Fully Secure Unbounded Revocable Attribute-Based Encryption in Prime Order Bilinear Groups via Subset Difference Method

Pratish Datta, Ratna Dutta, and Sourav Mukhopadhyay

Department of Mathematics
Indian Institute of Technology Kharagpur
Kharagpur-721302, India
{pratishdatta,ratna,sourav}@maths.iitkgp.ernet.in

Abstract. Providing an efficient revocation mechanism for attribute-based encryption (ABE) is of utmost importance since over time an user's credentials may be revealed or expired. All previously known revocable ABE (RABE) constructions (a) essentially utilize the complete subtree (CS) scheme for revocation purpose, (b) are bounded in the sense that the size of the public parameters depends linearly on the size of the attribute universe and logarithmically on the number of users in the system, and (c) are either selectively secure, which seems unrealistic in a dynamic system such as RABE, or fully secure but built in a composite order bilinear group setting, which is undesirable from the point of view of both efficiency and security. This paper presents the *first fully secure unbounded* RABE using *subset difference* (SD) mechanism for revocation which greatly improves the broadcast efficiency compared to the CS scheme. Our RABE scheme is built on a prime order bilinear group setting resulting in practical computation cost, and its security depends on the Decisional Linear assumption.

Keywords: attribute-based encryption, revocable attribute-based encryption, key revocation, subset difference method, prime order bilinear groups.

1 Introduction

Attribute-based encryption (ABE) enforces an access control mechanism over encrypted data using access policies and ascribed attributes among private keys and ciphertexts. The notion of ABE was first put forward by Sahai and Waters [20] and refined by many subsequent works [9], [3], [18], [22]. In a (key-policy) ABE system, an encryptor may specify a set of attributes, which could be any keyword describing the ciphertext, directly while encrypting a certain plaintext. A user in the system possesses a key associated with an access policy, stating what kind of ciphertext it can decrypt. Users' private keys are distributed from a trusted key generation center. In such a system, a user can decrypt a ciphertext if the policy associated with its key satisfies the attribute set associated with the ciphertext. ABE has been employed in several important real-life applications including pay-TV system with package policy and cloud based data-sharing services.

While using in a practical situation, an ABE scheme must be supported with an efficient revocation mechanism that can handle dynamic credentials of users, since a user's credentials can be revealed or expired. *Revocable attribute-based encryption* (RABE) is an extension of ABE that provides enhanced control over dynamic user-credentials. Observe that in many ABE based practical applications it is extremely important to be able to revoke individual users rather than a class of users possessing some particular attribute at a time. For instance, in a pay-TV system it is essential for the broadcasting company to have the control to restrict a user from viewing subsequently broadcasted programs once its subscription expires without affecting other subscribed users.

In 2008, Boldyreva et al. [4] constructed the first RABE scheme. In order to implement revocation, their RABE scheme uses a key update approach that roughly works as follows: The encryptor will encrypt a plaintext with an attribute set Γ as usual, and in addition, it also specifies the time slot attribute for which the ciphertext can be decrypted. The key generation center manages the revoked user list and periodically announces a key update material at each time slot, so that only non-revoked users can update their keys and use it to decrypt ciphertexts encrypted for the legitimate time. This revocation approach termed as *indirect* revocation in the literature. The prime disadvantage of this approach is that, the key update phase can be a bottleneck, since it requires communication from the key generation center to all non-revoked users at all time periods and the encryptor cannot get a direct control on the revocation list which may be unsuitable for many practical applications. For instance, in pay-TV system, the encryptor, i.e., the TV program distributor company needs direct control on the revocation list to revoke pirate keys instantly without bothering other legitimate customers. In order to avoid this bottleneck of key update phase, subsequent works [2], [1], [19] adopted a *direct* revocation methodology, which allows encryptors to specify the revocation list directly when encrypting and thus enables instant revocation rather than requiring a key update phase.

Although, a number of RABE schemes exist in the literature [4], [2], [1], [19], the main design principle of these constructions essentially follows that of Boldyreva et al. [4] and employs the complete subtree (CS) scheme of Naor et al. [15] for user revocation. Replacing the CS technique by the *subset difference method* (SD) [15] or the *layered subset difference method* (LSD) [10] can reduce the size of the ciphertext component meant for enforcing revocation from $O(\widehat{r}\log\frac{N_{\max}}{\widehat{r}})$ to $O(\widehat{r})$ where $N_{\max}$ and $\widehat{r}$ respectively denote the total number of users and number of revoked users. This can provide significant improvement in the broadcast efficiency particularly when the number of users present in the system is very large compared to the number of revoked users.

Recently Lee et al. [12], utilized the SD scheme to manage revocation for identity-based encryption (IBE) and pointed out that their technique for RIBE cannot be extended to realize RABE via SD scheme.

Further, a desirable property of a RABE scheme is its *independence* from the size of the attribute universe and the total number of users supported by the system. In all previous RABE schemes [2], [1], [19], [4], the public parameter size grows linearly with the number of attributes in the attribute universe and logarithmically with the number of users. Moreover, all previous RABE schemes except [19] provide only *selective security* which seems unrealistic in a dynamic system such as RABE. Although [19] achieves *full security*, it is built on a composite order bilinear group setting under non-standard assumptions. Note that the bit length of group elements is very large, as well as, group-operations and pairing computations are prohibitively slow [8] in composite order bilinear groups than a comparable prime order group.

Our Contributions

Communication bandwidth is of greater concern than user storage in present day applications like pay-TV and many others. Our goal in this work is to explore applicability of SD mechanism in ABE setting to design an *unbounded* RABE with *reduced communication bandwidth* and simultaneously achieve *full security.* The starting point of our work is the ABE construction of Okamoto et al. [17]. However, integrating SD revocation with ABE or replacing CS scheme by SD technique to construct a broadcast efficient RABE seems to be a quite challenging task.

We construct the *first fully secure unbounded* (key-policy) RABE scheme supporting *direct* revocation employing *subset difference* (SD) mechanism. Towards this end, we build a rather non-trivial technique in order to integrate the SD scheme with the ABE construction of [17]. On a more positive note, the full security of our RABE scheme is based on standard assumption, namely, the *Decisional Linear* (DLIN) assumption. Most importantly, due to the use of *prime order bilinear group* and *subset difference* revocation scheme, the *broadcast efficiency* of our RABE is much *higher* compared to the existing constructions [2], [1], [19] with reasonable computation cost. Furthermore, the proposed RABE scheme is the *first to achieve constant size public parameters* and thus overcomes the bottleneck of accommodating large attribute universe and an unbounded number of users.

Note that, as opposed to the CS scheme, an assigned key for a subset in SD scheme depends on the keys of some other subsets. This interdependence of keys makes the use of SD method in attribute-based setting quite challenging. We assign uniformly and independently chosen secrets to each users and split that secret into two parts– one for the access structure and the other for revocation. This latter part is further subdivided into random secrets to solve the complex key assignment problem of the SD method. We integrate SD with ABE by enforcing the condition that an user can retrieve the first part of its secret if and only if its access structure is satisfied by the set of attributes specified in the ciphertext, and all the subdivided components of the other part of its secret can be extracted if and only if its subscription is valid according to the conditions of SD scheme.

For proving security of our RABE scheme, the main intricacy of this work, we utilize the (*extended*) *dual system encryption* methodology over *dual pairing vector spaces* introduced in [17]. However, in order to adopt the technique of [17], for our RABE scheme, we extend some of the problems and methodology such as *indexing* and *consistent randomness amplification* employed in [17].

Although we use the monotone version of the ABE scheme of [17] to present our RABE construction for simplicity, we would like to mention that our technique can also be applied to combine the original non-monotone ABE scheme of [17] with the SD method.

2 Preliminaries

2.1 Notations

- $y \xleftarrow{\$} A$: y is randomly selected from A according to its distribution, when A is a random variable, and y is uniformly selected from A, when A is a set.
- $\mathcal{G} \to x$: x is the output of the algorithm or experiment $\mathcal{G}$.
- $\vec{x}$: a vector $(x_1, \ldots, x_n) \in \mathbb{F}_q^n$ of length n for some $n \in \mathbb{N}$.
- $\boldsymbol{x}$: an element of vector space $\mathbb{V} \neq \mathbb{F}_q^n$.
- $\mathsf{span}\langle \boldsymbol{b}_1, \ldots, \boldsymbol{b}_m \rangle \subseteq \mathbb{V}$: the subspace of $\mathbb{V}$ generated by $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_m\} \subseteq \mathbb{V}$.
- $\mathsf{span}\langle \vec{x}_1, \ldots, \vec{x}_m \rangle \subseteq \mathbb{F}_q^n$: the subspace of $\mathbb{F}_q^n$ spanned by $\{\vec{x}_1, \ldots, \vec{x}_m\} \subseteq \mathbb{F}_q^n$.
- $(x_1, \ldots, x_m)_{\mathbb{B}}$: $\sum_{i=1}^{m} x_i \boldsymbol{b}_i$ that is a linear combination of vectors in $\mathbb{B} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_m\} \subseteq \mathbb{V}$ with scalars $x_1, \ldots, x_m$.
- $\mathsf{GL}(m, \mathbb{F}_q)$: The general linear group of degree m over $\mathbb{F}_q$.

2.2 Dual Pairing Vector Spaces by Direct Product of Symmetric Pairing Groups

Definition 1 (Symmetric Bilinear Pairing Groups). *A symmetric bilinear pairing group $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ is a tuple of a prime q, cyclic additive group $\mathbb{G}$ and multiplicative group $\mathbb{G}_T$ of order q each, $G \neq 0 \in \mathbb{G}$, and a polynomial time computable non-degenerate bilinear pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, i.e., $e(sG, tG) = e(G, G)^{st}$ for all $s, t \in \mathbb{F}_q$ (bilinearity) and $e(G, G) \neq 1$ (non-degeneracy). Let $\mathcal{G}_{\mathsf{bpg}}$ be an algorithm that takes input 1^λ and outputs a description of bilinear pairing group $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ with security parameter λ.*

Definition 2 (Dual Pairing Vector Spaces (DPVS)). *As introduced in* [16]*, a dual pairing vector space* (DPVS) *$(q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, E)$ by a direct product of symmetric pairing groups $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ is a tuple of prime q, n dimensional vector space $\mathbb{V} = \mathbb{G}^n = \overbrace{\mathbb{G} \times \ldots \times \mathbb{G}}^{n}$ over $\mathbb{F}_q$, cyclic* *group* $\mathbb{G}_T$ *of order* q*, canonical basis* $\mathbb{A} = \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n\}$ *of* $\mathbb{V}$*, where* $\boldsymbol{a}_i = (\overbrace{0, \ldots, 0}^{i-1}, G, \overbrace{0, \ldots, 0}^{n-i})$*, and pairing* $E : \mathbb{V} \times \mathbb{V} \to \mathbb{G}_T$*. The pairing* E *is defined by* $E(\boldsymbol{x}, \boldsymbol{y}) = \prod_{i=1}^{n} e(G_i, H_i) \in \mathbb{G}_T$ *where* $\boldsymbol{x} = (G_1, \ldots, G_n) \in \mathbb{V}$ *and* $\boldsymbol{y} = (H_1, \ldots, H_n) \in \mathbb{V}$*. The map* E *is non-degenerate bilinear, i.e.,* $E(s\boldsymbol{x}, t\boldsymbol{y}) = E(\boldsymbol{x}, \boldsymbol{y})^{st}$ *for* $s, t \in \mathbb{F}_q$*, and if* $E(\boldsymbol{x}, \boldsymbol{y}) = 1$ *for all* $\boldsymbol{y} \in \mathbb{V}$*, then* $\boldsymbol{x} = 0$*. For all* i *and* j*,* $E(\boldsymbol{a}_i, \boldsymbol{a}_j) = e(G, G)^{\delta_{i,j}}$ *where* $\delta_{i,j} = 1$ *if* $i = j$*, and 0 otherwise, and* $e(G, G) \neq 1 \in \mathbb{G}_T$*.* DPVS *generation algorithm* $\mathcal{G}_{\mathsf{dpvs}}$ *takes input* 1^λ $(\lambda \in \mathbb{N})$*,* $n \in \mathbb{N}$ *together with* $\mathsf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e)$*, and outputs a description of* $\mathsf{param}_{\mathbb{V}} = (q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, E)$ *with security parameter* λ *and* n*-dimensional* $\mathbb{V}$*. It can be constructed by using* $\mathcal{G}_{\mathsf{bpg}}$ *as a subroutine.*

For matrix $\boldsymbol{W} = (w_{i,j})_{i,j=1,\ldots,n} \in \mathbb{F}_q^{n \times n}$ *and element* $\boldsymbol{x} = (G_1, \ldots, G_n)$ *in* n*-dimensional* $\mathbb{V}$*,* $\boldsymbol{xW}$ *denotes* $\left(\sum_{i=1}^{n} G_i w_{i,1}, \ldots, \sum_{i=1}^{n} G_i w_{i,n}\right) = \left(\sum_{i=1}^{n} w_{i,1} G_i, \ldots, \sum_{i=1}^{n} w_{i,n} G_i\right)$ *by a natural multiplication of an* n*-dimensional row vector and an* $n \times n$ *matrix. Thus it satisfies an associative law, i.e.,* $(\boldsymbol{xW}_1)\boldsymbol{W}_2 = \boldsymbol{x}(\boldsymbol{W}_1\boldsymbol{W}_2)$*.*

In Figure 1 we describe *random dual orthogonal basis generator* $\mathcal{G}_{\mathsf{ob}}$, which is used as a subroutine in our RABE scheme.

$\mathcal{G}_{\mathsf{ob}}(1^\lambda, (n_t)_{t=0,1})$: This algorithm performs the following operations:

- Generate $(\mathsf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\$} \mathcal{G}_{\mathsf{bpg}}(1^\lambda)$, $\psi \xleftarrow{\$} \mathbb{F}_q^\times$, where $\mathbb{F}_q^\times = \mathbb{F}_q \backslash \{0\}$.
- For $t = 0, 1$ execute the following:
  - Obtain $\mathsf{param}_{\mathbb{V}_t} = (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, E) \xleftarrow{\$} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, n_t, \mathsf{param}_{\mathbb{G}})$ such that $\mathbb{V}_t = \overbrace{\mathbb{G} \times \ldots \times \mathbb{G}}^{n_t}$ and $\mathbb{A}_t = \{\boldsymbol{a}_{t,1}, \ldots, \boldsymbol{a}_{t,n_t}\}$ is the canonical basis of $\mathbb{V}_t$.
  - Choose $\boldsymbol{X}_t = (\chi_{t,i,j})_{i,j=1,\ldots,n_t} \xleftarrow{\$} \mathsf{GL}(n_t, \mathbb{F}_q)$.
  - Compute $\boldsymbol{X}_t^* = (\vartheta_{t,i,j})_{i,j=1,\ldots,n_t} = \psi(\boldsymbol{X}_t^\intercal)^{-1}$, where $\boldsymbol{Y}^\intercal$ denotes transpose of the matrix $\boldsymbol{Y}$. Hereafter, $\vec{\chi}_{t,i}$ and $\vec{\vartheta}_{t,i}$ represent the i-th rows of $\boldsymbol{X}_t$ and $\boldsymbol{X}_t^*$ respectively, for $i = 1, \ldots, n_t$. Note that, for $i, i' = 1, \ldots, n_t$, $\vec{\chi}_{t,i} \cdot \vec{\vartheta}_{t,i'} = \sum_{j=1}^{n_t} \chi_{t,i,j} \vartheta_{t,i',j} = \psi$, if $i = i'$, and 0, otherwise.
  - Set $\boldsymbol{b}_{t,i} = (\vec{\chi}_{t,i})_{\mathbb{A}_t} = \sum_{j=1}^{n_t} \chi_{t,i,j} \boldsymbol{a}_{t,j} = (\chi_{t,i,1} G, \ldots, \chi_{t,i,n_t} G)$, $\boldsymbol{b}_{t,i}^* = (\vec{\vartheta}_{t,i})_{\mathbb{A}_t} = \sum_{j=1}^{n_t} \vartheta_{t,i,j} \boldsymbol{a}_{t,j} = (\vartheta_{t,i,1} G, \ldots, \vartheta_{t,i,n_t} G)$ for $i = 1, \ldots n_t$, and define $\mathbb{B}_t = \{\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,n_t}\}$, $\mathbb{B}_t^* = \{\boldsymbol{b}_{t,1}^*, \ldots, \boldsymbol{b}_{t,n_t}^*\}$.
- Compute $g_T = e(G, G)^\psi$ and set $\mathsf{param} = (\{\mathsf{param}_{\mathbb{V}_t}\}_{t=0,1}, g_T)$.
- Return $(\mathsf{param}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,1})$.

Observe that

$$
\begin{aligned}
E(\boldsymbol{b}_{t,i}, \boldsymbol{b}_{t,i'}^*) =& E((\vec{\chi}_{t,i})_{\mathbb{A}_t}, (\vec{\vartheta}_{t,i'})_{\mathbb{A}_t}) \\
=& E((\chi_{t,i,1} G, \ldots, \chi_{t,i,n_t} G), (\vartheta_{t,i',1} G, \ldots, \vartheta_{t,i',n_t} G)) \\
=& \prod_{j=1}^{n_t} e(G, G)^{\chi_{t,i,j} \vartheta_{t,i',j}} = e(G, G)^{\vec{\chi}_{t,i} \cdot \vec{\vartheta}_{t,i'}} \\
=& g_T, \text{ if } i = i', \text{ and } 0, \text{ otherwise for } t = 0, 1; i, i' = 1, \ldots, n_t.
\end{aligned}
$$

Fig. 1: Dual orthogonal basis generator $\mathcal{G}_{\mathsf{ob}}(1^\lambda, (n_t)_{t=0,1})$

Henceforth, for simplicity, we denote $n = n_1$, $\mathbb{V} = \mathbb{V}_1$, $\mathbb{A} = \mathbb{A}_1$, $\mathbb{B} = \mathbb{B}_1$ and $\mathbb{B}^* = \mathbb{B}_1^*$ for variables with $t = 1$.

2.3 Complexity Assumptions Derived from the Decisional Linear (DLIN) Assumption

Definition 3 (DLIN: Decisional Linear Assumption). *The* DLIN *problem is to guess* $\beta \in \{0,1\}$*, given* $\varrho = (\mathsf{param}_{\mathbb{G}}, G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta) \xleftarrow{\$} \mathcal{G}_\beta^{\mathsf{DLIN}}(1^\lambda)$*, where*

$$\begin{gathered}\mathcal{G}_\beta^{\mathsf{DLIN}}(1^\lambda):\ (\mathbf{param}_{\mathbb{G}} = (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\$} \mathcal{G}_{\mathsf{bpg}}(1^\lambda),\\ \kappa, \delta, \xi, \sigma \xleftarrow{\$} \mathbb{F}_q,\ Y_0 = (\delta + \sigma)G,\ Y_1 \xleftarrow{\$} \mathbb{G},\\ \text{return } \varrho = (\mathsf{param}_{\mathbb{G}}, G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta),\end{gathered}$$

For a probabilistic machine $\mathcal{F}$*, we define the advantage of* $\mathcal{F}$ *for the* DLIN *problem as:*

$$\mathsf{Adv}_{\mathcal{F}}^{\mathsf{DLIN}}(\lambda) = \left|\mathsf{Pr}\left[\mathcal{F}(1^\lambda, \varrho) \to 1 | \varrho \xleftarrow{\$} \mathcal{G}_0^{\mathsf{DLIN}}(1^\lambda)\right] - \mathsf{Pr}\left[\mathcal{F}(1^\lambda, \varrho) \to 1 | \varrho \xleftarrow{\$} \mathcal{G}_1^{\mathsf{DLIN}}(1^\lambda)\right]\right|.$$

The DLIN *assumption states that for any probabilistic polynomial-time adversary* $\mathcal{F}$*, the advantage* $\mathsf{Adv}_{\mathcal{F}}^{\mathsf{DLIN}}(\lambda)$ *is negligible in* λ*.*

The validity of the DLIN assumption in the generic group model has been established by Boneh et al. [5].

Definition 4 (Problem 1). *Problem 1 is to guess* $\beta \in \{0,1\}$ *given* $\varrho = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}}_0^*, \widehat{\mathbb{B}}, \widehat{\mathbb{B}}^*, \boldsymbol{e}_{\beta,0}, \{\boldsymbol{e}_{\beta,t,i}\}_{t=1,\ldots,d;i=1,2}, \{\boldsymbol{e}_{\beta,d+\upsilon,\varpi,i}\}_{\upsilon=1,2,\varpi=1,\ldots,\widehat{r}_{\max},i=1,2}) \xleftarrow{\$} \mathcal{G}_\beta^{\mathsf{P1}}(1^\lambda, d, \widehat{r}_{\max})$*, where*

```
$\mathcal{G}_\beta^{\mathsf{P1}}(1^\lambda, d, \widehat{r}_{\max}):\ (\mathsf{param}, \{\mathbb{B}_0, \mathbb{B}_0^*\}, \{\mathbb{B}, \mathbb{B}^*\}) \xleftarrow{\$} \mathcal{G}_{\mathsf{ob}}(1^\lambda, (n_0 = 5, n = 16))$,
$\varphi_0, \omega \xleftarrow{\$} \mathbb{F}_q,\ \tau \xleftarrow{\$} \mathbb{F}_q^\times$,
$\widehat{\mathbb{B}}_0 = \{\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}\},\ \widehat{\mathbb{B}} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_4, \boldsymbol{b}_{13}, \boldsymbol{b}_{14}\}$,
$\widehat{\mathbb{B}}_0^* = \{\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*, \boldsymbol{b}_{0,4}^*\},\ \widehat{\mathbb{B}}^* = \{\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_4^*, \boldsymbol{b}_{11}^*, \boldsymbol{b}_{12}^*\}$,
$\boldsymbol{e}_{0,0} = (\omega, 0, 0, 0, \varphi_0)_{\mathbb{B}_0},\ \boldsymbol{e}_{1,0} = (\omega, \tau, 0, 0, \varphi_0)_{\mathbb{B}_0}$,
$\vec{e}_1 = (1,0),\ \vec{e}_2 = (0,1) \in \mathbb{F}_q^2$,

for $t = 1, \ldots, d$,
    $\boldsymbol{Z}_t \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$,
    for $i = 1, 2$,
        $\sigma_{t,i}, \varphi_{t,i,1}, \varphi_{t,i,2} \xleftarrow{\$} \mathbb{F}_q$,
        $\boldsymbol{e}_{0,t,i} = (\sigma_{t,i}(1,t), \omega\vec{e}_i, 0^6, 0^2, \varphi_{t,i,1}, \varphi_{t,i,2}, 0^2)_{\mathbb{B}}$,
        $\boldsymbol{e}_{1,t,i} = (\sigma_{t,i}(1,t), \omega\vec{e}_i, \tau\vec{e}_i, 0^2, \tau\vec{e}_i\boldsymbol{Z}_t, 0^2, \varphi_{t,i,1}, \varphi_{t,i,2}, 0^2)_{\mathbb{B}}$,

for $\upsilon = 1, 2; \varpi = 1, \ldots, \widehat{r}_{\max}$,
    $\boldsymbol{Z}_{d+\upsilon,\varpi} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$,
    for $i = 1, 2$,
        $\sigma_{d+\upsilon,\varpi,i}, \varphi_{d+\upsilon,\varpi,i,1}, \varphi_{d+\upsilon,\varpi,i,2} \xleftarrow{\$} \mathbb{F}_q$,
        $\boldsymbol{e}_{0,d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega\vec{e}_i, 0^6, 0^2, \varphi_{d+\upsilon,\varpi,i,1}, \varphi_{d+\upsilon,\varpi,i,2}, 0^2)_{\mathbb{B}}$,
        $\boldsymbol{e}_{1,d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega\vec{e}_i, \tau\vec{e}_i, 0^2, \tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \varphi_{d+\upsilon,\varpi,i,1}, \varphi_{d+\upsilon,\varpi,i,2}, 0^2)_{\mathbb{B}}$,
return $\varrho = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}}_0^*, \widehat{\mathbb{B}}, \widehat{\mathbb{B}}^*, \boldsymbol{e}_{\beta,0}, \{\boldsymbol{e}_{\beta,t,i}\}_{t=1,\ldots,d;i=1,2}, \{\boldsymbol{e}_{\beta,d+\upsilon,\varpi,i}\}_{\upsilon=1,2;\varpi=1,\ldots,\widehat{r}_{\max};i=1,2})$.
```

For a probabilistic adversary $\mathcal{B}$ *the advantage of* $\mathcal{B}$ *for Problem 1 is given by*

$$\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P1}}(\lambda) = \left|\mathsf{Pr}\left[\mathcal{B}(1^\lambda, \varrho) \to 1 | \varrho \xleftarrow{\$} \mathcal{G}_0^{\mathsf{P1}}(1^\lambda, d, \widehat{r}_{\max})\right] - \mathsf{Pr}\left[\mathcal{B}(1^\lambda, \varrho) \to 1 | \varrho \xleftarrow{\$} \mathcal{G}_1^{\mathsf{P1}}(1^\lambda, d, \widehat{r}_{\max})\right]\right|.$$

Lemma 1. *Problem 1 is computationally intractable under the* DLIN *assumption. Formally, for any probabilistic polynomial-time adversary* $\mathcal{B}$ *there exist probabilistic machines* $\mathcal{F}_1$*,* $\mathcal{F}_2$*,* $\mathcal{F}_3$*,* *whose running times are essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}}(\lambda) \leq \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_1}(\lambda) + \sum_{p=1}^{d}\sum_{j=1}^{2}\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}j}}(\lambda) + \sum_{\upsilon=1}^{2}\sum_{\varpi=1}^{\widehat{r}_{\max}}\sum_{j=1}^{2}\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j}(\lambda) + \epsilon,$$

where $\mathcal{F}_{2\text{-}p\text{-}j}(\cdot) = \mathcal{F}_2(p, j, \cdot)$, $\mathcal{F}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j}(\cdot) = \mathcal{F}_3(d+\upsilon, \varpi, j, \cdot)$ and $\epsilon = [5 + 10d + 20\widehat{r}_{\max}]/q$.

Definition 5 (Problem 2). *Problem 2 is to guess $\beta \in \{0,1\}$, given $(\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}}^*_0, \widehat{\mathbb{B}}, \widehat{\mathbb{B}}^*, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \{\boldsymbol{h}^*_{\beta,t,i}, \boldsymbol{e}_{t,i}\}_{t=1,\dots,d;i=1,2}, \{\boldsymbol{h}^*_{\beta,d+\upsilon,\varpi,i}, \boldsymbol{e}_{d+\upsilon,\varpi,i}\}_{\upsilon=1,2;\varpi=1,\dots,\aleph;i=1,2}) \xleftarrow{\$} \mathcal{G}^{\mathsf{P2}}_{\beta}(1^\lambda, d, N_{\max}, \widehat{r}_{\max})$, where*

$$\mathcal{G}^{\mathsf{P2}}_{\beta}(1^\lambda, d, N_{\max}, \widehat{r}_{\max}): \ (\mathsf{param}, \{\mathbb{B}_0, \mathbb{B}^*_0\}, \{\mathbb{B}, \mathbb{B}^*\}) \xleftarrow{\$} \mathcal{G}_{\mathsf{ob}}(1^\lambda, (n_0 = 5, n = 16)),$$
$$\gamma, \eta_0, \varphi_0, \omega \xleftarrow{\$} \mathbb{F}_q, \ \ \tau, \delta \xleftarrow{\$} \mathbb{F}_q^{\times},$$
$$\widehat{\mathbb{B}}_0 = \{\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}\}, \ \ \widehat{\mathbb{B}} = \{\boldsymbol{b}_1, \dots, \boldsymbol{b}_4, \boldsymbol{b}_{13}, \boldsymbol{b}_{14}\},$$
$$\widehat{\mathbb{B}}^*_0 = \{\boldsymbol{b}^*_{0,1}, \dots, \boldsymbol{b}^*_{0,4}\}, \ \ \widehat{\mathbb{B}}^* = \{\boldsymbol{b}^*_1, \dots, \boldsymbol{b}^*_4, \boldsymbol{b}^*_{11}, \boldsymbol{b}^*_{12}\},$$
$$\boldsymbol{h}^*_{0,0} = (\gamma, 0, 0, \eta_0, 0)_{\mathbb{B}^*_0}, \ \ \boldsymbol{h}^*_{1,0} = (\gamma, \delta, 0, \eta_0, 0)_{\mathbb{B}^*_0}, \ \ \boldsymbol{e}_0 = (\omega, \tau, 0, 0, \varphi_0)_{\mathbb{B}_0},$$
$$\vec{e}_1 = (1,0), \vec{e}_2 = (0,1) \in \mathbb{F}_q^2,$$

$$\textit{for } t = 1, \dots, d,$$
$$\quad \boldsymbol{Z}_t \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q), \ \ \boldsymbol{U}_t = (\boldsymbol{Z}_t^{-1})^{\mathsf{T}},$$
$$\quad \textit{for } i = 1,2,$$
$$\quad\quad \mu_{t,i}, \sigma_{t,i}, \eta_{t,i,1}, \eta_{t,i,2}, \varphi_{t,i,1}, \varphi_{t,i,2} \xleftarrow{\$} \mathbb{F}_q,$$
$$\quad\quad \boldsymbol{h}^*_{0,t,i} = (\mu_{t,i}(t,-1), \gamma\vec{e}_i, \quad 0^4, \quad\quad 0^2, \quad \eta_{t,i,1}, \eta_{t,i,2}, 0^2, 0^2)_{\mathbb{B}^*},$$
$$\quad\quad \boldsymbol{h}^*_{1,t,i} = (\mu_{t,i}(t,-1), \gamma\vec{e}_i, \quad 0^4, \quad \delta\vec{e}_i\boldsymbol{U}_t, \eta_{t,i,1}, \eta_{t,i,2}, 0^2, 0^2)_{\mathbb{B}^*},$$
$$\quad\quad \boldsymbol{e}_{t,i} = \ (\sigma_{t,i}(1,t), \omega\vec{e}_i, \ \tau\vec{e}_i, 0^2, \tau\vec{e}_i\boldsymbol{Z}_t, 0^2, \varphi_{t,i,1}, \varphi_{t,i,2}, 0^2)_{\mathbb{B}},$$

$$\textit{for } \upsilon = 1, 2; \varpi = 1, \dots, \log^2 N_{\max} + \widehat{r}_{\max},$$
$$\quad \boldsymbol{Z}_{d+\upsilon,\varpi} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q), \ \ \boldsymbol{U}_{d+\upsilon,\varpi} = (\boldsymbol{Z}^{-1}_{d+\upsilon,\varpi})^{\mathsf{T}},$$
$$\quad \textit{for } i = 1, 2,$$
$$\quad\quad \mu_{d+\upsilon,\varpi,i}, \sigma_{d+\upsilon,\varpi,i}, \eta_{d+\upsilon,\varpi,i,1}, \eta_{d+\upsilon,\varpi,i,2}, \varphi_{d+\upsilon,\varpi,i,1}, \varphi_{d+\upsilon,\varpi,i,,2} \xleftarrow{\$} \mathbb{F}_q,$$
$$\quad\quad \boldsymbol{h}^*_{0,d+\upsilon,\varpi,i} = (\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1), \gamma\vec{e}_i, \quad 0^4, \quad\quad 0^2, \quad \eta_{d+\upsilon,\varpi,i,1}, \eta_{d+\upsilon,\varpi,i,2}, 0^2, 0^2)_{\mathbb{B}^*},$$
$$\quad\quad \boldsymbol{h}^*_{1,d+\upsilon,\varpi,i} = (\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1), \gamma\vec{e}_i, \quad 0^4, \quad \delta\vec{e}_i\boldsymbol{U}_{d+\upsilon,\varpi}, \eta_{d+\upsilon,\varpi,i,1}, \eta_{d+\upsilon,\varpi,i,2}, 0^2, 0^2)_{\mathbb{B}^*},$$
$$\quad\quad \boldsymbol{e}_{d+\upsilon,\varpi,i} = \ (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega\vec{e}_i, \ \tau\vec{e}_i, 0^2, \tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \varphi_{d+\upsilon,\varpi,i,1}, \varphi_{d+\upsilon,\varpi,i,2}, 0^2)_{\mathbb{B}},$$
$$\textit{return } \ \varrho = \ (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}}^*_0, \widehat{\mathbb{B}}, \widehat{\mathbb{B}}^*, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \{\boldsymbol{h}^*_{\beta,t,i}, \boldsymbol{e}_{t,i}\}_{t=1,\dots,d;i=1,2},$$
$$\quad\quad \{\boldsymbol{h}^*_{\beta,d+\upsilon,\varpi,i}, \boldsymbol{e}_{d+\upsilon,\varpi,i}\}_{\upsilon=1,2;\varpi=1,\dots,\aleph;i=1,2}).$$

For a probabilistic adversary $\mathcal{B}$, the advantage of $\mathcal{B}$ for Problem 2 is given by

$$\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda) = \Bigg| \mathsf{Pr}\left[\mathcal{B}(1^\lambda, \varrho) \to 1 | \varrho \xleftarrow{\$} \mathcal{G}^{\mathsf{P2}}_0(1^\lambda, d, N_{\max}, \widehat{r}_{\max})\right] - \mathsf{Pr}\left[\mathcal{B}(1^\lambda, \varrho) \to 1 | \varrho \xleftarrow{\$} \mathcal{G}^{\mathsf{P2}}_1(1^\lambda, d, N_{\max}, \widehat{r}_{\max})\right] \Bigg|.$$

Lemma 2. *Problem 2 is computationally intractable under the* DLIN *assumption. More formally, for any probabilistic polynomial-time adversary $\mathcal{B}$, there exist probabilistic machines $\mathcal{F}_1, \mathcal{F}_{2\text{-}1}, \dots, \mathcal{F}_{2\text{-}11}$, whose running times are essentially the same as that of $\mathcal{B}$, such that for* *any security parameter* λ,

$$\begin{aligned}
&\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda) \le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_1}(\lambda) + \sum_{j=1}^{2}\left[\sum_{p=1}^{d}\left\{\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}1\text{-}j}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}2\text{-}j}}(\lambda) + \sum_{\substack{l=1\\ l\neq p}}^{d+2}\Big(\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}3\text{-}j\text{-}l}}(\lambda) +\right.\right.\\
&\left.\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}4\text{-}j\text{-}l}}(\lambda)\Big) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}5\text{-}j}}(\lambda)\right\} + \sum_{\upsilon=1}^{2}\sum_{\varpi=1}^{\aleph}\left\{\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}6\text{-}j}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}7\text{-}j}}(\lambda) +\right.\\
&\qquad\sum_{\substack{l=1\\ l\neq d+\upsilon}}^{d+2}\Big(\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}j\text{-}l}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}9\text{-}j\text{-}l}}(\lambda)\Big) +\\
&\qquad\left.\left.\sum_{\substack{\iota=1\\ \iota\neq\varpi}}^{\aleph}\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}10\text{-}j\text{-}\iota}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}11\text{-}j}}(\lambda)\right\}\right] + \epsilon,
\end{aligned}$$

where

$$\begin{aligned}
&\mathcal{F}_{2\text{-}p\text{-}1\text{-}j}(\cdot) = \mathcal{F}_{2\text{-}1}(p, j, \cdot), \mathcal{F}_{2\text{-}p\text{-}2\text{-}j}(\cdot) = \mathcal{F}_{2\text{-}2}(p, j, \cdot),\\
&\mathcal{F}_{2\text{-}p\text{-}3\text{-}j\text{-}l}(\cdot) = \mathcal{F}_{2\text{-}3}(p, j, l, \cdot), \mathcal{F}_{2\text{-}p\text{-}4\text{-}j\text{-}l}(\cdot) = \mathcal{F}_{2\text{-}4}(p, j, l, \cdot),\\
&\mathcal{F}_{2\text{-}p\text{-}5\text{-}j}(\cdot) = \mathcal{F}_{2\text{-}5}(p, j, \cdot), \mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}6\text{-}j}(\cdot) = \mathcal{F}_{2\text{-}6}(d+\upsilon, \varpi, j, \cdot) \text{ etc.},
\end{aligned}$$

$\aleph = \log^2 N_{\max} + \widehat{r}_{\max}$ *and* $\epsilon = \left[5 + 40d + 10d^2 + 2\aleph(30 + 10d + 10\aleph)\right]/q$.

