# New Proof Techniques for DLIN-Based Adaptively Secure Attribute-Based Encryption and Their Application

Katsuyuki Takashima
Mitsubishi Electric, Takashima.Katsuyuki@aj.MitsubishiElectric.co.jp

October 22, 2015

**Abstract.** We propose *adaptively secure* attribute-based encryption (ABE) schemes for boolean formulas over large universe attributes from the *decisional linear (DLIN) assumption*, which allow *an arbitrary number of attribute reuse* in an available formula *without the previously employed redundant multiple encoding technique*. Based on the key-policy (KP-)ABE scheme, we have an *adaptively secure communication-efficient non-interactive verifiable computation (NI-VC) from DLIN*. While any previous adaptive NI-VC from a static assumption has multiplicatively dependent communication cost on the input variable multiplicity, we remove the dependency. For achieving the results, we develop a new encoding method for access policy matrix for ABE, by *decoupling linear secret sharing (LSS)* into its matrix and randomness, and *partially randomizing* the LSS shares in simulation. The new techniques are of independent interest and we expect it will find another application than ABE.

**Keywords:** Attribute-Based Encryption, Verifiable Computation, Unbounded Multi-Use Attributes in Policy, Adaptive Security, Static Assumption

## 1 Introduction

### 1.1 Backgrounds

*Attribute-based encryption* (ABE) introduced by Sahai and Waters [25] presents an advanced vision for encryption and provides more flexible and fine-grained access control in sharing and distributing sensitive data than traditional symmetric and public-key encryption as well as recent identity-based encryption. In ABE systems, either one of the parameters for encryption and secret key is a set of attributes, and the other is an access policy (structure) over a universe of attributes, e.g., a secret key for a user is associated with an access policy and a ciphertext is associated with a set of attributes. A secret key with a policy can decrypt a ciphertext associated with a set of attributes, iff the attribute set satisfies the policy. If the access policy is for a secret key (resp. for encryption), it is called key-policy ABE (KP-ABE) (resp. ciphertext-policy ABE (CP-ABE)).

All the existing *practical* ABE schemes have been constructed by (bilinear) pairing groups, and the largest class of relations supported by the ABE schemes is (non-monotone or arithmetic) span programs [14, 15, 8, 3] (or (non-monotone) span programs with inner-product relations [20]). While general polynomial size circuits are supported [13, 7] recently, they are much less efficient than the

pairing-based ABE schemes and non-practical when the relations are limited to span programs. Hereafter, we focus on pairing-based ABE with span program access structures. An example of such span program predicate over attributes is given by (Institute = Univ. A) AND ((Department = Biology) OR (Position = Professor)), which we simply denote by $X_1 \wedge (X_2 \vee X_3)$ where $X_1 :=$ Univ. A, $X_2 :=$ Biology and $X_3 :=$ Professor. We define attribute-multiplicity $k$ for a predicate as the maximum number of appearances of attribute variables, i.e., $k = 2$ for predicate $(X_1 \wedge X_2) \vee (X_1 \wedge X_3) \vee (X_2 \wedge X_4)$ since $X_1$ and $X_2$ appear twice and others appear just once. Our aim is to achieve *short ciphertexts (resp. keys)*, in particular, short size *independent of the attribute-multiplicity in an access policy* in expressive (adaptively secure) KP-ABE (resp. CP-ABE). ABE with unbounded attribute-multiplicity is called "multi-use" ABE scheme in the literatures ([16, 20] etc.).

Adaptive security for ABE is the standard and realistic, and then desirable security notion. Previously, either efficiency or security is sacrificed for achieving the multi-use property in adaptively secure ABE. See adaptively secure ABE given in Table 1 (and Table 2).

In previous *static* assumption based schemes [16, 20, 8], for allowing arbitrary reuse of attributes in a policy in the adaptive security setting, for example, in KP-ABE, multiple ciphertext components whose number is linear in the attribute multiplicity $k$ for available policies are necessary, which leads to a very long ciphertext. More precisely, the same information representing attribute set $\Gamma$ is duplicated over *multiple* ciphertext components depending on the multiplicity $k$. (See OT10 and CGW15 KP-ABE schemes in Table 1.)

Lewko-Waters [18] first constructed adaptively-secure CP-ABE and KP-ABE schemes for span programs with allowing arbitrary reuse of attributes in a policy *without the above redundant multiple encoding technique*. While Lewko-Waters's (CP-)ABE scheme ([18] and subsequent work [2, 3] in Table 1) shows an interesting approach to allowing arbitrary reuse of attributes in a policy, the security is proven only based on *q-type assumptions* with $q$ the maximum number of key queries. However, the assumptions (and also the associated schemes) suffered a special attack which was presented by Cheon [10] at Eurocrypt 2006. Consequently, it is very desirable that the $q$-type assumption should be replaced by a *static* (non-$q$ type) assumption with keeping compact ciphertexts.

Moreover, we note that there exist *no multi-use* CP-ABE scheme with short, i.e., non-redundant, secret keys *even in the selective security setting* from *a static assumption* (Table 2). Now, an important open question is:

> *Is there an adaptively secure KP-(resp. CP-)ABE scheme from a static (standard) assumption whose ciphertext (resp. secret key) size does not depend on the maximum attribute-multiplicity k of available policies ?*

This work makes a significant step for addressing the problem.

Recently, non-interactive verifiable computation (NI-VC) for ensuring correct delegated computation of a (boolean) function $f$ has been extensively studied, and several approaches exist. One interesting approach is a generic conversion

**Table 1.** Comparison with the existing pairing-based multi-use KP-ABE schemes, where PK, SK, CT stand for public key, secret key, ciphertext, respectively, and $n', n, \ell, r, k$ represent the number of attributes in CT, the max of $n'$, the number of rows in access matrix in SK, the max of the number of columns in access matrix in SK, (the max of) the "attribute-multiplicity" of an access matrix in SK, respectively. The fourth row describes the warm-up scheme in Section 5.3.

| | Security | Assump. | PK size | SK size | CT size |
|---|---|---|---|---|---|
| GPSW06[14] | selective | DBDH | $O(n)\|\mathbb{G}\|$ | $O(\ell)\|\mathbb{G}\|$ | $O(n')\|\mathbb{G}\|$ |
| Tak14 [26] | semi-adaptive | DLIN | $O(n)\|\mathbb{G}\|$ | $O(\ell n)\|\mathbb{G}\|$ | $O(1)\|\mathbb{G}\|$ |
| (Warm-up) | | | | $O(\ell)\|\mathbb{G}\|$ | $O(n)\|\mathbb{G}\|$ |
| OT10[20] | adaptive | DLIN | $O(n)\|\mathbb{G}\|$ | $O(\ell)\|\mathbb{G}\|$ | $O(kn')\|\mathbb{G}\|$ |
| LW12 [18] | | $\ell$-Parallel BDHE ($+\alpha$) | $O(n)\|\mathbb{G}\|$ | $O(\ell)\|\mathbb{G}\|$ | $O(n')\|\mathbb{G}\|$ |
| Att15 [2, 3] | | EDHE3 & 4 parametrized by $n, \ell, r$ | $O(n)\|\mathbb{G}\|$ | $O(\ell n)\|\mathbb{G}\|$ | $O(1)\|\mathbb{G}\|$ |
| CGW15 [8] | | $s$-Lin for $\forall s$ | $O(n)\|\mathbb{G}\|$ for $s=2$ | $O(\ell)\|\mathbb{G}\|$ for $s=2$ | $O(kn')\|\mathbb{G}\|$ for $s=2$ |
| Proposed | adaptive | DLIN | $O(n+r)\|\mathbb{G}\|$ | $O(\ell)\|\mathbb{G}\|$ | $O(n+r)\|\mathbb{G}\|$ |

to NI-VC (in the pre-processing model) from KP-ABE [24, 9]. An important security requirement is soundness against a malicious server. So, the security should reflect the adversary's adaptive selection of the target function $f$. However, since all previous KP-ABEs have the above mentioned drawback, *no* NI-VC constructions achieve *adaptively secure communication-efficient (i.e., independent from the input variable multiplicity $k$)* NI-VC from a static (standard) assumption, where the input variable multiplicity $k$ is defined for each function $f$, e.g., $f = (X_1 \wedge X_2) \vee (X_1 \wedge X_3) \vee (X_2 \wedge X_4)$ has $k = 2$ as for KP-ABE. We address the following open question affirmatively.

> *Is there an adaptively secure NI-VC (with pre-processing) from a static (standard) assumption whose communication cost does not depend on the maximum input multiplicity $k$ of available functions ?*

### 1.2 Our Results

We obtain the following results.

- We propose an adaptively secure *multi-use* KP-ABE construction for boolean formulas over large universe attribute matching predicates *with short ciphertexts from the DLIN assumption* (in Section 5). The size of a ciphertext for attributes *does not (multiplicatively) depend on the attribute multiplicity $k$ in available access structures*, but has only an *additive* dependence on some size parameter $r$ of access structures. For comparison with existing ones, refer to Table 1.

**Table 2.** Comparison with the existing pairing-based multi-use CP-ABE schemes, where PK, SK, CT stand for public key, secret key, ciphertext, respectively, and $n', n, \ell, r, k$ represent the number of attributes in SK, the max of $n'$, the number of rows in access matrix in CT, the max of the number of columns in access matrix in CT, (the max of) the "attribute-multiplicity" of an access matrix in CT, respectively.

| | Security | Assump. | PK size | SK size | CT size |
|---|---|---|---|---|---|
| Wat11[28] Scheme 2 | selective | $\nu$-BDHE | $O(n)\|\mathbb{G}\|$ | $O(kn')\|\mathbb{G}\|$ | $O(\ell)\|\mathbb{G}\|$ |
| Wat11[28] Scheme 3 | | DBDH | $O(nr)\|\mathbb{G}\|$ | $O(kn'+r)\|\mathbb{G}\|$ | $O(\ell^2)\|\mathbb{G}\|$ |
| YAH15 [29] [1] | | parame-terized | $O((n\ell)^2\lambda)\|\mathbb{G}\|$ | $O((n\ell)^4\lambda^2)\|\mathbb{G}\|$ | $O(1)\|\mathbb{G}\|$ |
| OT10 [20] | adaptive | DLIN | $O(n)\|\mathbb{G}\|$ | $O(kn')\|\mathbb{G}\|$ | $O(\ell)\|\mathbb{G}\|$ |
| LW12 [18] | | $\ell$-Parallel BDHE $(+\alpha)$ | $O(n)\|\mathbb{G}\|$ | $O(n')\|\mathbb{G}\|$ | $O(\ell)\|\mathbb{G}\|$ |
| CGW15 [8] | | $s$-Lin for $\forall s$ | $O(n)\,\|\mathbb{G}\|$ for $s=2$ | $O(kn')\|\mathbb{G}\|$ for $s=2$ | $O(\ell)\|\mathbb{G}\|$ for $s=2$ |
| Proposed | adaptive | DLIN | $O(n+r)\|\mathbb{G}\|$ | $O(n+r)\|\mathbb{G}\|$ | $O(\ell)\|\mathbb{G}\|$ |

- We also propose an adaptively secure multi-use CP-ABE construction for the same access structures as the above KP-ABE with short keys from DLIN. The CP-ABE scheme is obtained from the above KP-ABE by the natural dual conversion, in particular, the keys *do not depend on the attribute multiplicity in available access structures*. We note that it is *the first multi-use CP-ABE construction with short keys from a static assumption even including the selective secure schemes* (Table 2). For the concrete scheme, see Appendix D.
- We obtain an *adaptively secure communication-efficient NI-VC (with preprocessing) from a static assumption, i.e., DLIN*, which is obtained by converting our KP-ABE to NI-VC. The communication cost does not depend on the maximum input multiplicity $k$, which addresses the above open problem. For comparison of our NI-VC and existing (pairing-based) ones, refer to Table 3 in Section 6.2.

We used two techniques, decoupling of linear secret sharing (LSS) into two (dual) components, i.e., span program matrix and randomness, and the partial randomization of LSS. A new sparse matrix machinery (Section 4) underlies them. The techniques can be extended naturally to arithmetic span programs (ASP), then, our results can be extended to ASP based ABE proposed by Ishai and Wee [15].

[1] Since $k \leq \ell$, the size of secret keys of the YAH15 scheme [29] is very large compared with others. Also, in [1], a *selective-secure* constant-size ciphertext, but, large secret keys CP-ABE scheme was proposed, recently.

### 1.3 Key Techniques

Our results are related to KP- and CP-ABEs, however, for simplicity, we mainly treat on KP-ABE, since it is a base scheme for NI-VC. According to a new framework introduced by Attrapadung, doubly selective security (i.e., selective and co-selective) leads to achieving adaptive one. Since selective security is easily obtained in KP-ABE, we should concentrate on achieving co-selectively secure KP-ABE below.

We extend the sparse matrix technique on dual pairing vector spaces (DPVS) developed in [22, 26]. Based on [26], we have DLIN-based, multi-use and *semi-adaptively* secure KP-ABE with short ciphertext size. We give the underlying scheme in Section 5.3 (as a warm-up), and extend it to our adaptive one. Here, access structure $\mathbb{S}$ is given by $\ell \times r$ matrix $M$ and each row $M_i \in \mathbb{F}_q^r$ of the matrix is associated to an attribute value by a map $\rho$, i.e., labeled with attributes $v_i := \rho(i)$. For a fixed special (all-one) vector $\vec{1}$, $\mathbb{S}$ accepts attribute set $\Gamma$ iff $\vec{1} \in \mathsf{span}\langle M_i \,|\, v_i \in \Gamma\rangle$. First, to achieve short ciphertexts in the underlying KP-ABE, attributes $\Gamma := \{x_j\}_{j=1,\ldots,n'}$ are encoded in an $n$-dimensional (with $n \geq n'+1$) vector $\vec{y} := (y_1, \ldots, y_n)$ such that $\sum_{j=0}^{n-1} y_{n-j} z^j = z^{n-1-n'} \prod_{j=1}^{n'} (z - x_j)$. Each (non-zero) attribute value $v_i$ (for $i = 1, \ldots, \ell$) associated with a row of access structure matrix $M$ (in $\mathbb{S}$) is encoded as $\vec{v}_i := (v_i^{n-1}, \ldots, v_i, 1)$, so $\vec{y} \cdot \vec{v}_i = v_i^{n-1-n'} \prod_{j=1}^{n'} (v_i - x_j)$, and the value of inner product is equal to zero if and only if $v_i = x_j$ for some $j$, i.e., $v_i \in \Gamma$. Here, the relation between $\mathbb{S}$ and $\Gamma$ is determined by the multiple inner product values $\vec{y} \cdot \vec{v}_i$ for one vector $\vec{y}$ which is equivalent to $\Gamma$. In previous work (e.g., [26]), a ciphertext vector element $\boldsymbol{c}_1$ is encoded with $\omega\vec{y}$ (for random $\omega$), and key vector elements $\boldsymbol{k}_i^*$ are encoded with $\vec{v}_i$ and shared secret values $M_i \cdot \vec{f}$ $(i = 1, \ldots, \ell)$ for a central secret $\vec{1} \cdot \vec{f}$ with uniformly random $\vec{f}$, respectively. We change the encoding method for our new proof method as indicated below.

**Basic Idea: Decoupling of LSS matrix from randomness** Secret keys in all previous KP-ABE schemes contain shared secret values $s_0 := \vec{1} \cdot \vec{f}$ and $s_i := M_i \cdot \vec{f}$, which means that randomness $\vec{f}$ is determined at the key generation phase. Namely, for pre-challenge queried keys (in simulation), the challenge $\vec{y}$ is not yet revealed to the challenger, i.e., simulator, at the query phase. We consider this is a main bottleneck for achieving compact ciphertext for multi-use of leaf attributes $v_i$ for rows $M_i$ of access matrix. For addressing the problem, we change an encoding method of LSS (Fig. 1). First, we decouple LSS encoding into LSS matrix and randomness, and randomness is encoded on the ciphertext side. Precisely, in the secret key, concatenated $\mathcal{V}_i := (\theta_i \vec{v}_i, \xi M_i) \in \mathbb{F}_q^{n+r}$ are encoded in the $i$-th component $\boldsymbol{k}_i^*$ for $i = 1, .., \ell$ with random $\theta_i, \xi$. We note that the key component $\boldsymbol{k}_i^*$ has no randomness for LSS (except for connecting randomness $\xi$), instead, LSS matrix $M := (M_i)_{i=1}^{\ell}$ is directly encoded in $\{\boldsymbol{k}_i^*\}$. In ciphertext, $\mathcal{Y} := (\omega\vec{y}, \vec{f}) \in \mathbb{F}_q^{n+r}$ is encoded. Hence, in decryption, inner-product values are

$$\mathcal{Y} \cdot \mathcal{V}_i = \omega\theta_i(\vec{y} \cdot \vec{v}_i) + \xi M_i \cdot \vec{f} = \omega\theta_i(\vec{y} \cdot \vec{v}_i) + \xi s_i \quad \text{for } i = 1, \ldots, \ell,$$

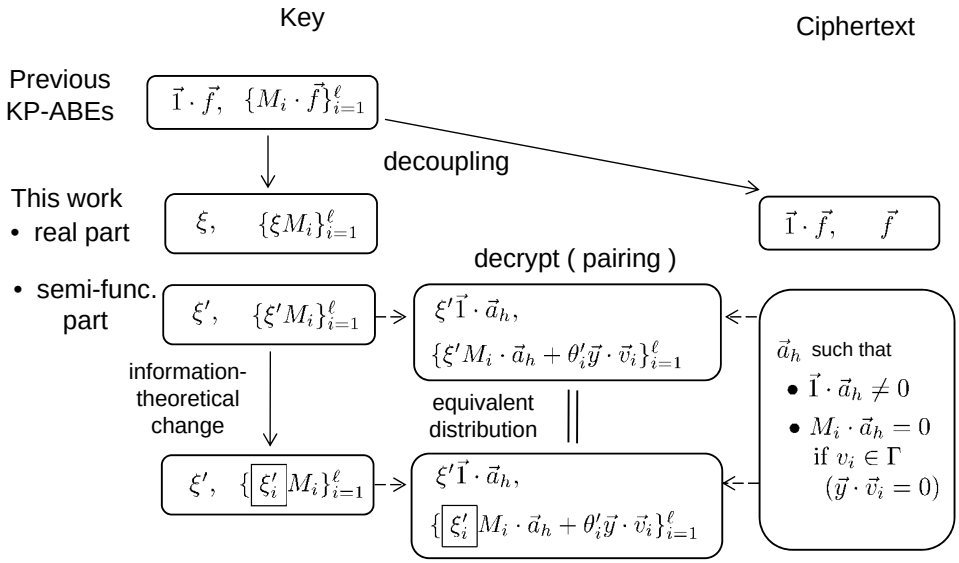


**Fig. 1.** Decoupling of LSS matrix from randomness and partial LSS randomization in semi-functional parts. Here, $(M=(M_i),\rho)$ is an access structure, uniformly random $\vec{f}\xleftarrow{\cup}\mathbb{F}_q^r, \xi, \xi', \xi_i', \theta_i'\xleftarrow{\cup}\mathbb{F}_q$, $\vec{y}:=(y_1,\ldots,y_n)$ such that $\sum_{j=0}^{n-1} y_{n-j}z^j = z^{n-1-n'}\prod_{j=1}^{n'}(z-x_j)$, and $\vec{v}_i:=(v_i^{n-1},\ldots,v_i,1)$ for $\rho(i)=v_i$.

therefore, if $\vec{y}\cdot\vec{v}_i=0$, secret share $\xi s_i$ for central secret $\xi s_0$ is obtained, and if $\vec{y}\cdot\vec{v}_i\neq 0$, $s_i$ is totally hidden from the decryptor since $\theta_i$ is freshly random.

