,j}) = \left((z^{(k)}_{i,j})^{-1}\right)^T$, where z is defined as in the Basic Problem 4. Then,

$$\begin{aligned}
\boldsymbol{d}_2^{*(0)} &= (0, z z_0, 0, 0, 0)_{\mathbb{B}^{*(0)}}, \\
\boldsymbol{d}_{n_k+i}^{*(k)} &= (\overbrace{0^{n_k}}^{n_k}, \overbrace{z z^{(k)}_{i,1}, \ldots, z z^{(k)}_{i,n_k}}^{n_k}, \overbrace{0^{n_k}}^{n_k}, \overbrace{0}^{1})_{\mathbb{B}^{*(k)}},\ i = 1, \ldots, n_k,\ k = 1, 2; \\
\boldsymbol{d}_2^{(0)} &= (0, z^{-1} u_0, 0, 0, 0)_{\mathbb{B}^{(0)}}, \\
\boldsymbol{d}_{n_k+i}^{(k)} &= (\overbrace{0^{n_k}}^{n_k}, \overbrace{z^{-1} u^{(k)}_{i,1}, \ldots, z^{-1} u^{(k)}_{i,n_k}}^{n_k}, \overbrace{0,}^{1} \overbrace{0}^{1})_{\mathbb{B}^{(k)}},\ i = 1, \ldots, n_k,\ k = 1, 2.
\end{aligned}$$

Then, $\mathcal{C}$ computes

$$\begin{aligned}
\mathbb{D}^{*(0)} &= (\boldsymbol{b}_1^{*(0)}, \boldsymbol{d}_2^{*(0)}, \boldsymbol{b}_3^{*(0)}, \boldsymbol{b}_4^{*(0)}, \boldsymbol{b}_5^{*(0)}), \\
\widehat{\mathbb{D}}^{(0)} &= (\boldsymbol{b}_1^{(0)}, \boldsymbol{b}_3^{(0)}, \boldsymbol{b}_4^{(0)}, \boldsymbol{b}_5^{(0)}), \\
\mathbb{D}^{*(k)} &= (\boldsymbol{b}_1^{*(k)}, \ldots, \boldsymbol{b}_{n_k}^{*(k)}, \boldsymbol{d}_{n_k+1}^{*(k)}, \ldots, \boldsymbol{d}_{2n_k}^{*(k)}, \boldsymbol{b}_{2n_k+1}^{*(k)}, \ldots, \boldsymbol{b}_{3n_k+1}^{*(k)}), \\
\widehat{\mathbb{D}}^{(k)} &= (\boldsymbol{b}_1^{(k)}, \ldots, \boldsymbol{b}_{n_k}^{(k)}, \boldsymbol{b}_{2n_k+1}^{(k)}, \ldots, \boldsymbol{b}_{3n_k+1}^{(k)}), \text{for} k = 1, 2.
\end{aligned}$$

Finally, $\mathcal{C}$ hands $(\mathsf{param}_{\vec{n}}, \widehat{\mathbb{D}}^{(0)}, \mathbb{D}^{*(0)}, \boldsymbol{y}_{\beta}^{*(0)}, \boldsymbol{f}^{(0)}, \{\widehat{\mathbb{D}}^{(k)}, \mathbb{D}^{*(k)}, \{\boldsymbol{y}_{\beta,i}^{*(k)}, \boldsymbol{f}_i^{(k)}\}_{i=1,\ldots,n_k}\}_{k=1,2})$ over to $\mathcal{B}$ and if $\mathcal{B}$ outputs β' forwards this bit as its own output. For π in the Basic Problem 4 let $\pi' = z\pi$. Then, with respect to $\pi', \mathbb{D}^{(k)}, \mathbb{D}^{*(k)}$, $k = 0, 1, 2$, the above input to $\mathcal{B}$ has the same distribution as the instance of Problem 4. □

Proof of Lemma 17: The proof of Lemma 17 was given in [20].