Problems 1 and 2 are extended from Problems 1-ABE and 2-ABE in [17] respectively. We provide the proofs of Lemmas 1 and 2 respectively in Appendix B and Appendix C.

Remark 1. Note that in Problems 1 and 2 the coefficients of $\{\boldsymbol{b}_7, \boldsymbol{b}_8\}$, $\{\boldsymbol{b}^*_7, \boldsymbol{b}^*_8\}$ and $\{\boldsymbol{b}_{15}, \boldsymbol{b}_{16}\}$, $\{\boldsymbol{b}^*_{15}, \boldsymbol{b}^*_{16}\}$ are always set to zero. One may wonder about the requirement of these additional subspaces. However, note that these subspaces would be extremely important while reducing Problems 1 and 2 to the DLIN problem (see the proof of Lemma 39 in Appendix C).

2.4 Monotone Span Programs and Access Structures

Definition 6 (Monotone Span Programs). *Let* $\{p_1, \ldots, p_m\}$ *be a set of variables. A monotone span program over* $\mathbb{F}_q$ *is a labeled matrix* $\widehat{\boldsymbol{M}} = (\boldsymbol{M}, \rho)$ *where* $\boldsymbol{M}$ *is a* $\ell \times r$ *matrix over* $\mathbb{F}_q$ *and* ρ *is a labeling of rows of* $\boldsymbol{M}$ *by variables from* $\{p_1, \ldots, p_m\}$ *(every row is labeled by one variable), i.e.,* $\rho : \{1, \ldots, \ell\} \to \{p_1, \ldots, p_m\}$.

A monotone span program accepts or rejects an input by the following criterion. For every input sequence $\varsigma \in \{0, 1\}^m$*, define the submatrix* $\boldsymbol{M}_\varsigma$ *of* $\boldsymbol{M}$ *consisting of those rows whose labels are set to 1 by the input* ς*, i.e., rows labeled by some* p_i *such that* $\varsigma_i = 1$*. The span program* $\widehat{\boldsymbol{M}}$ *accepts* ς *if and only if* $\overrightarrow{1} \in \mathsf{span}\langle \boldsymbol{M}_\varsigma \rangle = \mathsf{span}\langle \boldsymbol{M}_j | \rho(j) = p_i \ \wedge \ \varsigma_i = 1 \rangle$*, i.e., some linear combination of the rows of* $\boldsymbol{M}_\varsigma$ *gives the all one vector* $\overrightarrow{1}$*. Monotone span programs compute monotone Boolean functions.*

We assume that no row $\boldsymbol{M}_i$ $(i = 1, \ldots, \ell)$ of the matrix $\boldsymbol{M}$ is $\overrightarrow{0}$. We now define a monotone access structure that is essentially the same as the one defined in [17]. This access structure will be employed in our proposed RABE scheme.

Definition 7 (Monotone Access Structure). *Let* $\mathbb{U}$ *be an universe of attributes each of which is expressed by a pair of attribute id and value of attribute, i.e.,* $\mathbb{U} = \{(t, A_t) | t \in \{1, \ldots, d\} \wedge A_t \in \mathbb{F}_q\}$*. We now define such an attribute* (t, A_t) *to be a variable* p_t *of a monotone span program* $\widehat{\boldsymbol{M}} = (\boldsymbol{M}, \rho)$*, i.e.,* $p_t = (t, A_t)$*. A monotone access structure* $\mathbb{S}$ *is monotone span program*

$\widehat{\boldsymbol{M}} = (\boldsymbol{M}, \rho)$ along with variables $p_t = (t, A_t)$ for $t \in \{1, \ldots, d\}$, i.e., $\mathbb{S} = (\boldsymbol{M}, \rho)$ such that $\rho : \{1, \ldots, \ell\} \to \{p_1, \ldots, p_d\}$.

Let Γ be a set of attributes, i.e., $\Gamma \subseteq \mathbb{U}$. The access structure $\mathbb{S} = (\boldsymbol{M}, \rho)$ accepts Γ if and only if $\vec{1} \in \mathsf{span}\langle \boldsymbol{M}_i | \rho(i) = (t, A_t) \in \Gamma \rangle$.

We now describe a linear secret-sharing scheme for a monotone access structure or span program.

Definition 8 (Linear Secret-Sharing Scheme). *Let $\widehat{\boldsymbol{M}} = (\boldsymbol{M}, \rho)$ be a monotone span program where $\boldsymbol{M}$ is an $\ell \times r$ matrix over $\mathbb{F}_q$ and ρ is a labeling of the rows of $\boldsymbol{M}$. Let column vector $\vec{f}^{\intercal} = (f_1, \ldots, f_r)^{\intercal} \xleftarrow{\$} \mathbb{F}_q^r$. Then, $s_0 = \vec{1} \cdot \vec{f}^{\intercal} = \sum_{k=1}^{r} f_k$ is the secret to be shared, and $\vec{s}^{\intercal} = (s_1, \ldots, s_\ell)^{\intercal} = \boldsymbol{M} \cdot \vec{f}^{\intercal}$ is the vector of ℓ shares of the secret s_0 and the share s_i belongs to $\rho(i)$.*

If span program $\widehat{\boldsymbol{M}} = (\boldsymbol{M}, \rho)$ accepts $\varsigma \in \{0,1\}^m$ or access structure $\mathbb{S} = (\boldsymbol{M}, \rho)$ accepts Γ, i.e., $\vec{1} \in \mathsf{span}\langle \boldsymbol{M}_i | \varsigma_{\rho(i)} = 1 \rangle$ or $\vec{1} \in \mathsf{span}\langle \boldsymbol{M}_i | \rho(i) \in \Gamma \rangle$, then there exist constants $\{\alpha_i \in \mathbb{F}_q | i \in I\}$ such that $I \subseteq \{i | i \in \{1, \ldots, \ell\} \ \wedge \ [\varsigma_{\rho(i)} = 1 \text{ or } \rho(i) \in \Gamma]\}$ and $\sum_{i \in I} \alpha_i s_i = s_0$. Furthermore, these constants $\{\alpha_i\}$ can be computed in time polynomial in the size of matrix $\boldsymbol{M}$.

2.5 The Notion of Revocable Attribute-Based Encryption (RABE)

• ***Syntax of Revocable Attribute-Based Encryption***: As described in [1], [19], a (key-policy) revocable attribute-based encryption (RABE) scheme that is associated with the attribute universe $\mathbb{U}$, a collection of admissible access structures $\mathfrak{S}$ and message space $\mathbb{M}$, consists of the following algorithms:

$\mathsf{RABE.Setup}(1^\lambda, N_{\max})$: Taking as input a security parameter 1^λ and the maximum number of users $N_{\max}$, the key generation center publishes public parameters PP and a state ST, while generates a master secret key MK for itself.

$\mathsf{RABE.GenKey}(\mathsf{PP}, \mathsf{MK}, \mathsf{ST}, ID, \mathbb{S})$: The key generation center takes as input the public parameters PP, the master secret key MK, the state ST, an user identity ID and the access structure $\mathbb{S} = (\boldsymbol{M}, \rho) \in \mathfrak{S}$ of that user. It provides a private key $\mathsf{SK}_{\mathbb{S},ID}$ to that user and publishes an updated state ST.

$\mathsf{RABE.Encrypt}(\mathsf{PP}, \mathsf{ST}, \Gamma, \mathsf{RL}, M)$: On input the public parameters PP, the state ST, an attribute set $\Gamma \subseteq \mathbb{U}$, a set of revoked user identities RL and a message $M \in \mathbb{M}$, the encryptor outputs a ciphertext $\mathsf{CT}_{\Gamma,\mathsf{RL}}$.

$\mathsf{RABE.Decrypt}(\mathsf{CT}_{\Gamma,\mathsf{RL}}, \mathsf{SK}_{\mathbb{S},ID}, \mathbb{S}, ID, \mathsf{PP}, \mathsf{ST})$: A user takes as input a ciphertext $\mathsf{CT}_{\Gamma,\mathsf{RL}}$, its private key $\mathsf{SK}_{\mathbb{S},ID}$, its access structure $\mathbb{S}$, user identity ID, the public parameters PP and the state ST. It obtains an encrypted message M or the distinguished symbol $\perp$.

• ***Correctness***: The correctness of RABE is defined as follows: For all $\mathsf{PP}, \mathsf{ST}, \mathsf{MK}$ generated by $\mathsf{RABE.Setup}(1^\lambda, N_{\max})$, $\mathsf{SK}_{\mathbb{S},ID}$ generated by $\mathsf{RABE.GenKey}(\mathsf{PP}, \mathsf{MK}, \mathsf{ST}, ID, \mathbb{S})$ for any $\mathbb{S}, ID$, $\mathsf{CT}_{\Gamma,\mathsf{RL}}$ generated by $\mathsf{RABE.Encrypt}(\mathsf{PP}, \mathsf{ST}, \Gamma, \mathsf{RL}, M)$ for any Γ, RL and M, it is required that

- If $\mathbb{S}$ accepts Γ and $ID \notin \mathsf{RL}$, then $\mathsf{RABE.Decrypt}(\mathsf{CT}_{\Gamma,\mathsf{RL}}, \mathsf{SK}_{\mathbb{S},ID}, \mathbb{S}, ID, \mathsf{PP}, \mathsf{ST}) = M$.
- If $\mathbb{S}$ does not accept Γ or $ID \in \mathsf{RL}$, then $\mathsf{RABE.Decrypt}(\mathsf{CT}_{\Gamma,\mathsf{RL}}, \mathsf{SK}_{\mathbb{S},ID}, \mathbb{S}, ID, \mathsf{PP}, \mathsf{ST} = \perp$ with all but negligible probability.

• ***Security Model***: The security of RABE under chosen plaintext attacks (CPA) is defined in terms of the following experiment between a challenger $\mathcal{B}$ and a probabilistic polynomial-time adversary $\mathcal{A}$:

Setup: $\mathcal{B}$ generates a master secret key MK, a state ST, and public parameters PP by running $\mathsf{RABE.Setup}(1^\lambda, N_{\max})$. It keeps MK to itself and gives PP, ST to $\mathcal{A}$.

Phase 1: $\mathcal{A}$ adaptively requests a polynomial number of private keys for access structure-user identity pairs $(\mathbb{S}_1, ID_1), \ldots, (\mathbb{S}_{\widehat{q}_1}, ID_{\widehat{q}_1})$, and $\mathcal{B}$ gives the corresponding private keys $\mathsf{SK}_{\mathbb{S}_1,ID_1}, \ldots, \mathsf{SK}_{\mathbb{S}_{\widehat{q}_1},ID_{\widehat{q}_1}}$ along with the updated state ST to $\mathcal{A}$ by executing $\mathsf{RABE.Genkey}(\mathsf{PP}, \mathsf{MK}, \mathsf{ST}, ID_\imath, \mathbb{S}_\imath)$ for $\imath = 1, \ldots, \widehat{q}_1$.

Challenge: $\mathcal{A}$ submits a challenge attribute set $\mathit{\Gamma}^*$, a revocation list RL^*, and two challenge messages M_0^*, M_1^* with equal length satisfying the following restriction: If a private key query for an access structure-user identity pair $(\mathbb{S}_\imath, ID_\imath)$ such that $\mathbb{S}_\imath$ accepts $\mathit{\Gamma}^*$ was requested, then $ID_\imath$ must belong to RL^*. $\mathcal{B}$ flips a random coin $b \in \{0,1\}$ and gives the challenge ciphertext CT^* to $\mathcal{A}$ by performing $\mathsf{RABE.Encrypt}(\mathsf{PP}, \mathsf{ST}, \mathit{\Gamma}^*, \mathsf{RL}^*, M_b^*)$.

Phase 2: $\mathcal{A}$ may continue to request a polynomial number of additional private key queries for access structure-user identity pairs $(\mathbb{S}_{\widehat{q}_1+1}, ID_{\widehat{q}_1+1}), \ldots, (\mathbb{S}_{\widehat{q}}, ID_{\widehat{q}})$ subject to the same restriction as before, and $\mathcal{B}$ gives the corresponding private keys to $\mathcal{A}$.

Guess: Finally, $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$ and wins the game if $b = b'$.

The advantage of $\mathcal{A}$ is defined as $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{RABE,IND\text{-}CPA}}(\lambda) = |\mathsf{Pr}[b = b'] - 1/2|$ where the probability is taken over all the randomness of the experiment. An RABE scheme is fully secure under chosen plaintext attacks if for all probabilistic polynomial-time adversary $\mathcal{A}$, the advantage of $\mathcal{A}$ in the above experiment is negligible in the security parameter λ.

2.6 The Subset difference Revocation Scheme

• **Some Notations Related to Full Binary Tree**: A full binary tree $\mathcal{T}$ is a tree data structure where each node except the leaf nodes has two child nodes. We define some notations concerning a full binary tree that will be used in the subsequent discussions:

- $N_{\max}$: The number of leaf nodes in $\mathcal{T}$. The number of all nodes in $\mathcal{T}$ is $2N_{\max} - 1$.
- ν_i: A node in $\mathcal{T}$ for any i, $1 \leq i \leq 2N_{\max} - 1$.
- D_i: The depth of a node ν_i, i.e., the length of the path from the root node to the node ν_i. The root node is at depth zero. The depth of $\mathcal{T}$ is the length of the path from the root node to a leaf node.
- T_i: A subtree of $\mathcal{T}$ that is rooted at ν_i for any node ν_i in $\mathcal{T}$.
- $T_{i,j}$: The subtree $T_i \backslash T_j$ for any two nodes ν_i, ν_j in $\mathcal{T}$ such that ν_j is a descendant of ν_i, i.e., all nodes that are descendants of ν_i but not of ν_j.
- S_i: The set of leaf nodes in T_i.
- $S_{i,j}$: The set of leaf nodes in $T_{i,j}$, i.e., $S_{i,j} = S_i \backslash S_j$.
- L_i: An identifier for a node ν_i in $\mathcal{T}$, that is a fixed and unique string. The identifier of each node in the tree is assigned as follows: Each edge in the tree is assigned with 0 or 1 depending on whether the edge connects a node to its left or right child node. The identifier L_i of a node ν_i is the bit string obtained by reading all the labels of edges in the path from the root to the node ν_i.
- $(L_i \| D_j)$: The integer representation of the string formed by concatenation of the binary representation of D_j, the depth of the node ν_j, with L_i, the identifier of ν_i for any subset $S_{i,j}$ of leaf nodes defined by the nodes ν_i and ν_j.
- $(L_i \| L_j)$: The integer representation of the string obtained by concatenating L_j with L_i for any subset $S_{i,j}$ of leaf nodes defined by the nodes ν_i and ν_j.
- $ST(\mathcal{T}, R)$ (or simply $ST(R)$): The Steiner Tree induced by a subset R of leaf nodes and the root node of the full binary tree $\mathcal{T}$, i.e., the minimal subtree of $\mathcal{T}$ that connects all the leaf nodes in R and the root node.

• **Subset Difference Method**: The *subset difference* (SD) revocation method is a special instance of a general methodology for revocation schemes proposed by Naor et al., known as the subset cover (SC) framework [15]. The well-known complete subtree (CS) scheme used in all the previous RABE schemes [4], [1], [19] is another instance of the subset cover framework. The original subset cover framework consists of a subset assignment part and a key assignment part. As in [12], in this paper, we define the subset cover framework by using the subset assignment part only. The formal definition of subset cover framework is given as follows:

Definition 9 (Subset Cover Framework). *A subset cover* (SC) *scheme for the set* $\mathcal{N} = \{1, \ldots, N_{\max}\}$ *of users consists of the following probabilistic polynomial-time algorithms:*

$\mathsf{SC.Setup}(N_{\max})$*: The trusted authority takes as input the maximum number* $N_{\max}$ *of users and publishes a collection* $\mathcal{S}$ *of subsets* $S_1, \ldots, S_w$ *where* $S_i \subseteq \mathcal{N}$*.*

$\mathsf{SC.Assign}(\mathcal{S}, u)$*: On input the collection* $\mathcal{S}$ *and a user serial number* $u \in \mathcal{N}$*, the trusted authority provides a private set* $\mathsf{PV}_u = \{S_{j_1}, \ldots, S_{j_v}\}$ *to the user with serial number* u*.*

$\mathsf{SC.Cover}(\mathcal{S}, R)$*: Taking as input the collection* $\mathcal{S}$ *and a revoked set* $R \subset \mathcal{N}$ *of users, a cover generator outputs a covering set* $\mathsf{CV}_R = \{S_{i_1}, \ldots, S_{i_z}\}$*, that is a partition of the non-revoked users* $\mathcal{N} \backslash R$ *into disjoined subsets* $S_{i_1}, \ldots, S_{i_z}$ *such that* $\mathcal{N} \backslash R = \cup_{l=1}^{z} S_{i_l}$*.*

$\mathsf{SC.Match}(\mathsf{CV}_R, \mathsf{PV}_u)$*: A user takes as input a covering set* $\mathsf{CV}_R = \{S_{i_1}, \ldots, S_{i_z}\}$ *together with its private set* $\mathsf{PV}_u = \{S_{j_1}, \ldots, S_{j_v}\}$ *and obtains* $(S_{i_l}, S_{j_{l'}})$ *such that* $S_{i_l} \in \mathsf{CV}_R$*,* $u \in S_{i_l}$*, and* $S_{j_{l'}} \in \mathsf{PV}_u$*, or obtains* $\perp$*.*

- ***Correctness***: *The correctness of subset cover framework is defined as follows: For all* $\mathcal{S}$ *generated by* SC.Setup*, all* PV_u *generated by* SC.Assign*, and any* R*, it is required that:*

  – *If* $u \notin R$*, then* $\mathsf{SC.Match}(\mathsf{CV}_R, \mathsf{PV}_u) = (S_{i_l}, S_{j_{l'}})$ *such that* $S_{i_l} \in \mathsf{CV}_R$*,* $u \in S_{i_l}$ *and* $S_{j_{l'}} \in \mathsf{PV}_u$*.*

  – *If* $u \in R$*, then* $\mathsf{SC.Match}(\mathsf{CV}_R, \mathsf{PV}_u) = \perp$*.*

Note that the exact conditions of the subsets output by the matching algorithm is defined by the specific instance of the SC *scheme.*

As mentioned earlier, the SD scheme is a particular instance of the SC scheme and it was proposed by Naor et al. [15] as an improvement on the CS scheme. The SD scheme is described as follows:

$\mathsf{SD.Setup}(N_{\max})$: The trusted authority takes as input the maximum number $N_{\max}$ of users. Let $N_{\max} = 2^{n_{\max}}$ for simplicity. It first sets a full binary tree $\mathcal{T}$ of depth $n_{\max}$. Each user is assigned to a different leaf node in $\mathcal{T}$. The collection $\mathcal{S}$ of SD scheme is the set of all subsets $S_{j,k} = S_j \backslash S_k$ where ν_j, ν_k are nodes in $\mathcal{T}$, ν_k is a descendant of ν_j; where S_j (resp. S_k) is the set of leaf nodes of the subtree rooted at ν_j (resp. ν_k). It publishes the full binary tree $\mathcal{T}$.

$\mathsf{SD.Assign}(\mathcal{T}, u)$: Taking as input the tree $\mathcal{T}$ and a user serial number $u \in \mathcal{N}$, the trusted authority computes the private set PV_u for the user u as follows: Let $\nu(u)$ be the leaf node of $\mathcal{T}$ that is assigned to the user u. Let $(\nu_{l_0}, \ldots, \nu_{l_{n_{\max}}})$ be the path from the root node ν_{l_0} to the leaf node $\nu_{l_{n_{\max}}} = \nu(u)$. It first sets a private set $\mathsf{PV}_u = \varnothing$. For all $j, k \in \{l_0, \ldots, l_{n_{\max}}\}$ such that ν_k is a descendant of ν_j, it adds into PV_u the subset $S_{j,k}$ defined by the two nodes ν_j and ν_k.

$\mathsf{SD.Cover}(\mathcal{T}, R)$: On input the tree $\mathcal{T}$ and a revoked set R of users, a cover generator proceeds as follows: It first sets a subtree T as $ST(R)$, and then it outputs a covering set CV_R built iteratively by removing nodes from T until T consists of just a single node as described below:

1. It finds two leaf nodes ν_j and ν_k in T such that the least-common-ancestor ν of ν_j and ν_k does not contain any other leaf nodes of T in its subtree. Note that such a pair (ν_j, ν_k) can always be found. Let ν_l and ν_m be the two child nodes of ν such that ν_j is a descendant of ν_l and ν_k is a descendant of ν_m. If there is only one leaf node left, it makes $\nu_j = \nu_k$ to be that leaf node, ν to be the root of T and $\nu_l = \nu_m = \nu$.
2. If $\nu_l \neq \nu_j$, then it adds the subset $S_{l,j}$ to CV_R; likewise, if $\nu_m \neq \nu_k$, then it adds the subset $S_{m,k}$ to CV_R.
3. It removes from T all the descendants of ν and makes ν a leaf node.

$\mathsf{SD.Match}(\mathsf{CV}_R, \mathsf{PV}_u)$: A user takes as input a covering set CV_R and its private set PV_u. If it finds two subsets $S_{j,k}$ and $S_{j',k'}$ such that $S_{j,k} \in \mathsf{CV}_R$, $S_{j',k'} \in \mathsf{PV}_u$, and $(j = j') \wedge (D_k = D_{k'}) \wedge (k \neq k')$, then it outputs $(S_{j,k}, S_{j',k'})$. Otherwise, it obtains $\perp$.

The correctness property of SD scheme is formally stated by the following lemma:

Lemma 3. *If $u \notin R$, then there exists a unique pair of subsets $(S_{j,k}, S_{j',k'})$ such that $S_{j,k} \in \mathsf{CV}_R$, $S_{j',k'} \in \mathsf{PV}_u$ and $(j = j') \wedge (D_k = D_{k'}) \wedge (k \neq k')$ holds. Otherwise, such a pair of subsets cannot be found.*

Observation: Note that for any fixed pair (ν_j, D) of node and depth value, there is at most one subset $S_{j,k} \in \mathsf{CV}_R$ and at most one subset $S_{j,k'} \in \mathsf{PV}_u$ such that $D_k = D_{k'} = D$. Moreover, the defining nodes ν_j, ν_k of the subsets $S_{j,k} \in \mathsf{CV}_R$ are all distinct. This observation is very important in our RABE construction.

Lemma 4 ([15]). *Let $N_{\max}$ be the number of leaf nodes in a full binary tree and $\widehat{r}$ be the size of a revoked set. In the* SD *scheme, the size of a private set is $O(\log^2 N_{\max})$ and the size of a covering set is at most $2\widehat{r} - 1$.*

Remark 2. The covering algorithm of the SD scheme is only defined for $\widehat{r} \geq 1$. One simple way to handle the case $\widehat{r} = 0$ is to use a dummy user that is always revoked. In the SD scheme, the size of the covering set is at most $2\widehat{r} - 1$, but it is rough worst-case analysis and the size is always smaller than that of the CS scheme since a subset in the CS scheme is also a subset in the SD scheme [15]. The better analysis of this covering set size is given by Martin et al. [14].

Note that the layered subset difference scheme (LSD) was proposed by Halevy and Shamir [10] to reduce the size of a private set in the SD scheme. The SD scheme in a cryptosystem generally can be replaced by the LSD scheme since the LSD scheme is a special case of the SD scheme.

Lemma 5 ([10]). *Let $N_{\max}$ be the number of leaf nodes in a full binary tree and $\widehat{r}$ be the size of a revoked set. In the* LSD *scheme, the size of a private set is $O(\log^{1.5} N_{\max})$ and the size of a covering set is at most $4\widehat{r} - 2$.*

3 Our RABE Scheme

3.1 Construction

Let $\mathcal{N} = \{1, \ldots, N_{\max}\}$ be the universe of user key serial numbers. Let d be the total number of attributes in the attribute universe $\mathbb{U} = \{(t, A_t) | t \in \{1, \ldots, d\} \wedge A_t \in \mathbb{F}_q\}$. Further, assume that $\widehat{r}_{\max}$ be the maximum of $\sharp\mathsf{CV}_{\mathsf{RI}}$, the cardinality of the covering set $\mathsf{CV}_{\mathsf{RI}}$, for all revoked set $\mathsf{RI} \subset \mathcal{N}$ used in the system. Our RABE scheme supports access structures $\mathbb{S} = (\boldsymbol{M}, \rho)$ defined in Section 2.4, the collection of which is denoted by $\mathfrak{S}$. In the proposed RABE scheme, we assume that ρ is injective for $\mathbb{S} = (\boldsymbol{M}, \rho) \in \mathfrak{S}$. The message space in our RABE scheme is $\mathbb{M} = \mathbb{G}_T$. Our RABE scheme is described as follows:

$\mathsf{RABE.Setup}(1^\lambda, N_{\max})$: The key generation center takes as input a security parameter 1^λ and the maximum number $N_{\max}$ of users and proceeds as follows:

1. It first runs $\mathcal{G}_{\mathsf{ob}}(1^\lambda, (n_0 = 5, n = 16))$, described in Figure 1, to obtain $(\mathsf{param} = (\mathsf{param}_{\mathbb{V}_0}, \mathsf{param}_{\mathbb{V}}, g_T), \{\mathbb{B}_0, \mathbb{B}_0^*\},$
$\{\mathbb{B}, \mathbb{B}^*\})$. It sets $\widehat{\mathbb{B}}_0 = \{\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}\}$, $\widehat{\mathbb{B}} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_4, \boldsymbol{b}_{13}, \boldsymbol{b}_{14}\}$, $\widehat{\mathbb{B}}_0^* = \{\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*, \boldsymbol{b}_{0,4}^*\}$, $\widehat{\mathbb{B}}^* = \{\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_4^*, \boldsymbol{b}_{11}^*, \boldsymbol{b}_{12}^*\}$.
2. It obtains $\mathcal{T}$ by running $\mathsf{SD.Setup}(N_{\max})$. Let $\mathcal{S}$ be the collection of all subsets $S_{j,k}$ of $\mathcal{T}$. It initializes the user list $\mathsf{UL} = \varnothing$.
3. It publishes the public parameters $\mathsf{PP} = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}})$, and a state $\mathsf{ST} = (\mathcal{T}, \mathsf{UL})$, while it keeps the master secret key $\mathsf{MK} = (\widehat{\mathbb{B}}_0^*, \widehat{\mathbb{B}}^*)$.

$\mathsf{RABE.GenKey}(\mathsf{PP}, \mathsf{MK}, \mathsf{ST}, \mathbb{S}, ID)$: Taking as input the public parameters PP, the master secret key MK, the state $\mathsf{ST} = (\mathcal{T}, \mathsf{UL})$, an access structure $\mathbb{S} = (\boldsymbol{M}, \rho) \in \mathfrak{S}$ such that $\boldsymbol{M}$ is an $\ell \times r$ matrix and ρ is a labeling of the rows of $\boldsymbol{M}$, and an user identity ID, the key generation center provides a private key to the corresponding user as follows:

1. It first chooses $\vec{f} \xleftarrow{\$} \mathbb{F}_q^r$, computes $\vec{s}^{\mathsf{T}} = (s_1, \ldots, s_\ell)^{\mathsf{T}} = \boldsymbol{M} \cdot \vec{f}^{\mathsf{T}}$, $s_0' = \vec{1} \cdot \vec{f}^{\mathsf{T}}$, selects $\eta_0, s_0'' \xleftarrow{\$} \mathbb{F}_q$, and sets $s_0 = s_0' + s_0''$. Note that, $s_1, \ldots, s_\ell$ are shares of s_0'. Next, it computes

   $$\boldsymbol{k}_0^* = (-s_0, 1, \eta_0)_{\widehat{\mathbb{B}}_0^*} = -s_0 \boldsymbol{b}_{0,1}^* + \boldsymbol{b}_{0,3}^* + \eta_0 \boldsymbol{b}_{0,4}^* = (-s_0, 0, 1, \eta_0, 0)_{\mathbb{B}_0^*}.$$

   For $i = 1, \ldots, \ell$, if $\rho(i) = (t, A_t)$, it chooses $\mu_i, \theta_i, \eta_{i,1}, \eta_{i,2} \xleftarrow{\$} \mathbb{F}_q$ and computes

   $$\begin{aligned}\boldsymbol{k}_i^* =& (\mu_i(t, -1), s_i + \theta_i A_t, -\theta_i, \eta_{i,1}, \eta_{i,2})_{\widehat{\mathbb{B}}^*} \\ =& \mu_i t \boldsymbol{b}_1^* - \mu_i \boldsymbol{b}_2^* + (s_i + \theta_i A_t) \boldsymbol{b}_3^* - \theta_i \boldsymbol{b}_4^* + \eta_{i,1} \boldsymbol{b}_{11}^* + \eta_{i,2} \boldsymbol{b}_{12}^* \\ =& (\mu_i(t, -1), s_i + \theta_i A_t, -\theta_i, 0^6, \eta_{i,1}, \eta_{i,2}, 0^2, 0^2)_{\mathbb{B}^*}.\end{aligned}$$

   We mention that, $\boldsymbol{k}_0^*$ and $\boldsymbol{k}_i^*$ are actually some linear combinations of vectors in $\widehat{\mathbb{B}}_0^*$ and $\widehat{\mathbb{B}}^*$ respectively, where $\widehat{\mathbb{B}}_0^*$ and $\widehat{\mathbb{B}}^*$ are extractable from MK. However, for ease of discussion, we have represented $\boldsymbol{k}_0^*$ and $\boldsymbol{k}_i^*$ as linear combinations of vectors in $\mathbb{B}_0^*$ and $\mathbb{B}^*$ respectively taking the coefficients of vectors in $\mathbb{B}_0^* \backslash \widehat{\mathbb{B}}_0^*$ and $\mathbb{B}^* \backslash \widehat{\mathbb{B}}^*$ as zeros. Hereafter, a similar notation will be followed for representing linear combinations in $\mathbb{V}_0$ and $\mathbb{V}$.
2. It assigns the user identity ID to a leaf node $\nu(u)$ in $\mathcal{T}$ that is not yet assigned, where $u \in \mathcal{N}$ is a serial number that is assigned to ID. It saves (ID, u) to UL. Next it obtains PV_u by running $\mathsf{SD.Assign}(\mathcal{T}, u)$.
3. For each $S_{j,k} \in \mathsf{PV}_u$, it performs the following operations: It first selects $s_{j,k,1}, s_{j,k,2} \xleftarrow{\$} \mathbb{F}_q$ such that $s_0'' = s_{j,k,1} + s_{j,k,2}$, i.e., it breaks s_0'' into two random parts. It further chooses $\mu_{j,k,1}, \mu_{j,k,2}, \theta_{j,k}, \eta_{j,k,1,1}, \eta_{j,k,1,2}, \eta_{j,k,2,1}, \eta_{j,k,2,2}, \xleftarrow{\$} \mathbb{F}_q$ and computes

   $$\begin{aligned}\boldsymbol{k}_{j,k,1}^* &= (\mu_{j,k,1}(d+1, -1),\ s_{j,k,1} + \theta_{j,k}(L_j \| D_k), -\theta_{j,k},\ 0^6, \eta_{j,k,1,1}, \eta_{j,k,1,2}, 0^2, 0^2)_{\mathbb{B}^*} \\ \boldsymbol{k}_{j,k,2}^* &= (\mu_{j,k,2}(d+2, -1), \quad s_{j,k,2}((L_j \| L_k), -1), \quad 0^6, \eta_{j,k,2,1}, \eta_{j,k,2,2}, 0^2, 0^2)_{\mathbb{B}^*}.\end{aligned}$$

4. Finally, it publishes the updated state $\mathsf{ST} = (\mathcal{T}, \mathsf{UL})$ and provides a private key

   $$\mathsf{SK}_{\mathbb{S},ID} = (\mathsf{PV}_u, \boldsymbol{k}_0^*, \{\boldsymbol{k}_i^*\}_{i=1,\ldots,\ell}, \{\boldsymbol{k}_{j,k,1}^*, \boldsymbol{k}_{j,k,2}^*\}_{S_{j,k} \in \mathsf{PV}_u})$$

   to the user.