**New Proof Techniques: Partial LSS randomization in simulation and new underlying lemma** At the top level of strategy of the security proof, we follow the dual system encryption methodology proposed by Waters [27]. The above change of encoding enables the simulator to simulate the randomness of LSS depending on both of the $h$-th queried access structure $\mathbb{S}:=(M,\rho)$ and attributes $\Gamma:=\{x_t\}$ (equivalently, vector $\vec{y}$). The simulated randomness $\vec{a}_h$ is *not fully random in* $\mathbb{F}_q^r$, but satisfies $M_i\cdot\vec{a}_h=0$ if $v_i\in\Gamma$ and $\vec{1}\cdot\vec{a}_h\neq 0$. Such a vector exists since $\Gamma$ does not satisfy $\mathbb{S}$, and it has been used for security in previous works, for example, in [14]. In ciphertext, the concatenated vector $\mathcal{Y}':=(\omega'\vec{y},\vec{a}_h)\in\mathbb{F}_q^{n+r}$ is encoded in the semi-functional space. And, in the semi-functional space of the $h$-th queried key, $\mathcal{V}_i':=(\theta_i'\vec{v}_i,\xi' M_i)\in\mathbb{F}_q^{n+r}$ are encoded in the $i$-th component $\boldsymbol{k}_i^*$ for $i=1,..,\ell$. Since $\mathcal{V}_i'$ is *independent of* $\Gamma$, it can be simulated for the pre-challenge key. Then,

$$\mathcal{Y}'\cdot\mathcal{V}_i'=\omega'\theta_i'(\vec{y}\cdot\vec{v}_i)+\xi' M_i\cdot\vec{a_h}=\begin{cases}0 & \text{if } \vec{y}\cdot\vec{v}_i=0,\\ \omega'\theta_i'(\vec{y}\cdot\vec{v}_i)+\xi' M_i\cdot\vec{a_h} & \text{if } \vec{y}\cdot\vec{v}_i\neq 0,\end{cases}$$

for $i=1,\ldots,\ell$. Here, if $\vec{y}\cdot\vec{v}_i\neq 0$, $\mathcal{Y}'\cdot\mathcal{V}_i'$ is uniformly random and independent from other variables since $\theta_i'$ are freshly random. Let $\mathcal{V}_i'':=(\theta_i'\vec{v}_i,\xi_i' M_i)\in\mathbb{F}_q^{n+r}$

with uniformly random $\xi'_i$ which are independent of each other for $i = 1, \ldots, \ell$.

$$\mathcal{Y}' \cdot \mathcal{V}''_i = \omega' \theta'_i (\vec{y} \cdot \vec{v}_i) + \xi'_i M_i \cdot \vec{a_h} = \begin{cases} 0 & \text{if } \vec{y} \cdot \vec{v}_i = 0, \\ \omega' \theta'_i (\vec{y} \cdot \vec{v}_i) + \xi'_i M_i \cdot \vec{a_h} & \text{if } \vec{y} \cdot \vec{v}_i \neq 0, \end{cases}$$

for $i = 1, \ldots, \ell$. Again, if $\vec{y} \cdot \vec{v}_i \neq 0$, $\mathcal{Y}' \cdot \mathcal{V}''_i$ is uniformly random and independent of other variables. That is, $\mathcal{Y}' \cdot \mathcal{V}'_i$ and $\mathcal{Y}' \cdot \mathcal{V}''_i$ are equivalently distributed. Therefore, we can conceptually change $\mathcal{V}'_i$ which contains variable $\xi'$ to $\mathcal{V}''_i$ with *no* $\xi'$ (Lemma 10) by using the pairwise independence lemma for *new sparse* matrices (Lemma 3). For the new underlying sparse matrix technique, refer to the beginning of Section 4. We stress that $\mathcal{V}''_i$ *are also independent of the challenge attributes* $\Gamma$, and then can be used in the pre-challenge key simulation.

In this way, we can sequentially eliminate the randomness $\xi'$ from all key components, $\boldsymbol{k}^*_i$ for $i = 1, .., \ell$, *except for* $\boldsymbol{k}^*_0$, and finally, $\xi'$ remains only in the central element $\boldsymbol{k}^*_0$, and the inner-product of the semi-functional parts of $\boldsymbol{k}^*_0$ and the corresponding ciphertext component is uniformly random value $\xi' \vec{1} \cdot \vec{a}_h$ since $\vec{1} \cdot \vec{a}_h \neq 0$. So, the proof proceeds successfully. Refer to Section 5.5 for the details.

### 1.4 Notations

When $A$ is a random variable or distribution, $y \xleftarrow{\mathsf{R}} A$ denotes that $y$ is randomly selected from $A$ according to its distribution. When $A$ is a set, $y \xleftarrow{\mathsf{U}} A$ denotes that $y$ is uniformly selected from $A$. We denote the finite field of order $q$ by $\mathbb{F}_q$, and $\mathbb{F}_q \setminus \{0\}$ by $\mathbb{F}_q^\times$. A vector symbol denotes a vector representation over $\mathbb{F}_q$, e.g., $\vec{y}$ denotes $(y_1, \ldots, y_n) \in \mathbb{F}_q^n$. For two vectors $\vec{y} = (y_1, \ldots, y_n)$ and $\vec{v} = (v_1, \ldots, v_n)$, $\vec{y} \cdot \vec{v}$ denotes the inner-product $\sum_{i=1}^n y_i v_i$. The vector $\vec{0}$ is abused as the zero vector in $\mathbb{F}_q^n$ for any $n$. $X^{\mathrm{T}}$ denotes the transpose of matrix $X$. A bold face letter denotes an element of vector space $\mathbb{V}$, e.g., $\boldsymbol{x} \in \mathbb{V}$. When $\boldsymbol{b}_i \in \mathbb{V}$ ($i = 1, \ldots, n$), $\mathsf{span}\langle \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \rangle \subseteq \mathbb{V}$ (resp. $\mathsf{span}\langle \vec{x}_1, \ldots, \vec{x}_n \rangle$) denotes the subspace generated by $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ (resp. $\vec{x}_1, \ldots, \vec{x}_n$). For bases $\mathbb{B} := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_N)$ and $\mathbb{B}^* := (\boldsymbol{b}^*_1, \ldots, \boldsymbol{b}^*_N)$, $(x_1, \ldots, x_N)_{\mathbb{B}} := \sum_{i=1}^N x_i \boldsymbol{b}_i$ and $(y_1, \ldots, y_N)_{\mathbb{B}^*} := \sum_{i=1}^N y_i \boldsymbol{b}^*_i$. $\vec{e}_j$ denotes the canonical basis vector $(\overbrace{0 \cdots 0}^{j-1}, 1, \overbrace{0 \cdots 0}^{n+r-j}) \in \mathbb{F}_q^{n+r}$ for positive integers $n$ and $r$. $GL(n, \mathbb{F}_q)$ denotes the general linear group of degree $n$ over $\mathbb{F}_q$.

## 2 Dual Pairing Vector Spaces (DPVS)

In this paper, for simplicity of description, we will present the proposed schemes on the symmetric version of dual pairing vector spaces (DPVS) [19] constructed using symmetric bilinear pairing groups given in Def. 1. Owing to the abstraction of DPVS, the presentation and the security proof of the proposed schemes are essentially the same as those on the asymmetric version of DPVS.

**Definition 1.** *"Symmetric bilinear pairing groups"* $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ *are a tuple of a prime* $q$*, cyclic additive group* $\mathbb{G}$ *and multiplicative group* $\mathbb{G}_T$ *of order* $q$*,* $G \neq 0 \in \mathbb{G}$*, and a polynomial-time computable nondegenerate bilinear pairing* $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ *i.e.,* $e(sG, tG) = e(G, G)^{st}$ *and* $e(G, G) \neq 1$*. Let* $\mathcal{G}_{\mathsf{bpg}}$ *be an*

*algorithm that takes input* $1^\lambda$ *and outputs a description of bilinear pairing groups* $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ *with security parameter* $\lambda$.

*"Dual pairing vector spaces (DPVS)" of dimension* $N$ *by a direct product of symmetric pairing groups* $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ *are given by prime* $q$, $N$*-dimensional vector space* $\mathbb{V} := \overbrace{\mathbb{G} \times \cdots \times \mathbb{G}}^{N}$ *over* $\mathbb{F}_q$, *cyclic group* $\mathbb{G}_T$ *of order* $q$, *and pairing* $e : \mathbb{V} \times \mathbb{V} \to \mathbb{G}_T$. *The pairing 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* $\boldsymbol{y} := (H_1, \ldots, H_N) \in \mathbb{V}$. *This is nondegenerate bilinear i.e.,* $e(s\boldsymbol{x}, t\boldsymbol{y}) = e(\boldsymbol{x}, \boldsymbol{y})^{st}$ *and if* $e(\boldsymbol{x}, \boldsymbol{y}) = 1$ *for all* $\boldsymbol{y} \in \mathbb{V}$, *then* $\boldsymbol{x} = \boldsymbol{0}$.

# 3 Definitions of KP-ABE

## 3.1 Span Programs and Access Structures

**Definition 2 (Span Programs [5]).** $\mathcal{U}$ *(*$\subset \{0,1\}^*$*) is a universe, a set of attributes, which is expressed by a value of attribute, i.e.,* $v \in \mathbb{F}_q^\times (:= \mathbb{F}_q \setminus \{0\})$. *A span program over* $\mathbb{F}_q$ *is a labeled matrix* $\mathbb{S} := (M, \rho)$ *where* $M$ *is a* $(\ell \times r)$ *matrix over* $\mathbb{F}_q$ *and* $\rho$ *is a labeling of the rows of* $M$ *by literals from* $\{v, v', \ldots\}$ *(every row is labeled by one literal), i.e.,* $\rho : \{1, \ldots, \ell\} \to \{v, v', \ldots\}$. *A span program accepts or rejects an input by the following criterion. Let* $\Gamma$ *be a set of attributes, i.e.,* $\Gamma := \{x_j\}_{1 \le j \le n'}$ *(*$x_j \in \mathbb{F}_q^\times$*). The span program* $\mathbb{S}$ *accepts* $\Gamma$ *if and only if* $\vec{1} \in \mathsf{span}\langle (M_i)_{\rho(i) = v_i \in \Gamma} \rangle$, *i.e., some linear combination of the rows* $(M_i)_{\rho(i) \in \Gamma}$ *gives the all one vector* $\vec{1}$.

No row $M_i$ $(i = 1, \ldots, \ell)$ of the matrix $M$ is $\vec{0}$. We now construct a secret-sharing scheme for a (monotone) span program.

**Definition 3.** *A secret-sharing scheme for span program* $\mathbb{S} := (M, \rho)$ *is:*

1. *Let* $M$ *be* $\ell \times r$ *matrix. Let column vector* $\vec{f} := (f_1, \ldots, f_r) \xleftarrow{\mathsf{U}} \mathbb{F}_q^r$. *Then,* $s_0 := \vec{1} \cdot \vec{f} = \sum_{k=1}^{r} f_k$ *is the secret to be shared, and* $\vec{s} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}}$ *is the* $\ell$ *shares of the secret* $s_0$ *and the share* $s_i$ *belongs to* $\rho(i)$.
2. *If span program* $\mathbb{S} := (M, \rho)$ *accepts* $\Gamma$, *i.e.,* $\vec{1} \in \mathsf{span}\langle (M_i)_{\rho(i) \in \Gamma} \rangle$, *there exist constants* $\{\alpha_i \in \mathbb{F}_q \mid i \in I\}$ *such that* $I \subseteq \{i \in \{1, \ldots, \ell\} \mid \rho(i) \in \Gamma\}$ *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 the matrix* $M$.

## 3.2 Key-Policy Attribute-Based Encryption (KP-ABE)

In key-policy attribute-based encryption (KP-ABE), encryption (resp. a secret key) is associated with attributes $\Gamma$ (resp. access structure $\mathbb{S}$). Relation $R$ for KP-ABE is defined as $R(\mathbb{S}, \Gamma) = 1$ iff access structure $\mathbb{S}$ accepts $\Gamma$.

**Definition 4 (Key-Policy Attribute-Based Encryption: KP-ABE).** *A key-policy attribute-based encryption scheme consists of probabilistic polynomial-time algorithms* $\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}$ *and* $\mathsf{Dec}$. *They are given as follows:*

$\mathsf{Setup}$ *takes as input security parameter* $1^\lambda$*, a bound* $n$ *on the number of attributes per ciphertext and a bound* $r$ *on the number of columns of an access matrix in a secret key. It outputs public parameters* $\mathsf{pk}$ *and master secret key* $\mathsf{sk}$.

$\mathsf{KeyGen}$ *takes as input public parameters* $\mathsf{pk}$*, master secret key* $\mathsf{sk}$*, and access structure* $\mathbb{S} := (M, \rho)$*. It outputs a corresponding secret key* $\mathsf{sk}_\mathbb{S}$.

$\mathsf{Enc}$ *takes as input public parameters* $\mathsf{pk}$*, message* $m$ *in some associated message space* $\mathsf{msg}$*, and a set of attributes,* $\Gamma := \{x_j\}_{j=1}^{n'}$*. It outputs a ciphertext* $\mathsf{ct}_\Gamma$.

$\mathsf{Dec}$ *takes as input public parameters* $\mathsf{pk}$*, secret key* $\mathsf{sk}_\mathbb{S}$ *for access structure* $\mathbb{S}$*, and ciphertext* $\mathsf{ct}_\Gamma$ *that was encrypted under a set of attributes* $\Gamma$*. It outputs either* $m' \in \mathsf{msg}$ *or the distinguished symbol* $\perp$.

A KP-ABE scheme should have the correctness: for all $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda, n, r)$, all access structures $\mathbb{S}$, all secret keys $\mathsf{sk}_\mathbb{S} \xleftarrow{\mathsf{R}} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathbb{S})$, all messages $m$, all attribute sets $\Gamma$, all ciphertexts $\mathsf{ct}_\Gamma \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m, \Gamma)$, it holds that $m = \mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_\mathbb{S}, \mathsf{ct}_\Gamma)$ if $\mathbb{S}$ accepts $\Gamma$. Otherwise, it holds with negligible probability.

**Remark 1** According to the explicit construction of span programs from boolean formulas (e.g., Appendix of [17]), while appending AND gate gets $r$ (and $\ell$) larger, appending OR gate gets only $\ell$ larger. Therefore, for example, our ABE can deal with any arbitrarily long $t$-DNF for a fixed $t$ like $(X_1 \wedge \ldots \wedge X_t) \vee \cdots \vee (Y_1 \wedge \ldots \wedge Y_t)$, where unbounded multi-use of attributes for $X_1, \ldots, Y_t$ is allowed.

**Definition 5 (Adaptive Security).** *The model for defining the adaptively payload-hiding security of KP-ABE under chosen plaintext attack is given by the following game:*

**Setup** *In the adaptive security, the challenger runs the setup,* $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda, n, r)$*, and gives public parameters* $\mathsf{pk}$ *to the adversary.*

**Phase 1** *The adversary is allowed to adaptively issue a polynomial number of key queries,* $\mathbb{S}$*, to the challenger. The challenger gives* $\mathsf{sk}_\mathbb{S} \xleftarrow{\mathsf{R}} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathbb{S})$ *to the adversary.*

**Challenge** *The adversary submits two messages* $m^{(0)}, m^{(1)}$*, and a challenge attribute set,* $\Gamma$*, provided that no* $\mathbb{S}$ *queried to the challenger in Phase 1 accepts* $\Gamma$*. The challenger flips a coin* $b \xleftarrow{\mathsf{U}} \{0, 1\}$*, and computes* $\mathsf{ct}_\Gamma^{(b)} \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m^{(b)}, \Gamma)$*. It gives* $\mathsf{ct}_\Gamma^{(b)}$ *to the adversary.*

**Phase 2** *Phase 1 is repeated with the restriction that no queried* $\mathbb{S}$ *accepts challenge* $\Gamma$.

**Guess** *The adversary outputs a guess* $b'$ *of* $b$*, and wins if* $b' = b$.

*The advantage of adversary* $\mathcal{A}$ *in the adaptive game is defined as* $\mathsf{Adv}_\mathcal{A}^{\mathsf{KP\text{-}ABE}}(\lambda) := \Pr[\mathcal{A} \text{ wins}] - 1/2$ *for any* $\lambda$*. A KP-ABE scheme is adaptively payload-hiding secure if all poly-time adversaries have at most a negligible advantage in the game.*

**Remark 2** The challenge $\Gamma$ is declared by the adversary just before **Phase 1** (resp. before **Setup**) in the semi-adaptive (resp. selective) game, and the corresponding security notions are defined in the similar manner as above.

## 4 Special Matrix Subgroups

Let $\tilde{n} := n + r$. Lemmas 1–3 are key lemmas for the security proof for our KP- and CP-ABE schemes.

We start by a motivational argument for introducing our new sparse matrix technique. Previous sparse matrices in DPVS [22, 26] are given by the form in Eq. (4), whose diagonal element except for the first one is the same denoted by $u$. For achieving our information theoretical change from $(\mathcal{Y}', \mathcal{V}'_i)$ to $(\mathcal{Y}', \mathcal{V}''_i)$ described in Section 1.3, we use one more randomness in diagonal elements, i.e., two random $u_1$ and $u_2$, as given in Eq. (1). More precisely, random $U \xleftarrow{\mathsf{U}} \mathcal{H}(n, r, \mathbb{F}_q)$ acts on $\mathbb{F}_q^{n+r} = \mathbb{F}_q^n \times \mathbb{F}_q^r$ by using different scalars $u_1$ and $u_2$ on the first $\mathbb{F}_q^n$ and the second $\mathbb{F}_q^r$ respectively. The new sparse matrix action is the key fact for proving Lemmas 3 and 10.