$\mathsf{RABE.Encrypt}(\mathsf{PP}, \mathsf{ST}, \mathit{\Gamma}, \mathsf{RL}, M)$: On input the public parameters PP, the state $\mathsf{ST} = (\mathcal{T}, \mathsf{UL})$, an attribute set $\mathit{\Gamma} \subseteq \mathbb{U}$, a revocation list RL of user identities and a message $M \in \mathbb{G}_T$, the encryptor executes the following steps:

1. It first extracts g_T from PP, chooses $\omega, \zeta, \varphi_0 \xleftarrow{\$} \mathbb{F}_q$ and computes

   $$\boldsymbol{c}_0 = (\omega, 0, \zeta, 0, \varphi_0)_{\mathbb{B}_0},$$
   $$\text{and } c = g_T^{\zeta} M.$$

2. For all $(t, A_t) \in \mathit{\Gamma}$, it selects $\sigma_t, \varphi_{t,1}, \varphi_{t,2} \xleftarrow{\$} \mathbb{F}_q$, and computes

   $$\boldsymbol{c}_t = (\sigma_t(1, t), \omega(1, A_t), 0^6, 0^2, \varphi_{t,1}, \varphi_{t,2}, 0^2)_{\mathbb{B}}.$$

3. Then it defines the revoked user serial number set $\mathsf{RI} \subseteq \mathcal{N}$ from RL by using UL. Next it obtains the covering set $\mathsf{CV}_{\mathsf{RI}}$ by executing $\mathsf{SD.Cover}(\mathcal{T}, \mathsf{RI})$.

4. For each $S_{j,k} \in \mathsf{CV}_{\mathsf{RI}}$, it performs the following steps: It chooses $\sigma_{j,k,1}, \sigma_{j,k,2}, \varphi_{j,k,1,1}, \varphi_{j,k,1,2}, \varphi_{j,k,2,1}, \varphi_{j,k,2,2} \xleftarrow{\$} \mathbb{F}_q$ and computes
$$\begin{aligned}\boldsymbol{c}_{j,k,1} &= (\sigma_{j,k,1}(1, d+1), \omega(1, (L_j \| D_k)), 0^6, 0^2, \varphi_{j,k,1,1}, \varphi_{j,k,1,2}, 0^2)_{\mathbb{B}},\\ \boldsymbol{c}_{j,k,2} &= (\sigma_{j,k,2}(1, d+2), \omega(1, (L_j \| L_k)), 0^6, 0^2, \varphi_{j,k,2,1}, \varphi_{j,k,2,2}, 0^2)_{\mathbb{B}}.\end{aligned}$$
5. The encryptor outputs the ciphertext
$$\mathsf{CT}_{\Gamma,\mathsf{RL}} = (\mathsf{CV}_{\mathsf{RI}}, c, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,A_t)\in\Gamma}, \{\boldsymbol{c}_{j,k,1}, \boldsymbol{c}_{j,k,2}\}_{S_{j,k}\in\mathsf{CV}_{\mathsf{RI}}}).$$

$\mathsf{RABE.Decrypt}(\mathsf{CT}_{\Gamma,\mathsf{RL}}, \mathsf{SK}_{\mathbb{S},ID}, \mathbb{S}, ID, \mathsf{PP}, \mathsf{ST})$: A user takes as input a ciphertext $\mathsf{CT}_{\Gamma,\mathsf{RL}} = (\mathsf{CV}_{\mathsf{RI}}, c, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,A_t)\in\Gamma}, \{\boldsymbol{c}_{j,k,1}, \boldsymbol{c}_{j,k,2}\}_{S_{j,k}\in\mathsf{CV}_{\mathsf{RI}}})$, its private key $\mathsf{SK}_{\mathbb{S},ID} = (\mathsf{PV}_u, \boldsymbol{k}_0^*, \{\boldsymbol{k}_i^*\}_{i=1,\ldots,\ell}, \{\boldsymbol{k}_{j,k,1}^*, \boldsymbol{k}_{j,k,2}^*\}_{S_{j,k}\in\mathsf{PV}_u})$, its access structure $\mathbb{S} = (\boldsymbol{M}, \rho)$, user identity ID, the public parameters PP and the state ST. It proceeds as follows:

1. If $\mathbb{S}$ accepts Γ, then it computes I and $\{\alpha_i\}_{i\in I}$ such that $\vec{1} = \sum_{i\in I} \alpha_i \boldsymbol{M}_i$ and hence $s_0' = \sum_{i\in I} \alpha_i s_i$, where $\boldsymbol{M}_i$ is the i-th row of $\boldsymbol{M}$ and $I \subseteq \{i \in \{1, \ldots, \ell\} | \rho(i) = (t, A_t) \in \Gamma\}$. Otherwise, it obtains $\perp$.
2. If $ID \notin \mathsf{RL}$ for $(ID, u) \in \mathsf{UL}$, then it obtains $(S_{j,k}, S_{j',k'})$ by running $\mathsf{SD.Match}(\mathsf{CV}_{\mathsf{RI}}, \mathsf{PV}_u)$ such that $S_{j,k} \in \mathsf{CV}_{\mathsf{RI}}$, $S_{j',k'} \in \mathsf{PV}_u$ and $(j = j') \wedge (D_k = D_{k'}) \wedge (k \neq k')$. Otherwise, it outputs $\perp$.
3. It computes $\pi' = \prod_{\substack{i\in I\\ \rho(i)=(t,A_t)}} E(\boldsymbol{c}_t, \boldsymbol{k}_i^*)^{\alpha_i}$, $\pi'' = E(\boldsymbol{c}_{j,k,1}, \boldsymbol{k}_{j',k',1}^*) E(\boldsymbol{c}_{j,k,2}, \boldsymbol{k}_{j',k',2}^*)^{\frac{1}{(L_{j'} \| L_{k'}) - (L_j \| L_k)}}$ and $\pi = E(\boldsymbol{c}_0, \boldsymbol{k}_0^*)\pi'\pi''$.
4. It retrieves the message as $M = c/\pi$.

3.2 Correctness

Let $\mathsf{SK}_{\mathbb{S},ID} = (\mathsf{PV}_u, \boldsymbol{k}_0^*, \{\boldsymbol{k}_i^*\}_{i=1,\ldots,\ell}, \{\boldsymbol{k}_{j,k,1}^*, \boldsymbol{k}_{j,k,2}^*\}_{S_{j,k}\in\mathsf{PV}_u})$ be a private key for an user with identity ID together with an access structure $\mathbb{S} = (\boldsymbol{M}, \rho)$, and $\mathsf{CT}_{\Gamma,\mathsf{RL}} = (\mathsf{CV}_{\mathsf{RI}}, c, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,A_t)\in\Gamma}, \{\boldsymbol{c}_{j,k,1}, \boldsymbol{c}_{j,k,2}\}_{S_{j,k}\in\mathsf{CV}_{\mathsf{RI}}})$ be a ciphertext for an attribute set Γ together with a revocation list RL of user identities. If $ID \notin \mathsf{RL}$, then a pair of subsets $(S_{j,k}, S_{j',k'})$ such that $S_{j,k} \in \mathsf{CV}_{\mathsf{RI}}$, $S_{j',k'} \in \mathsf{PV}_u$ and $(j = j') \wedge (D_k = D_{k'}) \wedge (k \neq k')$, i.e., $(L_j \| D_k) = (L_{j'} \| D_{k'})$ and $(L_j \| L_k) \neq (L_{j'} \| L_{k'})$, can be found from the correctness of the SD scheme (Lemma 3). Now,

$$\begin{aligned}\pi' &= \prod_{\substack{i\in I\\ \rho(i)=(t,A_t)}} E(\boldsymbol{c}_t, \boldsymbol{k}_i^*)^{\alpha_i}\\
&= \prod_{\substack{i\in I\\ \rho(i)=(t,A_t)}} E\left((\sigma_t(1,t), \omega(1, A_t), 0^8, \varphi_{t,1}, \varphi_{t,2}, 0^2)_{\mathbb{B}}, (\mu_i(t,-1), (s_i + \theta_i A_t, -\theta_i, 0^6, \eta_{i,1}, \eta_{i,2}, 0^4)_{\mathbb{B}^*}\right)^{\alpha_i}\\
&= \prod_{\substack{i\in I\\ \rho(i)=(t,A_t)}} \left[E(\boldsymbol{b}_1, \boldsymbol{b}_1^*)^{\sigma_t\mu_i t} E(\boldsymbol{b}_2, \boldsymbol{b}_2^*)^{-\sigma_t\mu_i t} E(\boldsymbol{b}_3, \boldsymbol{b}_3^*)^{(s_i+\theta_i A_t)\omega} E(\boldsymbol{b}_4, \boldsymbol{b}_4^*)^{-\theta_i\omega A_t}\right]^{\alpha_i}\\
&= \prod_{\substack{i\in I\\ \rho(i)=(t,A_t)}} \left[g_T^{\sigma_t\mu_i t} g_T^{-\sigma_t\mu_i t} g_T^{(s_i+\theta_i A_t)\omega} g_T^{-\theta_i\omega A_t}\right]^{\alpha_i} = \prod_{\substack{i\in I\\ \rho(i)=(t,A_t)}} g_T^{\omega\alpha_i s_i} = g_T^{\sum_{\substack{i\in I\\ \rho(i)=(t,A_t)}} \omega\alpha_i s_i} = g_T^{\omega s_0'},\end{aligned}$$

as $\sum\limits_{\substack{i\in I\\ \rho(i)=(t,A_t)}} \alpha_i s_i = s'_0$, since $\mathbb{S}=(\boldsymbol{M},\rho)$ accepts Γ. Similarly,

$\pi'' = g_T^{\omega s_{j',k',1}} g_T^{\omega s_{j',k',2}\frac{(L_{j'}\|L_{k'})-(L_j\|L_k)}{(L_{j'}\|L_{k'})-(L_j\|L_k)}} = g_T^{\omega s''_0}$, as $s_{j',k',1}+s_{j',k',2}=s''_0$, since $S_{j',k'}\in\mathsf{PV}_u$. Thus,

$$\pi = g_T^{\left[-\omega s_0+\zeta+\omega s'_0+\omega s''_0\right]} = g_T^{\left[\omega(-s_0+s'_0+s''_0)+\zeta\right]} = g_T^{\zeta}.$$

So, $c/\pi = g_T^{\zeta} M/g^{\zeta} = M$.

4 Security Analysis

Theorem 1. *The* RABE *scheme, introduced in Section 3, is fully secure against chosen plaintext attacks* (CPA) *under the* DLIN *assumption. More precisely, for any probabilistic polynomial-time adversary $\mathcal{A}$, there exists probabilistic machines $\mathcal{F}_{1\text{-}1},\ldots,\mathcal{F}_{1\text{-}3}$, $\mathcal{F}_{2\text{-}1\text{-}1},\ldots,\mathcal{F}_{2\text{-}1\text{-}12}$, $\mathcal{F}_{2\text{-}2\text{-}1},\ldots,$ $\mathcal{F}_{2\text{-}2\text{-}12}$ whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ,*

$$\begin{aligned}
&\mathsf{Adv}_{\mathcal{A}}^{\mathsf{RABE,IND\text{-}CPA}}(\lambda) \leq \mathsf{Adv}_{\mathcal{F}_{1\text{-}1}}^{\mathsf{DLIN}}(\lambda) + \sum_{p=1}^{d}\sum_{j=1}^{2}\mathsf{Adv}_{\mathcal{F}_{1\text{-}2\text{-}p\text{-}j}}^{\mathsf{DLIN}}(\lambda) + \sum_{\upsilon=1}^{2}\sum_{\varpi=1}^{\widehat{r}_{\max}}\sum_{j=1}^{2}\mathsf{Adv}_{\mathcal{F}_{1\text{-}3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j}}^{\mathsf{DLIN}}(\lambda)+\\
&\sum_{h=1}^{\widehat{q}}\sum_{i=1}^{2}\Bigg[\mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}1}}^{\mathsf{DLIN}}(\lambda) + \sum_{j=1}^{2}\Bigg[\sum_{p=1}^{d}\Bigg\{\mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}2\text{-}j}}^{\mathsf{DLIN}}(\lambda) + \mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}3\text{-}j}}^{\mathsf{DLIN}}(\lambda) + \sum_{\substack{l=1\\ l\neq p}}^{d+2}\Bigg(\mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}4\text{-}j\text{-}l}}^{\mathsf{DLIN}}(\lambda)\ +\\
&\qquad\qquad \mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}5\text{-}j\text{-}l}}^{\mathsf{DLIN}}(\lambda)\Bigg) + \mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}6\text{-}j}}^{\mathsf{DLIN}}(\lambda)\Bigg\} + \sum_{\upsilon=1}^{2}\sum_{\varpi=1}^{\aleph}\Bigg\{\mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}(d+\upsilon)\text{-}\varpi\text{-}7\text{-}j}}^{\mathsf{DLIN}}(\lambda)+\\
&\qquad \mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}j}}^{\mathsf{DLIN}}(\lambda) + \sum_{\substack{l=1\\ l\neq d+\upsilon}}^{d+2}\Bigg(\mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}(d+\upsilon)\text{-}\varpi\text{-}9\text{-}j\text{-}l}}^{\mathsf{DLIN}}(\lambda) + \mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}(d+\upsilon)\text{-}\varpi\text{-}10\text{-}j\text{-}l}}^{\mathsf{DLIN}}(\lambda)\Bigg)+\\
&\qquad\qquad \sum_{\substack{\iota=1\\ \iota\neq\varpi}}^{\aleph}\mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}(d+\upsilon)\text{-}\varpi\text{-}11\text{-}j\text{-}\iota}}^{\mathsf{DLIN}}(\lambda) + \mathsf{Adv}_{\mathcal{F}_{2\text{-}h\text{-}i\text{-}(d+\upsilon)\text{-}\varpi\text{-}12\text{-}j}}^{\mathsf{DLIN}}(\lambda)\Bigg\}\Bigg]\Bigg] + \epsilon,
\end{aligned}$$

where

$$\mathcal{F}_{1\text{-}2\text{-}p\text{-}j}(\cdot)=\mathcal{F}_{1\text{-}2}(p,j,\cdot),\ \ \mathcal{F}_{1\text{-}3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j}(\cdot)=\mathcal{F}_{1\text{-}3}(d+\upsilon,\varpi,j,\cdot),$$

and, for $i=1,2,$

$$\begin{aligned}
&\mathcal{F}_{2\text{-}h\text{-}i\text{-}1}(\cdot)=\mathcal{F}_{2\text{-}i\text{-}1}(h,\cdot),\ \ \mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}2\text{-}j}(\cdot)=\mathcal{F}_{2\text{-}i\text{-}2}(h,p,j,\cdot),\\
&\mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}3\text{-}j}(\cdot)=\mathcal{F}_{2\text{-}i\text{-}3}(h,p,j,\cdot),\ \ \mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}4\text{-}j\text{-}l}(\cdot)=\mathcal{F}_{2\text{-}i\text{-}4}(h,p,j,l,\cdot),\\
&\mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}5\text{-}j\text{-}l}(\cdot)=\mathcal{F}_{2\text{-}i\text{-}5}(h,p,j,l,\cdot),\ \ \mathcal{F}_{2\text{-}h\text{-}i\text{-}p\text{-}6\text{-}j}(\cdot)=\mathcal{F}_{2\text{-}i\text{-}6}(h,p,j,\cdot),\\
&\mathcal{F}_{2\text{-}h\text{-}i\text{-}(d+\upsilon)\text{-}\varpi\text{-}7\text{-}j}(\cdot)=\mathcal{F}_{2\text{-}i\text{-}7}(h,d+\upsilon,\varpi,j,\cdot)\ \mathcal{F}_{2\text{-}h\text{-}i\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}j}(\cdot)=\mathcal{F}_{2\text{-}i\text{-}8}(h,d+\upsilon,\varpi,j,\cdot),
\end{aligned}$$

etc., $\widehat{q}$ is the maximum number of $\mathcal{A}$'s private key queries, d is the size of the attribute universe used in the system, $N_{\max}$ is the upper bound of user key serial numbers, $\widehat{r}_{\max}$ is the maximum size of a covering set of non-revoked users used in the system, $\aleph = \log^2 N_{\max} + \widehat{r}_{\max}$, and $\epsilon = \left[6+10d+20\widehat{r}_{\max}+14\widehat{q}+80d\widehat{q}+20d^2\widehat{q}+4\widehat{q}\aleph(30+10d+10\aleph)\right]/q$.

Proof. At the top level of strategy of the security proof, we follow the dual system encryption methodology over dual pairing vector space (DPVS) described in [16], [17]. To prove the security of our RABE scheme, we use Problem 1 and 2.

To prove Theorem 1, we consider the following games. In Game 0, a part framed by a box indicates positions of coefficients to be changed in a subsequent game. In the other games, a part framed by a box indicates coefficients which were changed in a transition from the previous game. Games proceed as follows:

$$\mathsf{Game}\ 0 \implies \mathsf{Game}\ 1 \implies \{\mathsf{Game}\ 2\text{-}h\text{-}1 \implies \mathsf{Game}\ 2\text{-}h\text{-}2 \implies \mathsf{Game}\ 2\text{-}h\text{-}3\}_{h=1,\dots,\widehat{q}} \implies \mathsf{Game}\ 3$$

Game 0: Game 0 is the original security game, i.e., the reply to a key query for an access structure-user identity pair $(\mathbb{S}_\imath = (\boldsymbol{M}^{(\imath)}, \rho_\imath), ID_\imath)$ is given by $\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath} = (\mathsf{PV}_{u_\imath}, \boldsymbol{k}_0^*, \{\boldsymbol{k}_i^*\}_{i=1,\dots,\ell}, \{\boldsymbol{k}_{j,k,1}^*, \boldsymbol{k}_{j,k,2}^*\}_{S_{j,k} \in \mathsf{PV}_{u_\imath}})$, where

$$\boldsymbol{k}_0^* = (-s_0, \boxed{0}, 1, \eta_0, 0)_{\mathbb{B}_0^*}, \tag{1}$$

for $i = 1, \dots, \ell$ such that $\rho_\imath(i) = (t, A_t)$,

$$\boldsymbol{k}_i^* = (\mu_i(t, -1), s_i + \theta_i A_t, -\theta_i, 0^4, \boxed{0^2}, \eta_{i,1}, \eta_{i,2}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{2}$$

in which $\nu_i, \theta_i, \eta_0, \eta_{i,1}, \eta_{i,2} \xleftarrow{\$} \mathbb{F}_q$, $s_0 = s_0' + s_0''$, $s_0' = \vec{1} \cdot \vec{f}^{\mathsf{T}}$, $\vec{s}^{\mathsf{T}} = (s_1, \dots, s_\ell)^{\mathsf{T}} = \boldsymbol{M}^{(\imath)} \cdot \vec{f}^{\mathsf{T}}$, $s_0'' \xleftarrow{\$} \mathbb{F}_q$, $\vec{f} \xleftarrow{\$} \mathbb{F}_q^r$, $\boldsymbol{M}^{(\imath)}$ being an $\ell \times r$ matrix, and for all $S_{j,k} \in \mathsf{PV}_{u_\imath}$, such that $\nu(u_\imath)$ is the leaf node of $\mathcal{T}$ assigned to $ID_\imath$,

$$\boldsymbol{k}_{j,k,1}^* = (\mu_{j,k,1}(d+1, -1), s_{j,k,1} + \theta_{j,k}(L_j \| D_k), -\theta_{j,k}, 0^4, \boxed{0^2}, \eta_{j,k,1,1}, \eta_{j,k,1,2}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{3}$$

$$\boldsymbol{k}_{j,k,2}^* = (\mu_{j,k,2}(d+2, -1), \quad s_{j,k,2}((L_j \| L_k), -1), \quad 0^4, \boxed{0^2}, \eta_{j,k,2,1}, \eta_{j,k,2,2}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{4}$$

such that $s_{j,k,1}, s_{j,k,2}, \mu_{j,k,1}, \mu_{j,k,2}, \theta_{j,k}, \eta_{j,k,1,1}, \eta_{j,k,1,2}, \eta_{j,k,2,1}, \eta_{j,k,2,2} \xleftarrow{\$} \mathbb{F}_q$ so that $s_0'' = s_{j,k,1} + s_{j,k,2}$.
The challenge ciphertext for challenge plaintext(M_0^*, M_1^*), attribute set $\Gamma^* = \{(t, A_t) | 1 \le t \le d\}$ and revocation list RL^* is given by $\mathsf{CT}^* = (\mathsf{CV}_{\mathsf{RI}^*}, c, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t, A_t \in \Gamma^*}, \{\boldsymbol{c}_{j,k,1}, \boldsymbol{c}_{j,k,2}\}_{S_{j,k} \in \mathsf{CV}_{\mathsf{RI}^*}})$ where

$$c = g_T^{\zeta} M_b^*, b \xleftarrow{\$} \{0, 1\}, \tag{5}$$

$$\boldsymbol{c}_0 = (\omega, \boxed{0}, \boxed{\zeta}, 0, \varphi_0)_{\mathbb{B}_0}, \tag{6}$$

for $(t, A_t) \in \Gamma^*$,

$$\boldsymbol{c}_t = (\sigma_t(1, t), \omega(1, A_t), \boxed{0^2}, 0^2, \boxed{0^2}, 0^2, \varphi_{t,1}, \varphi_{t,2}, 0^2)_{\mathbb{B}}, \tag{7}$$

such that $\omega, \zeta, \varphi_0, \sigma_t, \varphi_{t,1}, \varphi_{t,2} \xleftarrow{\$} \mathbb{F}_q$, and for all $S_{j,k} \in \mathsf{CV}_{\mathsf{RI}^*}$, RI^* being the set of revoked user serial numbers obtained from RL^*,

$$\boldsymbol{c}_{j,k,1} = (\sigma_{j,k,1}(1, d+1), \omega(1, (L_j \| D_k)), \boxed{0^2}, 0^2, \boxed{0^2}, 0^2, \varphi_{j,k,1,1}, \varphi_{j,k,1,2}, 0^2)_{\mathbb{B}} \tag{8}$$

$$\boldsymbol{c}_{j,k,2} = (\sigma_{j,k,2}(1, d+2), \omega(1, (L_j \| L_k)), \boxed{0^2}, 0^2, \boxed{0^2}, 0^2, \varphi_{j,k,2,1}, \varphi_{j,k,2,2}, 0^2)_{\mathbb{B}}, \tag{9}$$

such that $\sigma_{j,k,1}, \sigma_{j,k,2}, \varphi_{j,k,1,1}, \varphi_{j,k,1,2}, \varphi_{j,k,2,1}, \varphi_{j,k,2,2} \xleftarrow{\$} \mathbb{F}_q$. Note that there is at most one $S_{j,k}$ in $\mathsf{CV}_{\mathsf{RI}^*}$ for any j and any k, as mentioned in Section 2.6.

Game 1: This game is identical to Game 0 except that the components of the challenge ciphertext CT^* is computed as follows:

$$c = g_T^{\zeta} M_b^*, \ b \xleftarrow{\$} \{0, 1\}, \tag{10}$$

$$\boldsymbol{c}_0 = (\omega, \boxed{\tau}, \zeta, 0, \varphi_0)_{\mathbb{B}_0}, \tag{11}$$

for $(t, A_t) \in \Gamma^*$,

$$\boldsymbol{c}_t = (\sigma_t(1, t), \omega(1, A_t), \boxed{\tau(1, A_t)}, 0^2, \boxed{\tau(1, A_t) \cdot \boldsymbol{Z}_t}, 0^2, \varphi_{t,1}, \varphi_{t,2}, 0^2)_{\mathbb{B}}, \tag{12}$$

where $\tau \xleftarrow{\$} \mathbb{F}_q^{\times}$, $\boldsymbol{Z}_t \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$. For all $S_{j,k} \in \mathsf{CV}_{\mathsf{RI}^*}$,

$$\begin{aligned} \boldsymbol{c}_{j,k,1} &= (\sigma_{j,k,1}(1, d+1), \omega(1, (L_j \| D_k)), \boxed{\tau(1, (L_j \| D_k))}, 0^2, \\ &\qquad \boxed{\tau(1, (L_j \| D_k)) \cdot \boldsymbol{Z}_{d+1,j,k}}, 0^2, \varphi_{j,k,1,1}, \varphi_{j,k,1,2}, 0^2)_{\mathbb{B}}, \qquad (13) \\ \boldsymbol{c}_{j,k,2} &= (\sigma_{j,k,2}(1, d+2), \omega(1, (L_j \| L_k)), \boxed{\tau(1, (L_j \| L_k))}, 0^2, \\ &\qquad \boxed{\tau(1, (L_j \| L_k)) \cdot \boldsymbol{Z}_{d+2,j,k}}, 0^2, \varphi_{j,k,2,1}, \varphi_{j,k,2,2}, 0^2)_{\mathbb{B}}, \qquad (14) \end{aligned}$$

where $\boldsymbol{Z}_{d+1,j,k}, \boldsymbol{Z}_{d+2,j,k} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$. All other variables are generated as in $\mathsf{Game}\ 0$.

Game 2-*h*-1 ($h = 1, \ldots, \widehat{q}$): We denote $\mathsf{Game}\ 1$ as $\mathsf{Game}\ 2\text{-}0\text{-}3$. $\mathsf{Game}\ 2\text{-}h\text{-}1$ is the same as $\mathsf{Game}\ 2\text{-}(h-1)\text{-}3$ other than the components of the h-th queried key for access structure-user identity pair ($\mathbb{S}_h = (\boldsymbol{M}^{(h)}, \rho_h), ID_h$) are constructed as follows:

$$\begin{aligned} &\boldsymbol{k}_0^* = (-s_0, \boxed{-a_0}, 1, \eta_0, 0)_{\mathbb{B}_0^*} \qquad (15) \\ &\text{for } i = 1, \ldots, \ell \text{ such that } \rho_h(i) = (t, A_t), \\ &\boldsymbol{k}_i^* = (\mu_i(t, -1), s_i + \theta_i A_t, -\theta_i, 0^4, \boxed{(a_i + \pi_i A_t, -\pi_i) \cdot \boldsymbol{U}_t}, \eta_{i,1}, \eta_{i,2}, 0^2, 0^2)_{\mathbb{B}^*}, \qquad (16) \end{aligned}$$

where $\vec{g} \xleftarrow{\$} \mathbb{F}_q^r$, $a_0 = a_0' + a_0''$, $a_0' = \vec{1} \cdot \vec{g}^{\intercal}$, $(a_1, \ldots, a_\ell)^{\intercal} = \boldsymbol{M}^{(h)} \cdot \vec{g}^{\intercal}$, $a_0'' \xleftarrow{\$} \mathbb{F}_q$, $\boldsymbol{U}_t = (\boldsymbol{Z}_t^{-1})^{\intercal}$ for $\boldsymbol{Z}_t \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$, $\pi_i \xleftarrow{\$} \mathbb{F}_q$ for $i = 1, \ldots, \ell$, $\boldsymbol{M}^{(h)}$ being an $\ell \times r$ matrix. For all $S_{j,k} \in \mathsf{PV}_{u_h}$,

$$\begin{aligned} \boldsymbol{k}_{j,k,1}^* &= (\mu_{j,k,1}(d+1, -1), s_{j,k,1} + \theta_{j,k}(L_j \| D_k), -\theta_{j,k}, 0^4, \\ &\qquad \boxed{(a_{j,k,1} + \pi_{j,k}(L_j \| D_k), -\pi_{j,k}) \cdot \boldsymbol{U}_{d+1,j,k}}, \eta_{j,k,1,1}, \eta_{j,k,1,2}, 0^2, 0^2)_{\mathbb{B}^*} \qquad (17) \\ \boldsymbol{k}_{j,k,2}^* &= (\mu_{j,k,2}(d+2, -1), \quad s_{j,k,2}((L_j \| L_k), -1), \quad 0^4, \\ &\qquad \boxed{(a_{j,k,2}((L_j \| L_k), -1) \cdot \boldsymbol{U}_{d+2,j,k}}, \eta_{j,k,2,1}, \eta_{j,k,2,2}, 0^2, 0^2)_{\mathbb{B}^*}, \qquad (18) \end{aligned}$$

where $a_{j,k,1}, a_{j,k,2}, \pi_{j,k} \xleftarrow{\$} \mathbb{F}_q$ such that $a_{j,k,1} + a_{j,k,2} = a_0''$, $\boldsymbol{U}_{d+1,j,k} = (\boldsymbol{Z}_{d+1,j,k}^{-1})^{\intercal}, \boldsymbol{U}_{d+2,j,k} = (\boldsymbol{Z}_{d+2,j,k}^{-1})^{\intercal}$ for $\boldsymbol{Z}_{d+1,j,k}, \boldsymbol{Z}_{d+2,j,k} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$. All the other variables are constructed as in $\mathsf{Game}\ 2\text{-}(h-1)\text{-}3$.

Game 2-*h*-2 ($h = 1, \ldots, \widehat{q}$): This game is similar to $\mathsf{Game}\ 2\text{-}h\text{-}1$ with the exception that the component $\boldsymbol{k}_0^*$ of the h-th queried key for an access structure-user identity pair ($\mathbb{S}_h = (\boldsymbol{M}^{(h)}, \rho_h), ID_h$) is computed as follows:

$$\boldsymbol{k}_0^* = (-s_0, \boxed{r_0}, 1, \eta_0, 0)_{\mathbb{B}_0^*}, \qquad (19)$$

where $r_0 \xleftarrow{\$} \mathbb{F}_q$, and all the other variables are as in $\mathsf{Game}\ 2\text{-}h\text{-}1$.

Game 2-*h*-3 ($h = 1, \ldots, \widehat{q}$): This game is almost identical to $\mathsf{Game}\ 2\text{-}h\text{-}2$ except that the components of the h-th queried key for an access structure-user identity pair ($\mathbb{S}_h = (\boldsymbol{M}^{(h)}, \rho_h), ID_h$) is constructed as follows:

$$\begin{aligned} &\boldsymbol{k}_0^* = (-s_0, r_0, 1, \eta_0, 0)_{\mathbb{B}_0^*} \qquad (20) \\ &\text{for } i = 1, \ldots, \ell, \\ &\boldsymbol{k}_i^* = (\mu_i(t, -1), s_i + \theta_i A_t, -\theta_i, 0^4, \boxed{0^2}, \eta_{i,1}, \eta_{i,2}, 0^2, 0^2)_{\mathbb{B}^*}, \qquad (21) \end{aligned}$$

and for all $S_{j,k} \in \mathsf{PV}_{u_h}$,

$$\boldsymbol{k}^*_{j,k,1} = (\mu_{j,k,1}(d+1,-1), s_{j,k,1} + \theta_{j,k}(L_j \| D_k), -\theta_{j,k}, 0^4, \boxed{0^2}, \eta_{j,k,1,1}, \eta_{j,k,1,2}, 0^2, 0^2)_{\mathbb{B}^*} \quad (22)$$
$$\boldsymbol{k}^*_{j,k,2} = (\mu_{j,k,2}(d+2,-1), \quad s_{j,k,2}((L_j \| L_k), -1), \quad 0^4, \boxed{0^2}, \eta_{j,k,2,1}, \eta_{j,k,2,2}, 0^2, 0^2)_{\mathbb{B}^*}, (23)$$

where $r_0 \xleftarrow{\$} \mathbb{F}_q$, and all the other variables are generated as in Game 2-h-2.

Game 3: This game is similar to Game 2-$\widehat{q}$-3 with the only exception that the components $\boldsymbol{c}_0$ and c of the challenge ciphertext CT^* are computed as follows:

$$\boldsymbol{c}_0 = (\omega, \tau, \boxed{\zeta'}, 0, \varphi_0)_{\mathbb{B}_0}, \quad (24)$$
$$c = g_T^{\zeta} M_b^*, \quad (25)$$

where $\zeta' \xleftarrow{\$} \mathbb{F}_q$ (i.e., independent from $\zeta \xleftarrow{\$} \mathbb{F}_q$), and all other variables are generated as in Game 2-$\widehat{q}$-3.

Let $\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(2\text{-}h\text{-}\jmath)}_{\mathcal{A}}(\lambda)$ $(h = 1, \ldots, \widehat{q};\ \jmath = 1, 2, 3)$ and $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 0, 1, 2-h-$\jmath$ and 3 respectively. Clearly, $\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda)$ is equivalent to $\mathsf{Adv}^{\mathsf{RABE,IND\text{-}CPA}}_{\mathcal{A}}(\lambda)$ and $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$.

Lemmas 6–10 will evaluate the gaps between pairs of $\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)$, $\{\mathsf{Adv}^{(2\text{-}h\text{-}1)}_{\mathcal{A}}(\lambda), \ldots, \mathsf{Adv}^{(2\text{-}h\text{-}3)}_{\mathcal{A}}(\lambda)\}_{h=1,\ldots,\widehat{q}}$, and $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)$. From these lemmas we obtain,

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{RABE,IND\text{-}CPA}}_{\mathcal{A}}(\lambda) &= \mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) \\
&\le \left|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)\right| + \sum_{h=1}^{\widehat{q}} \Bigg[\left|\mathsf{Adv}^{(2\text{-}(h-1)\text{-}3)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2\text{-}h\text{-}1)}_{\mathcal{A}}(\lambda)\right| + \\
&\quad \sum_{\jmath=1}^{2} \left|\mathsf{Adv}^{(2\text{-}h\text{-}\jmath)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2\text{-}h\text{-}(\jmath+1))}_{\mathcal{A}}(\lambda)\right| \Bigg] + \left|\mathsf{Adv}^{(2\text{-}\widehat{q}\text{-}3)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)\right| + \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) \\
&\le \mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}_1}(\lambda) + \sum_{h=1}^{\widehat{q}} \left[\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}_{2\text{-}h\text{-}1}}(\lambda) + \mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}_{2\text{-}h\text{-}2}}(\lambda) \right] + (4\widehat{q} + 1)/q.
\end{aligned}$$

Therefore, from lemmas 1 and 2, we obtain the upper bound of $\mathsf{Adv}^{\mathsf{RABE,IND\text{-}CPA}}_{\mathcal{A}}(\lambda)$. This completes the proof of Theorem 1. □

Lemma 6. *For any probabilistic polynomial-time 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,*

$$\left|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)\right| \le \mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}_1}(\lambda).$$

Lemma 7. *For any probabilistic polynomial-time adversary $\mathcal{A}$, there exists probabilistic machine $\mathcal{B}_{2\text{-}1}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ,*

$$\left|\mathsf{Adv}^{(2\text{-}(h-1)\text{-}3)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2\text{-}h\text{-}1)}_{\mathcal{A}}(\lambda)\right| \le \mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}_{2\text{-}h\text{-}1}}(\lambda) + 2/q,$$

where $\mathcal{B}_{2\text{-}h\text{-}1}(\cdot) = \mathcal{B}_{2\text{-}1}(h, \cdot)$.

Lemma 8. *For any adversary $\mathcal{A}$, for any security parameter λ, $\mathsf{Adv}^{(2\text{-}h\text{-}1)}_{\mathcal{A}}(\lambda) = \mathsf{Adv}^{(2\text{-}h\text{-}2)}_{\mathcal{A}}(\lambda)$.*

Lemma 9. *For any probabilistic polynomial-time adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_{2\text{-}2}$ whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ,*

$$\left|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}2)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}3)}(\lambda)\right| \leq \mathsf{Adv}_{\mathcal{B}_{2\text{-}h\text{-}2}}^{\mathsf{P2}}(\lambda) + 2/q,$$

where $\mathcal{B}_{2\text{-}h\text{-}2}(\cdot) = \mathcal{B}_{2\text{-}2}(h, \cdot)$.

Lemma 10. *For any adversary $\mathcal{A}$, for any security parameter λ,*

$$\left|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\widehat{q}\text{-}3)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)\right| \leq 1/q.$$

The proofs of Lemmas 6–10 are given in Appendix A.

5 Efficiency

Table 1 and 2 compare our RABE scheme with the fully secure RABE scheme [19] and the selectively secure construction [1] which are currently the best known results for RABE. Note that we do not consider [4] for comparison as, unlike ours, it uses a key update mechanism for managing revocation.

Table 1: Communication and storage comparison

RABE	$\sharp$PP	$\sharp\mathsf{SK}_{\mathbb{S},ID}$	$\sharp\mathsf{CT}_{\Gamma,\mathsf{RL}}$	$\sharp\mathsf{param}_{\mathbb{G}}$	Security	Complexity Assumptions
[1]	$d + \log N_{\max} + 1$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$	$2(\ell + 1)\log N_{\max}$ in $\mathbb{G}$	$\sharp\Gamma + \sharp\mathsf{RI}\log\frac{N_{\max}}{\sharp\mathsf{RI}}$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$	q (prime)	Selective	DBDH
[19]	$d + \log N_{\max} + 1$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$	$2\ell + 2\log N_{\max}$ in $\mathbb{G}$	$1 + \sharp\Gamma + \sharp\mathsf{RI}\log\frac{N_{\max}}{\sharp\mathsf{RI}}$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$	n (composite)	Full	SD, GSD, Composite DH
Ours	111 in $\mathbb{G}$, 1 in $\mathbb{G}_T$	$5 + 16\ell +$ $16[\log^2 N_{\max} + \log N_{\max}]$ in $\mathbb{G}$	$16\sharp\Gamma + 64\sharp\mathsf{RI} - 27$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$	q (prime)	Full	DLIN

Here, DBDH, SD, GSD and Composite DH respectively stand for the Decisional Bilinear Diffie-Hellman, Subgroup Decision, Generalized Subgroup Decision and Composite Diffie-Hellman assumptions.

Table 2: Computation comparison

RABE	RABE.Setup	RABE.GenKey	RABE.Encrypt	RABE.Decrypt
[1]	1in $\mathbb{G}_T$;1	$(\log N_{\max} + 1)[\ell(d + 4) +$ $\log N_{\max} + 4]$ in $\mathbb{G}$	$1 + (d + 4)\sharp\Gamma + \sharp\mathsf{RI}\log\frac{N_{\max}}{\sharp\mathsf{RI}}(\log N_{\max} + 2)$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$	$2\ell + 2$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$; $2\ell + 2$
[19]	$d + \log N_{\max}$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$;1	$3\ell + \log^2 N_{\max} +$ $4\log N_{\max} + 4$ in $\mathbb{G}$	$1 + \sharp\Gamma + \sharp\mathsf{RI}\log\frac{N_{\max}}{\sharp\mathsf{RI}}(\log N_{\max} + 2)$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$	2ℓ in $\mathbb{G}$; $2\ell + 2$
Ours	111 in $\mathbb{G}$, 1 in $\mathbb{G}_T$;1	$10 + 96\ell +$ $96(\log^2 N_{\max} + \log N_{\max})$ in $\mathbb{G}$	$80\sharp\Gamma + 160\sharp\mathsf{RI} - 49$ in $\mathbb{G}$, 1 in $\mathbb{G}_T$	$16\ell + 16$ in $\mathbb{G}$; $16\ell + 37$

Here, '$x; y$' denotes 'x many exponentiations and y many pairings'.

Our scheme is the *first fully secure unbounded* RABE construction built in *prime order* bilinear group setting. We note the following facts:

- Our RABE protocol is the first to achieve constant public parameter size and thus, unlike [1], [19], can accommodate large attribute universe and an unbounded number of users.
- Our scheme provides full security instead of selective security achieved in [1] at the expense of some amount of efficiency loss. Note that it is usual to compromise in efficiency in order to achieve better security [6], [21].

- Our scheme uses prime order bilinear group as opposed to composite order bilinear group used in the fully secure RABE construction of [19]. As noted by Freeman [8], [13], the only known instantiation of composite order bilinear group uses elliptic curves (or more generally, abelian varieties) over finite fields. Since the elliptic curve group order must be infeasible to factor, it must be at least 1024 bits. On the other hand, the size of a prime order elliptic curve group that provides an equivalent level of security is only 160 bits which is almost 7 times smaller. This difference in the group order results in great reduction of the private key and ciphertext sizes when compared in terms of bit length.
- On a more positive note, we employ the subset difference (SD) method which always provides a smaller covering set compared to the complete subtree (CS) scheme [12] and all previous RABE constructions use CS scheme. Consequently, we could achieve a much smaller ciphertext size, particularly when dealing with large number of users in the system, at the expense of an admissible increase in the private key size.
- Regarding computational efficiency of our RABE scheme, note that due to the excessive bit length of the group order, group-operations and pairing computations are prohibitively slow on composite order elliptic curves[8], [13]. In particular, an exponentiation is nearly 25 times slower and a pairing computation is roughly 50 times slower on a 1024 bit composite order elliptic curve than the corresponding operations on a comparable prime order curve [8]. In this light, we can readily see from Table 2 that the computational cost of private key generation and encryption algorithms of our RABE scheme is very close to that of [19], the only existing RABE scheme with full security, and the decryption algorithm is much faster than that of [19]. Note that in the table we do not consider the cost of generating private sets and covering sets, since those are roughly the same in all the three schemes.

6 Conclusion

In this paper, we have developed the *first fully secure unbounded* RABE scheme in *prime order bilinear groups* that realizes user revocation through the *subset difference* (SD) scheme – a more efficient variant of the subset cover (SC) framework of Naor et al. [15] as compared to the complete subtree (CS) scheme used in all previous RABE constructions. The full security of our RABE scheme has been proven under the DLIN assumption by extending the technique of dual system encryption over dual pairing vector spaces described in [16], [17]. Due to the application of prime order bilinear groups and the SD scheme, our RABE scheme is highly *broadcast efficient.* Note that, like the RIBE scheme of [12], our RABE scheme can also be integrated with the layered subset difference (LSD) scheme [10] which is an improved variant of SD. One may attempt to construct an RABE scheme that specifies the decryption policies along with the revocation list in the ciphertext rather than incorporating the access structure in the private keys of the users. Moreover, it would be interesting to investigate the use the dynamic version of SD [7] in order to overcome the problem of maintaining a large static binary tree within the state. The possibility to apply SD in designing more advanced primitives such as revocable storage attribute-based encryption (RSABE) or revocable storage predicate encryption (RSPE) [11] is another interesting direction of research.

References

1. Attrapadung, N., Imai, H.: Attribute-based encryption supporting direct/indirect revocation modes. In: Cryptography and Coding, pp. 278–300. Springer (2009)
2. Attrapadung, N., Imai, H.: Conjunctive broadcast and attribute-based encryption. In: Pairing-Based Cryptography–Pairing 2009, pp. 248–265. Springer (2009)
3. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: Security and Privacy, 2007. SP'07. IEEE Symposium on. pp. 321–334. IEEE (2007)

4. Boldyreva, A., Goyal, V., Kumar, V.: Identity-based encryption with efficient revocation. In: Proceedings of the 15th ACM conference on Computer and communications security. pp. 417–426. ACM (2008)
5. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Advances in Cryptology–CRYPTO 2004. pp. 41–55. Springer (2004)
6. Boneh, D., Franklin, M.: Identity-based encryption from the weil pairing. In: Advances in CryptologyŮCRYPTO 2001. pp. 213–229. Springer (2001)
7. Chen, W., Ge, Z., Zhang, C., Kurose, J., Towsley, D.: On dynamic subset difference revocation scheme. In: NETWORKING 2004. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications, pp. 743–758. Springer (2004)
8. Freeman, D.M.: Converting pairing-based cryptosystems from composite-order groups to prime-order groups. In: Advances in Cryptology–EUROCRYPT 2010, pp. 44–61. Springer (2010)
9. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM conference on Computer and communications security. pp. 89–98. ACM (2006)
10. Halevy, D., Shamir, A.: The lsd broadcast encryption scheme. In: Advances in CryptologyŮCRYPTO 2002, pp. 47–60. Springer (2002)
11. Lee, K., Choi, S.G., Lee, D.H., Park, J.H., Yung, M.: Self-updatable encryption: Time constrained access control with hidden attributes and better efficiency. In: Advances in Cryptology-ASIACRYPT 2013, pp. 235–254. Springer (2013)
12. Lee, K., Lee, D.H., Park, J.H.: Efficient revocable identity-based encryption via subset difference methods. IACR Cryptology ePrint Archive 2014, 132 (2014)
13. Lewko, A.: Tools for simulating features of composite order bilinear groups in the prime order setting. In: Advances in Cryptology–EUROCRYPT 2012, pp. 318–335. Springer (2012)
14. Martin, T., Martin, K., Wild, P.: Establishing the broadcast efficiency of the subset difference revocation scheme. Designs, Codes and Cryptography 51(3), 315–334 (2009), `http://dx.doi.org/10.1007/s10623-008-9263-x`
15. Naor, D., Naor, M., Lotspiech, J.: Revocation and tracing schemes for stateless receivers. In: Advances in CryptologyŮCRYPTO 2001. pp. 41–62. Springer (2001)
16. Okamoto, T., Takashima, K.: Fully secure functional encryption with general relations from the decisional linear assumption. In: Advances in Cryptology–CRYPTO 2010, pp. 191–208. Springer (2010)
17. Okamoto, T., Takashima, K.: Fully secure unbounded inner-product and attribute-based encryption. In: Advances in Cryptology–ASIACRYPT 2012, pp. 349–366. Springer (2012)
18. Ostrovsky, R., Sahai, A., Waters, B.: Attribute-based encryption with non-monotonic access structures. In: Proceedings of the 14th ACM conference on Computer and communications security. pp. 195–203. ACM (2007)
19. Qian, J.l., Dong, X.l.: Fully secure revocable attribute-based encryption. Journal of Shanghai Jiaotong University (Science) 16, 490–496 (2011)
20. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Advances in Cryptology–EUROCRYPT 2005, pp. 457–473. Springer (2005)
21. Waters, B.: Dual system encryption: Realizing fully secure ibe and hibe under simple assumptions. In: Advances in Cryptology-CRYPTO 2009, pp. 619–636. Springer (2009)
22. Waters, B.: Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In: Public Key Cryptography–PKC 2011, pp. 53–70. Springer (2011)

A Proofs of Lemmas 6–10

• **Proof of Lemma 6**: In order to prove Lemma 6, we construct a probabilistic machine $\mathcal{B}_1$ against Problem 1 using an adversary $\mathcal{A}$ in a security game (Game 0 or 1) as a black box as follows:

$\mathcal{B}_1$ is given a Problem 1 instance

$$\varrho = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}}_0^*, \widehat{\mathbb{B}}, \widehat{\mathbb{B}}^*, \boldsymbol{e}_{\beta,0}, \{\boldsymbol{e}_{\beta,t,i}\}_{t=1,\ldots,d;i=1,2}, \{\boldsymbol{e}_{\beta,d+\upsilon,\varpi,i}\}_{\upsilon=1,2;\varpi=1,\ldots,\widehat{r}_{\max};i=1,2}).$$

$\mathcal{B}_1$ plays a role of the challenger in the security game against adversary $\mathcal{A}$ as follows:

Setup: $\mathcal{B}_1$ provides $\mathcal{A}$ with the public parameters $\mathsf{PP} = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}})$ of Game 0 (and 1), where $\widehat{\mathbb{B}}_0 = \{\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}\}$, $\widehat{\mathbb{B}} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_4, \boldsymbol{b}_{13}, \boldsymbol{b}_{14}\}$ and a state $\mathsf{ST} = (\mathcal{T}, \mathsf{UL})$, where $\mathcal{T}$ is a full binary tree having $N_{\max}$ leaf nodes and $\mathsf{UL} = \varnothing$.

Phase 1: When a private key query is issued by $\mathcal{A}$ for an access structure-user identity pair $(\mathbb{S}_\imath = (\boldsymbol{M}^{(\imath)}, \rho_\imath), ID_\imath)$ such that $\boldsymbol{M}^{(\imath)}$ is an $\ell \times r$ matrix, $\mathcal{B}_1$ answers normal key

$$\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath} = (\mathsf{PV}_{u_\imath}, \boldsymbol{k}_0^*, \{\boldsymbol{k}_i^*\}_{i=1,\ldots,\ell}, \{\boldsymbol{k}_{j,k,1}^*, \boldsymbol{k}_{j,k,2}^*\}_{S_{j,k} \in \mathsf{PV}_{u_\imath}}),$$

as defined in equations (1)–(4) that is computed using $\widehat{\mathbb{B}}_0^*, \widehat{\mathbb{B}}^*$ of the Problem 1 instance, and the updated state ST.

Challenge: When $\mathcal{B}_1$ receives a challenge from $\mathcal{A}$ with challenge plaintexts M_0^*, M_1^*, an attribute set $\mathit{\Gamma}^* = \{(t, A_t) | 1 \leq t \leq d\}$ and set of revoked user identities RL^*, $\mathcal{B}_1$ computes the challenge ciphertext as follows:

1. It first chooses $\zeta \xleftarrow{\$} \mathbb{F}_q$ and computes
$$\boldsymbol{c}_0 = \boldsymbol{e}_{\beta,0} + \zeta \boldsymbol{b}_{0,3}.$$
2. For all $(t, A_t) \in \mathit{\Gamma}^*$, it sets
$$\boldsymbol{c}_t = \boldsymbol{e}_{\beta,t,1} + A_t \boldsymbol{e}_{\beta,t,2}.$$
3. It defines the revoked user serial number set $\mathsf{RI}^* \subseteq \mathcal{N}$ from RL^* by using UL. Next it obtains $\mathsf{CV}_{\mathsf{RI}^*} = \{S_{j_1,k_1}, \ldots, S_{j_m,k_m}\}$, where $m \leq \widehat{r}_{\max}$, by running $\mathsf{SD.Cover}(\mathcal{T}, \mathsf{RI}^*)$. Note that, as explained in Section 2.6, here j_ϖ and k_ϖ are all distinct. For all $S_{j_\varpi,k_\varpi} \in \mathsf{CV}_{\mathsf{RI}^*}$, $\mathcal{B}_1$ sets
$$\boldsymbol{c}_{j_\varpi,k_\varpi,1} = \boldsymbol{e}_{\beta,d+1,\varpi,1} + (L_{j_\varpi} \| D_{k_\varpi}) \boldsymbol{e}_{\beta,d+1,\varpi,2},$$
$$\boldsymbol{c}_{j_\varpi,k_\varpi,2} = \boldsymbol{e}_{\beta,d+2,\varpi,1} + (L_{j_\varpi} \| L_{k_\varpi}) \boldsymbol{e}_{\beta,d+2,\varpi,2}.$$
4. It selects $b \xleftarrow{\$} \{0,1\}$ and computes $c = g_T^\zeta M_b^*$.
5. It returns the challenge ciphertext
$$\mathsf{CT}^* = (\mathsf{CV}_{\mathsf{RI}^*}, c, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,A_t) \in \mathit{\Gamma}^*}, \{\boldsymbol{c}_{j_\varpi,k_\varpi,1}, \boldsymbol{c}_{j_\varpi,k_\varpi,2}\}_{S_{j_\varpi,k_\varpi} \in \mathsf{CV}_{\mathsf{RI}^*}}).$$

Phase 2: $\mathcal{A}$ may continue to query private keys, and $\mathcal{B}_1$ executes the same procedure as in Phase 1.

Guess: $\mathcal{A}$ finally outputs bit b'. If $b = b'$, $\mathcal{B}_1$ outputs $\beta' = 1$. Otherwise, $\mathcal{B}_1$ outputs $\beta' = 0$.

It is straightforward that the distribution by $\mathcal{B}_1$'s simulation given a Problem 1 instance with $\beta \in \{0,1\}$, is equivalent to that in Game 0 or Game 1, according as $\beta = 0$ or $\beta = 1$ respectively. □

• **Proof of Lemma 7**: In order to prove Lemma 7, we construct a probabilistic machine $\mathcal{B}_{2\text{-}1}$ against Problem 2 using an adversary $\mathcal{A}$ in a security game (Game 2-$(h-1)$-3 or 2-h-1) as a black box. $\mathcal{B}_{2\text{-}1}$ is given an integer h and a Problem 2 instance

$$\varrho = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}}_0^*, \widehat{\mathbb{B}}, \widehat{\mathbb{B}}^*, \boldsymbol{h}_{\beta,0}^*, \boldsymbol{e}_0, \{\boldsymbol{h}_{\beta,t,i}^*, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d; i=1,2},$$
$$\{\boldsymbol{h}_{\beta,d+\upsilon,\varpi,i}^*, \boldsymbol{e}_{d+\upsilon,\varpi,i}\}_{\upsilon=1,2; \varpi=1,\ldots,\log^2 N_{\max}+\widehat{r}_{\max}; i=1,2}).$$

$\mathcal{B}_{2\text{-}1}$ plays a role of the challenger in the security game against adversary $\mathcal{A}$ as follows:

Setup: $\mathcal{B}_{2\text{-}1}$ provides $\mathcal{A}$ the public parameters $\mathsf{PP} = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}})$ of Game 2-$(h-1)$-3 (and 2-h-1), where $\widehat{\mathbb{B}}_0$ and $\widehat{\mathbb{B}}$ are obtained from the Problem 2 instance, and a state $\mathsf{ST} = (\mathcal{T}, \mathsf{UL})$, where $\mathcal{T}$ is a full binary tree with $N_{\max}$ leaf nodes and $\mathsf{UL} = \varnothing$.

Phase 1: When the $\imath$-th private key query is issued by $\mathcal{A}$ for access structure-user identity pair $(\mathbb{S}_\imath = (\boldsymbol{M}^{(\imath)}, \rho_\imath), ID_\imath)$ such that $\boldsymbol{M}^{(\imath)}$ is an $\ell \times r$ matrix, $\mathcal{B}_{2\text{-}1}$ answers as follows:

(a) When $1 \leq \imath \leq h-1$, $\mathcal{B}_{2\text{-}1}$ answers semi-functional key

$$\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath} = (\mathsf{PV}_{u_\imath}, \boldsymbol{k}_0^*, \{\boldsymbol{k}_i^*\}_{i=1,\ldots,\ell}, \{\boldsymbol{k}_{j,k,1}^*, \boldsymbol{k}_{j,k,2}^*\}_{S_{j,k} \in \mathsf{PV}_{u_\imath}}),$$

as defined in equations (20)–(23), that is computed using $\widehat{\mathbb{B}}_0^*$ and $\widehat{\mathbb{B}}^*$ of the Problem 2 instance.

(b) When $\imath = h$, $\mathcal{B}_{2\text{-}1}$ generates the queried key as follows:

1. It picks $\widetilde{\pi}_t, \widetilde{\pi}_{d+1,\log N_{\max}(\iota-1)+\varepsilon}, \xi_t, \xi_{d+1,\log N_{\max}(\iota-1)+\varepsilon}, \widetilde{g}_k, \widetilde{g}_0, \widetilde{\xi}_k, \widetilde{\xi}_0 \xleftarrow{\$} \mathbb{F}_q$ for $t = 1, \ldots, d$; $\iota, \varepsilon = 1, \ldots, \log N_{\max}$; $k = 1, \ldots, r$.
2. It computes

$$\begin{aligned} &\text{for } k = 1, \ldots, r, \\ &\widetilde{\boldsymbol{p}}_{\beta,0,k}^* = \widetilde{g}_k \boldsymbol{h}_{\beta,0}^* + \widetilde{\xi}_k \boldsymbol{b}_{0,1}^*, \\ &\widetilde{\boldsymbol{p}}_{\beta,0,0}^* = \widetilde{g}_0 \boldsymbol{h}_{\beta,0}^* + \widetilde{\xi}_0 \boldsymbol{b}_{0,1}^*, \end{aligned}$$

and sets

$$\boldsymbol{k}_0^* = -\sum_{k=1}^{r} \widetilde{\boldsymbol{p}}_{\beta,0,k}^* - \widetilde{\boldsymbol{p}}_{\beta,0,0}^* + \boldsymbol{b}_{0,3}^*.$$

3. For $t = 1, \ldots, d$, it computes

$$\begin{aligned} &\text{for } \jmath = 1, 2, \\ &\boldsymbol{p}_{\beta,t,\jmath}^* = \widetilde{\pi}_t \boldsymbol{h}_{\beta,t,\jmath}^* + \xi_t \boldsymbol{b}_{2+\jmath}^* \text{ and} \\ &\text{for } k = 1, \ldots, r, \\ &\widetilde{\boldsymbol{p}}_{\beta,t,k}^* = \widetilde{g}_k \boldsymbol{h}_{\beta,t,1}^* + \widetilde{\xi}_k \boldsymbol{b}_3^*. \end{aligned}$$

Next, for each $i = 1, \ldots, \ell$, it sets

$$\boldsymbol{k}_i^* = A_t \boldsymbol{p}_{\beta,t,1}^* - \boldsymbol{p}_{\beta,t,2}^* + \sum_{k=1}^{r} \boldsymbol{M}_{i,k}^{(h)} \widetilde{\boldsymbol{p}}_{\beta,t,k}^*,$$

where $\rho_h(i) = (t, A_t)$, for $i = 1, \ldots, \ell$, and $(\boldsymbol{M}_{i,k}^{(h)})_{i=1,\ldots,\ell;k=1,\ldots,r} = \boldsymbol{M}^{(h)}$.

4. For each $S_{j_\iota,k_\varepsilon} \in \mathsf{PV}_{u_h}$, it performs the following operations: It first selects $\widetilde{g}_{j_\iota,k_\varepsilon,1}$, $\widetilde{g}_{j_\iota,k_\varepsilon,2}, \widetilde{\xi}_{j_\iota,k_\varepsilon,1}, \widetilde{\xi}_{j_\iota,k_\varepsilon,2} \xleftarrow{\$} \mathbb{F}_q$ such that $\widetilde{g}_{j_\iota,k_\varepsilon,1} + \widetilde{g}_{j_\iota,k_\varepsilon,2} = \widetilde{g}_0$ and $\widetilde{\xi}_{j_\iota,k_\varepsilon,1} + \widetilde{\xi}_{j_\iota,k_\varepsilon,2} = \widetilde{\xi}_0$, and computes for $\jmath = 1, 2$,

$$\begin{aligned} \boldsymbol{p}_{\beta,j_\iota,k_\varepsilon,1,\jmath}^* &= \widetilde{\pi}_{d+1,\log N_{\max}(\iota-1)+\varepsilon} \boldsymbol{h}_{\beta,d+1,\log N_{\max}(\iota-1)+\varepsilon,\jmath}^* + \xi_{d+1,\log N_{\max}(\iota-1)+\varepsilon} \boldsymbol{b}_{2+\jmath}^*, \\ \widetilde{\boldsymbol{p}}_{\beta,j_\iota,k_\varepsilon,1}^* &= \widetilde{g}_{j_\iota,k_\varepsilon,1} \boldsymbol{h}_{\beta,d+1,\log N_{\max}(\iota-1)+\varepsilon,1}^* + \widetilde{\xi}_{j_\iota,k_\varepsilon,1} \boldsymbol{b}_3^*, \\ \widetilde{\boldsymbol{p}}_{\beta,j_\iota,k_\varepsilon,2,\jmath}^* &= \widetilde{g}_{j_\iota,k_\varepsilon,2} \boldsymbol{h}_{\beta,d+2,\log N_{\max}(\iota-1)+\varepsilon,\jmath}^* + \widetilde{\xi}_{j_\iota,k_\varepsilon,2} \boldsymbol{b}_{2+\jmath}^*. \end{aligned}$$

Then it obtains

$$\begin{aligned} \boldsymbol{k}_{j_\iota,k_\varepsilon,1}^* &= (L_{j_\iota} \| D_{k_\varepsilon}) \boldsymbol{p}_{\beta,j_\iota,k_\varepsilon,1,1}^* - \boldsymbol{p}_{\beta,j_\iota,k_\varepsilon,1,2}^* + \widetilde{\boldsymbol{p}}_{\beta,j_\iota,k_\varepsilon,1}^*, \\ \boldsymbol{k}_{j_\iota,k_\varepsilon,2}^* &= (L_{j_\iota} \| L_{k_\varepsilon}) \widetilde{\boldsymbol{p}}_{\beta,j_\iota,k_\varepsilon,2,1}^* - \widetilde{\boldsymbol{p}}_{\beta,j_\iota,k_\varepsilon,2,2}^*. \end{aligned}$$

5. It returns the queried key as

$$\mathsf{SK}_{\mathbb{S}_h, ID_h} = (\mathsf{PV}_{u_h}, \boldsymbol{k}_0^*, \{\boldsymbol{k}_i^*\}_{i=1,\ldots,\ell}, \{\boldsymbol{k}_{j_\iota,k_\varepsilon,1}^*, \boldsymbol{k}_{j_\iota,k_\varepsilon,2}^*\}_{S_{j_\iota,k_\varepsilon} \in \mathsf{PV}_{u_h}}).$$

(c) When $h+1 \leq \imath \leq \widehat{q}$, $\mathcal{B}_{2\text{-}1}$ answers normal key

$$\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath} = (\mathsf{PV}_{u_\imath}, \boldsymbol{k}_0^*, \{\boldsymbol{k}_i^*\}_{i=1,\ldots,\ell}, \{\boldsymbol{k}_{j,k,1}^*, \boldsymbol{k}_{j,k,2}^*\}_{S_{j,k} \in \mathsf{PV}_{u_\imath}})$$

as defined in equations (1)–(4), that is computed using $\widehat{\mathbb{B}}_0^*, \widehat{\mathbb{B}}^*$ of the Problem 2 instance.

Challenge: When $\mathcal{B}_{2\text{-}1}$ receives a challenge from $\mathcal{A}$ with challenge plaintexts M_0^*, M_1^*, an attribute set $\Gamma^* = \{(t, A_t) | 1 \le t \le d\}$ and revocation list RL^*, $\mathcal{B}_{2\text{-}1}$ computes the challenge ciphertext as described below. For the ease of explanation, we assume that the h-th private key query occurs before the challenge is submitted. The simulation would be similar for the other case.

1. It computes
$$\boldsymbol{c}_0 = \boldsymbol{e}_0 + \zeta \boldsymbol{b}_{0,3} + \boldsymbol{q}_0,$$
where $\zeta \xleftarrow{\$} \mathbb{F}_q, \boldsymbol{q}_0 \xleftarrow{\$} \mathsf{span}\langle \boldsymbol{b}_{0,5} \rangle$.
2. For all $(t, A_t) \in \Gamma^*$, it sets
$$\boldsymbol{c}_t = \boldsymbol{e}_{t,1} + A_t \boldsymbol{e}_{t,2} + \boldsymbol{q}_t,$$
where $\boldsymbol{q}_t \xleftarrow{\$} \mathsf{span}\langle \boldsymbol{b}_{13}, \boldsymbol{b}_{14} \rangle$.
3. It defines the revoked user serial number set $\mathsf{RI}^* \subseteq \mathcal{N}$ from RL^* by using UL. Next it obtains $\mathsf{CV}_{\mathsf{RI}^*} = \{S_{j_1,k_1}, \ldots, S_{j_m,k_m}\}$, where $m \le \hat{r}_{\max}$, by performing $\mathsf{SD.Cover}(\mathcal{T}, \mathsf{RI}^*)$. Note that, here $j_{\iota'}$'s and $k_{\iota'}$'s are all distinct. For all $S_{j_{\iota'},k_{\iota'}} \in \mathsf{CV}_{\mathsf{RI}^*}$, $\mathcal{B}_{2\text{-}1}$ sets
$$\boldsymbol{c}_{j_{\iota'},k_{\iota'},1} = \boldsymbol{e}_{d+1,\iota',1} + (L_{j_{\iota'}} \| D_{k_{\iota'}}) \boldsymbol{e}_{d+1,\iota',2} + \boldsymbol{q}_{d+1,\iota'},$$
$$\boldsymbol{c}_{j_{\iota'},k_{\iota'},2} = \boldsymbol{e}_{d+2,\iota',1} + (L_{j_{\iota'}} \| L_{k_{\iota'}}) \boldsymbol{e}_{d+2,\iota',2} + \boldsymbol{q}_{d+2,\iota'},$$
where $\boldsymbol{q}_{d+1,\iota'}, \boldsymbol{q}_{d+2,\iota'} \xleftarrow{\$} \mathsf{span}\langle \boldsymbol{b}_{13}, \boldsymbol{b}_{14} \rangle$. While computing the above values for any $S_{j_{\iota'},k_{\iota'}} \in \mathsf{CV}_{\mathsf{RI}^*}$, if it is found that $(j_{\iota'} = j_\iota) \ \wedge \ (D_{k_{\iota'}} = D_{k_\varepsilon})$ for some $S_{j_\iota,k_\varepsilon} \in \mathsf{PV}_{u_h}$ corresponding to the h-th private key query, then the values $\boldsymbol{e}_{d+1,\iota',\jmath}, \boldsymbol{e}_{d+2,\iota',\jmath}$ for $\jmath = 1, 2$, that are used in the above computation are respectively $\boldsymbol{e}_{d+1,\log N_{\max}(\iota-1)+\varepsilon,\jmath}, \boldsymbol{e}_{d+2,\log N_{\max}(\iota-1)+\varepsilon,\jmath}$, where $1 \le \iota, \varepsilon \le \log N_{\max}$. Otherwise, fresh $\boldsymbol{e}_{d+1,\iota',\jmath}, \boldsymbol{e}_{d+2,\iota',\jmath}$ for $\jmath = 1, 2$ with $\iota' \ne \log N_{\max}(\iota - 1) + \varepsilon$, $1 \le \iota, \varepsilon \le \log N_{\max}$, are used.
4. It chooses $b \xleftarrow{\$} \{0, 1\}$ and computes $c = g_T^{\zeta} M_b^*$.
5. It returns the challenge ciphertext
$$\mathsf{CT}^* = (\mathsf{CV}_{\mathsf{RI}^*}, c, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,A_t) \in \Gamma^*}, \{\boldsymbol{c}_{j_{\iota'},k_{\iota'},1}, \boldsymbol{c}_{j_{\iota'},k_{\iota'},2}\}_{S_{j_{\iota'},k_{\iota'}} \in \mathsf{CV}_{\mathsf{RI}^*}}).$$

Phase 2: $\mathcal{A}$ may continue to request additional private keys. $\mathcal{B}_{2\text{-}1}$ performs the same procedure as in Case (c) of Phase 1. Note that, here we have assumed that the h-th key query occurred before the challenge has been submitted. The modification for the other case is straightforward.

Guess: $\mathcal{A}$ finally outputs a bit b'. If $b = b'$, then $\mathcal{B}_{2\text{-}1}$ outputs $\beta' = 1$. Otherwise, $\mathcal{B}_{2\text{-}1}$ outputs $\beta' = 0$.