For positive integers $n$ and $r$, let

$$\mathcal{H}(n, r, \mathbb{F}_q) := \left\{ \begin{pmatrix} u'_1 & & & & & \\ u'_2 & u_1 & & & & \\ \vdots & & \ddots & & & \\ u'_n & & & u_1 & & \\ u'_{n+1} & & & & u_2 & \\ \vdots & & & & & \ddots \\ u'_{n+r} & & & & & & u_2 \end{pmatrix} \middle| \begin{array}{l} u_1, u_2, u'_l \in \mathbb{F}_q \\ \quad \text{for } l = 1, \ldots, n+r, \\ \text{a blank element} \\ \text{in the matrix} \\ \text{denotes } 0 \in \mathbb{F}_q \end{array} \right\}, \quad (1)$$

and $\mathcal{H}(n, r, \mathbb{F}_q)^\times := \mathcal{H}(n, r, \mathbb{F}_q) \cap GL(\tilde{n}, \mathbb{F}_q)$.

**Lemma 1.** *$\mathcal{H}(n, r, \mathbb{F}_q)^\times$ is a subgroup of $GL(\tilde{n}, \mathbb{F}_q)$, where $\tilde{n} := n + r$.*

Lemma 1 is directly verified from the definition of groups. □

Let

$$X_{i,j} := \begin{pmatrix} \mu'_{i,j,1} & & & & & \\ \mu'_{i,j,2} & \mu_{i,j,1} & & & & \\ \vdots & & \ddots & & & \\ \mu'_{i,j,n} & & & \mu_{i,j,1} & & \\ \mu'_{i,j,n+1} & & & & \mu_{i,j,2} & \\ \vdots & & & & & \ddots \\ \mu'_{i,j,n+r} & & & & & & \mu_{i,j,2} \end{pmatrix} \begin{array}{l} \in \mathcal{H}(n, r, \mathbb{F}_q) \\ \text{for } i, j = \\ 1, \ldots, 5 \end{array} \quad (2)$$

and using $X_{i,j}$, we define

$$\mathcal{L}(5, n, r, \mathbb{F}_q) := \left\{ X := \begin{pmatrix} X_{1,1} & \cdots & X_{1,5} \\ \vdots & & \vdots \\ X_{5,1} & \cdots & X_{5,5} \end{pmatrix} \middle| \begin{array}{l} X_{i,j} \\ \in \mathcal{H}(n, r, \mathbb{F}_q) \\ \text{for } i, j = \\ 1, \ldots, 5 \end{array} \right\} \bigcap GL(5\tilde{n}, \mathbb{F}_q). \quad (3)$$

**Lemma 2.** *$\mathcal{L}(5, n, r, \mathbb{F}_q)$ is a subgroup of $GL(5\tilde{n}, \mathbb{F}_q)$.*

Lemma 2 is given in a similar manner as Lemma 2 in the full version of [22].

Next is a generalization of Lemma 6 in [22].

**Lemma 3.** *Let* $\vec{e}_j := (0,\dots,0,\overset{j}{1},0,\dots,0) \in \mathbb{F}_q^{n+r}$. *For all* $\vec{v} = (v_1,\dots,v_n,0,\dots,0) \in \mathsf{span}\langle\vec{e}_1,..,\vec{e}_n\rangle \setminus \mathsf{span}\langle\vec{e}_1\rangle$, $\vec{\kappa} = (0,\dots,0,\kappa_1,\dots,\kappa_r) \in \mathsf{span}\langle\vec{e}_{n+1},..,\vec{e}_{n+r}\rangle$ *and* $\pi \in \mathbb{F}_q$, *let*

$$W_{\vec{v},\vec{\kappa},\pi} := \{(\vec{w},\vec{z}) \in (\mathsf{span}\langle\vec{e}_1,\vec{v},\vec{\kappa}\rangle \setminus \mathsf{span}\langle\vec{e}_1\rangle) \times (\mathbb{F}_q^{n+r} \setminus \mathsf{span}\langle\vec{e}_1\rangle^{\perp}) \mid \vec{w}\cdot\vec{z} = \pi\}.$$

*For all* $(\vec{v},\vec{\kappa},\vec{x}) \in (\mathsf{span}\langle\vec{e}_1,..,\vec{e}_n\rangle \setminus \mathsf{span}\langle\vec{e}_1\rangle) \times \mathsf{span}\langle\vec{e}_{n+1},..,\vec{e}_{n+r}\rangle \times \big(\mathbb{F}_q^{n+r} \setminus \mathsf{span}\langle\vec{e}_1\rangle^{\perp}\big)$, *for all* $(\vec{w},\vec{z}) \in W_{\vec{v},\vec{\kappa},((\vec{v}+\vec{\kappa})\cdot\vec{x})}$, *it holds that* $\Pr[\,(\vec{v}+\vec{\kappa})U = \vec{w} \wedge \vec{x}Z = \vec{z}\,] = 1/\sharp W_{\vec{v},\vec{\kappa},((\vec{v}+\vec{\kappa})\cdot\vec{x})}$, *where* $U \xleftarrow{\mathsf{U}} \mathcal{H}(n,r,\mathbb{F}_q)^{\times}$ *and* $Z := (U^{-1})^{\mathrm{T}}$.

For the proof of Lemma 3, we define a subset of $\mathcal{H}(n,r,\mathbb{F}_q)$,

$$\mathcal{H}(n+r,0,\mathbb{F}_q) = \left\{ \left( \begin{array}{ccc} u'_1 & & \\ u'_2 & u & \\ \vdots & & \ddots \\ u'_{n+r} & & u \end{array} \right) \middle| \begin{array}{l} u, u'_l \in \mathbb{F}_q \\ \quad \text{for } l = 1,..,n+r, \\ \text{a blank element} \\ \text{in the matrix} \\ \text{denotes } 0 \in \mathbb{F}_q \end{array} \right\} \subset \mathcal{H}(n,r,\mathbb{F}_q), \quad (4)$$

and $\mathcal{H}(n+r,0,\mathbb{F}_q)^{\times} := \mathcal{H}(n+r,0,\mathbb{F}_q) \cap GL(n+r,\mathbb{F}_q)$ $(\subset \mathcal{H}(n,r,\mathbb{F}_q)^{\times})$.

For the subgroup $\mathcal{H}(n+r,0,\mathbb{F}_q)^{\times}$, a sparse matrix version of pairwise independence lemma was obtained in the following form [22].

**Lemma 4 (Lemma 6 in [22], Adapted).** *Let* $\vec{e}_1 := (1,0,\dots,0) \in \mathbb{F}_q^{n+r}$. *For all* $\vec{v} \in \mathbb{F}_q^{n+r} \setminus \mathsf{span}\langle\vec{e}_1\rangle$ *and* $\pi \in \mathbb{F}_q$, *let*

$$W_{\vec{v},\pi} := \{(\vec{w},\vec{z}) \in (\mathsf{span}\langle\vec{e}_1,\vec{v}\rangle \setminus \mathsf{span}\langle\vec{e}_1\rangle) \times (\mathbb{F}_q^{n+r} \setminus \mathsf{span}\langle\vec{e}_1\rangle^{\perp}) \mid \vec{w}\cdot\vec{z} = \pi\}.$$

*For all* $(\vec{v},\vec{x}) \in \big(\mathbb{F}_q^{n+r} \setminus \mathsf{span}\langle\vec{e}_1\rangle\big) \times \big(\mathbb{F}_q^{n+r} \setminus \mathsf{span}\langle\vec{e}_1\rangle^{\perp}\big)$, *for all* $(\vec{w},\vec{z}) \in W_{\vec{v},(\vec{v}\cdot\vec{x})}$, *it holds that* $\Pr[\,\vec{v}U' = \vec{w} \wedge \vec{x}Z' = \vec{z}\,] = 1/\sharp W_{\vec{v},(\vec{v}\cdot\vec{x})}$, *where* $U' \xleftarrow{\mathsf{U}} \mathcal{H}(n+r,0,\mathbb{F}_q)^{\times}$ *and* $Z' := (U'^{-1})^{\mathrm{T}}$.

We also define a diagonal subgroup $\mathcal{K} := \{D_\gamma := \mathsf{diag}(\overbrace{1,\dots,1}^{n},\overbrace{\gamma,\dots,\gamma}^{r}) \,|\, \gamma \in \mathbb{F}_q^{\times}\} \subset \mathcal{H}(n,r,\mathbb{F}_q)^{\times}$.

**Lemma 5.** *For* $n \geq 2$, *there is a natural bijection: let* $\mathcal{K}\cdot\mathcal{H}(n+r,0,\mathbb{F}_q)^{\times} := \{D_\gamma \cdot H \,|\, D_\gamma \in \mathcal{K}, H \in \mathcal{H}(n+r,0,\mathbb{F}_q)^{\times}\}$, *then, it holds that* $\mathcal{H}(n,r,\mathbb{F}_q)^{\times} = \mathcal{K}\cdot\mathcal{H}(n+r,0,\mathbb{F}_q)^{\times}$.

More precisely, the above is a semi-direct product: $\mathcal{H}(n,r,\mathbb{F}_q)^{\times} = \mathcal{H}(n+r,0,\mathbb{F}_q)^{\times} \rtimes \mathcal{K}$. However, we do not need the fact.

*Proof of Lemma 5.* Let $\varphi : \mathcal{K} \times \mathcal{H}(n+r,0,\mathbb{F}_q)^{\times} \ni (D_\gamma, H) \mapsto D_\gamma \cdot H \in \mathcal{H}(n,r,\mathbb{F}_q)^{\times}$. Surjectivity of $\varphi$ is trivial. If $D_\gamma \cdot H = I_{n+r}$, then $u_1 = 1$ in $H$, and $\gamma = 1$, i.e., $D_\gamma = I_{n+r}$. Then, $H = I_{n+r}$. That is, $\varphi$ is injective. □

We can prove Lemma 3 by using the product structure given in Lemma 5.

*Proof of Lemma 3.* From Lemma 5, $U = D_\gamma \cdot U'$ is generated as $\gamma \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}$ and $U' \xleftarrow{\mathsf{U}} \mathcal{H}(n+r,0,\mathbb{F}_q)^{\times}$. Then, $Z = (U^{-1})^{\mathrm{T}} = D_{\gamma^{-1}} \cdot Z'$ where $Z' := (U'^{-1})^{\mathrm{T}}$. Let

$\vec{x} = \vec{x}_1 + \vec{x}_2$ where $\vec{x}_1 \in \mathsf{span}\langle \vec{e}_1, .., \vec{e}_n \rangle$ and $\vec{x}_2 \in \mathsf{span}\langle \vec{e}_{n+1}, .., \vec{e}_{n+r} \rangle$. We obtain $(\vec{v}+\vec{\kappa}) \cdot U = (\vec{v}+\vec{\kappa}) \cdot (D_\gamma \cdot U') = (\vec{v}+\gamma\vec{\kappa}) \cdot U'$ and $\vec{x} \cdot Z = (\vec{x}_1 + \vec{x}_2) \cdot D_{\gamma^{-1}} \cdot Z' = (\vec{x}_1 + \gamma^{-1}\vec{x}_2) \cdot Z'$. By applying Lemma 4 to $(\vec{v}' := \vec{v}+\gamma\vec{\kappa},\ \vec{x}' := \vec{x}_1 + \gamma^{-1}\vec{x}_2)$ and $(U', Z')$, we obtain our version of pairwise independence lemma, Lemma 3. □

# 5 Adaptively Secure Multi-Use KP-ABE Scheme with Short Ciphertexts

## 5.1 Key Ideas in Constructing the Proposed KP-ABE Scheme

We extend the techniques developed in [26]. In the paper, Takashima presented a semi-adaptively secure KP-ABE scheme with constant-size ciphertexts by using sparse matrix DPVS approach. An underlying construction of our proposed scheme is given in Section 5.3. The underlying scheme is considered as a dual form of the scheme in [26] since the sparse basis matrix is used in a dual manner. Hence, while [26] scheme has size $O(1)$ ciphertexts and size $O(\ell n)$ keys, the underlying one has size $O(n)$ ciphertexts and size $O(\ell)$ keys (Table 1 in Section 1.2). In particular, a key consists of $\ell$ vector elements which are compressed into $O(1)$ group elements and a ciphertext consists of 1 vector element of $5n$-dimension.

As mentioned in Introduction, the top level idea of our construction is the decoupling technique of LSS encoding. The underlying scheme has a usual encoding of LSS, i.e., encoding a central secret $s_0$ and shares $s_i$. Therefore, the comprehension of the construction idea of the underlying one is necessary for understanding our proposed one. In this section, we will explain key ideas of constructing the underlying and our KP-ABE schemes. First, we will show how size $O(n)$ ciphertexts and size $O(\ell)$ keys can be achieved in the underlying scheme, where the IPE scheme given in [22] is used as a building block. Here, we will use a simplified (or toy) version of the underlying KP-ABE scheme, for which the security is no more ensured in the standard model under the DLIN assumption.

A ciphertext in the simplified KP-ABE scheme consists of two vector elements, $(\boldsymbol{c}_0, \boldsymbol{c}_1) \in \mathbb{G}^5 \times \mathbb{G}^n$, and $c_T \in \mathbb{G}_T$. A secret key consists of $\ell + 1$ vector elements, $(\boldsymbol{k}_0^*, \boldsymbol{k}_1^*, \ldots, \boldsymbol{k}_\ell^*) \in \mathbb{G}^5 \times (\mathbb{G}^n)^\ell$ for access structure $\mathbb{S} := (M, \rho)$, where the number of rows of $M$ is $\ell$ and $\boldsymbol{k}_i^*$ with $i \geq 1$ corresponds to the $i$-th row. Therefore, to achieve shorter secret keys, we have to compress $\boldsymbol{k}_i^* \in \mathbb{G}^n$ to a constant size in $n$. We now employ a special form of basis generation matrix, $X := \begin{pmatrix} \mu_1' & & \\ \mu_2' & \mu & \\ \vdots & & \ddots \\ \mu_n' & & \mu \end{pmatrix} \in \mathcal{H}(n, 0, \mathbb{F}_q)$ of Eq. (1) in Section 4, where $\mu, \mu_1', \ldots, \mu_n' \xleftarrow{\cup} \mathbb{F}_q$ and a blank in the matrix denotes $0 \in \mathbb{F}_q$. The master secret key (DPVS basis) is $\mathbb{B}^* := \begin{pmatrix} \boldsymbol{b}_1^* \\ \vdots \\ \boldsymbol{b}_n^* \end{pmatrix} := \begin{pmatrix} \mu_1' G & & \\ \mu_2' G & \mu G & \\ \vdots & & \ddots \\ \mu_n' G & & \mu G \end{pmatrix}$. Let the $i$-th

component of a secret key associated with $\mathbb{S} := (M := (M_i)_{i=1}^{\ell},\ \rho)$ consists of $\boldsymbol{k}_i^* := (\theta_i v_i^{n-1} + s_i,\ \theta_i v_i^{n-2}, \ldots, \theta_i v_i, \theta_i)_{\mathbb{B}^*} = (\theta_i v_i^{n-1} + s_i)\boldsymbol{b}_1^* + \theta_i(v_i^{n-2}\boldsymbol{b}_2^* + \cdots + v_i\boldsymbol{b}_{n-1}^* + \boldsymbol{b}_n^*) = \left(\left(\theta_i(\sum_{j=1}^{n} v_i^{n-j}\mu_j') + s_i\mu_1'\right)G,\ v_i^{n-2}\theta_i\mu G, \ldots, \theta_i\mu G\right)$, where $v_i := \rho(i), \theta_i \xleftarrow{\cup} \mathbb{F}_q, \vec{f} \xleftarrow{\cup} \mathbb{F}_q^r$ and $s_i := M_i \cdot \vec{f}$. Then, $\boldsymbol{k}_i^*$ can be compressed to only *two* group elements $\left(K_{i,1}^* := \left(\theta_i(\sum_{j=1}^{n} v_i^{n-j}\mu_j') + s_i\mu_1'\right)G,\ K_{i,2}^* := \theta_i\mu G\right)$ as well as $v_i$, since $\boldsymbol{k}_i^*$ can be obtained by $(K_{i,1}^*, v_i^{n-2}K_{i,2}^*, \ldots, v_iK_{i,2}^*, K_{i,2}^*)$ (note that $v_i^j K_{i,2}^* = v_i^j\theta_i\mu G$ for $j = 0, \ldots, n-2$). That is, the $i$-th component of a secret key (excluding $v_i$) can be just two group elements, or the size is constant in $n$, then $(\boldsymbol{k}_i^*)_{i=0}^{\ell}$ can be compressed into size $O(\ell)$.

Let $\mathbb{B} := (\boldsymbol{b}_i)$ be the dual orthonormal basis of $\mathbb{B}^* := (\boldsymbol{b}_i^*)$, and $\mathbb{B}$ be the public key in the simplified KP-ABE scheme. We specify $(\boldsymbol{c}_0, \boldsymbol{k}_0^*, c_T)$ such that $e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) = g_T^{\zeta - \xi s_0}$ and $c_T := g_T^{\zeta} m \in \mathbb{G}_T$ with $s_0$ is a center secret of shares $\{s_i\}_{i=1,\ldots,\ell}$ associated with access structure $\mathbb{S}$, which are embedded into $\{\boldsymbol{k}_i^*\}_{i=1,\ldots,\ell}$ as indicated above. We also set a ciphertext for $\Gamma := \{x_1, \ldots, x_{n'}\}$ as $\boldsymbol{c}_1 := (\omega\vec{y})_{\mathbb{B}}$ where $\vec{y} := (y_1, \ldots, y_n)$ such that $\sum_{j=0}^{n-1} y_{n-j} z^j = z^{n-1-n'} \prod_{j=1}^{n'}(z - x_j)$, and $\omega \xleftarrow{\cup} \mathbb{F}_q$. From the dual orthonormality of $\mathbb{B}$ and $\mathbb{B}^*$, if $\mathbb{S}$ accepts $\Gamma$, there exists a system of coefficients $\{\alpha_i\}_{\rho(i)\in\Gamma}$ such that $e(\boldsymbol{c}_1, \boldsymbol{k}'^*) = g_T^{\xi s_0}$, where $\boldsymbol{k}'^* := \sum_{\rho(i)\in\Gamma} \alpha_i \boldsymbol{k}_i^*$. Hence, a decryptor can compute $g_T^{\xi s_0}$ if and only if $\mathbb{S}$ accepts $\Gamma$, i.e., can obtain plaintext $m$. We can extend the simplified KP-ABE to a *semi-adaptively* secure KP-ABE scheme under the DLIN assumption, which is shown in Section 5.3. The security proof is given in a similar manner to [26]. The provably secure scheme has the same asymptotic sizes of keys and ciphertexts, i.e., $O(\ell)$-sized keys and $O(n)$-sized ciphertexts.

Our goal is to construct an *adaptively* secure KP-ABE with a comparable asymptotic data sizes, i.e., $O(\ell)$-sized keys and $O(n + r)$-sized ciphertexts, from the underlying one. We use a decoupling technique of LSS matrix from randomness for achieving the goal. First, we enlarge the space from $O(n)$ to $O(n + r)$ dimension. As described in Fig. 1, a uniformly random vector $\vec{f} \in \mathbb{F}_q^r$ for LSS is encoded on the ciphertext component $\boldsymbol{c}_1$. In the simplified scheme, $\boldsymbol{c}_1 := (\omega\vec{y}, \vec{f})_{\mathbb{B}} \in \mathbb{G}^{n+r}$ where $\vec{y} \in \mathbb{F}_q^r$ is defined as above. For encoding each row $M_i$ of access matrix $M$ on $\boldsymbol{k}_i^*$, the above matrix $X$ is extended to a $(n+r)\times(n+r)$ matrix in $\mathcal{H}(n, r, \mathbb{F}_q)$ (Eq. (1)), then the master secret key is given by

$$\mathbb{B}^* := \begin{pmatrix} \boldsymbol{b}_1^* \\ \vdots \\ \\ \boldsymbol{b}_n^* \\ \boldsymbol{b}_{n+1}^* \\ \vdots \\ \boldsymbol{b}_{n+r}^* \end{pmatrix} := \begin{pmatrix} \mu_1'G & & & & & \\ \mu_2'G & \mu_1 G & & & & \\ \vdots & & \ddots & & & \\ \mu_n'G & & & \mu_1 G & & \\ \mu_{n+1}'G & & & & \mu_2 G & \\ \vdots & & & & & \ddots \\ \mu_{n+r}'G & & & & & & \mu_2 G \end{pmatrix} \text{ where } \mu_1, \mu_2, \mu_1', \ldots,$$

$\mu_{n+r}' \xleftarrow{\cup} \mathbb{F}_q$. Here, note that two independent diagonal elements $\mu_1, \mu_2$ are used for the first $n$-dimension and the second $r$-dimension. (Refer to the ar-

gument given in the beginning of Section 4.) Hence, $\boldsymbol{k}_i^*$ is given by $\boldsymbol{k}_i^* := (\theta_i \vec{v}_i, \xi M_i)_{\mathbb{B}^*}$. We note $\boldsymbol{k}_i^*$ is compressed to three group elements as before, i.e., $K_{i,1}^* := \big(\theta_i(\sum_{l=1}^n v_i^{n-l}\mu_l') + \xi(\sum_{l=1}^r M_{i,l}\mu_{n+l}')\big)\, G,\ K_{i,2}^* := \theta_i\mu_1 G,\ K_{i,3}^* := \xi\mu_2 G$ for $i = 1, .., \ell$, and the secret key size is $O(\ell)$. The pairing value of $\boldsymbol{c}_1$ and $\boldsymbol{k}_i^*$ is $e(\boldsymbol{c}_1, \boldsymbol{k}_i^*) = g_T^{\omega\theta_i \vec{y}\cdot\vec{v}_i + \xi M_i\cdot\vec{f}} = g_T^{\omega\theta_i \vec{y}\cdot\vec{v}_i + \xi s_i}$ where $s_i := M_i \cdot \vec{f}$. These values are equivalent to the previous underlying scheme. Therefore, the decryption algorithm is the same as before.

We then explain how our *full* KP-ABE scheme is constructed on the above-mentioned simplified KP-ABE scheme. The target of designing the full KP-ABE scheme is to achieve the adaptive security *under the DLIN assumption*. Here, we adopt and extend a strategy initiated in [20], in which the dual system encryption methodology is employed in a modular or hierarchical manner. That is, three top level assumptions, the security of Problems 1–3, are directly used in the dual system encryption methodology and the assumptions are reduced to a primitive assumption, the DLIN assumption.

To meet the requirements for applying to the dual system encryption methodology and reducing to the DLIN assumption, the underlying vector space is five times greater than that of the above-mentioned simplified scheme. For example, $\boldsymbol{k}_i^* := (\ \theta_i\vec{v}_i,\ \xi M_i,\ 0^{2n+2r},\ \psi_i\vec{v}_i,\ \eta_i M_i,\ 0^{n+r}\ )_{\mathbb{B}^*}$ for $\rho(i) = v_i$, $\boldsymbol{c}_1 = (\ \omega\vec{y},\ \vec{f},\ 0^{2n+2r},\ 0^{n+r},\ \vec{\varphi}_1\ )_{\mathbb{B}}$ with $\vec{\varphi}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n+r}$, and $X := \begin{pmatrix} X_{1,1} \cdots X_{1,5} \\ \vdots \quad\quad \vdots \\ X_{5,1} \cdots X_{5,5} \end{pmatrix} \in \mathcal{L}(5, n, r, \mathbb{F}_q)$ of Eq. (3) in Section 4, where each $X_{i,j}$ is of the form of $X \in \mathcal{H}(n, r, \mathbb{F}_q)$ in the simplified scheme. The vector space consists of four orthogonal subspaces, i.e., real encoding part, hidden part, secret key randomness part, and ciphertext randomness part. The simplified KP-ABE scheme corresponds to the first real encoding part.

A key fact in the security reduction is that $\mathcal{L}(5, n, r, \mathbb{F}_q)$ is a *subgroup* of $GL(5(n+r), \mathbb{F}_q)$ (Lemma 2), which enables a *random-self-reducibility* argument for reducing the intractability of Problems 1–3 to the DLIN assumption. For the reduction, see [22]. We employ a new simulation technique in dual system encryption using random vector $\vec{f}$ in $\boldsymbol{c}_1$. For the details, refer to the proof outline in Section 5.5.

### 5.2 Dual Orthonormal Basis Generator

We describe random dual orthonormal basis generator $\mathcal{G}_{\mathsf{ob}}^{\mathsf{KP}}$ below, which is used as a subroutine in the proposed KP-ABE scheme.

$$
\begin{aligned}
&\mathcal{G}_{\mathsf{ob}}^{\mathsf{KP}}(1^\lambda, 5, (n, r)) : \quad \mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda),\\
&\quad N_0 := 5,\ N_1 := 5(n+r),\\
&\quad \mathsf{param}_{\mathbb{V}_t} := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N_t, \mathsf{param}_{\mathbb{G}}) \ \text{ for } t = 0, 1,\\
&\quad \psi \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\ g_T := e(G, G)^\psi,\ \mathsf{param}_{(n,r)} := ((n, r),\ \{\mathsf{param}_{\mathbb{V}_t}\}_{t=0,1},\ g_T),
\end{aligned}
$$

$X_0 := (\chi_{0,i,j})_{i,j=1,\ldots,5} \xleftarrow{\mathsf{U}} GL(N_0, \mathbb{F}_q)$, $X_1 \xleftarrow{\mathsf{U}} \mathcal{L}(5, n, r, \mathbb{F}_q)$, hereafter,
$\{\mu_{i,j,\iota}, \mu'_{i,j,l}\}_{l=1,\ldots,n+r}^{i,j=1,\ldots,5;\iota=1,2}$ denotes non-zero entries of $X_1$ as in Eq. (2),
$\boldsymbol{b}^*_{0,i} := (\chi_{0,i,1}, .., \chi_{0,i,5})_{\mathbb{A}} = \sum_{j=1}^{5} \chi_{0,i,j} \boldsymbol{a}_j$ for $i = 1, .., 5$, $\mathbb{B}^*_0 := (\boldsymbol{b}^*_{0,1}, .., \boldsymbol{b}^*_{0,5})$,
$B^*_{i,j,\iota} := \mu_{i,j,\iota} G$, $B'^*_{i,j,l} := \mu'_{i,j,l} G$ for $i, j = 1, \ldots, 5; \iota = 1, 2; l = 1, \ldots, n+r$,
for $t = 0, 1$, $(\vartheta_{t,i,j})_{i,j=1,\ldots,N_t} := \psi \cdot (X_t^{\mathrm{T}})^{-1}$,
$\boldsymbol{b}_{t,i} := (\vartheta_{t,i,1}, .., \vartheta_{t,i,N_t})_{\mathbb{A}} = \sum_{j=1}^{N_t} \vartheta_{t,i,j} \boldsymbol{a}_j$ for $i = 1, .., N_t$, $\mathbb{B}_t := (\boldsymbol{b}_{t,1}, .., \boldsymbol{b}_{t,N_t})$,
return $(\mathsf{param}_{(n,r)}, \mathbb{B}_0, \mathbb{B}^*_0, \mathbb{B}_1, \{B^*_{i,j,\iota}, B'^*_{i,j,l}\}_{l=1,\ldots,n+r}^{i,j=1,\ldots,5;\iota=1,2})$.

**Remark 3** Let sparse block matrix

$$\left.\begin{array}{l} \begin{pmatrix} \boldsymbol{b}^*_{1,(i-1)(n+r)+1} \\ \vdots \\ \boldsymbol{b}^*_{1,i(n+r)} \end{pmatrix} := (X_{i,1} \cdot G \cdots X_{i,5} \cdot G) \quad \text{for } i = 1, \ldots, 5, \\ \text{and } \mathbb{B}^*_1 := (\boldsymbol{b}^*_{1,1}, \ldots, \boldsymbol{b}^*_{1,5(n+r)}), \end{array}\right\} \tag{5}$$

$\mathbb{B}_1$ is the dual orthonormal basis of $\mathbb{B}^*_1$, i.e., $e(\boldsymbol{b}_{1,i}, \boldsymbol{b}^*_{1,i}) = g_T$ and $e(\boldsymbol{b}_{1,i}, \boldsymbol{b}^*_{1,j}) = 1$ for $1 \leq i \neq j \leq 5(n+r)$.

### 5.3 Warm-Up: Underlying Semi-adaptively Secure Construction

As a warm-up, we describe a semi-adaptively secure KP-ABE scheme, which is a dual construction of [26] whose secret keys are compressed by using a sparse matrix while [26] scheme has compressed ciphertexts. Namely, we use the sparse matrix in a dual manner of [26]. We refer to Section 1.4 for notations on DPVS.

$\mathsf{Setup}(1^\lambda, n)$ : / * $N_0 := 5$, $N_1 := 5n$ * /
$(\mathsf{param}_n, \mathbb{B}_0, \mathbb{B}^*_0, \mathbb{B}_1, \{B^*_{i,j,\iota}, B'^*_{i,j,l}\}_{l=1,\ldots,n}^{i,j=1,\ldots,5;\,\iota=1,2}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{KP}}_{\mathsf{ob}}(1^\lambda, 5, (n, 0))$,
$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,2}, \boldsymbol{b}_{0,5})$, $\widehat{\mathbb{B}}^*_0 := (\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,2}, \boldsymbol{b}^*_{0,4})$,
$\widehat{\mathbb{B}}_1 := (\boldsymbol{b}_{1,1}, .., \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,4n+1}, .., \boldsymbol{b}_{1,5n})$,
return $\mathsf{pk} := (1^\lambda, \mathsf{param}_n, \{\widehat{\mathbb{B}}_t\}_{t=0,1})$, $\mathsf{sk} := (\widehat{\mathbb{B}}^*_0, \{B^*_{i,j,\iota}, B'^*_{i,j,l}\}_{\iota=1,2;\,l=1,\ldots,n}^{i=1,4;j=1,\ldots,5})$.
$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathbb{S} := (M, \rho))$ : $\vec{f} \xleftarrow{\mathsf{U}} \mathbb{F}_q^r$, $s_0 := \vec{1} \cdot \vec{f}$, $\eta_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$,
$\boldsymbol{k}^*_0 := (1, s_0, 0, \eta_0, 0)_{\mathbb{B}^*_0}$,
for $i = 1, \ldots, \ell$,
if $\rho(i) = v_i$, $\vec{v}_i := (v_i^{n-1}, \ldots, v_i, 1)$, $s_i := M_i \cdot \vec{f}$, $\theta_i, \psi_i, \eta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q$,
for $j = 1, \ldots, 5$, $K^*_{i,1,j} := \sum_{l=1}^{n} v_{i,l}(\theta_i B'^*_{1,j,l} + \psi_i B'^*_{5,j,l}) + s_i B'^*_{1,j,1} + \eta_i B'^*_{5,j,1}$,
$K^*_{i,2,j} := \theta_i B^*_{1,j,1} + \psi_i B^*_{5,j,1}$,
return $\mathsf{sk}_{\mathbb{S}} := (\mathbb{S}, \boldsymbol{k}^*_0, \{K^*_{i,1,j}, K^*_{i,2,j}\}_{i=1,\ldots,\ell;j=1,\ldots,5})$.
$\mathsf{Enc}(\mathsf{pk}, m, \Gamma := \{x_1, \ldots, x_{n'} \,|\, x_j \in \mathbb{F}_q^\times, n' \leq n-1\})$ :

$\vec{y} := (y_1, \ldots, y_n)$ such that $\sum_{j=0}^{n-1} y_{n-j} z^j = z^{n-1-n'} \prod_{j=1}^{n'} (z - x_j)$,

$\omega, \varphi_0, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{\varphi}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$, $\boldsymbol{c}_0 := (\zeta,\ \omega,\ 0,\ 0,\ \varphi_0)_{\mathbb{B}_0}$,

$\boldsymbol{c}_1 := (\overbrace{\quad \omega\vec{y}, \quad}^{n} \overbrace{\quad 0^{2n}, \quad}^{2n} \overbrace{\quad 0^n, \quad}^{n} \overbrace{\quad \vec{\varphi}_1 \quad}^{n})_{\mathbb{B}_1}$

$c_T := g_T^{\zeta} m$, $\mathsf{ct}_\Gamma := (\Gamma, \boldsymbol{c}_0, \boldsymbol{c}_1, c_T)$, return $\mathsf{ct}_\Gamma$.

$\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{\mathbb{S}} := (\mathbb{S},\ \boldsymbol{k}_0^*, \{K_{i,1,j}^*, K_{i,3,j}^*\}_{j=1,\ldots,5}^{i=1,\ldots,\ell}), \mathsf{ct}_\Gamma := (\Gamma, \boldsymbol{c}_0, \boldsymbol{c}_1, c_T))$ :

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma$, then compute $I$ and $\{\alpha_i\}_{i \in I}$ such that

$\vec{1} = \sum_{i \in I} \alpha_i M_i$, where $M_i$ is the $i$-th row of $M$, and

$I \subseteq \{i \in \{1, \ldots, \ell\} \quad | \quad [\rho(i) = v_i \ \wedge \ v_i \in \Gamma]\ \}$.

for $i \in I$, if $\rho(i) = v_i$, $\vec{v}_i := (v_{i,j})_n^{j=1} := (v_i^{n-1}, \ldots, v_i, 1)$,

$\boldsymbol{k}_i^* := (\overbrace{\quad K_{i,1,1}^*,\ v_{i,2}K_{i,2,1}^*, .., v_{i,n}K_{i,2,1}^* \quad}^{n} \cdots \overbrace{\quad K_{i,1,5}^*,\ v_{i,2}K_{i,2,5}^*, .., v_{i,n}K_{i,2,5}^* \quad}^{n})$,

that is, $\boldsymbol{k}_i^* := (\overbrace{\quad \theta_i\vec{v}_i + s_i\vec{e}_1, \quad}^{n} \overbrace{\quad 0^{2n}, \quad}^{2n} \overbrace{\quad \psi_i\vec{v}_i + \eta_i\vec{e}_1, \quad}^{n} \overbrace{\quad 0^n \quad}^{n})_{\mathbb{B}_1^*}$,

$\boldsymbol{k}'^* := \sum_{i \in I} \alpha_i \boldsymbol{k}_i^*$, $K := e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot e(\boldsymbol{c}_1, \boldsymbol{k}'^*)$, return $m' := c_T / K$.

**[Correctness]** If $\mathbb{S} := (M, \rho)$ accepts $\Gamma$, $K = e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot e(\boldsymbol{c}_1, \boldsymbol{k}'^*) = g_T^{-\omega s_0 + \zeta} g_T^{\omega \sum_{i \in I} \alpha_i s_i} = g_T^{\zeta}$ where $s_0 := \vec{1} \cdot \vec{f}$, $s_i := M_i \cdot \vec{f}$ for $i = 1, \ldots, \ell$.

We note that secret key $\mathsf{sk}_{\mathbb{S}}$ consists of $5\ell + 5$ group elements and ciphertext $\mathsf{ct}_\Gamma$ consists of $5n + 5$ group elements (and one $\mathbb{G}_T$ element).

The standard DLIN assumption is defined in Appendix A.

**Theorem 1.** *The proposed multi-use KP-ABE scheme is semi-adaptively payload-hiding against chosen plaintext attacks under the DLIN assumption.*

Theorem 1 is proven in a similar manner as in [26].

### 5.4 Proposed Adaptively Secure Construction

By decoupling LSS coefficients $s_i := M_i \cdot \vec{f} \in \mathbb{F}_q$ to $M_i \in \mathbb{F}_q^r$ in the key side and $\vec{f} \in \mathbb{F}_q^r$ in the ciphertext side of the underlying scheme, we obtain our proposed adaptively secure KP-ABE scheme. We refer to Section 1.4 for notations on DPVS.

$\mathsf{Setup}(1^\lambda,\ (n, r))$ : / * $N_0 := 5$, $N_1 := 5(n + r)$ * /

$(\mathsf{param}_{(n,r)}, \mathbb{B}_0, \mathbb{B}_0^*, \mathbb{B}_1, \{B_{i,j,\iota}^*, B_{i,j,l}'^*\}_{l=1,\ldots,n+r}^{i,j=1,\ldots,5;\ \iota=1,2}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{KP}}(1^\lambda, 5, (n, r))$,

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,2}, \boldsymbol{b}_{0,5})$, $\widehat{\mathbb{B}}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,2}^*, \boldsymbol{b}_{0,4}^*)$,

$\widehat{\mathbb{B}}_1 := (\boldsymbol{b}_{1,1}, .., \boldsymbol{b}_{1,n+r}, \boldsymbol{b}_{1,4(n+r)+1}, .., \boldsymbol{b}_{1,5(n+r)})$,

return $\mathsf{pk} := (1^\lambda, \mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1})$, $\mathsf{sk} := (\widehat{\mathbb{B}}_0^*, \{B_{i,j,\iota}^*, B_{i,j,l}'^*\}_{\iota=1,2;\ l=1,\ldots,n+r}^{i=1,4;j=1,\ldots,5})$.