One can readily verify that the distribution by $\mathcal{B}_{2\text{-}1}$'s simulation given a Problem 2 instance with $\beta \in \{0, 1\}$ is equivalent to that in Game 2-$(h-1)$-3 or Game 2-h-1 according as $\beta = 0$ or $\beta = 1$ respectively, except that γ defined in Problem 2 is zero, i.e., except with probability $1/q$. □

• **Proof of Lemma 8**: We make use of the following key probabilistic (or information-theoretic) lemma to prove Lemma 8:

Lemma 11 (Lemma 8 in [17]). *For $p \in \mathbb{F}_q$, let $\mathbb{C}_p = \{(\vec{x}, \vec{v}) | \vec{x} \cdot \vec{v} = p\} \subset \mathbb{V} \times \mathbb{V}^*$, where $\mathbb{V}$ is n dimensional vector space $\mathbb{F}_q^n$ and $\mathbb{V}^*$ its dual. For all $(\vec{x}, \vec{v}) \in \mathbb{C}_p$, for all $(\vec{r}, \vec{w}) \in \mathbb{C}_p$,*
$$\mathsf{Pr}[\vec{x} \cdot \boldsymbol{U} = \vec{r} \ \wedge \ \vec{v} \cdot \boldsymbol{Z} = \vec{w}] = \mathsf{Pr}[\vec{x} \cdot \boldsymbol{Z} = \vec{r} \ \wedge \ \vec{v} \cdot \boldsymbol{U} = \vec{w}] = \frac{1}{\sharp \mathbb{C}_p},$$
where $\boldsymbol{Z} \xleftarrow{\$} \mathsf{GL}(n, \mathbb{F}_q)$, $\boldsymbol{U} = (\boldsymbol{Z}^{-1})^{\mathsf{T}}$.

It is clear that the distribution of the public parameters and the $\imath$-th private key query's answer for $\imath \neq h$ in Game 2-h-1 and Game 2-h-2 are exactly the same, namely, in both games, for $\imath < h$ the $\imath$-th queried key are generated as per equations (20)–(23), and for $\imath > h$, those are formed according as equations (1)–(4). Therefore, to prove Lemma 8, we will show that the joint distribution of the h-th private key query's answer and the challenge ciphertext in Game 2-h-1 and Game 2-h-2 are equivalent in the view of the adversary $\mathcal{A}$. For this we will show that a_0 in equation (15) is uniformly and independently distributed from the other variables in the joint distribution of adversary $\mathcal{A}$'s view. Since $a_0 = a'_0 + a''_0$, where $a'_0 = \vec{1} \cdot \vec{g}^{\intercal}$, $a''_0 = a_{j,k,1} + a_{j,k,2}$, a_0 is related only to $(a_1, \ldots, a_\ell)^{\intercal} = \boldsymbol{M}^{(h)} \cdot \vec{g}^{\intercal}$, $\{a_{j,k,1}, a_{j,k,2}\}_{S_{j,k} \in \mathsf{PV}_{u_h}}$. Also,

$$\boldsymbol{U}_t = (\boldsymbol{Z}_t^{-1})^{\intercal} \text{ for } 1 \leq t \leq d \text{ with } \rho_h(i) = (t, A_t), \text{ and}$$
$$\boldsymbol{U}_{d+1,j,k} = (\boldsymbol{Z}_{d+1,j',k'}^{-1})^{\intercal}, \boldsymbol{U}_{d+2,j,k} = (\boldsymbol{Z}_{d+2,j',k'}^{-1})^{\intercal} \text{ for } S_{j,k} \in \mathsf{PV}_{u_h}; S_{j',k'} \in \mathcal{S}; (j' = j) \wedge (D_{k'} = D_k),$$

where $\mathcal{S}$ is the collection of all subsets $S_{j,k}$ of leaf nodes of $\mathcal{T}$. Thus, a_0 is dependent only on $\{\vec{w}_i\}_{i=1,\ldots,\ell}, \{\vec{w}_{j,k,1}, \vec{w}_{j,k,2}\}_{S_{j,k} \in \mathsf{PV}_{u_h}}$ and $\{\vec{r}_t\}_{1 \leq t \leq d; \rho_h(i)=(t,A_t)}, \{\vec{r}_{j',k',1}, \vec{r}_{j',k',2} | S_{j',k'} \in \mathcal{S}; (j' = j) \wedge (D_{k'} = D_k)$ for some $S_{j,k} \in \mathsf{PV}_{u_h}\}$, where

$$\begin{aligned}
&\text{for } i = 1, \ldots, \ell,\\
&\vec{w}_i = (a_i + \pi_i A_t, -\pi_i) \cdot \boldsymbol{U}_t,\\
&\text{for } S_{j,k} \in \mathsf{PV}_{u_h},\\
&\vec{w}_{j,k,1} = (a_{j,k,1} + \pi_{j,k}(L_j \| D_k), -\pi_{j,k}) \cdot \boldsymbol{U}_{d+1,j,k}, \vec{w}_{j,k,2} = (a_{j,k,2}((L_j \| L_k), -1) \cdot \boldsymbol{U}_{d+2,j,k};\\
&\text{and for } 1 \leq t \leq d \text{ with } \rho_h(i) = (t, A_t),\\
&\vec{r}_t = \tau(1, A_t) \cdot \boldsymbol{Z}_t,\\
&\text{for } S_{j',k'} \in \mathcal{S} \text{ with } (j' = j) \;\wedge\; (D_{k'} = D_k) \text{ for some } S_{j,k} \in \mathsf{PV}_{u_h},\\
&\vec{r}_{j',k',1} = \tau(1, (L_{j'} \| D_{k'}))\boldsymbol{Z}_{d+1,j',k'}, \vec{r}_{j',k',2} = \tau(1, (L_{j'} \| L_{k'})) \cdot \boldsymbol{Z}_{d+2,j',k'}.
\end{aligned}$$

With respect to the joint distribution of these variables there are the following cases. Note that for any $i \in \{1, \ldots, \ell\}$, $(\boldsymbol{Z}_t, \boldsymbol{U}_t)$ with $\rho_h(i) = (t, A_t)$ is independent from the other variables, since as per our assumption ρ_h is injective. Also for all $S_{j,k} \in \mathsf{PV}_{u_h}$ and $S_{j',k'} \in \mathcal{S}$ with $(j' = j) \wedge (D_{k'} = D_k)$, $(\boldsymbol{Z}_{d+1,j',k'}, \boldsymbol{U}_{d+1,j,k})$, $(\boldsymbol{Z}_{d+2,j',k'}, \boldsymbol{U}_{d+2,j,k})$ are independent from the other variables since these values are utilized for one particular $(j, D_k) = (j', D_{k'})$ pair which appears in both PV_{u_h} and $\mathsf{CV}_{\mathsf{RI}^*}$ at most once.

I. $[\rho_h(i) = (t, A_t) \;\wedge\; (t, A_t) \in \Gamma^*]$: In this case from Lemma 11, the joint distribution of $(\vec{w}_i, \vec{r}_t)$ is uniformly and independently distributed on $\mathbb{C}_{\tau a_i} = \{(\vec{x}, \vec{v}) | \vec{x} \cdot \vec{v} = \tau a_i\}$ (over $\boldsymbol{Z}_t \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$).

II. $[\rho_h(i) = (t, A_t) \wedge (t, A_t) \notin \Gamma^*]$: Here, the distribution of $\vec{w}_i$ is uniformly and independently distributed on $\mathbb{F}_q^2$ (over $\boldsymbol{Z}_t \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$).

III. $[S_{j,k} \in \mathsf{PV}_{u_h}, S_{j',k'} \in \mathsf{CV}_{\mathsf{RI}^*}$ such that $(j' = j) \;\wedge\; (D_{k'} = D_k)]$: Again from Lemma 11, the joint distribution of $(\vec{w}_{j,k,1}, \vec{r}_{j',k',1})$ is uniformly and independently distributed on $\mathbb{C}_{\tau a_{j,k,1}} = \{(\vec{x}, \vec{v}) | \vec{x} \cdot \vec{v} = \tau a_{j,k,1}\}$ (over $\boldsymbol{Z}_{d+1,j',k'} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$); and the joint distribution of $(\vec{w}_{j,k,2}, \vec{r}_{j',k',2})$ is uniformly and independently distributed on $\mathbb{C}_{\tau a_{j,k,2}((L_j \| L_k) - (L_{j'} \| L_{k'}))}$ or on $\mathbb{C}_0$ (over $\boldsymbol{Z}_{d+2,j',k'} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$) according as $k' \neq k$ or $k' = k$.

IV. $[S_{j,k} \in \mathsf{PV}_{u_h}$ does not satisfy the condition $j' = j$ for any $S_{j',k'} \in \mathsf{CV}_{\mathsf{RI}^*}]$: Then the distribution of $\vec{w}_{j,k,1}$ and $\vec{w}_{j,k,2}$ are uniformly distributed on $\mathbb{F}_q^2$ (over $\boldsymbol{Z}_{d+1,j',k'}, \boldsymbol{Z}_{d+2,j',k'} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$).

We then observe that the joint distribution of $a_0, \tau, \{\vec{w}_i\}_{i=1,\ldots,\ell}$, $\{\vec{r}_t\}_{1 \leq t \leq d; \rho_h(i)=(t,A_t)}$, and $\{\vec{w}_{j,k,1}, \vec{w}_{j,k,2}\}_{S_{j,k} \in \mathsf{PV}_{u_h}}$, $\{\vec{r}_{j',k',1}, \vec{r}_{j',k',2} | S_{j',k'} \in \mathcal{S}; (j' = j) \;\wedge\; (D_{k'} = D_k)$ for some $S_{j,k} \in \mathsf{PV}_{u_h}\}$ in Cases II and IV are obviously independent from a_0.

Now, as per the restriction of the RABE security game, the adversary $\mathcal{A}$ cannot query a private key for an access structure-user identity pair such that the access structure satisfies the attribute set Γ^* and, at the same time, the user identity is not contained in the revocation list RL^*. Due to this constraint on adversary $\mathcal{A}$'s private key queries, $\vec{1} \notin \mathsf{span}\langle \boldsymbol{M}_i^{(h)} | \rho_h(i) = (t, A_t) \in \Gamma^* \rangle$ or for all $S_{j,k} \in \mathsf{PV}_{u_h}$, there does not exist $S_{j',k'} \in \mathsf{CV}_{\mathsf{RI}^*}$ such that $(j' = j) \wedge (D_{k'} = D_k) \wedge (k' \neq k)$. Therefore, $a_0 = a_0' + a_0''$ where $a_0' = \vec{1} \cdot \vec{g}^{\mathsf{T}}$, $a_0'' = a_{j,k,1} + a_{j,k,2}$, is independent from the joint distribution of τ, $\{\tau a_i = \tau \boldsymbol{M}_i^{(h)} \cdot \vec{g}^{\mathsf{T}} | \rho_h(i) = (t, A_t) \in \Gamma^*\}$ (over the random selection of $\vec{g}$) and $\{\tau a_{j,k,1}, \tau a_{j,k,2}((L_j \| L_k) - (L_{j'} \| L_{k'})) | S_{j,k} \in \mathsf{PV}_{u_h}, S_{j',k'} \in \mathsf{CV}_{RI^*}, (j' = j) \wedge (D_{k'} = D_k)\}$ (where $\tau a_{j,k,2}((L_j \| L_k) - (L_{j'} \| L_{k'})) = 0$ if $k' = k$) over the subdivision of the randomly selected element a_0'' into random parts $a_{j,k,1}, a_{j,k,2}$; which can be given by $(\vec{w}_i, \vec{r}_t)$ in Case I, and $(\vec{w}_{j,k,1}, \vec{r}_{j',k',1}), (\vec{w}_{j,k,2}, \vec{r}_{j',k',2})$ in Case III.

Thus a_0 is uniformly and independently distributed from the other variables in the joint distribution. Hence, the view of the adversary $\mathcal{A}$ in Game 2-h-1 is the same as that in Game 2-h-2. □

• **Proof of Lemma 9**: In order to prove Lemma 9, we construct a probabilistic machine $\mathcal{B}_{2\text{-}2}$ against Problem 2 using an adversary $\mathcal{A}$ in a security game (Game 2-h-2 or 2-h-3) as a black box. $\mathcal{B}_{2\text{-}2}$ acts in the same way as $\mathcal{B}_{2\text{-}1}$ in the proof of Lemma 7 except the following two points:

I. In Step 2 of Case (b), in Phase 1, $\boldsymbol{k}_0^*$ is computed as

$$\boldsymbol{k}_0^* = -\sum_{k=1}^{r} \widetilde{\boldsymbol{p}}^*_{\beta,0,k} - \widetilde{\boldsymbol{p}}^*_{\beta,0,0} + r_0' \boldsymbol{b}^*_{0,2} + \boldsymbol{b}^*_{0,3},$$

where $r_0' \xleftarrow{\$} \mathbb{F}_q$; $\widetilde{\boldsymbol{p}}^*_{\beta,0,k}, \widetilde{\boldsymbol{p}}^*_{\beta,0,0}$ are computed from $\boldsymbol{h}^*_{\beta,0}$, $\boldsymbol{b}^*_{0,1}$, in the same way as in the proof of Lemma 7; and $\boldsymbol{b}^*_{0,2}, \boldsymbol{b}^*_{0,3}$ are obtained from the Problem 2 instance.

II. In the Guess step, if $b = b'$, $\mathcal{B}_{2\text{-}2}$ outputs $\beta' = 0$, otherwise, it outputs $\beta' = 1$.

When $\beta = 0$ or $\beta = 1$, it is straightforward that the distribution by $\mathcal{B}_{2\text{-}2}$'s simulation is equivalent to that in Game 2-h-2 or Game 2-h-3 respectively except that γ defined in Problem 2 is zero, i.e., except with probability $1/q$ in each case. □

• **Proof of Lemma 10**: To prove Lemma 10, we will show that the distribution $(\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}}, \{\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath}\}_{\imath=1,\ldots,\widehat{q}}, \mathsf{CT}^*)$ in Game 2-$\widehat{q}$-3 and that in Game 3 are equivalent, where $\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath}$ is the answer to the $\imath$-th private key query and CT^* is the challenge ciphertext. By definition of the two games, we only need to consider elements on $\mathbb{V}_0$ or $\mathbb{V}_0^*$, where $\mathbb{V}_0 = \mathsf{span}\langle \mathbb{B}_0 \rangle$, $\mathbb{V}_0^* = \mathsf{span}\langle \mathbb{B}_0^* \rangle$. We define new bases $\mathbb{D}_0$ of $\mathbb{V}_0$ and $\mathbb{D}_0^*$ of $\mathbb{V}_0^*$ as follows: We generate $\theta \xleftarrow{\$} \mathbb{F}_q$ and set

$$\boldsymbol{d}_{0,2} = (0, 1, -\theta, 0, 0)_{\mathbb{B}_0} = \boldsymbol{b}_{0,2} - \theta \boldsymbol{b}_{0,3},$$
$$\boldsymbol{d}^*_{0,3} = (0, \theta, 1, 0, 0)_{\mathbb{B}_0^*} = \boldsymbol{b}^*_{0,3} + \theta \boldsymbol{b}^*_{0,2}.$$

We define $\mathbb{D}_0 = \{\boldsymbol{b}_{0,1}, \boldsymbol{d}_{0,2}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,4}, \boldsymbol{b}_{0,5}\}, \mathbb{D}_0^* = \{\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,2}, \boldsymbol{d}^*_{0,3}, \boldsymbol{b}^*_{0,4}, \boldsymbol{b}^*_{0,5}\}$. We then easily verify that $\mathbb{D}_0$ and $\mathbb{D}_0^*$ are dual orthogonal bases and are distributed the same as the original bases $\mathbb{B}_0$ and $\mathbb{B}_0^*$.

The $\mathbb{V}_0$ components $(\{\boldsymbol{k}_0^{*(\imath)}\}_{\imath=1,\ldots,\widehat{q}}, \boldsymbol{c}_0)$ in queried keys, challenge ciphertext $(\{\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath}\}_{\imath=1,\ldots,\widehat{q}}, \mathsf{CT}^*)$ in Game 2-$\widehat{q}$-3 are expressed over $\mathbb{B}_0^*$ and $\mathbb{B}_0$ respectively as $\boldsymbol{k}_0^{*(\imath)} = (-s_0^{(\imath)}, r_0^{(\imath)}, 1, \eta_0^{(\imath)}, 0)_{\mathbb{B}_0^*}$ and $\boldsymbol{c}_0 = (\omega, \tau, \zeta, 0, \varphi_0)_{\mathbb{B}_0}$. Note that here we index the $\boldsymbol{k}_0^*$ component of the queried key $\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath}$ and all its forming coefficients by $\imath$ for ease of explanation. Then,

$$\boldsymbol{k}_0^{*(\imath)} = (-s_0^{(\imath)}, r_0^{(\imath)}, 1, \eta_0^{(\imath)}, 0)_{\mathbb{B}_0^*} = (-s_0^{(\imath)}, r_0^{(\imath)} - \theta, 1, \eta_0^{(\imath)}, 0)_{\mathbb{D}_0^*} = (-s_0^{(\imath)}, \vartheta_0^{(\imath)}, 1, \eta_0^{(\imath)}, 0)_{\mathbb{D}_0^*},$$

where $\vartheta_0^{(\imath)} = r_0^{(\imath)} - \theta$ which are uniformly, independently distributed since $r_0^{(\imath)} \xleftarrow{\$} \mathbb{F}_q$. Again,

$$\boldsymbol{c}_0 = (\omega, \tau, \zeta, 0, \varphi_0)_{\mathbb{B}_0} = (\omega, \tau, \zeta + \tau\theta, 0, \varphi_0)_{\mathbb{D}_0} = (\omega, \tau, \zeta', 0, \varphi_0)_{\mathbb{D}_0},$$

where $\zeta' = \zeta + \tau\theta$ which is uniformly, independently distributed since $\theta \xleftarrow{\$} \mathbb{F}_q$.

In the light of the adversary's view, both $\{\mathbb{B}_0, \mathbb{B}_0^*\}$ and $\{\mathbb{D}_0, \mathbb{D}_0^*\}$ are consistent with public parameters $\mathsf{PP} = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}})$. Therefore, $\{\mathsf{SK}_{\mathbb{S}_\imath, ID_\imath}\}_{\imath=1,\ldots,\widehat{q}}$ and CT^* can be expressed as keys and ciphertext in two ways, in Game 2-$\widehat{q}$-3 over bases $\{\mathbb{B}_0, \mathbb{B}_0^*\}$ and in Game 3 over bases $\{\mathbb{D}_0, \mathbb{D}_0^*\}$. Thus Game 2-$\widehat{q}$-3 can be conceptually changed to Game 3 if $\tau \neq 0$, i.e., except with probability $1/q$. □

B Proofs of Lemma 1

The proof of this lemma is based on the following result:

Definition 10 (Basic Problem 1). *Basic Problem 1 is to guess $\beta \in \{0, 1\}$, given $\varrho = (\mathsf{param}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \mathbb{B}, \widehat{\mathbb{B}}^*, \boldsymbol{e}_{\beta,0}, \{\boldsymbol{e}_{\beta,i}\}_{i=1,2}) \xleftarrow{\$} \mathcal{G}_\beta^{\mathsf{BP1}}(1^\lambda)$,*

$$\mathcal{G}_\beta^{\mathsf{BP1}}(1^\lambda): \ (\mathsf{param}, \{\mathbb{B}_0, \mathbb{B}_0^*\}, \{\mathbb{B}, \mathbb{B}^*\}) \xleftarrow{\$} \mathcal{G}_{\mathsf{ob}}(1^\lambda, (n_0 = 5, n = 16)),$$

$$\widehat{\mathbb{B}}_0^* = \{\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*, \ldots, \boldsymbol{b}_{0,5}^*\}, \widehat{\mathbb{B}}^* = \{\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_6^*, \boldsymbol{b}_9^*, \ldots, \boldsymbol{b}_{16}^*\},$$

$$\omega, \varphi_0 \xleftarrow{\$} \mathbb{F}_q, \tau \xleftarrow{\$} \mathbb{F}_q^\times, \boldsymbol{e}_{0,0} = (\omega, 0, 0, 0, \varphi_0)_{\mathbb{B}_0}, \boldsymbol{e}_{1,0} = (\omega, \tau, 0, 0, \varphi_0)_{\mathbb{B}_0},$$

$$\vec{e}_1 = (1, 0), \vec{e}_2 = (0, 1) \in \mathbb{F}_q^2,$$

for $i = 1, 2$,

$$\vec{\varphi}_i \xleftarrow{\$} \mathbb{F}_q^2,$$

$$\boldsymbol{e}_{0,i} = (0^2, \omega\vec{e}_i, \quad 0^6, \quad 0^2, \vec{\varphi}_i, 0^2)_{\mathbb{B}}$$

$$\boldsymbol{e}_{1,i} = (0^2, \omega\vec{e}_i, \tau\vec{e}_i, 0^4, 0^2, \vec{\varphi}_i, 0^2)_{\mathbb{B}},$$

return $\varrho = (\mathsf{param}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \mathbb{B}, \widehat{\mathbb{B}}^*, \boldsymbol{e}_{\beta,0}, \{\boldsymbol{e}_{\beta,i}\}_{i=1,2})$.

For a probabilistic adversary $\mathcal{C}$, the advantage of $\mathcal{C}$ for Basic Problem 1, is given by,

$$\mathsf{Adv}_{\mathcal{C}}^{\mathsf{BP1}}(\lambda) = \left| \mathsf{Pr}\left[\mathcal{C}(1^\lambda, \varrho) \to 1 | \varrho \xleftarrow{\$} \mathcal{G}_0^{\mathsf{BP1}}(1^\lambda)\right] - \mathsf{Pr}\left[\mathcal{C}(1^\lambda, \varrho) \to 1 | \varrho \xleftarrow{\$} \mathcal{G}_1^{\mathsf{BP1}}(1^\lambda)\right] \right|.$$

Lemma 12. *For any probabilistic polynomial-time 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, $\mathsf{Adv}_{\mathcal{C}}^{\mathsf{BP1}}(\lambda) \leq \mathsf{Adv}_{\mathcal{F}}^{\mathsf{DLIN}}(\lambda) + 5/q$.*

Basic Problem 1 defined above is essentially Basic Problem 1 of [17] with two additional dimensions. The proof of Lemma 12 is exactly identical to that of Lemma 34 in [17].

In order to prove Lemma 1 we consider the following experiments. Problem 1 is the hybrid of the following Experiment 0, 1, 2-1-1-1, $\ldots$, 2-d-2-2, 3-$(d+1)$-1-1-1, $\ldots$, 3-$(d+2)$-$\widehat{r}_{\max}$-2-2, i.e.,

$$\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P1}}(\lambda) = \left| \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(0)}(\lambda) \to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+2)\text{-}\widehat{r}_{\max}\text{-}2\text{-}2)}(\lambda) \to 1\right] \right|.$$

Therefore from Lemma 12 and the following lemmas we obtain Lemma 1.

<u>**Experiments**</u>: In Experiment 0, a part framed by a box indicates positions of coefficients to be changed in subsequent experiment. In the other experiments, a part framed by a box indicates coefficients which were changed in a transition from the previous experiment. Experiments proceed as follows:

$$\textit{Experiment } 0 \implies \textit{Experiment } 1 \implies \{\text{Experiment 2-}p\text{-}j\text{-}l\}_{p=1,\ldots,d;j=1,2;l=1,2} \implies$$
$$\{\text{Experiment 3-}(d+\upsilon)\text{-}\varpi\text{-}j\text{-}l\}_{\upsilon=1,2;\varpi=1,\ldots,\widehat{r}_{\max};j=1,2;l=1,2}$$

Experiment 0 ($\mathsf{Exp}_{\mathcal{B}}^{(0)}$): $\beta = 0$ case of Problem 1. For a probabilistic adversary $\mathcal{B}$, we define an experiment $\mathsf{Exp}_{\mathcal{B}}^{(0)}$ using Problem 1 generator $\mathcal{G}_{\beta}^{\mathsf{P1}}(1^{\lambda}, d, \widehat{r}_{\max})$ in Definition 4:

1. $\mathcal{B}$ is given $\varrho \xleftarrow{\$} \mathcal{G}_0^{\mathsf{P1}}(1^{\lambda}, d, \widehat{r}_{\max})$, i.e.,

$$\boldsymbol{e}_0 = (\omega, \boxed{0}, 0, 0, \varphi_0)_{\mathbb{B}_0}, \tag{26}$$

for $t = 1, \dots, d; i = 1, 2,$

$$\boldsymbol{e}_{t,i} = (\sigma_{t,i}(1, t), \omega \vec{e}_i, \boxed{0^2}, 0^2, \boxed{0^2}, 0^2, \vec{\varphi}_{t,i}, 0^2)_{\mathbb{B}}, \tag{27}$$

for $\upsilon = 1, 2; \varpi = 1, \dots, \widehat{r}_{\max}; i = 1, 2,$

$$\boldsymbol{e}_{d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega \vec{e}_i, \boxed{0^2}, 0^2, \boxed{0^2}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,i}, 0^2)_{\mathbb{B}}, \tag{28}$$

where all variables are generated as in Problem 1. Note that here we have dropped $\beta = 0$ suffix of $\boldsymbol{e}$'s for simplicity.

2. The output of the experiment is defined as $\beta' \leftarrow \mathcal{B}(1^{\lambda}, \varrho)$.

Experiment 1 ($\mathsf{Exp}_{\mathcal{B}}^{(1)}$): The same as Experiment 0 except that

$$\boldsymbol{e}_0 = (\omega, \boxed{\tau}, 0, 0, \varphi_0)_{\mathbb{B}_0}, \tag{29}$$

for $t = 1, \dots, d; i = 1, 2,$

$$\boldsymbol{e}_{t,i} = (\sigma_{t,i}(1, t), \omega \vec{e}_i, \boxed{\tau \vec{e}_i}, 0^2, 0^2, 0^2, \vec{\varphi}_{t,i}, 0^2)_{\mathbb{B}}, \tag{30}$$

for $\upsilon = 1, 2; \varpi = 1, \dots, \widehat{r}_{\max}; i = 1, 2,$

$$\boldsymbol{e}_{d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega \vec{e}_i, \boxed{\tau \vec{e}_i}, 0^2, 0^2, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,i}, 0^2)_{\mathbb{B}}, \tag{31}$$

where $\tau \xleftarrow{\$} \mathbb{F}_q^{\times}$ and all the other variables are generated as in Experiment 0.

Experiment 2-*p*-*j*-1 ($\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}j\text{-}1)}$ for $p = 1, \dots, d; j = 1, 2$): We denote Experiment 1 by Experiment 2-0-2-2. This experiment is similar to Experiment 2-($p-1$)-2-2 if $j = 1$, or Experiment 2-p-1-2 if $j = 2$ except that

$$\boldsymbol{e}_{p,j} = (\sigma_{p,j}(1, p), \omega \vec{e}_j, \tau \vec{e}_j, 0^2, \boxed{\widetilde{\sigma}_{p,j}(1, p)}, 0^2, \vec{\varphi}_{p,j}, 0^2)_{\mathbb{B}}, \tag{32}$$

where $\widetilde{\sigma}_{p,j} \xleftarrow{\$} \mathbb{F}_q$, and all the other variables are generated as in Experiment 2-($p-1$)-2-2 if $j = 1$, or Experiment 2-p-1-2 if $j = 2$.

Experiment 2-*p*-*j*-2 ($\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}j\text{-}2)}$ for $p = 1, \dots, d; j = 1, 2$): This experiment is equivalent to Experiment 2-p-j-1 with the exception that

$$\boldsymbol{e}_{p,j} = (\sigma_{p,j}(1, p), \omega \vec{e}_j, \tau \vec{e}_j, 0^2, \boxed{\tau(z_{p,j,1}, z_{p,j,2})}, 0^2, \vec{\varphi}_{p,j}, 0^2)_{\mathbb{B}}, \tag{33}$$

where $z_{p,j,1}, z_{p,j,2} \xleftarrow{\$} \mathbb{F}_q$, and all other variables are generated as in Experiment 2-p-j-1.

Experiment 3-($d+\upsilon$)-ϖ-*j*-1 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j\text{-}1)}$, for $\upsilon = 1, 2; \varpi = 1, \dots, \widehat{r}_{\max}; j = 1, 2$): We denote Experiment 2-d-2-2 by Experiment 3-($d+0$)-$\widehat{r}_{\max}$-2-2. This experiment is identical to Experiment 3-($d+\upsilon-1$)-$\widehat{r}_{\max}$-2-2 or Experiment 3-($d+\upsilon$)-($\varpi-1$)-2-2 if $j = 1$, or Experiment 3-($d+\upsilon$)-ϖ-1-2 if $j = 2$ except that

$$\boldsymbol{e}_{d+\upsilon,\varpi,j} = (\sigma_{d+\upsilon,\varpi,j}(1, d+\upsilon), \omega \vec{e}_j, \tau \vec{e}_j, 0^2, \boxed{\widetilde{\sigma}_{d+\upsilon,\varpi,j}(1, d+\upsilon)}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,j}, 0^2)_{\mathbb{B}}, \tag{34}$$

where $\widetilde{\sigma}_{d+\upsilon,\varpi,j} \xleftarrow{\$} \mathbb{F}_q$ and all the other variables are generated as in Experiment 3-($d+\upsilon-1$)-$\widehat{r}_{\max}$-2-2 or Experiment 3-($d+\upsilon$)-($\varpi-1$)-2-2 if $j = 1$, or Experiment 3-($d+\upsilon$)-ϖ-1-2 if $j = 2$.

Experiment 3-$(d+\upsilon)$-ϖ-j-2 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j\text{-}2)}$, for $\upsilon = 1,2; \varpi = 1,\ldots,\widehat{r}_{\max}; j = 1,2$): This experiment is the same as Experiment 3-$(d+\upsilon)$-ϖ-j-1 except that

$$\boldsymbol{e}_{d+\upsilon,\varpi,j} = (\sigma_{d+\upsilon,\varpi,j}(1,d+\upsilon), \omega\vec{e}_j, \tau\vec{e}_j, 0^2, \boxed{\tau(z_{d+\upsilon,\varpi,j,1}, z_{d+\upsilon,\varpi,j,2})}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,j}, 0^2)_{\mathbb{B}}, \tag{35}$$

where $z_{d+\upsilon,\varpi,j,1}, z_{d+\upsilon,\varpi,j,2} \xleftarrow{\$} \mathbb{F}_q$, and all the other variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-j-1.

Let $\boldsymbol{Z}_t = \begin{pmatrix} z_{t,1,1} & z_{t,1,2} \\ z_{t,2,1} & z_{t,2,2} \end{pmatrix}$ for $t = 1,\ldots,d$ and $\boldsymbol{Z}_{d+\upsilon,\varpi} = \begin{pmatrix} z_{d+\upsilon,\varpi,1,1} & z_{d+\upsilon,\varpi,1,2} \\ z_{d+\upsilon,\varpi,2,1} & z_{d+\upsilon,\varpi,2,2} \end{pmatrix}$ for $\upsilon = 1,2; \varpi = 1,\ldots,\widehat{r}_{\max}$. Then $\boldsymbol{e}_{t,j}, \boldsymbol{e}_{d+\upsilon,\varpi,j}$ in the final experiment (Experiment 3-$(d+2)$-$\widehat{r}_{\max}$-2-2) are expressed as

for $t = 1,\ldots,d; j = 1,2,$

$$\boldsymbol{e}_{t,j} = (\sigma_{t,j}(1,t), \omega\vec{e}_j, \tau\vec{e}_j, 0^2, \tau\vec{e}_j \cdot \boldsymbol{Z}_t, 0^2, \vec{\varphi}_{t,j}, 0^2)_{\mathbb{B}}, \tag{36}$$

for $\upsilon = 1,2; \varpi = 1,\ldots,\widehat{r}_{\max}; j = 1,2,$

$$\boldsymbol{e}_{d+\upsilon,\varpi,j} = (\sigma_{d+\upsilon,\varpi,j}(1,d+\upsilon), \omega\vec{e}_j, \tau\vec{e}_j, 0^2, \tau\vec{e}_j \cdot \boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,j}, 0^2)_{\mathbb{B}}, \tag{37}$$

where $\boldsymbol{Z}_t, \boldsymbol{Z}_{d+\upsilon,\varpi} \xleftarrow{\$} \mathsf{GL}(2,\mathbb{F}_q)$. Therefore, the distribution in Experiment 3-$(d+2)$-$\widehat{r}_{\max}$-2-2 and that in the $\beta = 1$ case of Problem 1 are equivalent except for the case that $\mathsf{det}(\boldsymbol{Z}_t) = 0$ for some t or $\mathsf{det}(\boldsymbol{Z}_{d+\upsilon,\varpi}) = 0$ for some υ, ϖ, i.e., except with probability $(d + 2\widehat{r}_{\max})/q$.

Lemmas: In the following we consider canonical (monomial) linear order in $\mathbb{N}^2$ and $\mathbb{N}^3$. For $(t_1,i_1),(t_2,i_2) \in \mathbb{N}^2$ and $(t_1,i_1,j_1),(t_2,i_2,j_2) \in \mathbb{N}^3$,

$$(t_1,i_1) < (t_2,i_2) \Leftrightarrow (t_1 < t_2) \text{ or } (t_1 = t_2 \text{ and } i_1 < i_2),$$
$$(t_1,i_1,j_1) < (t_2,i_2,j_2) \Leftrightarrow (t_1 < t_2) \text{ or } (t_1 = t_2 \text{ and } i_1 < i_2) \text{ or } (t_1 = t_2 \text{ and } i_1 = i_2 \text{ and } j_1 < j_2)$$

Lemma 13. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_1$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(0)}(\lambda) \to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(1)}(\lambda) \to 1\right]\right| \leq \mathsf{Adv}_{\mathcal{C}_1}^{\mathsf{BP1}}(\lambda).$$

Lemma 14. *For any probabilistic polynomial-time $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_2$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}(p-1)\text{-}2\text{-}2)}(\lambda) \to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}1\text{-}1)}(\lambda) \to 1\right]\right| \leq \mathsf{Adv}_{\mathcal{C}_{2\text{-}p\text{-}j}}^{\mathsf{BP1}}(\lambda)\ (j=1),$$
$$\text{or } \left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}1\text{-}2)}(\lambda) \to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}2\text{-}1)}(\lambda) \to 1\right]\right| \leq \mathsf{Adv}_{\mathcal{C}_{2\text{-}p\text{-}j}}^{\mathsf{BP1}}(\lambda)\ (j=2),$$

where $\mathcal{C}_{2\text{-}p\text{-}j}(\cdot) = \mathcal{C}_2(p,j,\cdot)$.