$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \mathbb{S} := (M,\rho)) :\quad \xi, \eta_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q, \qquad \boldsymbol{k}_0^* := (1,\ \xi,\ 0,\ \eta_0,\ 0)_{\mathbb{B}_0^*},$

$\quad$ for $i = 1, \ldots, \ell,\ $ if $\rho(i) = v_i,\ \ \vec{v}_i := (v_i^{n-1}, \ldots, v_i, 1),\ \ \theta_i, \psi_i, \eta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q,$

$\quad\ $ for $j = 1, \ldots, 5,$

$\qquad K_{i,1,j}^* := \sum_{l=1}^{n} v_{i,l}(\theta_i B_{1,j,l}'^* + \psi_i B_{5,j,l}'^*) + \sum_{l=1}^{r} M_{i,l}(\xi B_{1,j,n+l}'^* + \eta_i B_{5,j,n+l}'^*),$

$\qquad K_{i,2,j}^* := \theta_i B_{1,j,1}^* + \psi_i B_{5,j,1}^*, \quad K_{i,3,j}^* := \xi B_{1,j,2}^* + \eta_i B_{5,j,2}^*,$

$\quad$ return $\mathsf{sk}_{\mathbb{S}} := (\mathbb{S},\ \boldsymbol{k}_0^*,\ \{K_{i,1,j}^*, K_{i,2,j}^*, K_{i,3,j}^*\}_{i=1,\ldots,\ell; j=1,\ldots,5}).$

$\mathsf{Enc}(\mathsf{pk},\ m,\ \Gamma := \{x_1, \ldots, x_{n'} \mid x_j \in \mathbb{F}_q^{\times}, n' \le n-1\}) :$

$\quad \vec{y} := (y_1, \ldots, y_n)$ such that $\sum_{j=0}^{n-1} y_{n-j} z^j = z^{n-1-n'} \prod_{j=1}^{n'} (z - x_j),$

$\quad \vec{f} \xleftarrow{\mathsf{U}} \mathbb{F}_q^r,\ \omega, \varphi_0, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \ \vec{\varphi}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n+r}, \quad \boldsymbol{c}_0 := (\zeta,\ \vec{1} \cdot \vec{f},\ 0,\ 0,\ \varphi_0)_{\mathbb{B}_0},$

$$\boldsymbol{c}_1 := (\overbrace{\quad \omega\vec{y},\ \vec{f},\quad}^{n+r}\ \overbrace{\quad 0^{2n+2r},\quad}^{2n+2r}\ \overbrace{\quad 0^{n+r},\quad}^{n+r}\ \overbrace{\quad \vec{\varphi}_1 \quad}^{n+r})_{\mathbb{B}_1}$$

$\quad c_T := g_T^{\zeta} m,\ \ \mathsf{ct}_{\Gamma} := (\Gamma, \boldsymbol{c}_0, \boldsymbol{c}_1, c_T), \quad$ return $\mathsf{ct}_{\Gamma}.$

$\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{\mathbb{S}} := (\mathbb{S},\ \boldsymbol{k}_0^*, \{K_{i,1,j}^*, K_{i,2,j}^*, K_{i,3,j}^*\}_{j=1,\ldots,5}^{i=1,\ldots,\ell}), \mathsf{ct}_{\Gamma} := (\Gamma, \boldsymbol{c}_0, \boldsymbol{c}_1, c_T)) :$

$\quad$ If $\mathbb{S} := (M, \rho)$ accepts $\Gamma$, then compute $I$ and $\{\alpha_i\}_{i \in I}$ such that

$\qquad \vec{1} = \sum_{i \in I} \alpha_i M_i,$ where $M_i$ is the $i$-th row of $M,$ and

$\qquad I \subseteq \{i \in \{1, \ldots, \ell\} \quad | \quad [\rho(i) = v_i \ \wedge\ v_i \in \Gamma]\ \}.$

$\quad$ for $i \in I, \quad$ if $\rho(i) = v_i,\ \ \vec{v}_i := (v_{i,j})_n^{j=1} := (v_i^{n-1}, \ldots, v_i, 1),$

$$\boldsymbol{k}_i^* := (\overbrace{\begin{array}{l} K_{i,1,1}^*,\ v_{i,2}K_{i,2,1}^*, .., v_{i,n}K_{i,2,1}^*,\ M_{i,1}K_{i,3,1}^*, .., M_{i,r}K_{i,3,1}^*,\ \cdots \\ K_{i,1,5}^*,\ v_{i,2}K_{i,2,5}^*, .., v_{i,n}K_{i,2,5}^*,\ M_{i,1}K_{i,3,5}^*, .., M_{i,r}K_{i,3,5}^* \end{array}}^{n+r},$$

$$\text{that is, } \boldsymbol{k}_i^* := (\overbrace{\quad \theta_i\vec{v}_i,\ \xi M_i,\quad}^{n+r}\ \overbrace{\quad 0^{2n+2r},\quad}^{2n+2r}\ \overbrace{\quad \psi_i\vec{v}_i,\ \eta_i M_i,\quad}^{n+r}\ \overbrace{\quad 0^{n+r}\quad}^{n+r})_{\mathbb{B}_1^*},$$

$\quad \boldsymbol{k}'^* := \sum_{i \in I} \alpha_i \boldsymbol{k}_i^*,\ \ K := e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot e(\boldsymbol{c}_1, \boldsymbol{k}'^*), \quad$ return $\ m' := c_T / K.$

**[Correctness]** If $\mathbb{S} := (M, \rho)$ accepts $\Gamma$, $K = e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot e(\boldsymbol{c}_1, \boldsymbol{k}'^*) = g_T^{-\xi s_0 + \zeta} g_T^{\xi \sum_{i \in I} \alpha_i s_i} = g_T^{\zeta}$ where $s_0 := \vec{1} \cdot \vec{f},\ s_i := M_i \cdot \vec{f}$ for $i = 1, \ldots, \ell$.

We note that secret key $\mathsf{sk}_{\mathbb{S}}$ consists of $5\ell + 5$ group elements and ciphertext $\mathsf{ct}_{\Gamma}$ consists of $5(n + r) + 5$ group elements (and one $\mathbb{G}_T$ element).

### 5.5 Security of the Proposed KP-ABE

The standard DLIN assumption is defined in Appendix A.

**Theorem 2.** *The proposed multi-use KP-ABE scheme is adaptively payload-hiding against chosen plaintext attacks under the DLIN assumption.*

Let $\nu_1$ (resp. $\nu_2$) be (the maximum of) the number of pre-challenge (resp. post-challenge) key queries, and $\nu := \nu_1 + \nu_2$ the total number of key queries. $\ell$ is the maximum of the number of rows in access matrices (of key queries).

**Outline of the Proof of Theorem 2** At the top level strategy of the security proof, the dual system encryption by Waters [27] is employed, where ciphertexts and secret keys have two forms, *normal* and *semi-functional*. The real system uses only normal ciphertexts and normal secret keys, and semi-functional ciphertexts and keys are used only in subsequent security games for the security proof.

To prove this theorem, we employ Game 0 (original adaptive security game) through Game 4. Games proceed as follows:

Game 0
for $h = 1, \ldots, \nu_1$, /* Game 1 sequence */
  Game 1-$h$-1 $\rightarrow$ Game 1-$h$-2
  for $p = 1, \ldots, \ell$, /* Game 1-$h$-3 sequence */
    Game 1-$h$-3-$p$-1 $\rightarrow$ Game 1-$h$-3-$p$-2 $\rightarrow$ Game 1-$h$-3-$p$-3
  Game 1-$h$-4
Game 2
for $h = \nu_1 + 1, \ldots, \nu (= \nu_1 + \nu_2)$, /* Game 3 sequence */
  Game 3-$h$-1
  for $p = 1, \ldots, \ell$, /* Game 3-$h$-2 sequence */
    Game 3-$h$-2-$p$-1 $\rightarrow$ Game 3-$h$-2-$p$-2 $\rightarrow$ Game 3-$h$-2-$p$-3
  Game 3-$h$-3 $\rightarrow$ Game 3-$h$-4
Game 4

The security games consist of two main parts, Game 1 sequence for pre-challenge keys and Game 3 sequence for post-challenge keys. Two types of semi-functional forms for keys and ciphertexts are used in the two sequences, called *selective-policy* semi-functional and *selective-attributes* semi-functional, respectively.

Normal forms are given by Eq. (8) for ciphertexts and Eqs. (6) and (7) for keys. Notable properties of these forms are: LSS matrix $M := (M_i)$ is directly encoded in keys $\{\boldsymbol{k}_i^*\}_{i=1}^{\ell}$ and randomness for the LSS, $\vec{f}$, is encoded in ciphertext $\boldsymbol{c}_1$.

Game 1 sequence (for pre-challenge keys) The Game 1 sequence is parametrized by the pre-challenge key index $h = 1, \ldots, \nu_1$.

The simulator is first given access structure $\mathbb{S} := (M, \rho)$ for the $h$-th key query from the adversary, then given attributes $\Gamma := \{x_t\}$ for the challenge query. The key task of the simulator is to embed $\mathbb{S} := (M, \rho)$, i.e., encoded vector $\vec{v}_i$ and rows $M_i$ of $M$, into the challenge ciphertext appropriately. Since the policy $\mathbb{S}$ is first revealed to the simulator, we use selective-policy semi-functional keys and ciphertext in the sequence.

A selective-policy semi-functional ciphertext is given by Eq. (9) and selective-policy semi-functional key is given by Eqs. (10) and (7). Temporary form keys are given by Eqs. (11)–(14). Notable properties of these forms are:

- A selective-policy semi-functional key given by Eqs. (10) and (7) and all temporary form keys in the Game 1 sequence, Eqs. (11)–(14), are all *independent from the challenge attribute set* $\Gamma$.

– (Partial) randomness for LSS matrix, $\vec{a}_h$, in the challenge ciphertext is selected depending on access structure $\mathbb{S} := (M, \rho)$ in the $h$-th queried key (and challenge attributes $\Gamma$) such that $\vec{a}_h \xleftarrow{\mathsf{U}} \{\vec{a}_h \in \mathbb{F}_q^r \,|\, M_i \cdot \vec{a}_h = 0 \text{ if } v_i := \rho(i) \in \Gamma \text{ for all } i = 1, \ldots, \ell, \text{ and } \vec{1} \cdot \vec{a}_h \neq 0\}$.
– Randomness $\xi'$ in Eq. (10) for $\boldsymbol{k}_0^*$ and $\{\xi'_p\}_\ell^{p=1}$ in Eqs. (13) and (14) for $\{\boldsymbol{k}_p^*\}$ are independently and uniformly distributed in $\mathbb{F}_q$. Moreover, the variable $\xi'$ is independent from all the other variables, and this is the goal of the Game 1 sequence.

Game 3 sequence (for post-challenge keys) The Game 3 sequence is parametrized by the post-challenge key index $h = \nu_1 + 1, \ldots, \nu$.

The simulator is first given attributes $\Gamma := \{x_t\}$ for the challenge query from the adversary, then given access structure $\mathbb{S} := (M, \rho)$ for the $h$-th key query. The key task of the simulator is to embed $\Gamma := \{x_t\}$, i.e., encoded vector $\vec{y}$, into the reply to the $h$-th key query, appropriately. Since the attributes $\Gamma$ are first revealed to the simulator, we use selective-attributes semi-functional keys and ciphertext in the sequence.

A selective-attributes semi-functional ciphertext is given by Eqs. (15) and (16), and selective-attributes semi-functional key is given by Eqs. (19) and (7). Temporary form ciphertext is given by Eq. (18). Notable properties of these forms are:

– A selective-attributes semi-functional cipheretxt given by Eqs. (15) and (16) and the temporal form cipheretxt, Eq. (18), are all *independent from the $h$-th (and all) queried key policy* $\mathbb{S}$.
– Only key components $\boldsymbol{k}_p^*$ in the $h$-th queried key with $v_p := \rho(p) \notin \Gamma$ are additionally randomized by using a new $\theta''_p \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (in Game 3-$h$-2-$p$-3), which is determined by the $h$-th access structure $\mathbb{S}$ and challenge attributes $\Gamma$.
– Uniformly distributed randomness $\xi'' \in \mathbb{F}_q$ in Eq. (19) for $\boldsymbol{k}_0^*$ is independent from all the other variables, and this is the goal of the Game 3 sequence.

In Game 4, the challenge ciphertext is changed to non-functional form, component $c_T$ is independently distributed from other components $(\boldsymbol{c}_0, \boldsymbol{c}_1)$. In the final game, the advantage of the adversary is zero. As usual, we prove that the advantage gaps between neighboring games are negligible, using computational problems, Problems 1–3 and information-theoretical game changes. We have shown that the intractability of (complicated) Problems 1–3 is reduced to that of the DLIN Problem through several intermediate steps, or intermediate problems, as in [20, 22, 26].

**Proof of Theorem 2** To prove Theorem 2, we consider the following $(3\ell + 3)(\nu_1 + \nu_2) + 3$ 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 game from the previous game.

**Game 0 :** Original game. That is, the reply to a key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ is:

$$\boldsymbol{k}_0^* := (\ 1,\ \xi,\ \boxed{0},\ \eta_0,\ 0\ )_{\mathbb{B}_0^*}, \tag{6}$$

$$\text{for } i = 1, \ldots, \ell,\ \ \boldsymbol{k}_i^* := (\ \theta_i \vec{v}_i,\ \xi M_i,\ \boxed{0^{2n+2r}},\ \psi_i \vec{v}_i,\ \eta_i M_i,\ 0^{n+r}\ )_{\mathbb{B}_1^*}, \tag{7}$$

where $\vec{v}_i := (v_i^{n-1}, .., v_i, 1)$ if $\rho(i) = v_i$, $\xi, \eta_0, \eta_i, \theta_i, \psi_i \xleftarrow{\cup} \mathbb{F}_q$. The challenge ciphertext for plaintexts $(m^{(0)}, m^{(1)})$ and $\Gamma := \{x_1, .., x_{n'} \,|\, x_j \in \mathbb{F}_q^\times, n' \leq n-1\}$ is:

$$\left.\begin{aligned} \boldsymbol{c}_0 &:= (\ \boxed{\zeta},\ \vec{1} \cdot \vec{f},\ \boxed{0},\ 0,\ \varphi_0\ )_{\mathbb{B}_0}, \quad c_T := g_T^{\zeta} m^{(b)}, \\ \boldsymbol{c}_1 &:= (\ \omega \vec{y},\ \vec{f},\ \boxed{0^{2n+2r}},\ 0^{n+r},\ \vec{\varphi}_1\ )_{\mathbb{B}_1}, \end{aligned}\right\} \tag{8}$$

where $\vec{y} := (y_1, \ldots, y_n)$ such that $\sum_{j=0}^{n-1} y_{n-j} z^j = z^{n-1-n'} \prod_{j=1}^{n'} (z - x_j)$, and $b \xleftarrow{\cup} \{0,1\}; \zeta, \omega, \varphi_0 \xleftarrow{\cup} \mathbb{F}_q, \vec{f} \xleftarrow{\cup} \mathbb{F}_q^r, \vec{\varphi}_1 \xleftarrow{\cup} \mathbb{F}_q^{n+r}$.

**Game 1-$h$-1 $(h = 1, \ldots, \nu_1)$ :** Game 1-0-4 is Game 0. Same as Game 1-$(h-1)$-4 except that $\boldsymbol{c}_0$ and $\boldsymbol{c}_1$ in the challenge ciphertexts for $\Gamma := \{x_t\}$ are

$$\left.\begin{aligned} \boldsymbol{c}_0 &:= (\ \zeta,\ \vec{1} \cdot \vec{f},\ \boxed{\vec{1} \cdot \vec{a}_h},\ 0,\ \varphi_0)_{\mathbb{B}_0}, \\ \boldsymbol{c}_1 &:= (\ \omega \vec{y},\ \vec{f},\ \boxed{\omega' \vec{y},\ \vec{a}_h,\ \omega' \vec{y},\ \vec{a}_h},\ 0^{n+r},\ 0^{n+r},\ \vec{\varphi}_1)_{\mathbb{B}_1}, \end{aligned}\right\} \tag{9}$$

where $\omega' \xleftarrow{\cup} \mathbb{F}_q$, the $h$-th key query is for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ and $\vec{a}_h \xleftarrow{\cup} \{\vec{a}_h \in \mathbb{F}_q^r \,|\, M_i \cdot \vec{a}_h = 0 \text{ if } \vec{v}_i \cdot \vec{y} = 0 \text{ for all } i = 1, \ldots, \ell, \text{ and } \vec{1} \cdot \vec{a}_h \neq 0\}$, and all the other variables are generated as in Game 1-$(h-1)$-4.

**Game 1-$h$-2 $(h = 1, \ldots, \nu_1)$ :** Game 1-$h$-2 is the same as Game 1-$h$-1 except all $\boldsymbol{k}_i^*$ in the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ are:

$$\boldsymbol{k}_0^* := (\ \zeta,\ \xi,\ \boxed{\xi'},\ \eta_0,\ 0\ )_{\mathbb{B}_0^*}, \tag{10}$$

$$\text{for } i = 1, .., \ell,\ \boldsymbol{k}_i^* := (\ \theta_i \vec{v}_i, \xi M_i,\ \boxed{\theta_i' \vec{v}_i,\ \xi' M_i},\ 0^{n+r}, \psi_i \vec{v}_i, \eta_i M_i, 0^{n+r})_{\mathbb{B}_1^*}, \tag{11}$$

where $\theta_i', \xi' \xleftarrow{\cup} \mathbb{F}_q$ and all the other variables are generated as in Game 1-$h$-1.

**Game 1-$h$-3-$p$-1 $(h = 1, \ldots, \nu_1; p = 1, \ldots, \ell)$ :** Game 1-$h$-3-0-3 is Game 1-$h$-2. Game 1-$h$-3-$p$-1 is the same as Game 1-$h$-3-$(p-1)$-3 except $\boldsymbol{k}_p^*$ in the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ is:

$$\boldsymbol{k}_p^* := (\ \theta_p \vec{v}_p,\ \xi M_p,\ \boxed{0^{n+r},\ \theta_p' \vec{v}_p,\ \xi' M_p},\ \psi_p \vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{B}_1^*}, \tag{12}$$

where all the variables are generated as in Game 1-$h$-3-$(p-1)$-3.

**Game 1-$h$-3-$p$-2 $(h = 1, \ldots, \nu_1; p = 1, \ldots, \ell)$ :** Game 1-$h$-3-$p$-2 is the same as Game 1-$h$-3-$p$-1 except $\boldsymbol{k}_p^*$ in the the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ is:

$$\boldsymbol{k}_p^* := (\ \theta_p \vec{v}_p,\ \xi M_p,\ 0^{n+r},\ \theta'_p \vec{v}_p,\ \boxed{\xi'_p M_p},\ \psi_p \vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{B}_1^*}, \tag{13}$$

where $\xi'_p \xleftarrow{\cup} \mathbb{F}_q$ and all the other variables are generated as in Game 1-$h$-3-$p$-1.
**Game 1-$h$-3-$p$-3 $(h = 1, \ldots, \nu_1; p = 1, \ldots, \ell)$ :** Game 1-$h$-3-$p$-3 is the same as Game 1-$h$-3-$p$-2 except $\boldsymbol{k}_p^*$ in the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ is:

$$\boldsymbol{k}_p^* := (\ \theta_p \vec{v}_p,\ \xi M_p,\ \boxed{\theta'_p \vec{v}_p,\ \xi'_p M_p,\ 0^{n+r}},\ \psi_p \vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{B}_1^*}, \tag{14}$$

where all the variables are generated as in Game 1-$h$-3-$p$-2.

Note that in Game 1-$h$-3-$\ell$-3, the uniformly distributed variable $\xi'$ in $\boldsymbol{k}_0^*$ (Eq. (10)) is *independent from all the other variables*.

**Game 1-$h$-4 $(h = 1, \ldots, \nu_1)$ :** Game 1-$h$-4 is the same as Game 1-$h$-3-$\ell$-3 except $\boldsymbol{k}_i$ $(i = 1, \ldots, \ell)$ in the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ are:

for $i = 1, \ldots, \ell$,

$$\boldsymbol{k}_i^* := (\ \theta_i \vec{v}_i,\ \xi M_i,\ \boxed{0^{n+r}},\ 0^{n+r},\ \psi_i \vec{v}_i,\ \eta_i M_i,\ 0^{n+r})_{\mathbb{B}_1^*}, \quad (= \text{Eq. (7)})$$

where all the variables are generated as in Game 1-$h$-3-$\ell$-3.
**Game 2 :** Game 2 is the same as Game 1-$\nu_1$-4 except the challenge ciphertext is:

$$\boldsymbol{c}_0 := (\ \zeta,\ \vec{1} \cdot \vec{f},\ \boxed{\vec{1} \cdot \vec{f'}},\ 0,\ \varphi_0\ )_{\mathbb{B}_0}, \tag{15}$$

$$\boldsymbol{c}_1 := (\ \omega \vec{y},\ \vec{f},\ \omega' \vec{y},\ \boxed{\vec{f'}},\ \omega' \vec{y},\ \boxed{\vec{f'}},\ 0^{n+r},\ \vec{\varphi}_1\ )_{\mathbb{B}_1}, \tag{16}$$

where $\vec{f'} \xleftarrow{\cup} \mathbb{F}_q^r$ and all the other variables are generated as in Game 1-$\nu_1$-4.
**Game 3-$h$-1 $(h = \nu_1 + 1, \ldots, \nu)$ :** Game 3-$\nu_1$-4 is Game 2. Game 3-$h$-1 is the same as Game 3-$(h-1)$-4 except that all the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ is:

$$\boldsymbol{k}_0^* := (\ 1,\ \xi,\ \boxed{\xi'},\ \eta_0,\ 0\ )_{\mathbb{B}_0^*},$$

$$\text{for } i = 1, .., \ell,\ \boldsymbol{k}_i^* := (\ \theta_i \vec{v}_i, \xi M_i,\ \boxed{\theta'_i \vec{v}_i,\ \xi' M_i}, 0^{n+r}, \psi_i \vec{v}_i, \eta_i M_i, 0^{n+r})_{\mathbb{B}_1^*},$$

where $\xi', \theta'_i \xleftarrow{\cup} \mathbb{F}_q$, and all the other variables are generated as in Game 3-$(h-1)$-4.
**Game 3-$h$-2-$p$-1 $(h = \nu_1 + 1, \ldots, \nu; p = 1, \ldots, \ell)$ :** Game 3-$h$-2-0-3 is Game 3-$h$-1. Game 3-$h$-2-$p$-1 is the same as Game 3-$h$-2-$(p-1)$-3 except $\boldsymbol{k}_p^*$ in the reply to the $h$-th key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ is:

$$\text{if } \vec{v}_p \cdot \vec{y} \neq 0,\ \boldsymbol{k}_p^* := (\ \theta_p \vec{v}_p, \xi M_p,\ \boxed{0^{n+r},\ \theta'_p \vec{v}_p,\ \xi' M_p},\ \psi_p \vec{v}_p, \eta_p M_p, 0^{n+r})_{\mathbb{B}_1^*},$$

where all the variables are generated as in Game 3-$h$-2-$(p-1)$-3.

**Game 3-$h$-2-$p$-2 $(h = \nu_1 + 1, \ldots, \nu; p = 1, \ldots, \ell)$ :** Game 3-$h$-2-$p$-2 is the same as Game 3-$h$-2-$p$-1 except $\boldsymbol{k}_p^*$ in the reply to the $h$-th key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ is:

if $\vec{v}_p \cdot \vec{y} \neq 0$,

$$\boldsymbol{k}_p^* := (\ \theta_p \vec{v}_p, \xi M_p,\ 0^{n+r},\ \boxed{\theta_p''},\ \theta_p' \vec{v}_p^{\geq 2},\ \xi' M_p,\ \psi_p \vec{v}_p, \eta_p M_p,\ 0^{n+r})_{\mathbb{B}_1^*},$$

where $\theta_p'' \xleftarrow{\cup} \mathbb{F}_q$, $\vec{v}_p^{\geq 2} := (v_p^{n-2}, \ldots, v_p, 1) \in \mathbb{F}_q^{n-1}$ is the last $n-1$ entries of $\vec{v}_p$ for $v_p = \rho(p)$, and all the other variables are generated as in Game 3-$h$-2-$p$-1.

**Game 3-$h$-2-$p$-3 $(h = \nu_1 + 1, \ldots, \nu; p = 1, \ldots, \ell)$ :** Game 3-$h$-2-$p$-3 is the same as Game 3-$h$-2-$p$-2 except $\boldsymbol{k}_p^*$ in the reply to the $h$-th key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ is:

if $\vec{v}_p \cdot \vec{y} \neq 0$,

$$\boldsymbol{k}_p^* := (\ \theta_p \vec{v}_p, \xi M_p,\ \boxed{\theta_p'',\ \theta_p' \vec{v}_p^{\geq 2},\ \xi' M_p,\ 0^{n+r}},\ \psi_p \vec{v}_p, \eta_p M_p,\ 0^{n+r})_{\mathbb{B}_1^*}, \quad (17)$$

where all the variables are generated as in Game 3-$h$-2-$p$-2.

**Game 3-$h$-3 $(h = \nu_1 + 1, \ldots, \nu)$ :** Game 3-$h$-3 is the same as Game 3-$h$-2-$\ell$-3 except that $\boldsymbol{c}_1$ in the challenge ciphertext for $\Gamma := \{x_t\}$, and $(\boldsymbol{k}_i^*)_\ell^{i=0}$ in the reply to the $h$-th key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ are:

$$\boldsymbol{c}_1 := (\ \omega \vec{y},\ \vec{f},\ \boxed{\omega', 0^{n+r-1},\ \vec{z}},\ 0^{n+r},\ \vec{\varphi}_1\ )_{\mathbb{B}_1}, \quad (18)$$

$$\boldsymbol{k}_0^* := (\ 1,\ \xi,\ \boxed{\xi''},\ 0,\ \varphi_0\ )_{\mathbb{B}_0^*}, \quad (19)$$

for $i = 1, \ldots, \ell$, if $\vec{v}_i \cdot \vec{y} = 0$,

$$\boldsymbol{k}_i^* := (\theta_i \vec{v}_i, \xi M_i,\ \boxed{\tfrac{\xi'}{\omega'} M_i \cdot \vec{f}'},\ \theta_i' \vec{v}_i^{\geq 2},\ \xi' M_i, 0^{n+r}, \psi_i \vec{v}_i, \eta_i M_i, 0^{n+r})_{\mathbb{B}_1^*}, \quad (20)$$

where $\xi'' \xleftarrow{\cup} \mathbb{F}_q, \vec{z} \xleftarrow{\cup} \mathbb{F}_q^{n+r}$ and all the other variables are generated as in Game 3-$h$-2-$\ell$-3.

**Game 3-$h$-4 $(h = \nu_1 + 1, \ldots, \nu)$ :** Game 3-$h$-4 is the same as Game 3-$h$-3 except that $\boldsymbol{c}_1$ in the challenge ciphertext for $\Gamma := \{x_t\}$, and $(\boldsymbol{k}_i^*)_{i=1}^\ell$ in the reply to the $h$-th key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ are:

$$\boldsymbol{c}_1 := (\ \omega \vec{y},\ \vec{f},\ \boxed{\omega' \vec{y},\ \vec{f}',\ \omega' \vec{y},\ \vec{f}'},\ 0^{n+r},\ \vec{\varphi}_1\ )_{\mathbb{B}_1}, \qquad (= \text{Eq. (16)})$$

for $i = 1, \ldots, \ell$,

$$\boldsymbol{k}_i^* := (\ \theta_i \vec{v}_i, \xi M_i,\ \boxed{0^{n+r}},\ 0^{n+r},\ \psi_i \vec{v}_i, \eta_i M_i,\ 0^{n+r}\ )_{\mathbb{B}_1^*}, \quad (= \text{Eq. (7)})$$

where all the variables are generated as in Game 3-$h$-3.

**Game 4 :** Game 4 is the same as Game 3-$\nu$-4 except that $\boldsymbol{c}_0$ in the challenge ciphertext for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix is:

$$\boldsymbol{c}_0 := (\ \boxed{\zeta'},\ \vec{1} \cdot \vec{f},\ \vec{1} \cdot \vec{f}',\ \eta_0,\ 0)_{\mathbb{B}_0},$$

where $\zeta' \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (i.e., independent from all the other variables, in particular, from $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$), and all the other variables are generated as in Game 3-$\nu$-4.

We show lemmas that evaluate the gaps between pairs of the advantages of neighboring games. This completes the proof of Theorem 2. □

**Lemmas** We will show lemmas for evaluating advantage gaps between neighboring games. Intermediate problems, Problems 1–3, whose intractability is reduced to that of DLIN (Lemmas 22–24), are used below. Problem 1 (resp. 2) is a standard decisional subspace problem for ciphertexts (resp. keys) side [20] and Problem 3 swaps coefficients in the $2\tilde{n}$-dimensional semi-functional space (i.e., Problem 2 in [26]). All the problems are given in Appendix B.

**Lemma 6.** *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_1$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1\text{-}1\text{-}1)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}_1}(\lambda)$.*

*Proof.* Lemma 6 is proven in a similar manner to Lemma 4 in [20] by using a Problem 1 instance. In Game 0, all the queried keys are normal. As in a usual dual system encryption proof, we can transform a normal ciphertext to a semi-functional form Eq. (9) by using Problem 1. □

**Lemma 7.** *For any adversary $\mathcal{A}$, for any security parameter $\lambda$, $|\mathsf{Adv}^{(1\text{-}(h-1)\text{-}3\text{-}\ell\text{-}3)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1\text{-}h\text{-}1)}_{\mathcal{A}}(\lambda)| \leq \epsilon(\lambda)$, where $\epsilon(\lambda)$ is a negligible function.*

*Proof.* In Game 1-$(h-1)$-3-$\ell$-3, semi-functional parts of all key components $\boldsymbol{k}^*_i$ for $i \geq 1$ are zero. Therefore, the semi-functional part of the challenge cipheretxt $\boldsymbol{c}_1$ can be conceptually changed to any vector except for negligible probability. Therefore, we obtain $\boldsymbol{c}_1$ as in Eq. (9). □

**Lemma 8.** *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_2$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}^{(1\text{-}h\text{-}1)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1\text{-}h\text{-}2)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}_2}(\lambda)$.*

*Proof.* Lemma 8 is proven in a similar manner to Lemma 5 in [20] by using a Problem 2 instance. □

**Lemma 9.** *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_3$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}^{(1\text{-}h\text{-}3\text{-}(p-1)\text{-}3)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1\text{-}h\text{-}3\text{-}p\text{-}1)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{\mathsf{P3}}_{\mathcal{B}_3}(\lambda)$.*

*Proof.* Lemma 9 is proven in a similar manner to Lemma 8 in [26] by using a Problem 3 instance. Problem 3 is used for swapping coefficient vectors of key $\boldsymbol{k}^*_p$ in the first block in the semi-functional part to the second block. Therefore, by using Problem 3, we can change $\boldsymbol{k}^*_p$ in Eq. (11) to that in Eq. (12). □

**Lemma 10.** *For any adversary $\mathcal{A}$, for any security parameter $\lambda$, $|\mathsf{Adv}^{(1\text{-}h\text{-}3\text{-}p\text{-}1)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1\text{-}h\text{-}3\text{-}p\text{-}2)}_{\mathcal{A}}(\lambda)| \leq \epsilon(\lambda)$, where $\epsilon(\lambda)$ is a negligible function.*

*Proof.* To prove Lemma 10, we will show distribution $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \mathsf{ct}_{\Gamma})$ in Games 1-$h$-3-$p$-1 and 1-$h$-3-$p$-2 are equivalent. For that purpose, we define an intermediate game, Game 1-$h$-3-$p$-1', as

**Game 1-$h$-3-$p$-1' ($h = 1, \ldots, \nu_1; p = 1, \ldots, \ell$) :** Game 1-$h$-3-$p$-1' is the same as Game 1-$h$-3-$p$-1 except $\boldsymbol{k}_p^*$ in the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ and the challenge ciphertext $\boldsymbol{c}_1$ are:

$$\begin{aligned}
\boldsymbol{k}_p^* &:= (\ \theta_p \vec{v}_p,\ \xi M_p,\ 0^{n+r},\ \boxed{\vec{w}},\ \psi_p \vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{B}_1^*},\\
\boldsymbol{c}_1 &:= (\ \omega \vec{y},\ \vec{f},\ \omega' \vec{y},\ \vec{a}_h,\ \boxed{\vec{z}},\ 0^{n+r},\ \vec{\varphi}_1)_{\mathbb{B}_1},
\end{aligned}$$

where $\vec{v}_p' := (\vec{v}_p, 0^r), M_p' := (0^n, M_p)$ and if $\vec{y} \cdot \vec{v}_p = 0$, then $(\vec{w}, \vec{z}) \xleftarrow{\cup} W_0' := \{(\vec{w}, \vec{z}) \in \mathsf{span}\langle \vec{e}_1, \vec{v}_p', M_p' \rangle \times \mathbb{F}_q^{n+r} \mid \vec{w} \cdot \vec{z} = 0\}$, and if $\vec{y} \cdot \vec{v}_p \neq 0$, $(\vec{w}, \vec{z}) \xleftarrow{\cup} (\mathsf{span}\langle \vec{e}_1, \vec{v}_p', M_p' \rangle \times \mathbb{F}_q^{n+r}) \setminus W_0'$.

**Claim 1** *The distribution $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \mathsf{ct}_{\Gamma})$ in Game 1-h-3-p-1 and that in Game 1-h-3-p-1' are equivalent except with negligible probability.*

*Proof of Claim 1.* We will consider the distribution in Game 1-$h$-3-$p$-1. We define new dual orthonormal bases $(\mathbb{D}_1, \mathbb{D}_1^*)$ of $\mathbb{V}_1$. First, we generate matrix $U \xleftarrow{\cup} \mathcal{H}(n, r, \mathbb{F}_q) \cap GL(n + r, \mathbb{F}_q)$, and set $\tilde{n} := n + r$,

$$\begin{aligned}
&\begin{pmatrix} \boldsymbol{d}_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{d}_{1,3\tilde{n}} \end{pmatrix} := U^{-1} \cdot \begin{pmatrix} \boldsymbol{b}_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{b}_{1,3\tilde{n}} \end{pmatrix}, \begin{pmatrix} \boldsymbol{d}_{1,2\tilde{n}+1}^* \\ \vdots \\ \boldsymbol{d}_{1,3\tilde{n}}^* \end{pmatrix} := U^{\mathrm{T}} \cdot \begin{pmatrix} \boldsymbol{b}_{1,2\tilde{n}+1}^* \\ \vdots \\ \boldsymbol{b}_{1,3\tilde{n}}^* \end{pmatrix},\\
&\mathbb{D}_1 := (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,2\tilde{n}}, \boldsymbol{d}_{1,2\tilde{n}+1}, \ldots, \boldsymbol{d}_{1,3\tilde{n}}, \boldsymbol{b}_{1,3\tilde{n}+1}, \ldots, \boldsymbol{b}_{1,5\tilde{n}}),\\
&\mathbb{D}_1^* := (\boldsymbol{b}_{1,1}^*, \ldots, \boldsymbol{b}_{1,2\tilde{n}}^*, \boldsymbol{d}_{1,2\tilde{n}+1}^*, \ldots, \boldsymbol{d}_{1,3\tilde{n}}^*, \boldsymbol{b}_{1,3\tilde{n}+1}^*, \ldots, \boldsymbol{b}_{1,5\tilde{n}}^*).
\end{aligned}$$

Then, $\mathbb{D}_1$ and $\mathbb{D}_1^*$ are dual orthonormal bases. $\boldsymbol{k}_p^*$ in the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ and the challenge ciphertext $\boldsymbol{c}_1$ is expressed as

$$\begin{aligned}
\boldsymbol{k}_p^* &= (\ \theta_p \vec{v}_p,\ \xi M_p,\ 0^{n+r},\ \theta_p' \vec{v}_p,\ \xi' M_p,\ \psi_p \vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{B}_1^*}\\
&= (\ \theta_p \vec{v}_p,\ \xi M_p,\ 0^{n+r},\ (\theta_p' \vec{v}_p,\ \xi' M_p) \cdot Z,\ \psi_p \vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{D}_1^*}, \quad (21)\\
\boldsymbol{c}_1 &= (\ \omega \vec{y},\ \vec{f},\ \omega' \vec{y},\ \vec{a}_h,\ \omega' \vec{y},\ \vec{a}_h,\ 0^{n+r},\ \vec{\varphi}_1)_{\mathbb{B}_1}\\
&= (\ \omega \vec{y},\ \vec{f},\ \omega' \vec{y},\ \vec{a}_h,\ (\omega' \vec{y},\ \vec{a}_h) \cdot U,\ 0^{n+r},\ \vec{\varphi}_1)_{\mathbb{D}_1} \quad (22)
\end{aligned}$$

where $Z := (U^{-1})^{\mathrm{T}}$. From Lemma 3, the pair of coefficients $(\vec{w}, \vec{z}) := ((\omega' \vec{y},\ \vec{a}_h) \cdot U, (\theta_p' \vec{v}_p,\ \xi' M_p) \cdot Z)$ is uniformly distributed in

$$W_{\vec{v}_p', M_p', \pi} := \{(\vec{w}, \vec{z}) \in (\mathsf{span}\langle \vec{e}_1, \vec{v}_p', M_p' \rangle \backslash \mathsf{span}\langle \vec{e}_1 \rangle) \times (\mathbb{F}_q^{n+r} \backslash \mathsf{span}\langle \vec{e}_1 \rangle^{\perp}) \mid \vec{w} \cdot \vec{z} = \pi\}.$$

where $\pi := \omega' \theta_p' (\vec{v}_p \cdot \vec{y}) + \xi' M_p \cdot \vec{a}_h$, $\vec{v}_p' := (\vec{v}_p,\ 0^r)$ and $M_p' := (0^n,\ M_p)$. In particular, if $\vec{v}_p \cdot \vec{y} \neq 0$, then $\pi$ is independently and uniformly distributed since $\theta_p' \xleftarrow{\cup} \mathbb{F}_q$,

and $(\vec{w}, \vec{z})$ is independently and uniformly distributed in $(\mathsf{span}\langle \vec{e}_1, \vec{v}'_p, M'_p\rangle \times \mathbb{F}_q^{n+r}) \setminus W'_0$ (except with negligible probability). If $\vec{v}_p \cdot \vec{y} = 0$, then $M_p \cdot \vec{a}_h = 0$, and $(\vec{w}, \vec{z})$ is independently and uniformly distributed in $W'_0$.