Lemma 15. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}j\text{-}1)}(\lambda) \to 1\right] = \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}j\text{-}2)}(\lambda) \to 1\right].$$

Lemmas 13, 14 and 15 can be proven similarly as Lemmas 45, 46 and 47 of [17], respectively.

Lemma 16. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_3$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left.\begin{array}{l} \left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon-1)\text{-}\widehat{r}_{\max}\text{-}2\text{-}2)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}1\text{-}1\text{-}1)}(\lambda)\to 1\right]\right|\leq \mathsf{Adv}^{\mathsf{BP1}}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}1\text{-}j}}(\lambda) \\ \text{for } 1<\varpi\leq\widehat{r}_{\max}, \\ \left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}(\varpi-1)\text{-}2\text{-}2)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1\text{-}1)}(\lambda)\to 1\right]\right|\leq \mathsf{Adv}^{\mathsf{BP1}}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j}}(\lambda) \end{array}\right\}(j=1)$$

or for $1\leq\varpi\leq\widehat{r}_{\max}$,

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1\text{-}2)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}2\text{-}1)}(\lambda)\to 1\right]\right|\leq \mathsf{Adv}^{\mathsf{BP1}}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j}}(\lambda)\ (j=2),$$

where $\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j}(\cdot)=\mathcal{C}_3(d+\upsilon,\varpi,j,\cdot)$.

Proof. Given integers d,υ,ϖ,j and a $\mathsf{BP1}$ instance $(\mathsf{param},\mathbb{B}_0,\widehat{\mathbb{B}}_0^*,\mathbb{B},\widehat{\mathbb{B}}^*,\boldsymbol{e}_{\beta,0},\{\boldsymbol{e}_{\beta,i}\}_{i=1,2})$, $\mathcal{C}_3$ sets new dual orthogonal bases $\mathbb{D}=\{\boldsymbol{d}_1,\ldots,\boldsymbol{d}_{16}\}=\{\boldsymbol{b}_3,\boldsymbol{b}_4,\boldsymbol{b}_1,\boldsymbol{b}_2,\boldsymbol{b}_9,\boldsymbol{b}_{10},\boldsymbol{b}_7,\boldsymbol{b}_8,\boldsymbol{b}_5,\boldsymbol{b}_6,\boldsymbol{b}_{11},\ldots,\boldsymbol{b}_{16}\}$ and $\mathbb{D}^*=\{\boldsymbol{d}_1^*,\ldots,\boldsymbol{d}_{16}^*\}=\{\boldsymbol{b}_3^*,\boldsymbol{b}_4^*,\boldsymbol{b}_1^*,\boldsymbol{b}_2^*,\boldsymbol{b}_9^*,\boldsymbol{b}_{10}^*,\boldsymbol{b}_7^*,\boldsymbol{b}_8^*,\boldsymbol{b}_5^*,\boldsymbol{b}_6^*,\boldsymbol{b}_{11}^*,\ldots,\boldsymbol{b}_{16}^*\}$.

$\mathcal{C}_3$ then sets $\widehat{\mathbb{B}}_0=\{\boldsymbol{b}_{0,1},\boldsymbol{b}_{0,3},\boldsymbol{b}_{0,5}\},\widehat{\mathbb{B}}_0^{*\prime}=\{\boldsymbol{b}_{0,1}^*,\boldsymbol{b}_{0,3}^*,\boldsymbol{b}_{0,4}^*\},\widehat{\mathbb{D}}=\{\boldsymbol{d}_1,\ldots,\boldsymbol{d}_4,\boldsymbol{d}_{13},\boldsymbol{d}_{14}\},\widehat{\mathbb{D}}^*=\{\boldsymbol{d}_1^*,\ldots,\boldsymbol{d}_4^*,\boldsymbol{d}_{11}^*,\boldsymbol{d}_{12}^*\}$. Note that $\mathcal{C}_3$ can compute $\{\widehat{\mathbb{D}},\widehat{\mathbb{D}}^*\}$ from $\{\mathbb{B},\widehat{\mathbb{B}}^*\}$ of the $\mathsf{BP1}$ instance.

$\mathcal{C}_3$ then compute $\boldsymbol{e}_{t,i}$ for $t=1,\ldots,d;i=1,2$ and $\boldsymbol{e}_{d+\upsilon',\varpi',i}$ for $(d+\upsilon',\varpi',i)<(d+\upsilon,\varpi,j)$ as in equations (36), (37) respectively and for $(d+\upsilon,\varpi,j)<(d+\upsilon',\varpi',i)$ as in equation (31), using $\mathbb{D}$ together with $\widetilde{\omega},\sigma_{t,i},z_{t,i,1},z_{t,i,2},\varphi_{t,i,1},\varphi_{t,i,2},\sigma_{d+\upsilon',\varpi',i},z_{d+\upsilon',\varpi',i,1},z_{d+\upsilon',\varpi',i,2},\varpi_{d+\upsilon',\varpi',i,1},$ $\varpi_{d+\upsilon',\varpi',i,2}\xleftarrow{\$}\mathbb{F}_q$, $\widetilde{\tau}\xleftarrow{\$}\mathbb{F}_q^\times$. Using $\widetilde{\omega},\widetilde{\tau}$, $\mathcal{C}_3$ computes

$$\begin{aligned}\boldsymbol{g}_0&=(\widetilde{\omega},\widetilde{\tau},0,0,\varphi_0)_{\mathbb{B}_0},\\ \boldsymbol{g}_{d+\upsilon,\varpi,j}&=\boldsymbol{e}_{\beta,1}+(d+\upsilon)\boldsymbol{e}_{\beta,2}+\widetilde{\omega}\boldsymbol{d}_{2+j}+\widetilde{\tau}\boldsymbol{d}_{4+j},\end{aligned}$$

where $\varphi_0\xleftarrow{\$}\mathbb{F}q$.

$\mathcal{C}_3$ then gives

$$\varrho=(\mathsf{param},\widehat{\mathbb{B}}_0,\widehat{\mathbb{B}}_0^{*\prime},\widehat{\mathbb{D}},\widehat{\mathbb{D}}^*,\boldsymbol{g}_0,\{\boldsymbol{e}_{t,i}\}_{t=1,\ldots,d;i=1,2},\{\boldsymbol{e}_{d+\upsilon',\varpi',i}\}_{(d+\upsilon',\varpi',i)\lesseqgtr(d+\upsilon,\varpi,j)},\boldsymbol{g}_{d+\upsilon,\varpi,j})$$

to $\mathcal{B}$, and outputs $\beta'\in\{0,1\}$ if $\mathcal{B}$ outputs β'.

When $j=1$, if $\beta=0$ (resp. $\beta=1$), the distribution of ϱ is exactly same as that of instances in Experiment 3-$(d+\upsilon-1)$-$\widehat{r}_{\max}$-2-2 or Experiment 3-$(d+\upsilon)$-$(\varpi-1)$-2-2 (resp. Experiment 3-$(d+\upsilon)$-1-1-1 or Experiment 3-$(d+\upsilon)$-ϖ-1-1). On the other hand, when $j=2$, if $\beta=0$ (resp. $\beta=1$), the distribution of ϱ is exactly same as that of instances in Experiment 3-$(d+\upsilon)$-ϖ-1-2 (resp. Experiment 3-$(d+\upsilon)$-ϖ-2-1). □

Lemma 17. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j\text{-}1)}(\lambda)\to 1\right]=\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}j\text{-}2)}(\lambda)\to 1\right].$$

Proof. We generate $\boldsymbol{Z}\xleftarrow{\$}\mathsf{GL}(2,\mathbb{F}_q)$, $\boldsymbol{U}=(\boldsymbol{Z}^{-1})^{\mathsf{T}}$, and set $\begin{pmatrix}\boldsymbol{d}_9\\ \boldsymbol{d}_{10}\end{pmatrix}=\boldsymbol{U}^{\mathsf{T}}\cdot\begin{pmatrix}\boldsymbol{b}_9\\ \boldsymbol{b}_{10}\end{pmatrix}$ and $\begin{pmatrix}\boldsymbol{d}_9^*\\ \boldsymbol{d}_{10}^*\end{pmatrix}=$ $\boldsymbol{Z}\cdot\begin{pmatrix}\boldsymbol{b}_9^*\\ \boldsymbol{b}_{10}^*\end{pmatrix}$.

We then define $\mathbb{D}=(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_8,\boldsymbol{d}_9,\boldsymbol{d}_{10},\boldsymbol{b}_{11},\ldots,\boldsymbol{b}_{16}\}$, $\mathbb{D}^*=\{\boldsymbol{b}_1^*,\ldots,\boldsymbol{b}_8^*,\boldsymbol{d}_9^*,\boldsymbol{d}_{10}^*,\boldsymbol{b}_{11}^*,\ldots,\boldsymbol{b}_{16}^*\}$. Note that $\{\mathbb{D},\mathbb{D}^*\}$ are consistent with $\{\widehat{\mathbb{B}},\widehat{\mathbb{B}}^*\}$. Since,

for $t=1,\ldots,d;i=1,2$,

$$\begin{aligned}\boldsymbol{e}_{t,i}&=(\sigma_{t,i}(1,t),\omega\vec{e}_i,\tau\vec{e}_i,0^2,\tau\vec{z}_{t,i},0^2,\vec{\varphi}_{t,i},0^2)_{\mathbb{B}}\\ &=(\sigma_{t,i}(1,t),\omega\vec{e}_i,\tau\vec{e}_i,0^2,\tau\vec{z}'_{t,i},0^2,\vec{\varphi}_{t,i},0^2)_{\mathbb{D}},\end{aligned}$$

for $(d+\upsilon',\varpi',i) < (d+\upsilon,\varpi,j)$,

$$\begin{aligned}\boldsymbol{e}_{d+\upsilon',\varpi',i} &= (\sigma_{d+\upsilon',\varpi',i}(1,d+\upsilon'),\omega\vec{e}_i,\tau\vec{e}_i,0^2,\tau\vec{z}_{d+\upsilon',\varpi',i},0^2,\vec{\varphi}_{d+\upsilon',\varpi',i},0^2)_{\mathbb{B}}\\ &= (\sigma_{d+\upsilon',\varpi',i}(1,d+\upsilon'),\omega\vec{e}_i,\tau\vec{e}_i,0^2,\tau\vec{z}'_{d+\upsilon',\varpi',i},0^2,\vec{\varphi}_{d+\upsilon',\varpi',i},0^2)_{\mathbb{D}},\end{aligned}$$

for $(d+\upsilon',\varpi',i) = (d+\upsilon,\varpi,j)$,

$$\begin{aligned}\boldsymbol{e}_{d+\upsilon,\varpi,j} &= (\sigma_{d+\upsilon,\varpi,j}(1,d+\upsilon),\omega\vec{e}_j,\tau\vec{e}_j,0^2,\widetilde{\sigma}_{d+\upsilon,\varpi,j}(1,d+\upsilon),0^2,\vec{\varphi}_{d+\upsilon,\varpi,j},0^2)_{\mathbb{B}}\\ &= (\sigma_{d+\upsilon,\varpi,j}(1,d+\upsilon),\omega\vec{e}_j,\tau\vec{e}_j,0^2,\tau\vec{z}_{d+\upsilon,\varpi,j},0^2,\vec{\varphi}_{d+\upsilon,\varpi,j},0^2)_{\mathbb{D}},\end{aligned}$$

for $(d+\upsilon,\varpi,j) < (d+\upsilon',\varpi',i)$,

$$\begin{aligned}\boldsymbol{e}_{d+\upsilon',\varpi',i} &= (\sigma_{d+\upsilon',\varpi',i}(1,d+\upsilon'),\omega\vec{e}_i,\tau\vec{e}_i,0^2,0^2,0^2,\vec{\varphi}_{d+\upsilon',\varpi',i},0^2)_{\mathbb{B}}\\ &= (\sigma_{d+\upsilon',\varpi',i}(1,d+\upsilon'),\omega\vec{e}_i,\tau\vec{e}_i,0^2,0^2,0^2,\vec{\varphi}_{d+\upsilon',\varpi',i},0^2)_{\mathbb{D}},\end{aligned}$$

where $\vec{z}_{d+\upsilon,\varpi,j} = \tau^{-1}\widetilde{\sigma}_{d+\upsilon,\varpi,j}(1,d+\upsilon)\cdot\boldsymbol{Z}$ and $\vec{z}_{t,i} = (z_{t,i,1},z_{t,i,2})$, $\vec{z}'_{t,i} = \vec{z}_{t,i}\cdot\boldsymbol{Z}$ for $t = 1,\dots,d; i = 1,2$ and $\vec{z}_{d+\upsilon',\varpi',i} = (z_{d+\upsilon',\varpi',i,1},z_{d+\upsilon',\varpi',i,2})$, $\vec{z}'_{d+\upsilon',\varpi',i} = \vec{z}_{d+\upsilon',\varpi',i}\cdot\boldsymbol{Z}$ for $(d+\upsilon',\varpi',i) < (d+\upsilon,\varpi,j)$.

Therefore $\vec{z}'_{t,i}$ for $t = 1,\dots,d; i = 1,2$, $\vec{z}'_{d+\upsilon',\varpi',i}$ for $(d+\upsilon',\varpi',i) < (d+\upsilon,\varpi,j)$ and $\vec{z}_{d+\upsilon,\varpi,j}$ are uniformly and independently distributed. Hence, the joint distribution for Experiment 3-$(d+\upsilon)$-ϖ-j-1 and that for Experiment 3-$(d+\upsilon)$-ϖ-j-2 are equivalent. □

C Proof of Lemma 2

The proof of Lemma 2 is based on Lemma 12 and the following results:

Definition 11 (Basic Problem 2). *Basic Problem 2 is to guess* $\beta \in \{0,1\}$, *given* $\varrho = (\mathsf{param},\widehat{\mathbb{B}}_0,\mathbb{B}_0^*,\widehat{\mathbb{B}},\mathbb{B}^*,\boldsymbol{h}^*_{\beta,0},\boldsymbol{e}_0,\{\boldsymbol{h}^*_{\beta,i},\boldsymbol{e}_i\}_{i=1,2}) \xleftarrow{\$} \mathcal{G}_\beta^{\mathsf{BP2}}(1^\lambda)$, *where*

$$\begin{aligned}&\mathcal{G}_\beta^{\mathsf{BP2}}(1^\lambda):\ (\mathsf{param},\{\mathbb{B}_0,\mathbb{B}_0^*\},\{\mathbb{B},\mathbb{B}^*\}) \xleftarrow{\$} \mathcal{G}_{\mathsf{ob}}(1^\lambda,(n_0=5,n=16)),\\
&\quad \widehat{\mathbb{B}}_0 = \{\boldsymbol{b}_{0,1},\boldsymbol{b}_{0,3},\dots,\boldsymbol{b}_{0,5}\},\ \widehat{\mathbb{B}} = \{\boldsymbol{b}_1,\dots,\boldsymbol{b}_6,\boldsymbol{b}_9,\dots,\boldsymbol{b}_{16}\},\\
&\quad \gamma,\omega,\eta_0 \xleftarrow{\$} \mathbb{F}_q,\ \delta,\tau \xleftarrow{\$} \mathbb{F}_q^\times,\\
&\quad \boldsymbol{h}^*_{0,0} = (\gamma,0,0,\eta_0,0)_{\mathbb{B}_0^*},\ \boldsymbol{h}^*_{1,0} = (\gamma,\delta,0,\eta_0,0)_{\mathbb{B}_0^*},\ \boldsymbol{e}_0 = (\omega,\tau,0,0,0)_{\mathbb{B}_0}\\
&\quad \vec{e}_1 = (1,0),\ \vec{e}_2 = (0,1) \in \mathbb{F}_q^2,\\
&\quad \textit{for } i = 1,2,\\
&\qquad \vec{\eta}_i \xleftarrow{\$} \mathbb{F}_q^2,\\
&\qquad \boldsymbol{h}^*_{0,1} = (0^2,\gamma\vec{e}_i,\quad 0^6,\quad \vec{\eta}_i,0^2,0^2)_{\mathbb{B}^*},\\
&\qquad \boldsymbol{h}^*_{1,i} = (0^2,\gamma\vec{e}_i,\delta\vec{e}_i,0^4,\vec{\eta}_i,0^2,0^2)_{\mathbb{B}^*},\\
&\qquad \boldsymbol{e}_i = (0^2,\omega\vec{e}_i,\tau\vec{e}_i,0^4,0^2,0^2,0^2)_{\mathbb{B}},\\
&\quad \textit{return } \varrho = (\mathsf{param},\widehat{\mathbb{B}}_0,\mathbb{B}_0^*,\widehat{\mathbb{B}},\mathbb{B}^*,\boldsymbol{h}^*_{\beta,0},\boldsymbol{e}_0,\{\boldsymbol{h}^*_{\beta,i},\boldsymbol{e}_i\}_{i=1,2}).\end{aligned}$$

For a probabilistic adversary $\mathcal{C}$, *the advantage of* $\mathcal{C}$ *for Basic Problem 2,* $\mathsf{Adv}_{\mathcal{C}}^{\mathsf{BP2}}(\lambda)$, *is defined similarly as in Definition 10.*

Lemma 18. *For any probabilistic polynomial-time 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,

$$\mathsf{Adv}_{\mathcal{C}}^{\mathsf{BP2}}(\lambda) \le \mathsf{Adv}_{\mathcal{F}}^{\mathsf{DLIN}}(\lambda) + 5/q.$$

Definition 12 (Basic Problem 3-p for $p = 1,\dots,d+2$). *Basic Problem* 3-p *is to guess* $\beta \in \{0,1\}$, *given* $\varrho = (\mathsf{param},\mathbb{B}_0,\mathbb{B}_0^*,\widehat{\mathbb{B}},\mathbb{B}^*,\boldsymbol{e}_0,\{\boldsymbol{h}^*_{\beta,p,i},\boldsymbol{e}_i,\boldsymbol{g}_i\}_{i=1,2}) \xleftarrow{\$} \mathcal{G}_\beta^{\mathsf{BP3}\text{-}p}(1^\lambda,d+2)$, *where*

$$\begin{array}{l}
\mathcal{G}_\beta^{\mathsf{BP3}\text{-}p}(1^\lambda, d+2):\ (\mathsf{param},\{\mathbb{B}_0,\mathbb{B}_0^*\},\{\mathbb{B},\mathbb{B}^*\}) \xleftarrow{\$} \mathcal{G}_{\mathsf{ob}}(1^\lambda,(n_0=5,n=16)),\\
\quad \widehat{\mathbb{B}}=\{\boldsymbol{b}_1,\dots,\boldsymbol{b}_6,\boldsymbol{b}_{11},\dots,\boldsymbol{b}_{16}\},\ \tau\xleftarrow{\$}\mathbb{F}_q^\times,\ \boldsymbol{e}_0=(0,\tau,0,0,0)_{\mathbb{B}_0},\\
\quad \vec{e}_1=(1,0),\ \vec{e}_2=(0,1)\in\mathbb{F}_q^2,\\
\quad \textit{for } i=1,2,\\
\qquad \vec{\eta}_{p,i}\xleftarrow{\$}\mathbb{F}_q^2,\ \mu_{p,i},\theta_{p,i}\xleftarrow{\$}\mathbb{F}_q,\\
\qquad \begin{array}{rl}
\boldsymbol{h}^*_{0,p,i}= & (\ \mu_{p,i}(p,-1),0^2,\quad 0^6,\quad \vec{\eta}_{p,i},0^2,0^2)_{\mathbb{B}^*},\\
\boldsymbol{h}^*_{1,p,i}= & (\ \mu_{p,i}(p,-1),0^2,-\theta_{p,i}\vec{e}_i,\theta_{p,i}\vec{e}_i,0^2,\ \vec{\eta}_{p,i},0^2,0^2)_{\mathbb{B}^*},\\
\boldsymbol{e}_i= & (\quad 0^4,\quad \tau\vec{e}_i,\tau\vec{e}_i,0^2,\quad 0^2,\ 0^2,0^2)_{\mathbb{B}},\\
\boldsymbol{g}_i= & (\quad 0^4,\quad 0^4,\quad \tau\vec{e}_i,\quad 0^2,\ 0^2,0^2)_{\mathbb{B}},
\end{array}\\
\quad \textit{return } \varrho=(\mathsf{param},\mathbb{B}_0,\mathbb{B}_0^*,\widehat{\mathbb{B}},\mathbb{B}^*,\boldsymbol{e}_0,\{\boldsymbol{h}^*_{\beta,p,i},\boldsymbol{e}_i,\boldsymbol{g}_i\}_{i=1,2}).
\end{array}$$

For a probabilistic adversary $\mathcal{C}$, the advantage of $\mathcal{C}$ for Basic Problem 3-p*,* $\mathsf{Adv}_{\mathcal{C}_p}^{\mathsf{BP3}\text{-}p}(\lambda)$*, is defined in a similar fashion as in Definition 10, where* $\mathcal{C}_p(\cdot)=\mathcal{C}(p,\cdot)$*.*

Lemma 19. *For any Probabilistic polynomial-time 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,*

$$\mathsf{Adv}_{\mathcal{C}_p}^{\mathsf{BP3}\text{-}p}(\lambda)\le\sum_{j=1}^{2}\mathsf{Adv}_{\mathcal{F}_{p\text{-}j}}^{\mathsf{DLIN}}(\lambda)+10/q,$$

where $\mathcal{F}_{p\text{-}j}(\cdot)=\mathcal{F}(p,j,\cdot)$*.*

Definition 13 (Basic Problem 4-p for $p=1,\dots,d+2$). *Basic Problem* 4-p *is to guess* $\beta\in\{0,1\}$*, given* $\varrho=(\mathsf{param},\mathbb{B}_0,\mathbb{B}_0^*,\widehat{\mathbb{B}},\mathbb{B}^*,\{\boldsymbol{h}^*_{\beta,p,i}\}_{i=1,2})\xleftarrow{\$}\mathcal{G}_\beta^{\mathsf{BP4}\text{-}p}(1^\lambda,d+2)$*, where*

$$\begin{array}{l}
\mathcal{G}_\beta^{\mathsf{BP4}\text{-}p}(1^\lambda, d+2):\ (\mathsf{param},\{\mathbb{B}_0,\mathbb{B}_0^*\},\{\mathbb{B},\mathbb{B}^*\}) \xleftarrow{\$} \mathcal{G}_{\mathsf{ob}}(1^\lambda,(n_0=5,n=16)),\\
\quad \widehat{\mathbb{B}}=\{\boldsymbol{b}_1,\dots,\boldsymbol{b}_6,\boldsymbol{b}_9,\dots,\boldsymbol{b}_{16}\},\\
\quad \vec{e}_1=(1,0),\ \vec{e}_2=(0,1)\in\mathbb{F}_q^2,\\
\quad \textit{for } i=1,2,\\
\qquad \vec{\eta}_{p,i}\xleftarrow{\$}\mathbb{F}_q^2,\ \mu_{p,i},\theta_{p,i}\xleftarrow{\$}\mathbb{F}_q,\\
\qquad \begin{array}{rl}
\boldsymbol{h}^*_{0,p,i}= & (\mu_{p,i}(p,-1),0^2,\quad 0^6,\quad \vec{\eta}_{p,i},0^2,0^2)_{\mathbb{B}^*},\\
\boldsymbol{h}^*_{1,p,i}= & (\mu_{p,i}(p,-1),0^2,0^2,\theta_{p,i}\vec{e}_i,0^2,\ \vec{\eta}_{p,i},0^2,0^2)_{\mathbb{B}^*},
\end{array}\\
\quad \textit{return } \varrho=(\mathsf{param},\mathbb{B}_0,\mathbb{B}_0^*,\widehat{\mathbb{B}},\mathbb{B}^*,\{\boldsymbol{h}^*_{\beta,p,i}\}_{i=1,2}).
\end{array}$$

For any probabilistic adversary $\mathcal{C}$, the advantage of $\mathcal{C}$ for Basic Problem 4-p*,* $\mathsf{Adv}_{\mathcal{C}_p}^{\mathsf{BP4}\text{-}p}(\lambda)$*, is analogously defined as in Definition 10, where* $\mathcal{C}_p(\cdot)=\mathcal{C}(p,\cdot)$*.*

Lemma 20. *For any probabilistic polynomial-time 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,*

$$\mathsf{Adv}_{\mathcal{C}_p}^{\mathsf{BP4}\text{-}p}(\lambda)\le\sum_{j=1}^{2}\mathsf{Adv}_{\mathcal{F}_{p\text{-}j}}^{\mathsf{DLIN}}(\lambda)+10/q,$$

where $\mathcal{F}_{p\text{-}j}(\cdot)=\mathcal{F}(p,j,\cdot)$*.*

Definition 14 (Basic Problem 5-p for $p=1,\dots,d+2$). *Basic Problem* 5-p *is to guess* $\beta\in\{0,1\}$*, given* $\varrho=(\mathsf{param},\mathbb{B}_0,\mathbb{B}_0^*,\mathbb{B},\widehat{\mathbb{B}}^*,\boldsymbol{h}_0^*,\{\boldsymbol{h}^*_{p,i},\boldsymbol{e}_{\beta,l,i}\}_{l=1,\dots,p-1,p+1,\dots,d+2;i=1,2},\{\widetilde{\boldsymbol{h}}^*_j\}_{j=5,6,9,10})\xleftarrow{\$}\mathcal{G}_\beta^{\mathsf{BP5}\text{-}p}(1^\lambda,d+2)$*, where*

$$\begin{array}{l}
\mathcal{G}_\beta^{\mathsf{BP5}\text{-}p}(1^\lambda, d+2):\ (\mathsf{param},\{\mathbb{B}_0,\mathbb{B}_0^*\},\{\mathbb{B},\mathbb{B}^*\}) \xleftarrow{\$} \mathcal{G}_{\mathsf{ob}}(1^\lambda,(n_0=5,n=16)),\\
\quad \widehat{\mathbb{B}}^*=\{\boldsymbol{b}^*_1,\dots,\boldsymbol{b}^*_6,\boldsymbol{b}^*_9,\dots,\boldsymbol{b}^*_{16}\},\\
\quad \vec{e}_1=(1,0),\ \vec{e}_2=(0,1)\in\mathbb{F}_q^2,
\end{array}$$

```
$\delta \xleftarrow{\$} \mathbb{F}_q^{\times},\ \boldsymbol{h}_0^* = \delta \boldsymbol{b}_{0,2}^*,\ \widetilde{\boldsymbol{h}}_j^* = \delta \boldsymbol{b}_j^*,$ for $j = 5, 6, 9, 10,$
for $i = 1, 2,$
    $\vec{\eta}_{p,i} \xleftarrow{\$} \mathbb{F}_q^2,\ \mu_{p,i} \xleftarrow{\$} \mathbb{F}_q,$
    $\boldsymbol{h}_{p,i}^* = (\mu_{p,i}(p, -1), 0^2, 0^2, \delta \vec{e}_i, 0^2, \vec{\eta}_{p,i}, 0^2, 0^2)_{\mathbb{B}^*}$
    for $l = 1, \ldots, p-1, p+1, \ldots, d+2,$
        $\vec{\chi}_{l,i}, \vec{\varphi}_{l,i} \xleftarrow{\$} \mathbb{F}_q^2,\ \sigma_{l,i} \xleftarrow{\$} \mathbb{F}_q,$
        $\boldsymbol{e}_{0,l,i} = (\sigma_{l,i}(1, l), 0^2, \quad 0^6, \quad 0^2, \vec{\varphi}_{l,i}, 0^2)_{\mathbb{B}},$
        $\boldsymbol{e}_{1,l,i} = (\sigma_{l,i}(1, l), 0^2, 0^2, \vec{\chi}_{l,i}, 0^2, 0^2, \vec{\varphi}_{l,i}, 0^2)_{\mathbb{B}},$
return $\varrho = (\mathsf{param}, \mathbb{B}_0, \mathbb{B}_0^*, \mathbb{B}, \widehat{\mathbb{B}}^*, \boldsymbol{h}_0^*, \{\boldsymbol{h}_{p,i}^*, \boldsymbol{e}_{\beta,l,i}\}_{l=1,\ldots,p-1,p+1,\ldots,d+2; i=1,2}, \{\widetilde{\boldsymbol{h}}_j^*\}_{j=5,6,9,10}).$
```

For a probabilistic adversary $\mathcal{C}$, the advantage of $\mathcal{C}$ for Basic Problem 5-p, $\mathsf{Adv}_{\mathcal{C}_p}^{\mathsf{BP5}\text{-}p}(\lambda)$, *is similarly defined as in Definition 10.*

Lemma 21. *For any probabilistic polynomial-time 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,*

$$\mathsf{Adv}_{\mathcal{C}_p}^{\mathsf{BP5}\text{-}p}(\lambda) \leq \sum_{j=1}^{2} \sum_{\substack{l=1 \\ l \neq p}}^{d+2} \mathsf{Adv}_{\mathcal{F}_{p\text{-}j\text{-}l}}^{\mathsf{DLIN}}(\lambda) + 5(d+1)/q,$$

where $\mathcal{F}_{p\text{-}j\text{-}l}(\cdot) = \mathcal{F}(p, j, l, \cdot)$.

Basic Problems 2–5-p are essentially Basic Problems 2–5-p defined in [17] with two additional dimensions. The proofs of Lemmas 18–21 are similar to Lemmas 35, 36, 38 and 39 of [17] respectively.

In order to prove Lemma 2, we consider the following experiments. Problem 2 is the hybrid of the following experiments performed with a probabilistic adversary $\mathcal{B}$:

$$\begin{aligned} &\mathsf{Exp}_{\mathcal{B}}^{(0)} \implies \mathsf{Exp}_{\mathcal{B}}^{(1)} \implies \{\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}1)} \implies \ldots \implies \mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}8)}\}_{p=1,\ldots,d} \\ &\implies \{\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1)} \implies \ldots \implies \mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}7)} \implies \{\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}j\text{-}1)} \\ &\implies \mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}j\text{-}2)}\}_{\iota=1,\ldots,\varpi-1,\varpi+1,\ldots,\aleph; j=1,2} \implies \mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}9)}\}_{\upsilon=1,2; \varpi=1,\ldots,\aleph} \end{aligned}$$

Thus, $\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda) = \left|\mathsf{Pr}\left[\mathsf{Exp}^{(0)}_{\mathcal{B}} \to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}^{(3\text{-}(d+2)\text{-}\aleph\text{-}9)}_{\mathcal{B}} \to 1\right]\right|$. Therefore, from Lemmas 12, 18–21 and the following lemmas we obtain

$$
\begin{aligned}
&\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda) = \left|\mathsf{Pr}\left[\mathsf{Exp}^{(0)}_{\mathcal{B}}(\lambda) \to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}^{(3\text{-}(d+2)\text{-}\aleph\text{-}9)}_{\mathcal{B}}(\lambda) \to 1\right]\right| \\
&\le \mathsf{Adv}^{\mathsf{BP2}}_{\mathcal{C}_1}(\lambda) + \sum_{p=1}^{d}\left[\mathsf{Adv}^{\mathsf{BP3}\text{-}p}_{\mathcal{C}_{2\text{-}p\text{-}1}}(\lambda) + \mathsf{Adv}^{\mathsf{BP3}\text{-}p}_{\mathcal{C}_{2\text{-}p\text{-}2}}(\lambda) + \mathsf{Adv}^{\mathsf{BP5}\text{-}p}_{\mathcal{C}_{2\text{-}p\text{-}3}}(\lambda) + \mathsf{Adv}^{\mathsf{BP5}\text{-}p}_{\mathcal{C}_{2\text{-}p\text{-}4}}(\lambda) + \mathsf{Adv}^{\mathsf{BP4}\text{-}p}_{\mathcal{C}_{2\text{-}p\text{-}5}}(\lambda)\right] + \\
&\quad \sum_{\upsilon=1}^{2}\sum_{\varpi=1}^{\aleph}\left[\mathsf{Adv}^{\mathsf{BP3}\text{-}(d+\upsilon)}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1}}(\lambda) + \mathsf{Adv}^{\mathsf{BP3}\text{-}(d+\upsilon)}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}2}}(\lambda) + \mathsf{Adv}^{\mathsf{BP5}\text{-}(d+\upsilon)}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}3}}(\lambda) + \mathsf{Adv}^{\mathsf{BP5}\text{-}(d+\upsilon)}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}4}}(\lambda) + \right. \\
&\quad \left. \sum_{\substack{\iota=1\\ \iota\neq\varpi}}^{\aleph}\sum_{j=1}^{2}\mathsf{Adv}^{\mathsf{BP1}}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}5\text{-}\iota\text{-}j}}(\lambda) + \mathsf{Adv}^{\mathsf{BP4}\text{-}(d+\upsilon)}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}6}}(\lambda)\right] \\
&\le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_1}(\lambda) + \sum_{j=1}^{2}\left[\sum_{p=1}^{d}\left\{\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}1\text{-}j}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}2\text{-}j}}(\lambda) + \sum_{\substack{l=1\\ l\neq p}}^{d+2}\left(\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}3\text{-}j\text{-}l}}(\lambda) + \right.\right.\right. \\
&\quad \left.\left. \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}4\text{-}j\text{-}l}}(\lambda)\right) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}p\text{-}5\text{-}j}}(\lambda)\right\} + \sum_{\upsilon=1}^{2}\sum_{\varpi=1}^{\aleph}\left\{\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}6\text{-}j}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}7\text{-}j}}(\lambda) + \right. \\
&\quad \sum_{\substack{l=1\\ l\neq d+\upsilon}}^{d+2}\left(\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}j\text{-}l}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}9\text{-}j\text{-}l}}(\lambda)\right) + \\
&\quad \left.\left. \sum_{\substack{\iota=1\\ \iota\neq\varpi}}^{\aleph}\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}10\text{-}j\text{-}\iota}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_{2\text{-}(d+\upsilon)\text{-}\varpi\text{-}11\text{-}j}}(\lambda)\right\}\right] + \left[5 + 40d + 10d^2 + 2\aleph(30 + 10d + 10\aleph)\right]/q,
\end{aligned}
$$

and thus we get Lemma 2.