When $1 \le i \ne p \le \ell$, the $i$-th component of the $h$-th queried key $\boldsymbol{k}^*_i$ is

$$\left.\begin{array}{lll} \text{if } i < p, & \boldsymbol{k}^*_i = (\ \theta_i \vec{v}_i,\ \xi M_i,\ \theta'_i \vec{v}_i,\ \xi'_i M_i,\ 0^{n+r},\ \ldots)_{\mathbb{B}^*_1} \\ & \quad = (\ \theta_i \vec{v}_i,\ \xi M_i,\ \theta'_i \vec{v}_i,\ \xi'_i M_i,\ 0^{n+r},\ \ldots)_{\mathbb{D}^*_1}, \\ \text{if } i > p, & \boldsymbol{k}^*_i = (\ \theta_i \vec{v}_i,\ \xi M_i,\ \theta'_i \vec{v}_i,\ \xi' M_i,\ 0^{n+r},\ \ldots)_{\mathbb{B}^*_1} \\ & \quad = (\ \theta_i \vec{v}_i,\ \xi M_i,\ \theta'_i \vec{v}_i,\ \xi' M_i,\ 0^{n+r},\ \ldots)_{\mathbb{D}^*_1}. \end{array}\right\} \tag{23}$$

In the light of the adversary's view, $(\mathbb{D}_1, \mathbb{D}^*_1)$ is consistent with public key $\mathsf{pk} := (1^\lambda, \mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1})$. Moreover, since the RHS of Eqs. (21), (22) and (23) are in the same forms in those in Game 1-$h$-3-$p$-1', namely, Game 1-$h$-3-$p$-1 can be conceptually changed to Game 1-$h$-3-$p$-1' except with negligible probability. □

**Claim 2** *The distribution* $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}^{(j)*}_{\mathbb{S}}\}_{j=1,\ldots,\nu}, \mathsf{ct}_\Gamma)$ *in Game 1-h-3-p-2 and that in Game 1-h-3-p-1' are equivalent except with negligible probability.*

*Proof of Claim 2.* Claim 2 can be proven in a similar manner to Claim 1. Namely, we show that Game 1-$h$-3-$p$-2 can be conceptually changed to Game 1-$h$-3-$p$-1' except with negligible probability. In the proof, we consider a pair of coefficients $(\vec{w}, \vec{z}) := ((\omega' \vec{y},\ \vec{a}_h) \cdot U, (\theta'_p \vec{v}_p,\ \xi'_p M_p) \cdot Z)$ where a new randomness $\xi'_p$ is used instead of $\xi'$ in Claim 1. Lemma 3 shows the pair is uniformly distributed in $W_{\vec{v}'_p, M'_p, \pi}$ with the inner product value $\pi := \omega' \theta'_p (\vec{v}_p \cdot \vec{y}) + \xi'_p M_p \cdot \vec{a}_h$. Then, the same technique in Claim 1 is used in the rest of the proof of Claim 2. □

From Claims 1 and 2, the distribution $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}^{(j)*}_{\mathbb{S}}\}_{j=1,\ldots,\nu}, \mathsf{ct}_\Gamma)$ in Game 1-$h$-3-$p$-1 and that in Game 1-$h$-3-$p$-2 are equivalent except with negligible probability. This completes the proof of Lemma 10. □

**Lemma 11.** *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_4$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}^{(1\text{-}h\text{-}3\text{-}p\text{-}2)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1\text{-}h\text{-}3\text{-}p\text{-}3)}_{\mathcal{A}}(\lambda)| \le \mathsf{Adv}^{\mathsf{P3}}_{\mathcal{B}_4}(\lambda)$.*

*Proof.* Lemma 11 is proven in a similar manner to Lemma 9 by using a Problem 3 instance. □

**Lemma 12.** *For any adversary $\mathcal{A}$, there exist probabilistic machines $\mathcal{B}_{5\text{-}1}, \ldots, \mathcal{B}_{5\text{-}3}$ whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}^{(1\text{-}h\text{-}3\text{-}\ell\text{-}3)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1\text{-}h\text{-}4)}_{\mathcal{A}}(\lambda)| \le \sum_{i=1}^{\ell} (\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}_{5\text{-}i\text{-}1}}(\lambda) + \mathsf{Adv}^{\mathsf{P3}}_{\mathcal{B}_{5\text{-}i\text{-}2}}(\lambda) + \mathsf{Adv}^{\mathsf{P3}}_{\mathcal{B}_{5\text{-}i\text{-}3}}(\lambda))$, where $\mathcal{B}_{5\text{-}i\text{-}l}(\cdot) := \mathcal{B}_{5\text{-}l}(i, \cdot)$.*

*Proof.* We can change Game 1-$h$-3-$\ell$-3 to 1-$h$-4 by tracing the reverse transformations from Game 1-$h$-3-$\ell$-3 to Game 1-$h$-1 with $\boldsymbol{k}^*_0$ unchanged. Therefore, by combining Lemmas 11–7 in a reverse order, we obtain Lemma 12. □

**Lemma 13.** *For any adversary $\mathcal{A}$, for any security parameter $\lambda$, $|\mathsf{Adv}_{\mathcal{A}}^{(1\text{-}\nu_1\text{-}4)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2)}(\lambda)| \leq \epsilon(\lambda)$, where $\epsilon(\lambda)$ is a negligible function.*

*Proof.* Lemma 13 is proven in a similar manner to Lemma 7. □

**Lemma 14.** *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_6$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}_{\mathcal{A}}^{(3\text{-}(h-1)\text{-}4)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}1)}(\lambda)| \leq \mathsf{Adv}_{\mathcal{B}_6}^{\mathsf{P2}}(\lambda)$.*

*Proof.* Lemma 14 is proven in a similar manner to Lemma 8 by using a Problem 2 instance. □

**Lemma 15.** *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_7$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}2\text{-}(p-1)\text{-}3)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}2\text{-}p\text{-}1)}(\lambda)| \leq \mathsf{Adv}_{\mathcal{B}_7}^{\mathsf{P3}}(\lambda)$.*

*Proof.* Lemma 15 is proven in a similar manner to Lemma 9 by using a Problem 3 instance. □

**Lemma 16.** *For any adversary $\mathcal{A}$, for any security parameter $\lambda$, $|\mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}2\text{-}p\text{-}1)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}2\text{-}p\text{-}2)}(\lambda)| \leq \epsilon(\lambda)$, where $\epsilon(\lambda)$ is a negligible function.*

Lemma 16 is proven in a similar manner to Lemma 10 by using Lemma 3. Full proof is given in Appendix C.1.

**Lemma 17.** *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_8$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}2\text{-}p\text{-}2)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}2\text{-}p\text{-}3)}(\lambda)| \leq \mathsf{Adv}_{\mathcal{B}_8}^{\mathsf{P3}}(\lambda)$.*

*Proof.* Lemma 17 is proven in a similar manner to Lemma 9 by using a Problem 3 instance. □

**Lemma 18.** *For any adversary $\mathcal{A}$, for any security parameter $\lambda$, $|\mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}2\text{-}\ell\text{-}3)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}3)}(\lambda)| \leq \epsilon(\lambda)$, where $\epsilon(\lambda)$ is a negligible function.*

Lemma 18 is proven in a similar manner to Lemma 9 in the full version of [21] by using the technique called "one-dimensional localization of inner-product values". Full proof is given in Appendix C.2.

**Lemma 19.** *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_{9\text{-}1}, \ldots, \mathcal{B}_{9\text{-}3}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter $\lambda$, $|\mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}3)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3\text{-}h\text{-}4)}(\lambda)| \leq \sum_{i=1}^{\ell}(\mathsf{Adv}_{\mathcal{B}_{9\text{-}i\text{-}1}}^{\mathsf{P2}}(\lambda) + \mathsf{Adv}_{\mathcal{B}_{9\text{-}i\text{-}2}}^{\mathsf{P3}}(\lambda) + \mathsf{Adv}_{\mathcal{B}_{9\text{-}i\text{-}3}}^{\mathsf{P3}}(\lambda))$, where $\mathcal{B}_{9\text{-}i\text{-}l}(\cdot) := \mathcal{B}_{9\text{-}l}(i, \cdot)$.*

*Proof.* We can change Game 3-$h$-3 to 3-$h$-4 by tracing the reverse transformations from Game 3-$h$-3 to Game 3-$(h-1)$-4 with $\boldsymbol{k}_0^*$ unchanged. Therefore, by combining Lemmas 18–14 in a reverse order, we obtain Lemma 19. □

**Lemma 20.** *For any adversary $\mathcal{A}$, for any security parameter $\lambda$, $|\mathsf{Adv}_{\mathcal{A}}^{(3\text{-}\nu\text{-}4)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(4)}(\lambda)| \leq \epsilon(\lambda)$, where $\epsilon(\lambda)$ is a negligible function.*

Lemma 20 is proven in a similar manner to Lemma 7 in [20]. The full proof of Lemma 20 is given in Appendix C.3.

**Lemma 21.** *For any adversary $\mathcal{A}$, for any security parameter $\lambda$, $\mathsf{Adv}^{(4)}_{\mathcal{A}}(\lambda) = 0$.*

*Proof.* The value of $b$ is independent from the adversary's view in Game 4. Hence, $\mathsf{Adv}^{(4)}_{\mathcal{A}}(\lambda) = 0$. □

# 6 Publicly Verifiable Computation from Our KP-ABE

## 6.1 Definitions

**Definition 6 ([11, 24]).** *A publicly verifiable computation protocol for function class $\mathcal{F}$ (with preprocessing) consists of five-tuple of probabilistic polynomial-time algorithms* $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{ProbGen}, \mathsf{Compute}, \mathsf{Verify})$*:*

- $\mathsf{Setup}(1^\lambda) \to (\mathsf{PK}, \mathsf{MSK})$*: The randomized setup algorithm takes as input a security parameter $1^\lambda$, and outputs a short public key $\mathsf{PK}$ and master secret key $\mathsf{MSK}$.*
- $\mathsf{KeyGen}(\mathsf{MSK}, f) \to \mathsf{EK}_f$*: The randomized key generation algorithm takes as input a secret key $\mathsf{MSK}$ and a function $f \in \mathcal{F}$, and outputs a public evaluation key $\mathsf{EK}_f$, which will be used for the evaluation of the function $f$.*
- $\mathsf{ProbGen}(\mathsf{PK}, x) \to (\sigma_x, \mathsf{VK}_x)$*: The problem generation algorithm uses the public key $\mathsf{PK}$ to encode the function input $x \in \mathsf{Dom}(f)$ as a public value $\sigma_x$, which is given to the worker to compute with, and a public value $\mathsf{VK}_x$, which is used for verification.*
- $\mathsf{Compute}(\mathsf{EK}_f, \sigma_x) \to \sigma_{\mathsf{out}}$*: The deterministic worker algorithm uses the evaluation key $\mathsf{EK}_f$ together with the value $\sigma_x$ to compute a value $\sigma_{\mathsf{out}}$.*
- $\mathsf{Verify}(\mathsf{VK}_x, \sigma_{\mathsf{out}}) \to y$*: The deterministic verification algorithm uses the verification key $\mathsf{VK}_x$ and the worker's output $\sigma_{\mathsf{out}}$ to compute a string $y \in \{0,1\}^* \cup \perp$. Here, the special symbol $\perp$ signifies that the verification algorithm rejects the worker's answer $\sigma_{\mathsf{out}}$.*

**Correctness.** A publicly verifiable computation protocol is correct for a class of functions $\mathcal{F}$ if for any $(\mathsf{PK}, \mathsf{MSK}) \xleftarrow{R} \mathsf{Setup}(1^\lambda)$, any $f \in \mathcal{F}$, any $\mathsf{EK}_f \xleftarrow{R} \mathsf{KeyGen}(\mathsf{MSK}, f)$, any $x \in \mathsf{Dom}(f)$, any $(\sigma_x, \mathsf{VK}_x) \xleftarrow{R} \mathsf{ProbGen}(\mathsf{PK}, x)$, and any $\sigma_{\mathsf{out}} \xleftarrow{R} \mathsf{Compute}(\mathsf{EK}_f, \sigma_x)$, the verification algorithm $\mathsf{Verify}$ on input $\mathsf{VK}_x$ and $\sigma_{\mathsf{out}}$ outputs $y = f(x)$.

**Security** There are three notions of security (soundness) for publicly verifiable computation, depending on the level of adaptivity the client has in choosing the challenge instance $x^*$ with respect to $\mathsf{PK}$ and $\mathsf{EK}_f$ [9]:

- the weakest notion requires that $x^*$ be chosen independently of $(\mathsf{PK}, \mathsf{EK}_f)$. This is the notion achieved in [11] based on a selectively secure KP-ABE.

– an intermediate notion requires that $x^*$ by chosen independently of $\mathsf{EK}_f$, but may potentially depend on $\mathsf{PK}$. This is the notion achieved in [9] based on a semi-adaptively secure KP-ABE.
– the strongest notion allows $x^*$ to depend on both $\mathsf{PK}$ and $\mathsf{EK}_f$. It can be achieved based on an adaptively secure KP-ABE.

In [9], they mention that it is important that they allow client's input $x^*$ to depend on $\mathsf{PK}$ in order to achieve any meaningful notion of security, on the other hand, it seems reasonable to consider relaxed scenarios where the clients input does not depend on the server's private evaluation key $\mathsf{EK}_f$, since $\mathsf{EK}_f$ is only known to the server carrying out the computation. However, the soundness should be considered against *malicious server* possessing evaluation keys. Therefore, we consider semi-adaptive notion of soundness [9] is just a *weak guarantee* for the security of public VC, hence, our aim is to achieve adaptive soundness.

**Efficiency** A VC protocol needs to compute two functions $\mathsf{ProbGen}$ and $\mathsf{Verify}$ *(asymptotically) faster than the function $f$ itself.* More precisely, Chen-Wee [9] defines the efficiency requirement.

For the explicit description of adaptive soundness and efficiency requirement for NI-VC, see Appendix F.

### 6.2 Conversion to Adaptively Secure NI-VC from Our KP-ABE [24]

Below, we consider boolean function class $\mathcal{F}$, $f : \{0,1\}^n \to \{0,1\}$, for $n := n(\lambda)$. Let $\bar{f}(x) := 1$ iff $f(x) = 0$ and class $\bar{\mathcal{F}} := \{\bar{f} \mid f \in \mathcal{F}\}$. We construct public key VC protocol from $\mathsf{ABE} := (\mathsf{ABE.Setup}, \mathsf{ABE.KeyGen}, \mathsf{ABE.Enc}, \mathsf{ABE.Dec})$ for class $\mathcal{F} \cup \bar{\mathcal{F}}$. Let attribute set $[n] := \{1, \ldots, n\}$.

$\mathsf{Setup}(1^\lambda)$: For attribute set $[n]$ and a bound for row number $r$, generate two independent master key pairs: $(\mathsf{pk}_0, \mathsf{msk}_0) \leftarrow \mathsf{ABE.Setup}(1^\lambda, n, r)$, $(\mathsf{pk}_1, \mathsf{msk}_1) \leftarrow \mathsf{ABE.Setup}(1^\lambda, n, r)$, then set $\mathsf{PK} := (\mathsf{pk}_0, \mathsf{pk}_1, H)$ where $H$ is a one-way function, and $\mathsf{MSK} := (\mathsf{msk}_0, \mathsf{msk}_1)$. Output $(\mathsf{PK}, \mathsf{MSK})$.

$\mathsf{KeyGen}(\mathsf{MSK}, f)$: Generate secret keys for $\bar{f}$ and $f$: $\mathsf{sk}_{\bar{f}} \leftarrow \mathsf{ABE.KeyGen}(\mathsf{pk}_0, \mathsf{msk}_0, \bar{f})$, $\mathsf{sk}_f \leftarrow \mathsf{ABE.KeyGen}(\mathsf{pk}_1, \mathsf{msk}_1, f)$ then output evaluation key $\mathsf{EK}_f := (\mathsf{sk}_{\bar{f}}, \mathsf{sk}_f)$.

$\mathsf{ProbGen}(\mathsf{PK}, x)$: Sample two messages $m_0, m_1$ with the same length randomly. Generate ciphertexts: $\mathsf{ct}_{x,0} \leftarrow \mathsf{ABE.Enc}(\mathsf{pk}_0, x, m_0)$, $\mathsf{ct}_{x,1} \leftarrow \mathsf{ABE.Enc}(\mathsf{pk}_1, x, m_1)$. Output preprocessed value $\sigma_x := (\mathsf{ct}_{x,0}, \mathsf{ct}_{x,1})$ and verification key $\mathsf{VK}_x := (H(m_0), H(m_1))$.

$\mathsf{Compute}(\mathsf{EK}_f, \sigma_x)$: Decrypt two ciphertexts $\sigma_x := (\mathsf{ct}_{x,0}, \mathsf{ct}_{x,1})$ using $\mathsf{EK}_f := (\mathsf{sk}_{\bar{f}}, \mathsf{sk}_f)$: $m'_0 \leftarrow \mathsf{ABE.Dec}(\mathsf{pk}_0, \mathsf{sk}_{\bar{f}}, \mathsf{ct}_{x,0})$, $m'_1 \leftarrow \mathsf{ABE.Dec}(\mathsf{pk}_1, \mathsf{sk}_f, \mathsf{ct}_{x,1})$, and output the result $\sigma_{\mathsf{out}} := (m'_0, m'_1)$.

$\mathsf{Verify}(\mathsf{VK}_x, \sigma_{\mathsf{out}})$: Take verification key $\mathsf{VK}_x := (H(m_0), H(m_1))$ and the result $\sigma_{\mathsf{out}} := (m'_0, m'_1)$ as input, if $H(m_0) = H(m'_0)$, output 0, if $H(m_1) = H(m'_1)$, output 1, otherwise, output $\perp$.