<u>**Experiments**</u>: In Experiment 0, a part framed by a box indicates positions of coefficients to be changed in subsequent experiments. In the other experiments a part framed by a box indicates coefficients which were changed in a transition from the previous experiment. Experiments proceed as follows:

Experiment 0 ($\mathsf{Exp}^{(0)}_{\mathcal{B}}$): $\beta = 0$ case of Problem 2. For a probabilistic adversary $\mathcal{B}$, we define an experiment $\mathsf{Exp}^{(0)}_{\mathcal{B}}$ using Problem 2 generator $\mathcal{G}^{\mathsf{P2}}_{\beta}(1^{\lambda}, d, N_{\max}, \widehat{r}_{\max})$ in Definition 5 as follows:

1. $\mathcal{B}$ is given $\varrho \xleftarrow{\$} \mathcal{G}^{\mathsf{P2}}_{0}(1^{\lambda}, d, N_{\max}, \widehat{r}_{\max})$, i.e.,

$$\boldsymbol{h}^*_0 = (\gamma, \boxed{0}, 0, \eta_0, 0)_{\mathbb{B}^*_0}, \tag{38}$$

$$\boldsymbol{e}_0 = (\omega, \tau, 0, 0, \varphi_0)_{\mathbb{B}_0}, \tag{39}$$

for $t = 1, \ldots, d; i = 1, 2,$

$$\boldsymbol{h}^*_{t,i} = (\mu_{t,i}(t, -1), \gamma\vec{e}_i, \boxed{0^6}, \vec{\eta}_{t,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{40}$$

$$\boldsymbol{e}_{t,i} = (\sigma_{t,i}(1, t), \omega\vec{e}_i, \tau\vec{e}_i, \boxed{0^2}, \tau\vec{e}_i\boldsymbol{Z}_t, 0^2, \vec{\varphi}_{t,i}, 0^2)_{\mathbb{B}}, \tag{41}$$

for $\upsilon = 1, 2; \varpi = 1, \ldots, \aleph; i = 1, 2,$

$$\boldsymbol{h}^*_{d+\upsilon,\varpi,i} = (\mu_{d+\upsilon,\varpi,i}(d + \upsilon, -1), \gamma\vec{e}_i, \boxed{0^6}, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{42}$$

$$\boldsymbol{e}_{d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d + \upsilon), \omega\vec{e}_i, \tau\vec{e}_i, \boxed{0^2}, \tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,i}, 0^2)_{\mathbb{B}}. \tag{43}$$

2. The output of the experiment is defined as $\beta' \xleftarrow{\$} \mathcal{B}(1^{\lambda}, \varrho)$.

Experiment 1 ($\mathsf{Exp}^{(1)}_{\mathcal{B}}$): The same as Experiment 0, except that

$$
\begin{aligned}
&\boldsymbol{h}^*_0 = (\gamma, \boxed{\delta}, 0, \eta_0, 0)_{\mathbb{B}^*_0}, && (44)\\
&\text{for } t = 1, \ldots, d; i = 1, 2,\\
&\boldsymbol{h}^*_{t,i} = (\mu_{t,i}(t, -1), \gamma \vec{e}_i, \boxed{\delta \vec{e}_i}, 0^4, \vec{\eta}_{t,i}, 0^2, 0^2)_{\mathbb{B}^*}, && (45)\\
&\text{for } \upsilon = 1, 2; \varpi = 1, \ldots, \aleph; i = 1, 2,\\
&\boldsymbol{h}^*_{d+\upsilon,\varpi,i} = (\mu_{d+\upsilon,\varpi,i}(d+\upsilon, -1), \gamma \vec{e}_i, \boxed{\delta \vec{e}_i}, 0^4, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{B}^*}, && (46)
\end{aligned}
$$

where $\delta \xleftarrow{\$} \mathbb{F}_q^\times$ and all other variables are generated as in Experiment 0.

Experiment 2-p-1 ($\mathsf{Exp}^{(2\text{-}p\text{-}1)}_{\mathcal{B}}$, for $p = 1, \ldots, d$): We denote Experiment 1 as Experiment 2-0-8. This experiment is similar to Experiment 2-($p-1$)-8 except that

$$
\begin{aligned}
&\text{for } t = 1, \ldots, d; i = 1, 2,\\
&\boldsymbol{e}_{t,i} = (\sigma_{t,i}(1, t), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{\tau \vec{e}_i}, \tau \vec{e}_i \boldsymbol{Z}_t, 0^2, \vec{\varphi}_{t,i}, 0^2)_{\mathbb{B}}, && (47)\\
&\text{for } \upsilon = 1, 2; \varpi = 1, \ldots, \aleph; i = 1, 2,\\
&\boldsymbol{e}_{d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{\tau \vec{e}_i}, \tau \vec{e}_i \boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,i}, 0^2)_{\mathbb{B}}, && (48)
\end{aligned}
$$

where all the variables are generated as in Experiment 2-($p-1$)-8.

Experiment 2-p-2 ($\mathsf{Exp}^{(2\text{-}p\text{-}2)}_{\mathcal{B}}$ for $p = 1, \ldots, d$): Equivalent to Experiment 2-p-1 with the only exception that for $i = 1, 2$,

$$
\boldsymbol{h}^*_{p,i} = (\mu_{p,i}(p, -1), \gamma \vec{e}_i, \boxed{(\delta - \theta_{p,i}) \vec{e}_i, \theta_{p,i} \vec{e}_i}, 0^2, \vec{\eta}_{p,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{49}
$$

where $\theta_{p,i} \xleftarrow{\$} \mathbb{F}_q$, and all the other variables are generated as in Experiment 2-p-1.

Experiment 2-p-3 ($\mathsf{Exp}^{(2\text{-}p\text{-}3)}_{\mathcal{B}}$ for $p = 1, \ldots, d$): The same as Experiment as Experiment 2-p-2 with the difference that for $i = 1, 2$,

$$
\boldsymbol{h}^*_{p,i} = (\mu_{p,i}(p, -1), \gamma \vec{e}_i, \boxed{\theta_{p,i} \vec{e}_i, (\delta - \theta_{p,i}) \vec{e}_i}, 0^2, \vec{\eta}_{p,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{50}
$$

where all the variables are generated as in Experiment 2-p-2.

Experiment 2-p-4 ($\mathsf{Exp}^{(2\text{-}p\text{-}4)}_{\mathcal{B}}$ for $p = 1, \ldots, d$): Analogous to Experiment 2-p-3 except that for $i = 1, 2$,

$$
\boldsymbol{h}^*_{p,i} = (\mu_{p,i}(p, -1), \gamma \vec{e}_i, \boxed{0^2, \delta \vec{e}_i}, 0^2, \vec{\eta}_{p,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{51}
$$

where all the variables are generated as in Experiment 2-p-3.

Experiment 2-p-5 ($\mathsf{Exp}^{(2\text{-}p\text{-}5)}_{\mathcal{B}}$ for $p = 1, \ldots, d$): The same as Experiment 2-p-4 except that

$$
\begin{aligned}
&\text{for } l = 1, \ldots, p-1, p+1, \ldots, d; i = 1, 2,\\
&\boldsymbol{e}_{l,i} = (\sigma_{l,i}(1, l), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{\vec{\chi}_{l,i}}, \tau \vec{e}_i \boldsymbol{Z}_l, 0^2, \vec{\varphi}_{l,i}, 0^2)_{\mathbb{B}} && (52)\\
&\text{for } \upsilon = 1, 2; \varpi = 1, \ldots, \aleph; i = 1, 2,\\
&\boldsymbol{e}_{d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{\vec{\chi}_{d+\upsilon,\varpi,i}}, \tau \vec{e}_i \boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,i}, 0^2)_{\mathbb{B}}, && (53)
\end{aligned}
$$

where $\vec{\chi}_{l,i}, \vec{\chi}_{d+\upsilon,\varpi,i} \xleftarrow{\$} \mathbb{F}_q^2$ and all the other variables are generated as in Experiment 2-p-4.

Experiment 2-*p*-6 ($\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}6)}$ for $p = 1, \ldots, d$): Equivalent to Experiment 2-p-5 except that

$$\text{for } i = 1, 2,$$
$$\boldsymbol{h}^*_{p,i} = (\mu_{p,i}(p, -1), \gamma \vec{e}_i, 0^2, \boxed{\xi \vec{e}_i, \delta \vec{e}_i \boldsymbol{U}_p}, \vec{\eta}_{p,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{54}$$
$$\boldsymbol{e}_{p,i} = (\sigma_{p,i}(1, p), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{0^2}, \tau \vec{e}_i \boldsymbol{Z}_p, 0^2, \vec{\varphi}_{p,i}, 0^2)_{\mathbb{B}}, \tag{55}$$

where $\xi \xleftarrow{\$} \mathbb{F}_q$, $\boldsymbol{Z}_p \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$, $\boldsymbol{U}_p = (\boldsymbol{Z}_p^{-1})^{\mathsf{T}}$, and all other variables are generated as in Experiment 2-p-5.

Experiment 2-*p*-7 ($\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}7)}$ for $p = 1, \ldots, d$): The same as Experiment 2-p-6 with the exception that

$$\text{for } l = 1, \ldots, p-1, p+1, \ldots, d; i = 1, 2,$$
$$\boldsymbol{e}_{l,i} = (\sigma_{l,i}(1, l), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{0^2}, \tau \vec{e}_i \boldsymbol{Z}_l, 0^2, \vec{\varphi}_{l,i}, 0^2)_{\mathbb{B}}, \tag{56}$$
$$\text{for } \upsilon = 1, 2; \varpi = 1, \ldots, \aleph; i = 1, 2,$$
$$\boldsymbol{e}_{d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{0^2}, \tau \vec{e}_i \boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,i}, 0^2)_{\mathbb{B}}, \tag{57}$$

where all the variables are generated as in Experiment 2-p-6.

Experiment 2-*p*-8 (for $p = 1, \ldots, d$): The same as Experiment 2-p-7 except that for $i = 1, 2$,

$$\boldsymbol{h}^*_{p,i} = (\mu_{p,i}(p, -1), \gamma \vec{e}_i, 0^2, \boxed{0^2}, \delta \vec{e}_i \boldsymbol{U}_p, \vec{\eta}_{p,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{58}$$

where all the variables are generated as in Experiment 2-p-7.

Experiment 3-$(d+\upsilon)$-ϖ-1 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1)}$ for $\upsilon = 1, 2; \varpi = 1, \ldots, \aleph$): We denote Experiment 2-d-8 as Experiment 3-$(d+0)$-$\aleph$-9. This experiment is identical to Experiment 3-$(d+\upsilon-1)$-$\aleph$-9, if $\varpi = 1$, or Experiment 3-$(d+\upsilon)$-$(\varpi-1)$-9, if $1 < \varpi \leq \aleph$, except that

$$\text{for } t = 1, \ldots, d; i = 1, 2,$$
$$\boldsymbol{e}_{t,i} = (\sigma_{t,i}(1, t), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{\tau \vec{e}_i}, \tau \vec{e}_i \boldsymbol{Z}_t, 0^2, \vec{\varphi}_{t,i}, 0^2)_{\mathbb{B}}, \tag{59}$$
$$\text{for } \upsilon = 1, 2; \varpi = 1, \ldots, \aleph; i = 1, 2,$$
$$\boldsymbol{e}_{d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1, d+\upsilon), \omega \vec{e}_i, \tau \vec{e}_i, \boxed{\tau \vec{e}_i}, \tau \vec{e}_i \boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,i}, 0^2)_{\mathbb{B}}, \tag{60}$$

where all the variables are generated as in Experiment 3-$(d+\upsilon-1)$-$\aleph$-9, if $\varpi = 1$, or Experiment 3-$(d+\upsilon)$-$(\varpi-1)$-9, if $1 < \varpi \leq \aleph$.

Experiment 3-$(d+\upsilon)$-ϖ-2 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}2)}$ for $\upsilon = 1, 2; \varpi = 1, \ldots, \aleph$): The same as Experiment 3-$(d+\upsilon)$-ϖ-1 except that for $i = 1, 2$,

$$\boldsymbol{h}^*_{d+\upsilon,\varpi,i} = (\mu_{d+\upsilon,\varpi,i}(d+\upsilon, -1), \gamma \vec{e}_i, \boxed{(\delta - \theta_{d+\upsilon,\varpi,i}) \vec{e}_i, \theta_{d+\upsilon,\varpi,i} \vec{e}_i}, 0^2, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{61}$$

where $\theta_{d+\upsilon,\varpi,i} \xleftarrow{\$} \mathbb{F}_q$, and all the other variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-1.

Experiment 3-$(d+\upsilon)$-ϖ-3 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}3)}$ for $\upsilon = 1, 2; \varpi = 1, \ldots, \aleph$): Equivalent to Experiment 3-$(d+\upsilon)$-ϖ-2 with the exception that for $i = 1, 2$,

$$\boldsymbol{h}^*_{d+\upsilon,\varpi,i} = (\mu_{d+\upsilon,\varpi,i}(d+\upsilon, -1), \gamma \vec{e}_i, \boxed{\theta_{d+\upsilon,\varpi,i} \vec{e}_i, (\delta - \theta_{d+\upsilon,\varpi,i}) \vec{e}_i}, 0^2, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{62}$$

where all the variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-2.

Experiment 3-$(\boldsymbol{d+\upsilon})$-$\boldsymbol{\varpi}$-4 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}4)}$ for $\upsilon = 1,2; \varpi = 1,\ldots,\aleph$): The same as Experiment 3-$(d+\upsilon)$-ϖ-3 with the only exception that for $i = 1,2$,

$$\boldsymbol{h}^*_{d+\upsilon,\varpi,i} = (\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1), \gamma\vec{e}_i, \boxed{0^2, \delta\vec{e}_i}, 0^2, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{B}^*}, \tag{63}$$

where all the variables are generated as in Experiment 2-$(d+\upsilon)$-ϖ-3.

Experiment 3-$(\boldsymbol{d+\upsilon})$-$\boldsymbol{\varpi}$-5 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}5)}$ for $\upsilon = 1,2; \varpi = 1,\ldots,\aleph$): The same as Experiment 3-$(d+\upsilon)$-ϖ-4 except that

$$\begin{aligned}
&\text{for } l = 1,\ldots,d; i = 1,2,\\
&\boldsymbol{e}_{l,i} = (\sigma_{l,i}(1,l), \omega\vec{e}_i, \tau\vec{e}_i, \boxed{\vec{\chi}_{l,i}}, \tau\vec{e}_i\boldsymbol{Z}_l, 0^2, \vec{\varphi}_{l,i}, 0^2)_{\mathbb{B}}, && (64)\\
&\text{for } k = 1,\ldots,\upsilon-1,\upsilon+1,\ldots,2; \iota = 1,\ldots,\aleph; i = 1,2,\\
&\boldsymbol{e}_{d+k,\iota,i} = (\sigma_{d+k,\iota,i}(1,d+k), \omega\vec{e}_i, \tau\vec{e}_i, \boxed{\vec{\chi}_{d+k,\iota,i}}, \tau\vec{e}_i\boldsymbol{Z}_{d+k,\iota}, 0^2, \vec{\varphi}_{d+k,\iota,i}, 0^2)_{\mathbb{B}}, && (65)
\end{aligned}$$

where $\vec{\chi}_{l,i}, \vec{\chi}_{d+k,\iota,i} \xleftarrow{\$} \mathbb{F}_q^2$ and all the other variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-4.

Experiment 3-$(\boldsymbol{d+\upsilon})$-$\boldsymbol{\varpi}$-6 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}6)}$ for $\upsilon = 1,2; \varpi = 1,\ldots,\aleph$): The same as Experiment 3-$(d+\upsilon)$-ϖ-5 except that

$$\begin{aligned}
&\text{for } i = 1,2,\\
&\boldsymbol{h}^*_{d+\upsilon,\varpi,i} = (\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1), \gamma\vec{e}_i, 0^2, \boxed{\xi\vec{e}_i, \delta\vec{e}_i\boldsymbol{U}_{d+\upsilon,\varpi}}, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{B}^*}, && (66)\\
&\boldsymbol{e}_{d+\upsilon,\varpi,i} = (\sigma_{d+\upsilon,\varpi,i}(1,d+\upsilon), \omega\vec{e}_i, \tau\vec{e}_i, \boxed{0^2}, \tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\varpi}, 0^2, \vec{\varphi}_{d+\upsilon,\varpi,i}, 0^2)_{\mathbb{B}}, && (67)\\
&\text{for } \iota = 1,\ldots,\varpi-1,\varpi+1,\ldots,\aleph; i = 1,2,\\
&\boldsymbol{e}_{d+\upsilon,\iota,i} = (\sigma_{d+\upsilon,\iota,i}(1,d+\upsilon), \omega\vec{e}_i, \tau\vec{e}_i, \boxed{\vec{\chi}_{d+\upsilon,\iota,i}}, \tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\iota}, 0^2, \vec{\varphi}_{d+\upsilon,\iota,i}, 0^2)_{\mathbb{B}}, && (68)
\end{aligned}$$

where $\xi \xleftarrow{\$} \mathbb{F}_q$, $\boldsymbol{Z}_{d+\upsilon,\varpi} \xleftarrow{\$} \mathsf{GL}(2,\mathbb{F}_q)$, $\boldsymbol{U}_{d+\upsilon,\varpi} = (\boldsymbol{Z}_{d+\upsilon,\varpi}^{-1})^{\intercal}$, $\vec{\chi}_{d+\upsilon,\iota,i} \xleftarrow{\$} \mathbb{F}_q^2$, and all the other variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-5.

Experiment 3-$(\boldsymbol{d+\upsilon})$-$\boldsymbol{\varpi}$-7 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}7)}$ for $\upsilon = 1,2; \varpi = 1,\ldots,\aleph$): Analogous to Experiment 3-$(d+\upsilon)$-ϖ-6 with the exception that

$$\begin{aligned}
&\text{for } l = 1,\ldots,d; i = 1,2,\\
&\boldsymbol{e}_{l,i} = (\sigma_{l,i}(1,l), \omega\vec{e}_i, \tau\vec{e}_i, \boxed{0^2}, \tau\vec{e}_i\boldsymbol{Z}_l, 0^2, \vec{\varphi}_{l,i}, 0^2)_{\mathbb{B}}, && (69)\\
&\text{for } k = 1,\ldots,\upsilon-1,\upsilon+1,\ldots,2; \iota = 1,\ldots,\aleph; i = 1,2,\\
&\boldsymbol{e}_{d+k,\iota,i} = (\sigma_{d+k,\iota,i}(1,d+k), \omega\vec{e}_i, \tau\vec{e}_i, \boxed{0^2}, \tau\vec{e}_i\boldsymbol{Z}_{d+k,\iota}, 0^2, \vec{\varphi}_{d+k,\iota,i}, 0^2)_{\mathbb{B}}, && (70)
\end{aligned}$$

where all the variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-6.

Experiment 3-$(\boldsymbol{d+\upsilon})$-$\boldsymbol{\varpi}$-8-$\boldsymbol{\iota}$-$\boldsymbol{j}$-1 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}j\text{-}1)}$ for $\upsilon = 1,2; \varpi = 1,\ldots,\aleph; \iota = 1,\ldots,\varpi-1,\varpi+1,\ldots,\aleph; j = 1,2$): We view Experiment 3-$(d+\upsilon)$-$\varpi$-7 as Experiment 3-$(d+\upsilon)$-$\varpi$-8-0-2-2. This experiment is similar to Experiment 3-$(d+\upsilon)$-ϖ-8-$(\iota-1)$-2-2 (if $\iota \neq \varpi+1$) or Experiment 3-$(d+\upsilon)$-ϖ-8-$(\iota-2)$-2-2 (if $\iota = \varpi+1$), provided $j = 1$, or Experiment 3-$(d+\upsilon)$-ϖ-8-ι-1-2, provided $j = 2$, except that

$$\boldsymbol{e}_{d+\upsilon,\iota,j} = (\sigma_{d+\upsilon,\iota,j}(1,d+\upsilon), \omega\vec{e}_j, \tau\vec{e}_j, \boxed{\tilde{\sigma}_{d+\upsilon,\iota,j}(1,d+\upsilon)}, \tau\vec{e}_j\boldsymbol{Z}_{d+\upsilon,\iota}, 0^2, \vec{\varphi}_{d+\upsilon,\iota,j}, 0^2)_{\mathbb{B}}, \tag{71}$$

where $\widetilde{\sigma}_{d+\upsilon,\iota,j} \xleftarrow{\$} \mathbb{F}_q$ and all the other variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-8-$(\iota-1)$-2-2 or Experiment 3-$(d+\upsilon)$-ϖ-8-$(\iota-2)$-2-2, provided $j=1$, or Experiment 3-$(d+\upsilon)$-ϖ-8-ι-1-2, provided $j=2$.

Experiment 3-$(d{+}\upsilon)$-ϖ-8-ι-j-2 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}j\text{-}2)}$ for $\upsilon=1,2;\varpi=1,\ldots,\aleph;\iota=1,,\ldots,\varpi-1,\varpi+1,\ldots,\aleph;j=1,2$): The same as Experiment 3-$(d+\upsilon)$-ϖ-8-ι-j-1 with the exception that

$$\boldsymbol{e}_{d+\upsilon,\iota,j}=(\sigma_{d+\upsilon,\iota,j}(1,d+\upsilon),\omega\vec{e}_j,\tau\vec{e}_j,\boxed{0^2},\tau\vec{e}_j\boldsymbol{Z}_{d+\upsilon,\iota},0^2,\vec{\varphi}_{d+\upsilon,\iota,j},0^2)_{\mathbb{B}}, \tag{72}$$

where all the variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-8-ι-j-1.

Experiment 3-$(d+\upsilon)$-ϖ-9 ($\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}9)}$ for $\upsilon=1,2;\varpi=1,\ldots,\aleph$): This experiment is equivalent to Experiment 3-$(d+\upsilon)$-ϖ-8-$\aleph$-2-2, if $1\le\varpi\le\aleph-1$, or Experiment 3-$(d+\upsilon)$-ϖ-8-$(\aleph-1)$-2-2, if $\varpi=\aleph$, except that for $i=1,2$,

$$\boldsymbol{h}^*_{d+\upsilon,\varpi,i}=(\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1),\gamma\vec{e}_i,0^2,\boxed{0^2},\delta\vec{e}_i\boldsymbol{U}_{d+\upsilon,\varpi},\vec{\eta}_{d+\upsilon,\varpi,i},0^2,0^2)_{\mathbb{B}^*}, \tag{73}$$

where all the variables are generated as in Experiment 3-$(d+\upsilon)$-ϖ-8-$\aleph$-2-2, if $1\le\varpi\le\aleph-1$, or Experiment 3-$(d+\upsilon)$-ϖ-8-$(\aleph-1)$-2-2, if $\varpi=\aleph$.

Experiment 3-$(d+2)$-$\aleph$-9 is the $\beta=1$ case of Problem 2.

Lemmas: Here also we consider canonical (monomial) linear order in $\mathbb{N}^2$ and $\mathbb{N}^3$, as defined in Appendix B.

Lemma 22. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_1$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(0)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(1)}(\lambda)\to 1\right]\right|\le\mathsf{Adv}_{\mathcal{C}_1}^{\mathsf{BP2}}(\lambda).$$

Lemma 23. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}(p-1)\text{-}8)}(\lambda)\to 1\right]=\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}1)}(\lambda)\to 1\right].$$

Lemma 24. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{2\text{-}1}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}1)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}2)}(\lambda)\to 1\right]\right|\le\mathsf{Adv}_{\mathcal{C}_{2\text{-}p\text{-}1}}^{\mathsf{BP3}\text{-}p}(\lambda),$$

where $\mathcal{C}_{2\text{-}p\text{-}1}(\cdot)=\mathcal{C}_{2\text{-}1}(p,\cdot)$.

Lemma 25. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}2)}(\lambda)\to 1\right]=\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}3)}(\lambda)\to 1\right].$$

Lemma 26. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists probabilistic machine $\mathcal{C}_{2\text{-}2}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}3)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}4)}(\lambda)\to 1\right]\right|\le\mathsf{Adv}_{\mathcal{C}_{2\text{-}p\text{-}2}}^{\mathsf{BP3}\text{-}p}(\lambda),$$

where $\mathcal{C}_{2\text{-}p\text{-}2}(\cdot)=\mathcal{C}_{2\text{-}2}(p,\cdot)$.

Lemma 27. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{2\text{-}3}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}4)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}5)}(\lambda)\to 1\right]\right|\leq \mathsf{Adv}_{\mathcal{C}_{2\text{-}p\text{-}3}}^{\mathsf{BP5}\text{-}p}(\lambda),$$

where $\mathcal{C}_{2\text{-}p\text{-}3}(\cdot)=\mathcal{C}_{2\text{-}3}(p,\cdot)$.

Lemma 28. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}5)}(\lambda)\to 1\right]=\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}6)}(\lambda)\to 1\right].$$

Lemma 29. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{2\text{-}4}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}6)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}7)}(\lambda)\to 1\right]\right|\leq \mathsf{Adv}_{\mathcal{C}_{2\text{-}p\text{-}4}}^{\mathsf{BP5}\text{-}p}(\lambda),$$

where $\mathcal{C}_{2\text{-}p\text{-}4}(\cdot)=\mathcal{C}_{2\text{-}4}(p,\cdot)$.

Lemma 30. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{2\text{-}5}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}7)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(2\text{-}p\text{-}8)}(\lambda)\to 1\right]\right|\leq \mathsf{Adv}_{\mathcal{C}_{2\text{-}p\text{-}5}}^{\mathsf{BP4}\text{-}p}(\lambda),$$

where $\mathcal{C}_{2\text{-}p\text{-}5}(\cdot)=\mathcal{C}_{2\text{-}5}(p,\cdot)$.

The proofs of Lemmas 22–30 are similar to Lemmas 48–56 of [17] respectively with straightforward extension.

Lemma 31. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\begin{aligned}\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon-1)\text{-}\aleph\text{-}9)}(\lambda)\to 1\right]&=\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}1\text{-}1)}(\lambda)\to 1\right]\ (\varpi=1)\\ \text{or }\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}(\varpi-1)\text{-}9)}(\lambda)\to 1\right]&=\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1)}(\lambda)\right]\ (1<\varpi\leq\aleph).\end{aligned}$$

Lemma 32. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{3\text{-}1}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}2)}(\lambda)\to 1\right]\right|\leq \mathsf{Adv}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1}}^{\mathsf{BP3}\text{-}(d+\upsilon)}(\lambda),$$

where $\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}1}(\cdot)=\mathcal{C}_{3\text{-}1}(d+\upsilon,\varpi,\cdot)$.

Lemma 33. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}2)}(\lambda)\to 1\right]=\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}3)}(\lambda)\to 1\right].$$

Lemma 34. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{3\text{-}2}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}3)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}4)}(\lambda)\to 1\right]\right|\leq \mathsf{Adv}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}2}}^{\mathsf{BP3}\text{-}(d+\upsilon)}(\lambda),$$

where $\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}2}(\cdot)=\mathcal{C}_{3\text{-}2}(d+\upsilon,\varpi,\cdot)$.

Lemmas 31–34 can be proven in an analogous fashion as Lemmas 49–52 of [17] respectively by extending the simulation.

Lemma 35. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{3\text{-}3}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}4)}(\lambda)\to 1\right]-\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}5)}(\lambda)\to 1\right]\right|\le \mathsf{Adv}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}3}}^{\mathsf{BP5}\text{-}(d+\upsilon)}(\lambda),$$

where $\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}3}(\cdot)=\mathcal{C}_{3\text{-}3}(d+\upsilon,\varpi,\cdot)$.

Proof. $\mathcal{C}_{3\text{-}3}$ is given integers $d+\upsilon,\varpi$ and a $\mathsf{BP5}\text{-}(d+\upsilon)$ instance

$$(\mathsf{param},\mathbb{B}_0,\mathbb{B}_0^*,\mathbb{B},\widehat{\mathbb{B}}^*,\boldsymbol{h}_0^*,\{\boldsymbol{h}_{d+\upsilon,i}^*,\boldsymbol{e}_{\beta,l,i}\}_{l=1,\dots,d+\upsilon-1,d+\upsilon+1,\dots,d+2;i=1,2},\{\widetilde{\boldsymbol{h}}_j^*\}_{j=5,6,9,10}).$$

$\mathcal{C}_{3\text{-}3}$ calculates

$$\boldsymbol{g}_0=(\omega,\tau,0,0,\varphi_0)_{\mathbb{B}_0},$$

using $\mathbb{B}_0$ and $\omega,\varphi_0\xleftarrow{\$}\mathbb{F}_q,\tau\xleftarrow{\$}\mathbb{F}_q^\times$. $\mathcal{C}_{3\text{-}3}$ then calculates

$$\begin{aligned}
&\boldsymbol{p}_0^*=\boldsymbol{h}_0^*+(\gamma,0,0,\eta_0,0)_{\mathbb{B}_0^*},\\
&\text{for } l=1,\dots,d;i=1,2,\\
&\boldsymbol{p}_{l,i}^*=\sum_{j=1}^{2}\boldsymbol{U}_{l,i,j}\widetilde{\boldsymbol{h}}_{8+j}^*+(\mu_{l,i}(l,-1),\gamma\vec{e}_i,0^6,\vec{\eta}_{l,i},0^2,0^2)_{\mathbb{B}^*}\\
&\text{for } (d+k,\iota')<(d+\upsilon,\varpi);i=1,2,\\
&\boldsymbol{p}_{d+k,\iota',i}^*=\sum_{j=1}^{2}\boldsymbol{U}_{d+k,\iota',i,j}\widetilde{\boldsymbol{h}}_{8+j}^*+(\mu_{d+k,\iota',i}(d+k,-1),\gamma\vec{e}_i,0^6,\vec{\eta}_{d+k,\iota',i},0^2,0^2)_{\mathbb{B}^*},\\
&\text{for } (d+k,\iota')=(d+\upsilon,\varpi);i=1,2,\\
&\boldsymbol{p}_{d+\upsilon,\varpi,i}^*=\boldsymbol{h}_{d+\upsilon,i}^*+(0^2,\gamma\vec{e}_i,0^{12})_{\mathbb{B}^*},\\
&\text{for } (d+\upsilon,\varpi)<(d+k,\iota');i=1,2,\\
&\boldsymbol{p}_{d+k,\iota',i}^*=\sum_{j=1}^{2}\widetilde{\boldsymbol{h}}_{4+j}^*+(\mu_{d+k,\iota',i}(d+k,-1),\gamma\vec{e}_i,0^6,\vec{\eta}_{d+k,\iota',i},0^2,0^2)_{\mathbb{B}^*},\\
&\text{for } l=1,\dots,d;i=1,2,\\
&\boldsymbol{g}_{l,i}=\boldsymbol{e}_{\beta,l,i}+(0^2,\omega\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i\boldsymbol{Z}_l,0^6)_{\mathbb{B}},\\
&\text{for } k=1,\dots,\upsilon-1,\upsilon+1,\dots,2;\iota'=1,\dots,\aleph;i=1,2,\\
&\boldsymbol{g}_{d+k,\iota',i}=\alpha_{d+k,\iota',i}\boldsymbol{e}_{\beta,d+k,i}+(0^2,\omega\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i\boldsymbol{Z}_{d+k,\iota'},0^6)_{\mathbb{B}}+\sum_{j=1}^{2}\varphi_{d+k,\iota',i,j}\boldsymbol{b}_{12+j},\\
&\text{for } k=\upsilon;\iota'=1,\dots,\aleph;i=1,2,\\
&\boldsymbol{g}_{d+\upsilon,\iota',i}=(\sigma_{d+\upsilon,\iota',i}(1,d+\upsilon),\omega\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\iota'},0^2,\vec{\varphi}_{d+\upsilon,\iota',i},0^2)_{\mathbb{B}},
\end{aligned}$$

using $\mathbb{B}^*,\mathbb{B}$ and $\gamma,\eta_0,\mu_{l,i},\mu_{d+k,\iota',i},\alpha_{d+k,\iota',i},\varphi_{d+k,\iota',i,j},\sigma_{d+\upsilon,\iota',i}\xleftarrow{\$}\mathbb{F}_q$; $\vec{\eta}_{l,i},\vec{\eta}_{d+k,\iota',i},\vec{\varphi}_{d+\upsilon,\iota',i}\xleftarrow{\$}\mathbb{F}_q^2$; $\boldsymbol{Z}_l,\boldsymbol{Z}_{d+k,\iota'},\boldsymbol{Z}_{d+\upsilon,\iota'}\xleftarrow{\$}\mathsf{GL}(2,\mathbb{F}_q)$; $\boldsymbol{U}_l=(\boldsymbol{U}_{l,i,j})_{i,j}=(\boldsymbol{Z}_l^{-1})^\intercal,\boldsymbol{U}_{d+k,\iota'}=(\boldsymbol{U}_{d+k,\iota',i,j})_{i,j}=(\boldsymbol{Z}_{d+k,\iota'}^{-1})^\intercal$.

$\mathcal{C}_{3\text{-}3}$ then sets

$$\begin{aligned}
&\widehat{\mathbb{B}}_0=\{\boldsymbol{b}_{0,1},\boldsymbol{b}_{0,3},\boldsymbol{b}_{0,5}\},\ \widehat{\mathbb{B}}_0^*=\{\boldsymbol{b}_{0,1}^*,\dots,\boldsymbol{b}_{0,4}^*\},\\
&\widehat{\mathbb{B}}'=\{\boldsymbol{b}_1,\dots,\boldsymbol{b}_4,\boldsymbol{b}_{13},\boldsymbol{b}_{14}\},\ \widehat{\mathbb{B}}^*=\{\boldsymbol{b}_1^*,\dots,\boldsymbol{b}_4^*,\boldsymbol{b}_{11}^*,\boldsymbol{b}_{12}^*\}.
\end{aligned}$$

Next $\mathcal{C}_{3\text{-}3}$ gives

$$\varrho = (\mathsf{param}, \widehat{\mathbb{B}}_0, \widehat{\mathbb{B}}_0^*, \widehat{\mathbb{B}}', \widehat{\mathbb{B}}^*, \boldsymbol{p}_0^*, \boldsymbol{g}_0, \{\boldsymbol{p}_{l,i}^*, \boldsymbol{g}_{l,i}\}_{l=1,\ldots,d;i=1,2}, \{\boldsymbol{p}_{d+k,\iota',i}^*, \boldsymbol{g}_{d+k,\iota',i}\}_{k=1,2;\iota'=1,\ldots,\aleph;i=1,2})$$

to $\mathcal{B}$, and outputs $\beta' \in \{0,1\}$ if $\mathcal{B}$ outputs β'.

If $\beta = 0$ (resp. $\beta = 1$) , the distribution of ϱ is exactly same as that of instances in Experiment 3-$(d+\upsilon)$-ϖ-4 (resp. Experiment 3-$(d+\upsilon)$-ϖ-5). □

Lemma 36. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}5)}(\lambda) \to 1\right] = \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}6)}(\lambda) \to 1\right].$$

Proof. To prove lemma 36 we will show that the distribution $(\mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}}, \widehat{\mathbb{B}}^*\{\boldsymbol{h}_{t,i}^*, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d;i=1,2}, \{\boldsymbol{h}_{d+k,\iota',i}, \boldsymbol{e}_{d+k,\iota',i}\}_{k=1,2;\iota'=1,\ldots,\aleph;i=1,2})$ in Experiment 3-$(d+\upsilon)$-ϖ-5 and that in Experiment 3-$(d+\upsilon)$-ϖ-6 are equivalent. For that purpose, we define new dual orthogonal bases $\{\mathbb{D}, \mathbb{D}^*\}$ of $\mathbb{V}$ as follows: We generate $\widetilde{\xi}_i \xleftarrow{\$} \mathbb{F}_q^\times$, $\boldsymbol{Z}_{d+\upsilon,\varpi} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$, set $\boldsymbol{U}_{d+\upsilon,\varpi} = (\boldsymbol{Z}_{d+\upsilon,\varpi}^{-1})^\intercal$, and define

$$\begin{pmatrix} \boldsymbol{d}_7^* \\ \boldsymbol{d}_8^* \\ \boldsymbol{d}_9^* \\ \boldsymbol{d}_{10}^* \end{pmatrix} = \begin{pmatrix} \widetilde{\xi}\boldsymbol{I}_2 & -\boldsymbol{U}_{d+\upsilon,\varpi} \\ \\ \boldsymbol{0}_2 & \boldsymbol{I}_2 \end{pmatrix} \begin{pmatrix} \boldsymbol{b}_7^* \\ \boldsymbol{b}_8^* \\ \boldsymbol{b}_9^* \\ \boldsymbol{b}_{10}^* \end{pmatrix}, \quad \begin{pmatrix} \boldsymbol{d}_7 \\ \boldsymbol{d}_8 \\ \boldsymbol{d}_9 \\ \boldsymbol{d}_{10} \end{pmatrix} = \begin{pmatrix} \widetilde{\xi}^{-1}\boldsymbol{I}_2 & \boldsymbol{0}_2 \\ \\ \widetilde{\xi}^{-1}\boldsymbol{Z}_{d+\upsilon,\varpi}^{-1} & \boldsymbol{I}_2 \end{pmatrix} \begin{pmatrix} \boldsymbol{b}_7 \\ \boldsymbol{b}_8 \\ \boldsymbol{b}_9 \\ \boldsymbol{b}_{10} \end{pmatrix},$$

$$\mathbb{D} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_6, \boldsymbol{d}_7, \ldots, \boldsymbol{d}_{10}, \boldsymbol{b}_{11}, \ldots, \boldsymbol{b}_{16}\},$$
$$\mathbb{D}^* = \{\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_6^*, \boldsymbol{d}_7^*, \ldots, \boldsymbol{d}_{10}^*, \boldsymbol{b}_{11}^*, \ldots, \boldsymbol{b}_{16}^*\},$$

where $\boldsymbol{I}_2$ and $\boldsymbol{0}_2$ respectively denote the 2×2 identity and zero matrices. It can be easily verified that $\{\mathbb{D}, \mathbb{D}^*\}$ are dual orthogonal bases and are distributed the same as the original bases $\{\mathbb{B}, \mathbb{B}^*\}$. Below we show how $(\{\boldsymbol{h}_{t,i}^*, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d;i=1,2}, \{\boldsymbol{h}_{d+k,\iota',i}, \boldsymbol{e}_{d+k,\iota',i}\}_{k=1,2;\iota'=1,\ldots,\aleph;i=1,2})$ in Experiment 3-$(d+\upsilon)$-ϖ-5 can be expressed over bases $\{\mathbb{B}, \mathbb{B}^*\}$ and $\{\mathbb{D}, \mathbb{D}^*\}$ in two different ways.
For $t = 1, \ldots, d; i = 1, 2$,

$$\begin{aligned} \boldsymbol{h}_{t,i}^* =& (\mu_{t,i}(t,-1), \gamma\vec{e}_i, 0^4, \delta\vec{e}_i\boldsymbol{U}_t, \vec{\eta}_{t,i}, 0^2, 0^2)_{\mathbb{B}^*} \\ =& (\mu_{t,i}(t,-1), \gamma\vec{e}_i, 0^4, \delta\vec{e}_i\boldsymbol{U}_t, \vec{\eta}_{t,i}, 0^2, 0^2)_{\mathbb{D}^*}. \end{aligned}$$

For $(d+k, \iota') < (d+\upsilon, \varpi); i = 1, 2$,

$$\begin{aligned} \boldsymbol{h}_{d+k,\iota',i}^* =& (\mu_{d+k,\iota',i}(d+k,-1), \gamma\vec{e}_i, 0^4, \delta\vec{e}_i\boldsymbol{U}_{d+k,\iota'}, \vec{\eta}_{d+k,\iota',i}, 0^2, 0^2)_{\mathbb{B}^*} \\ =& (\mu_{d+k,\iota',i}(d+k,-1), \gamma\vec{e}_i, 0^4, \delta\vec{e}_i\boldsymbol{U}_{d+k,\iota'}, \vec{\eta}_{d+k,\iota',i}, 0^2, 0^2)_{\mathbb{D}^*}. \end{aligned}$$

For $(d+k, \iota') = (d+\upsilon, \varpi); i = 1, 2$,

$$\begin{aligned} \boldsymbol{h}_{d+\upsilon,\varpi,i}^* =& (\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1), \gamma\vec{e}_i, 0^2, \delta\vec{e}_i, 0^2, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{B}^*} \\ =& (\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1), \gamma\vec{e}_i, 0^2, \xi\vec{e}_i, \delta\vec{e}_i\boldsymbol{U}_{d+\upsilon,\varpi}, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{D}^*}, \end{aligned}$$

where $\xi = \widetilde{\xi}^{-1}\delta$.
For $(d+\upsilon, \varpi) < (d+k, \iota'); i = 1, 2$,

$$\begin{aligned} \boldsymbol{h}_{d+k,\iota',i}^* =& (\mu_{d+k,\iota',i}(d+k,-1), \gamma\vec{e}_i, \delta\vec{e}_i, 0^4, \vec{\eta}_{d+k,\iota',i}, 0^2, 0^2)_{\mathbb{B}^*} \\ =& (\mu_{d+k,\iota',i}(d+k,-1), \gamma\vec{e}_i, \delta\vec{e}_i, 0^4, \vec{\eta}_{d+k,\iota',i}, 0^2, 0^2)_{\mathbb{D}^*}. \end{aligned}$$

For $t = 1, \ldots, d; i = 1, 2$,

$$\begin{aligned} \boldsymbol{e}_{t,i} =& (\sigma_{t,i}(1,t), \omega\vec{e}_i, \tau\vec{e}_i, \vec{\chi}_{t,i}, \tau\vec{e}_i\boldsymbol{Z}_t, 0^2, \vec{\varphi}_{t,i}, 0^2)_{\mathbb{B}} \\ =& (\sigma_{t,i}(1,t), \omega\vec{e}_i, \tau\vec{e}_i, \vec{\chi}'_{t,i}, \tau\vec{e}_i\boldsymbol{Z}_t, 0^2, \vec{\varphi}_{t,i}, 0^2)_{\mathbb{D}}, \end{aligned}$$

where $\vec{\chi}'_{t,i} = \widetilde{\xi}(\vec{\chi}_{t,i} - \tau\vec{e}_i\boldsymbol{Z}_t\boldsymbol{Z}^{-1}_{d+\upsilon,\varpi})$.
For $k = 1,\dots,\upsilon-1,\upsilon+1,\dots,2; \iota' = 1,\dots,\aleph; i = 1,2$,

$$\begin{aligned}\boldsymbol{e}_{d+k,\iota',i} =&(\sigma_{d+k,\iota',i}(1,d+k),\omega\vec{e}_i,\tau\vec{e}_i,\vec{\chi}_{d+k,\iota',i},\tau\vec{e}_i\boldsymbol{Z}_{d+k,\iota'},0^2,\vec{\varphi}_{d+k,\iota',i},0^2)_{\mathbb{B}}\\ =&(\sigma_{d+k,\iota',i}(1,d+k),\omega\vec{e}_i,\tau\vec{e}_i,\vec{\chi}'_{d+k,\iota',i},\tau\vec{e}_i\boldsymbol{Z}_{d+k,\iota'},0^2,\vec{\varphi}_{d+k,\iota',i},0^2)_{\mathbb{D}},\end{aligned}$$

where $\vec{\chi}'_{d+k,\iota',i} = \widetilde{\xi}(\vec{\chi}_{d+k,\iota',i} - \tau\vec{e}_i\boldsymbol{Z}_{d+k,\iota'}\boldsymbol{Z}^{-1}_{d+\upsilon,\varpi})$.
For $k = \upsilon; \iota' = 1,\dots,\varpi-1,\varpi+1,\dots,\aleph; i = 1,2$,

$$\begin{aligned}\boldsymbol{e}_{d+\upsilon,\iota',i} =&(\sigma_{d+\upsilon,\iota',i}(1,d+\upsilon),\omega\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\iota'},0^2,\vec{\varphi}_{d+\upsilon,\iota',i},0^2)_{\mathbb{B}}\\ =&(\sigma_{d+\upsilon,\iota',i}(1,d+\upsilon),\omega\vec{e}_i,\tau\vec{e}_i,\vec{\chi}_{d+\upsilon,\iota',i},\tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\iota'},0^2,\vec{\varphi}_{d+\upsilon,\iota',i},0^2)_{\mathbb{D}},\end{aligned}$$

where $\vec{\chi}_{d+\upsilon,\iota',i} = \widetilde{\xi}(\tau\vec{e}_i - \tau e_i\boldsymbol{Z}_{d+\upsilon,\iota'}\boldsymbol{Z}^{-1}_{d+\upsilon,\varpi})$.
For $k = \upsilon; \iota' = \varpi; i = 1,2$,

$$\begin{aligned}\boldsymbol{e}_{d+\upsilon,\varpi,i} =&(\sigma_{d+\upsilon,\varpi,i}(1,d+\upsilon),\omega\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i,\tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\varpi},0^2,\vec{\varphi}_{d+\upsilon,\varpi,i},0^2)_{\mathbb{B}}\\ =&(\sigma_{d+\upsilon,\varpi,i}(1,d+\upsilon),\omega\vec{e}_i,\tau\vec{e}_i,0^2,\tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\varpi},0^2,\vec{\varphi}_{d+\upsilon,\varpi,i},0^2)_{\mathbb{D}}.\end{aligned}$$

Note that $\vec{\chi}'_{t,i}, \vec{\chi}'_{d+k,\iota',i}$ are uniformly, independently distributed since $\vec{\chi}_{t,i}, \vec{\chi}_{d+k,\iota',i} \xleftarrow{\$} \mathbb{F}_q^2$ $(t = 1,\dots,d; k \neq \upsilon)$. Also $\vec{\chi}_{d+\upsilon,\iota',i}$ are uniformly, independently distributed since $\widetilde{\xi} \xleftarrow{\$} \mathbb{F}_q$ and $\boldsymbol{Z}_{d+\upsilon,\iota'} \xleftarrow{\$} \mathsf{GL}(2,\mathbb{F}_q)$.

In the light of adversary's view, both $\{\mathbb{B},\mathbb{B}^*\}$ and $\{\mathbb{D},\mathbb{D}^*\}$ are consistent with $(\mathsf{param}_{\mathbb{V}},\widehat{\mathbb{B}},\widehat{\mathbb{B}}^*)$. Therefore, $(\{\boldsymbol{h}^*_{t,i},\boldsymbol{e}_{t,i}\}_{t=1,\dots,d;i=1,2},\{\boldsymbol{h}^*_{d+k,\iota',i},\boldsymbol{e}_{d+k,\iota',i}\}_{k=1,2;\iota'=1,\dots,\aleph;i=1,2})$ can be expressed in two ways, in Experiment 3-$(d+\upsilon)$-ϖ-5 over bases $\{\mathbb{B},\mathbb{B}^*\}$ and in Experiment 3-$(d+\upsilon)$-ϖ-6 over bases $\{\mathbb{D},\mathbb{D}^*\}$. Thus Experiment 3-$(d+\upsilon)$-$\varpi$-5 can be conceptually changed to Experiment 3-$(d+\upsilon)$-ϖ-6. □

Lemma 37. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{3\text{-}4}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}6)}(\lambda)\to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}7)}(\lambda)\to 1\right]\right| \leq \mathsf{Adv}^{\mathsf{BP5}\text{-}(d+\upsilon)}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}4}}(\lambda),$$

where $\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}4}(\cdot) = \mathcal{C}_{3\text{-}4}(d+\upsilon),\varpi,\cdot)$.

The proof of Lemma 37 is similar to that of Lemma 35.

Lemma 38. *For any probabilistic adversary $\mathcal{B}$, for any security parameter λ,*

$$\left.\begin{array}{l}\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}(\iota-1)\text{-}2\text{-}2)}(\lambda)\to 1\right] = \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}1\text{-}1)}(\lambda)\to 1\right]\ \ (\iota\neq\varpi+1)\\ \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}(\iota-2)\text{-}2\text{-}2)}(\lambda)\to 1\right] = \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}1\text{-}1)}(\lambda)\to 1\right]\ \ (\iota=\varpi+1)\end{array}\right\}\ (j=1),$$
$$\text{or } \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}1\text{-}2)}(\lambda)\to 1\right] = \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}2\text{-}1)}(\lambda)\to 1\right]\ \ (j=2).$$

Proof. Suppose that in experiment previous to Experiment 3-$(d+\upsilon)$-ϖ-8-ι-j-1,

$$\boldsymbol{e}_{d+\upsilon,\iota,j} = (\sigma_{d+\upsilon,\iota,j}(1,d+\upsilon),\omega\vec{e}_j,\tau\vec{e}_j,\vec{\chi}_{d+\upsilon,\iota,j},\tau\vec{e}_j\boldsymbol{Z}_{d+\upsilon,\iota},0^2,\vec{\varphi}_{d+\upsilon,\iota,j},0^2)_{\mathbb{B}},$$

where $\vec{\chi}_{d+\upsilon,\iota,j} = (\chi_{d+\upsilon,\iota,j,1},\chi_{d+\upsilon,\iota,j,2}) \xleftarrow{\$} \mathbb{F}_q^2$. We select $\widetilde{\sigma}_{d+\upsilon,\iota,j} \xleftarrow{\$} \mathbb{F}_q^{\times}$ and consider

$$\boldsymbol{Z} = \begin{pmatrix}\chi_{d+\upsilon,\iota,j,1}\widetilde{\sigma}^{-1}_{d+\upsilon,\iota,j} & 0\\ 0 & \chi_{d+\upsilon,\iota,j,2}\widetilde{\sigma}^{-1}_{d+\upsilon,\iota,j}(d+\upsilon)^{-1}\end{pmatrix}, \boldsymbol{U} = (\boldsymbol{Z}^{-1})^{\mathsf{T}}$$

and set

$$\begin{pmatrix} \boldsymbol{d}_7 \\ \boldsymbol{d}_8 \end{pmatrix} = \boldsymbol{Z} \begin{pmatrix} \boldsymbol{b}_7 \\ \boldsymbol{b}_8 \end{pmatrix}, \begin{pmatrix} \boldsymbol{d}_7^* \\ \boldsymbol{d}_8^* \end{pmatrix} = \boldsymbol{U} \begin{pmatrix} \boldsymbol{b}_7^* \\ \boldsymbol{b}_8^* \end{pmatrix}.$$

We define

$$\mathbb{D} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_6, \boldsymbol{d}_7, \boldsymbol{d}_8, \boldsymbol{b}_9, \ldots, \boldsymbol{b}_{16}\}, \mathbb{D}^* = \{\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_6^*, \boldsymbol{d}_7^*, \boldsymbol{d}_8^*, \boldsymbol{b}_9^*, \ldots, \boldsymbol{b}_{16}^*\}.$$

Note that $\{\mathbb{D}, \mathbb{D}^*\}$ are consistent with $\{\widehat{\mathbb{B}}, \widehat{\mathbb{B}}^*\}$.

Since the coefficients of $\{\boldsymbol{b}_7^*, \boldsymbol{b}_8^*\}$ in all of $\{\boldsymbol{h}_{t,i}^*\}_{t=1,\ldots,d;i=1,2}, \{\boldsymbol{h}_{d+k,\iota',i}\}_{(d+k,\iota')\lessgtr(d+\upsilon,\varpi);i=1,2}$ are zero, replacing $\mathbb{B}^*$ with $\mathbb{D}^*$ would not affect these elements. For $i = 1, 2$,

$$\begin{aligned} \boldsymbol{h}_{d+\upsilon,\varpi,i}^* =& (\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1), \gamma\vec{e}_i, 0^2, \xi_{d+\upsilon,\varpi,i}\vec{e}_i, \delta\vec{e}_i\boldsymbol{U}_{d+\upsilon,\varpi}, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{B}^*} \\ =& (\mu_{d+\upsilon,\varpi,i}(d+\upsilon,-1), \gamma\vec{e}_i, 0^2, \widetilde{\xi}_{d+\upsilon,\varpi,i}\vec{e}_i, \delta\vec{e}_i\boldsymbol{U}_{d+\upsilon,\varpi}, \vec{\eta}_{d+\upsilon,\varpi,i}, 0^2, 0^2)_{\mathbb{D}^*}, \end{aligned}$$

where $\widetilde{\xi}_{d+\upsilon,\varpi,1} = \xi_{d+\upsilon,\varpi,1}\chi_{d+\upsilon,\iota,j,1}\widetilde{\sigma}_{d+\upsilon,\iota,j}^{-1}, \widetilde{\xi}_{d+\upsilon,\varpi,2} = \xi_{d+\upsilon,\varpi,2}\chi_{d+\upsilon,\iota,j,2}\widetilde{\sigma}_{d+\upsilon,\iota,j}^{-1}(d+\upsilon)^{-1}$ are uniformly, independently distributed since $\xi_{d+\upsilon,\varpi,1}, \xi_{d+\upsilon,\varpi,2} \xleftarrow{\$} \mathbb{F}_q$.
Again, since the coefficients of $\{\boldsymbol{b}_7, \boldsymbol{b}_8\}$ in $\{\boldsymbol{e}_{t,i}\}_{t=1,\ldots,d;i=1,2}, \{\boldsymbol{e}_{d+k,\iota',i}\}_{k=1,\ldots,\upsilon-1,\upsilon+1,\ldots,2;\iota'=1,\ldots,\aleph;i=1,2}$, $\{\boldsymbol{e}_{d+\upsilon,\iota',i}\}_{\substack{(\iota',i)<(\iota,j) \\ \iota'\neq\varpi}}, \{\boldsymbol{e}_{d+\upsilon,\varpi,i}\}_{i=1,2}$ are all zero, replacing $\mathbb{B}$ with $\mathbb{D}$ would also not affect these elements. For $(\iota, j) < (\iota', i)$ such that $\iota' \neq \varpi$,

$$\begin{aligned} \boldsymbol{e}_{d+\upsilon,\iota',i} =& (\sigma_{d+\upsilon,\iota',i}(1, d+\upsilon), \omega\vec{e}_i, \tau\vec{e}_i, \vec{\chi}_{d+\upsilon,\iota',i}, \tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\iota'}, 0^2, \vec{\varphi}_{d+\upsilon,\iota',i}, 0^2)_{\mathbb{B}} \\ =& (\sigma_{d+\upsilon,\iota',i}(1, d+\upsilon), \omega\vec{e}_i, \tau\vec{e}_i, \vec{\chi}'_{d+\upsilon,\iota',i}, \tau\vec{e}_i\boldsymbol{Z}_{d+\upsilon,\iota'}, 0^2, \vec{\varphi}_{d+\upsilon,\iota',i}, 0^2)_{\mathbb{D}}, \end{aligned}$$

where $\vec{\chi}'_{d+\upsilon,\iota',i} = \vec{\chi}_{d+\upsilon,\iota',i}\boldsymbol{U}^\intercal$ is uniformly, independently distributed since $\vec{\chi}_{d+\upsilon,\iota',i} \xleftarrow{\$} \mathbb{F}_q^2$.

Finally,

$$\begin{aligned} \boldsymbol{e}_{d+\upsilon,\iota,j} =& (\sigma_{d+\upsilon,\iota,j}(1, d+\upsilon), \omega\vec{e}_j, \tau\vec{e}_j, \vec{\chi}_{d+\upsilon,\iota,j}, \tau\vec{e}_j\boldsymbol{Z}_{d+\upsilon,\iota}, 0^2, \vec{\varphi}_{d+\upsilon,\iota,j}, 0^2)_{\mathbb{B}} \\ =& (\sigma_{d+\upsilon,\iota,j}(1, d+\upsilon), \omega\vec{e}_j, \tau\vec{e}_j, \widetilde{\sigma}_{d+\upsilon,\iota,j}(1, d+\upsilon), \tau\vec{e}_j\boldsymbol{Z}_{d+\upsilon,\iota}, 0^2, \vec{\varphi}_{d+\upsilon,\iota,j}, 0^2)_{\mathbb{D}}, \end{aligned}$$

where $\widetilde{\sigma}_{d+\upsilon,\iota,j} \xleftarrow{\$} \mathbb{F}_q$.

Thus the joint distribution for Experiment 3-$(d+\upsilon)$-ϖ-8-ι-j-1 and that for its previous experiment are equivalent. □

Lemma 39. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{3\text{-}5}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\left|\Pr\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}j\text{-}1)}(\lambda) \to 1\right] - \Pr\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\iota\text{-}j\text{-}2)}(\lambda) \to 1\right]\right| \leq \mathsf{Adv}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}5\text{-}\iota\text{-}j}}^{\mathsf{BP1}}(\lambda),$$

where $\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}5\text{-}\iota\text{-}j}(\cdot) = \mathcal{C}_{3\text{-}5}(d+\upsilon, \varpi, \iota, j, \cdot)$.

Proof. Given integers $d, \upsilon, \varpi, \iota, j$, and a BP1 instance

$$(\mathsf{param}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \mathbb{B}, \widehat{\mathbb{B}}^*, \boldsymbol{e}_{\beta,0}, \{\boldsymbol{e}_{\beta,i}\}_{i=1,2}),$$

$\mathcal{C}_{3\text{-}5}$ sets new dual orthogonal bases

$$\begin{aligned} &\mathbb{D}_0 = \{\boldsymbol{d}_{0,1}, \ldots, \boldsymbol{d}_{0,5}\} = \{\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,5}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,4}, \boldsymbol{b}_{0,2}\}, \\ &\mathbb{D}_0^* = \{\boldsymbol{d}_{0,1}^*, \ldots, \boldsymbol{d}_{0,5}^*\} = \{\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,5}^*, \boldsymbol{b}_{0,3}^*, \boldsymbol{b}_{0,4}^*, \boldsymbol{b}_{0,2}^*\}, \\ &\mathbb{D} = \{\boldsymbol{d}_1, \ldots, \boldsymbol{d}_{16}\} = \{\boldsymbol{b}_3, \boldsymbol{b}_4, \boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_{15}, \boldsymbol{b}_{16}, \boldsymbol{b}_5, \boldsymbol{b}_6, \boldsymbol{b}_9, \boldsymbol{b}_{10}, \boldsymbol{b}_{11}, \ldots, \boldsymbol{b}_{14}, \boldsymbol{b}_7, \boldsymbol{b}_8\}, \\ &\mathbb{D}^* = \{\boldsymbol{d}_1^*, \ldots, \boldsymbol{d}_{16}^*\} = \{\boldsymbol{b}_3^*, \boldsymbol{b}_4^*, \boldsymbol{b}_1^*, \boldsymbol{b}_2^*, \boldsymbol{b}_{15}^*, \boldsymbol{b}_{16}^*, \boldsymbol{b}_5^*, \boldsymbol{b}_6^*, \boldsymbol{b}_9^*, \boldsymbol{b}_{10}^*, \boldsymbol{b}_{11}^*, \ldots, \boldsymbol{b}_{14}^*, \boldsymbol{b}_7^*, \boldsymbol{b}_8^*\}. \end{aligned}$$

$\mathcal{C}_{3\text{-}5}$ then defines

$$\begin{aligned}&\widehat{\mathbb{D}}_0 = \{\boldsymbol{d}_{0,1}, \boldsymbol{d}_{0,3}, \boldsymbol{d}_{0,5}\},\ \widehat{\mathbb{D}}_0^* = \{\boldsymbol{d}_{0,1}^*, \dots, \boldsymbol{d}_{0,4}^*\},\\ &\widehat{\mathbb{D}} = \{\boldsymbol{d}_1, \dots, \boldsymbol{d}_4, \boldsymbol{d}_{13}, \boldsymbol{d}_{14}\},\ \widehat{\mathbb{D}}^* = \{\boldsymbol{d}_1^*, \dots, \boldsymbol{d}_4^*, \boldsymbol{d}_{11}^*, \boldsymbol{d}_{12}^*\}.\end{aligned}$$

Note that $\mathcal{C}_{3\text{-}5}$ can calculate $\{\widehat{\mathbb{D}}_0, \widehat{\mathbb{D}}_0^*\}, \{\widehat{\mathbb{D}}, \widehat{\mathbb{D}}^*\}$ from $\{\mathbb{B}_0, \widehat{\mathbb{B}}_0^*\}, \{\mathbb{B}, \widehat{\mathbb{B}}^*\}$ in the $\mathsf{BP1}$ instance.

$\mathcal{C}_{3\text{-}5}$ then calculates

$$\begin{aligned}&\{\boldsymbol{e}_{t,i}\}_{t=1,\dots,d;i=1,2} \text{ as in equation (69)},\\ &\{\boldsymbol{e}_{d+k,\iota',i}\}_{k=1,\dots,\upsilon-1,\upsilon+1,\dots,2;\iota'=1,\dots,\aleph;i=1,2} \text{ as in equation (70)},\\ &\{\boldsymbol{e}_{d+\upsilon,\iota',i}\}_{\substack{(\iota',i)<(\iota,j)\\ \iota'\neq\varpi}} \text{ as in equation (72)},\\ &\{\boldsymbol{e}_{d+\upsilon,\iota',i}\}_{\substack{(\iota,j)<(\iota',i)\\ \iota'\neq\varpi}} \text{ as in equation (68) and}\\ &\{\boldsymbol{e}_{d+\upsilon,\varpi,i}\}_{i=1,2} \text{ as in equation (67)}\end{aligned}$$

using $\mathbb{D}$ and $\widetilde{\omega} \xleftarrow{\$} \mathbb{F}_q, \widetilde{\tau} \xleftarrow{\$} \mathbb{F}_q^\times$ and other necessary random elements from suitable spaces.

$\mathcal{C}_{3\text{-}5}$ also computes

$$\begin{aligned}&\boldsymbol{h}_0^* \text{ as in equation (44)},\\ &\{\boldsymbol{h}_{t,i}^*\}_{t=1,\dots,d;i=1,2} \text{ as in equation (58)},\\ &\{\boldsymbol{h}_{d+k,\iota',i}^*\}_{(d+k,\iota')<(d+\upsilon,\varpi);i=1,2} \text{ as in equation (73)},\\ &\{\boldsymbol{h}_{d+k,\iota',i}^*\}_{(d+\upsilon,\varpi)<(d+k,\iota');i=1,2} \text{ as in equation (46) and}\\ &\{\boldsymbol{h}_{d+\upsilon,\varpi,i}^*\}_{i=1,2} \text{ as in equation (66)}\end{aligned}$$

using $\mathbb{D}^*$ and required random elements from suitable spaces.

Using $\widetilde{\omega}, \widetilde{\tau}$, $\mathcal{C}_{3\text{-}5}$ computes

$$\begin{aligned}\boldsymbol{g}_0 &= (\widetilde{\omega}, \widetilde{\tau}, 0, 0, \varphi_0)_{\mathbb{D}_0},\\ \boldsymbol{g}_{d+\upsilon,\iota,j} &= \boldsymbol{e}_{\beta,1} + (d+\upsilon)\boldsymbol{e}_{\beta,2} + \widetilde{\omega}\boldsymbol{d}_{2+j} + \widetilde{\tau}\boldsymbol{d}_{4+j} + \widetilde{\tau}\sum_{m=1}^{2} \boldsymbol{Z}_{d+\upsilon,\iota,j,m}\boldsymbol{d}_{8+m},\end{aligned}$$

where $\varphi_0 \xleftarrow{\$} \mathbb{F}_q$ and $\boldsymbol{Z}_{d+\upsilon,\iota} = (\boldsymbol{Z}_{d+\upsilon,\iota,i,m})_{i,m=1,2} \xleftarrow{\$} \mathsf{GL}(2, \mathbb{F}_q)$ (has been chosen during the computation of $\boldsymbol{e}_{d+\upsilon,\iota,2}$ or $\boldsymbol{e}_{d+\upsilon,\iota,1}$ according as $j = 1$ or 2).

$\mathcal{C}_{3\text{-}5}$ then gives

$$\begin{aligned}\varrho =&(\mathsf{param}, \widehat{\mathbb{D}}_0, \widehat{\mathbb{D}}_0^*, \widehat{\mathbb{D}}, \widehat{\mathbb{D}}^*, \boldsymbol{h}_0^*, g_0, \{\boldsymbol{h}_{t,i}, \boldsymbol{e}_{t,i}\}_{t=1,\dots,d;i=1,2}, \{\boldsymbol{h}_{d+k,\iota',i}^*\}_{k=1,2;\iota'=1,\dots,\aleph;i=1,2},\\ &\{\boldsymbol{e}_{d+k,\iota',i}\}_{(d+k,\iota',i)\lesseqgtr(d+\upsilon,\iota,j)}, \boldsymbol{g}_{d+\upsilon,\iota,j})\end{aligned}$$

to $\mathcal{B}$ and outputs $\beta' \in \{0, 1\}$ if $\mathcal{B}$ outputs β'.

If $\beta = 1$ (resp. $\beta = 0$), the distribution of ϱ is exactly identical to that of instances in Experiment 3-$(d+\upsilon)$-ϖ-8-ι-j-1 (resp. Experiment 3-$(d+\upsilon)$-ϖ-8-ι-j-2). □

Lemma 40. *For any probabilistic polynomial-time adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{C}_{3\text{-}6}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ,*

$$\begin{aligned}&\text{for } 1 \leq \varpi \leq \aleph - 1,\\ &\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}8\text{-}\aleph\text{-}2\text{-}2)}(\lambda) \to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\varpi\text{-}9)}(\lambda) \to 1\right]\right| \leq \mathsf{Adv}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}6}}^{\mathsf{BP4}\text{-}(d+\upsilon)}(\lambda),\\ &\text{or, for } \varpi = \aleph,\\ &\left|\mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\aleph\text{-}8\text{-}(\aleph-1)\text{-}2\text{-}2)}(\lambda) \to 1\right] - \mathsf{Pr}\left[\mathsf{Exp}_{\mathcal{B}}^{(3\text{-}(d+\upsilon)\text{-}\aleph\text{-}9)}(\lambda) \to 1\right]\right| \leq \mathsf{Adv}_{\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}6}}^{\mathsf{BP4}\text{-}(d+\upsilon)}(\lambda),\end{aligned}$$

where $\mathcal{C}_{3\text{-}(d+\upsilon)\text{-}\varpi\text{-}6}(\cdot) = \mathcal{C}_{3\text{-}6}(d+\upsilon, \varpi, \cdot)$.

Lemma 40 can be proven analogously as Lemma 56 of [17] by extending the simulation.