**Correctness**: When compute keys, preprocessed data and result, correctly, if $f(x) = 0$, it holds $m'_0 = m_0$, and if $f(x) = 1$, it holds $m'_1 = m_1$ and $m'_0 \neq m_0$

**Table 3.** Comparison with existing pairing based (semi-)adaptively secure public key NI-VC schemes. PHGR13 deals with NC class. The others are obtained from KP-ABE using generic transformation given in Section 6.2, and deal with $\mathrm{NC}^1$. In the table, $|\mathbb{G}|$, $\lambda, n, \ell, k$ represent size of $\mathbb{G}$, security parameter, (the maximum of) input size of a boolean function $f$, size of $f$, maximum input multiplicity in available $f$, respectively. DLIN, KEA, $s$-Lin stand for Decisional LINear and Knowledge of Exponent Assumption, $s$-Linear, respectively.

| | Security | Assump. | Order of $\mathbb{G}$ | $\lvert\mathsf{EK}_f\rvert$ | Comm. cost in bits | Worker's complexity |
|---|---|---|---|---|---|---|
| CW14 [9] | semi-adaptive | non-para-metrized | composite | $O(\ell n)\,\lvert\mathbb{G}\rvert$ | $n + O(\lambda)$ | $O(\ell n)$ |
| Tak14 [26] | | DLIN | prime | $O(\ell n)\,\lvert\mathbb{G}\rvert$ | $n + O(\lambda)$ | $O(\ell n)$ |
| OT10 [20] | adaptive | DLIN | prime | $O(\ell)\,\lvert\mathbb{G}\rvert$ | $O(kn\lambda)$ | $O(\ell)$ |
| PHGR13 [12, 23] | | $\ell$-param. & KEA | prime | $O(\ell)\,\lvert\mathbb{G}\rvert$ | $n + O(\lambda)$ | $O(\ell)$ |
| Att15 [2, 3] | | $\ell$-para-metrized | prime | $O(\ell n)\,\lvert\mathbb{G}\rvert$ | $n + O(\lambda)$ | $O(\ell n)$ |
| CGW15 [8] | | $s$-Lin for $\forall s$ | prime | $O(\ell)\,\lvert\mathbb{G}\rvert$ for $s = 2$ | $O(kn\lambda)$ for $s = 2$ | $O(\ell)$ for $s = 2$ |
| Proposed | adaptive | DLIN | prime | $O(\ell)\,\lvert\mathbb{G}\rvert$ | $O((n + r)\lambda)$ | $O(\ell(n + r))$ |

except for negligible probability, we see the correctness of the VC scheme.

**Efficiency**: $\mathsf{ProbGen}$ encrypts $x$ and $\mathsf{Verify}$ computes the one-way function. For most KP-ABE schemes including one in Section 5, there exists a function $f_\lambda \in \mathcal{F}_\lambda$ for each security parameter $\lambda$, whose calculation time is more than polynomial $p(n, \lambda)$ of $n = n(\lambda), \lambda$. It means the efficiency requirement of the above VC. In particular, we note that $\mathsf{Verify}$ is very fast (one-way function evaluation).

Corresponding to three types of security for VC, three types of security for KP-ABE are defined: selective, semi-adaptive, adaptive security. In this paper, we focus adaptive security, and we mention the theorem below.

**Theorem 3.** *If KP-ABE scheme* $\mathsf{ABE}$ *is adaptively secure, the above VC protocol is adaptively secure (sound).*

The proof is a straightforward extension of Theorem 2 in [24].

**Remark 4** While our KP-ABE supports only monotone span programs, since it supports a *large universe* as underlying equality relations, it is enough to realize all boolean formula class by restricting the large universe to the $n$-element small universe, $[n]$, in an arbitrary manner. That is, our KP-ABE is enough to obtain an adaptively secure communication-efficient NI-VC by the above conversion.

We show a comparison table with our NI-VC and existing ones (Table 3).

## References


1. Agrawal, S., Chase, M.: A study of pair encoding: Predicate encryption in prime order groups. IACR Crypt. ePrint Archive (2015), http://eprint.iacr.org/2015/413
2. Attrapadung, N.: Dual system encryption via doubly selective security: Framework, fully secure functional encryption for regular languages, and more. In: EUROCRYPT. pp. 557–577 (2014)
3. Attrapadung, N.: Dual system encryption framework in prime-order groups. IACR Cryptology ePrint Archive 2015, 390 (2015), http://eprint.iacr.org/2015/390
4. Attrapadung, N., Yamada, S.: Duality in ABE: converting attribute based encryption for dual predicate and dual policy via computational encodings. In: CT-RSA 2015. pp. 87–105 (2015)
5. Beimel, A.: Secure schemes for secret sharing and key distribution. PhD Thesis, Israel Institute of Technology, Technion, Haifa (1996)
6. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M.K. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer (2004)
7. Boneh, D., Gentry, C., Gorbunov, S., Halevi, S., Nikolaenko, V., Segev, G., Vaikuntanathan, V., Vinayagamurthy, D.: Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In: EUROCRYPT 2014. pp. 533–556 (2014)
8. Chen, J., Gay, R., Wee, H.: Improved dual system ABE in prime-order groups via predicate encodings. In: EUROCRYPT 2015. pp. 595–624 (2015)
9. Chen, J., Wee, H.: Semi-adaptive attribute-based encryption and improved delegation for boolean formula. In: SCN. pp. 277–297 (2014)
10. Cheon, J.H.: Security analysis of the strong diffie-hellman problem. In: EUROCRYPT. pp. 1–11 (2006)
11. Gennaro, R., Gentry, C., Parno, B.: Non-interactive verifiable computing: Outsourcing computation to untrusted workers. In: CRYPTO. pp. 465–482 (2010)
12. Gennaro, R., Gentry, C., Parno, B., Raykova, M.: Quadratic span programs and succinct NIZKs without PCPs. In: EUROCRYPT. pp. 626–645 (2013)
13. Gorbunov, S., Vaikuntanathan, V., Wee, H.: Attribute-based encryption for circuits. In: STOC 2013. pp. 545–554 (2013)
14. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: ACM CCS 2006. pp. 89–98 (2006)
15. Ishai, Y., Wee, H.: Partial garbling schemes and their applications. In: ICALP 2014. pp. 650–662 (2014)
16. Lewko, A.B., Okamoto, T., Sahai, A., Takashima, K., Waters, B.: Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In: EUROCRYPT 2010. pp. 62–91 (2010), full version is available at http://eprint.iacr.org/2010/110
17. Lewko, A.B., Waters, B.: Decentralizing attribute-based encryption. In: EUROCRYPT 2011. pp. 568–588 (2011)
18. Lewko, A.B., Waters, B.: New proof methods for attribute-based encryption: Achieving full security through selective techniques. In: CRYPTO. pp. 180–198 (2012)
19. Okamoto, T., Takashima, K.: Hierarchical predicate encryption for inner-products. In: ASIACRYPT 2009. pp. 214–231 (2009)
20. Okamoto, T., Takashima, K.: Fully secure functional encryption with general relations from the decisional linear assumption. In: CRYPTO 2010. pp. 191–208 (2010), full version is available at http://eprint.iacr.org/2010/563

21. Okamoto, T., Takashima, K.: Efficient attribute-based signatures for non-monotone predicates in the standard model. IEEE T. Cloud Computing 2(4), 409–421 (2014), the preliminary version appeared in the proceedings of PKC 2011. Full version: http://eprint.iacr.org/2011/700
22. Okamoto, T., Takashima, K.: Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption. Des. Codes Cryptography 77(2-3), 725–771 (2015), the preliminary version appeared in CANS 2011.
23. Parno, B., Howell, J., Gentry, C., Raykova, M.: Pinocchio: Nearly practical verifiable computation. In: IEEE Symp. Security & Privacy. pp. 238–252 (2013)
24. Parno, B., Raykova, M., Vaikuntanathan, V.: How to delegate and verify in public: Verifiable computation from attribute-based encryption. In: TCC. pp. 422–439 (2012)
25. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: EUROCRYPT 2005. pp. 457–473 (2005)
26. Takashima, K.: Expressive attribute-based encryption with constant-size ciphertexts from the decisional linear assumption. In: SCN. pp. 298–317 (2014), full version is available at http://eprint.iacr.org/2014/207
27. Waters, B.: Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In: CRYPTO 2009. pp. 619–636 (2009)
28. Waters, B.: Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In: PKC 2011. pp. 53–70 (2011)
29. Yamada, S., Attrapadung, N., Hanaoka, G.: Conversions among several classes of predicate encryption and their applications. IACR Cryptology ePrint Archive (2015), http://eprint.iacr.org/2015/431, to appear in ASIACRYPT 2015

## A Decisional Linear (DLIN) Assumption

**Definition 7 (DLIN: Decisional Linear Assumption [6]).** *The DLIN problem is to guess* $\beta \in \{0,1\}$*, given* $(\mathsf{param}_{\mathbb{G}},\ G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta) \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{DLIN}}(1^\lambda)$*, where* $\mathcal{G}_\beta^{\mathsf{DLIN}}(1^\lambda) : \mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda), \kappa, \delta, \xi, \sigma \xleftarrow{\mathsf{U}} \mathbb{F}_q, Y_0 := (\delta+\sigma)G, Y_1 \xleftarrow{\mathsf{U}} \mathbb{G}, \mathrm{return}\ (\mathsf{param}_{\mathbb{G}}, G, \xi G, \kappa G,\ \delta\xi G, \sigma\kappa G, Y_\beta)$*, for* $\beta \xleftarrow{\mathsf{U}} \{0,1\}$*. For a probabilistic machine* $\mathcal{E}$*, we define the advantage of* $\mathcal{E}$ *for the DLIN problem as:* $\mathsf{Adv}_{\mathcal{E}}^{\mathsf{DLIN}}(\lambda) := \left|\mathsf{Pr}\left[\mathcal{E}(1^\lambda, \varrho) \to 1 \,\middle|\, \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathsf{DLIN}}(1^\lambda)\right] - \mathsf{Pr}\left[\mathcal{E}(1^\lambda, \varrho) \to 1 \,\middle|\, \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathsf{DLIN}}(1^\lambda)\right]\right|$*. The DLIN assumption is: For any probabilistic polynomial-time adversary* $\mathcal{E}$*, the advantage* $\mathsf{Adv}_{\mathcal{E}}^{\mathsf{DLIN}}(\lambda)$ *is negligible in* $\lambda$*.*

## B Problems 1–3 for the Proof of Theorem 2

**Definition 8 (Problem 1).** *Problem 1 is to guess* $\beta$*, given* $(\mathsf{param}_{\vec{n}}, \{\mathbb{B}_\iota, \widehat{\mathbb{B}}_\iota^*\}_{\iota=0,1}, \{\boldsymbol{e}_{\beta,i}\}_{i=0,\ldots,n+r}) \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{P1}}(1^\lambda, \vec{n})$*, where* $\vec{n} := (n, r)$ *and*

$\mathcal{G}_\beta^{\mathsf{P1}}(1^\lambda,\vec{n}) : (\mathsf{param}_{\vec{n}},\mathbb{B}_0,\mathbb{B}_0^*,\mathbb{B}_1,\{B^*_{i,j,\iota},B'^*_{i,j,l}\}_{l=1,\ldots,n+r}^{i,j=1,\ldots,5;\iota=1,2}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{KP}}(1^\lambda,5,\vec{n})$,
$\mathbb{B}_1^* := (\boldsymbol{b}^*_{1,1},\ldots,\boldsymbol{b}^*_{1,5(n+r)})$ is calculated from $\{B^*_{i,j,\iota},B'^*_{i,j,l}\}$,
$\widehat{\mathbb{B}}_0^* := (\boldsymbol{b}^*_{0,1},\boldsymbol{b}^*_{0,2},\boldsymbol{b}^*_{0,4},\boldsymbol{b}^*_{0,5})$, $\widehat{\mathbb{B}}_1^* := (\boldsymbol{b}^*_{1,1},..,\boldsymbol{b}^*_{1,n+r},\boldsymbol{b}^*_{1,2(n+r)+1},..,\boldsymbol{b}^*_{1,5(n+r)})$,
$\omega,\varphi_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times$, $\boldsymbol{e}_{0,0} := (0,\omega,0,0,\varphi_0)_{\mathbb{B}_0}$, $\boldsymbol{e}_{1,0} := (0,\omega,\tau,0,\varphi_0)_{\mathbb{B}_0}$,
for $i=1,\ldots,n+r$; $\vec{e}_i := (0^{i-1},1,0^{n+r-i}) \in \mathbb{F}_q^{n+r}$, $\vec{\varphi}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n+r}$,

$$\begin{array}{ll}
\boldsymbol{e}_{0,i} := ( & \overbrace{\omega\vec{e}_i,}^{n+r} \quad \overbrace{0^{2n+2r},}^{2n+2r} \quad \overbrace{0^{n+r},}^{n+r} \quad \overbrace{\vec{\varphi}_i}^{n+r} \quad )_{\mathbb{B}_1},\\
\boldsymbol{e}_{1,i} := ( & \omega\vec{e}_i, \quad \tau\vec{e}_i,\ 0^{n+r}, \quad 0^{n+r}, \quad \vec{\varphi}_i \quad )_{\mathbb{B}_1},
\end{array}$$

return $(\mathsf{param}_{(n,r)},\{\mathbb{B}_\iota,\widehat{\mathbb{B}}_\iota^*\}_{\iota=0,1},\{\boldsymbol{e}_{\beta,i}\}_{i=0,\ldots,n+r})$,

*for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. For a probabilistic adversary $\mathcal{B}$, we define the advantage of $\mathcal{B}$ as the quantity*
$\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P1}}(\lambda) := \left|\mathsf{Pr}\left[\mathcal{B}(1^\lambda,\varrho)\to 1 \,\middle|\, \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathsf{P1}}(1^\lambda,\vec{n})\right] - \mathsf{Pr}\left[\mathcal{B}(1^\lambda,\varrho)\to 1 \,\middle|\, \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathsf{P1}}(1^\lambda,\vec{n})\right]\right|$.

**Lemma 22.** *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{F}$, whose running time are essentially the same as that of $\mathcal{B}$, such that for any security parameter $\lambda$, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P1}}(\lambda) \le \mathsf{Adv}_{\mathcal{F}}^{\mathsf{DLIN}}(\lambda) + 5/q$.*

Lemma 22 is proven in a similar manner to Lemmas 1 in [20]. □

**Definition 9 (Problem 2).** *Problem 2 is to guess $\beta$, given $(\mathsf{param}_{\vec{n}},\{\widehat{\mathbb{B}}_\iota,\mathbb{B}_\iota^*\}_{\iota=0,1},\{\boldsymbol{h}^*_{\beta,i}\}_{i=0,\ldots,n+r},\{\boldsymbol{e}_i\}_{i=0,\ldots,2n+2r}) \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{P2}}(1^\lambda,\vec{n})$, where $\vec{n} := (n,r)$ and*

$\mathcal{G}_\beta^{\mathsf{P2}}(1^\lambda,\vec{n}) : (\mathsf{param}_{(n,r)},\mathbb{B}_0,\mathbb{B}_0^*,\mathbb{B}_1,\{B^*_{i,j,\iota},B'^*_{i,j,l}\}_{l=1,\ldots,n+r}^{i,j=1,\ldots,5;\iota=1,2}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{KP}}(1^\lambda,5,\vec{n})$,
$\mathbb{B}_1^* := (\boldsymbol{b}^*_{1,1},\ldots,\boldsymbol{b}^*_{1,5(n+r)})$ is calculated from $\{B^*_{i,j,\iota},B'^*_{i,j,l}\}$,
$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1},\boldsymbol{b}_{0,2},\boldsymbol{b}_{0,4},\boldsymbol{b}_{0,5})$, $\widehat{\mathbb{B}}_1 := (\boldsymbol{b}^*_{1,1},..,\boldsymbol{b}^*_{1,n+r},\boldsymbol{b}^*_{1,2(n+r)+1},..,\boldsymbol{b}^*_{1,5(n+r)})$,
$\omega,\delta,\varphi_0,\delta_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\rho \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times$, $\boldsymbol{h}^*_{0,0} := (0,\delta,0,\delta_0,0)_{\mathbb{B}_0^*}$, $\boldsymbol{h}^*_{1,0} := (0,\delta,\rho,\delta_0,0)_{\mathbb{B}_0^*}$,
$\boldsymbol{e}_0 := (0,\omega,\tau,0,\varphi_0)_{\mathbb{B}_0}$,
for $i=1,\ldots,n+r$; $\vec{e}_i := (0^{i-1},1,0^{n+r-i}) \in \mathbb{F}_q^{n+r}$, $\delta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{\varphi}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n+r}$,

$$\begin{array}{ll}
\boldsymbol{h}^*_{0,i} := ( & \overbrace{\delta\vec{e}_i,}^{n+r} \quad \overbrace{0^{2n+2r},}^{2n+2r} \quad \overbrace{\delta_i\vec{e}_i,}^{n+r} \quad \overbrace{0^{n+r}}^{n+r} \quad )_{\mathbb{B}_1^*}\\
\boldsymbol{h}^*_{1,i} := ( & \delta\vec{e}_i, \quad \rho\vec{e}_i,\ 0^{n+r}, \quad \delta_i\vec{e}_i, \quad 0^{n+r} \quad )_{\mathbb{B}_1^*}\\
\boldsymbol{e}_i := ( & \omega\vec{e}_i, \quad \tau\vec{e}_i,\ 0^{n+r}, \quad 0^{n+r}, \quad \vec{\varphi}_i \quad )_{\mathbb{B}_1},
\end{array}$$

$\boldsymbol{e}_{n+r+i} := \tau\boldsymbol{b}_{1,2n+2r+i}$,
return $(\mathsf{param}_{(n,r)},\{\widehat{\mathbb{B}}_\iota,\mathbb{B}_\iota^*\}_{\iota=0,1},\{\boldsymbol{h}^*_{\beta,i}\}_{i=0,\ldots,n+r},\{\boldsymbol{e}_i\}_{i=0,\ldots,2n+2r})$,

*for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. For a probabilistic adversary $\mathcal{B}$, the advantage of $\mathcal{B}$ for Problem 2, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P2}}(\lambda)$, is similarly defined as in Definition 8.*

**Lemma 23.** *For any adversary $\mathcal{B}$, there exists a probabilistic machine $\mathcal{F}$, whose running time are essentially the same as that of $\mathcal{B}$, such that for any security parameter $\lambda$, $\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda) \leq \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}}(\lambda) + 5/q$.*

Lemma 23 is proven in a similar manner to Lemmas 2 in [20]. □

**Definition 10 (Problem 3).** *Problem 3 is to guess $\beta$, given $(\mathsf{param}_n, \mathbb{B}_0, \mathbb{B}^*_0, \boldsymbol{h}^*_0, \boldsymbol{e}_0, \widehat{\mathbb{B}}_1, \mathbb{B}^*_1, \{\boldsymbol{h}^*_{\beta,i}, \boldsymbol{e}_i\}_{i=1,\ldots,n+r}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{P3}}_{\beta}(1^\lambda, \vec{n})$, where $\vec{n} := (n, r)$ and*

$$
\begin{aligned}
&\mathcal{G}^{\mathsf{P3}}_{\beta}(1^\lambda, \vec{n}) : (\mathsf{param}_{(n,r)}, \mathbb{B}_0, \mathbb{B}^*_0, \mathbb{B}_1, \{B^*_{i,j,\iota}, B'^*_{i,j,l}\}^{i,j=1,\ldots,5;\iota=1,2}_{l=1,\ldots,n+r}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{KP}}_{\mathsf{ob}}(1^\lambda, 5, \vec{n}),\\
&\quad \mathbb{B}^*_1 := (\boldsymbol{b}^*_{1,1}, \ldots, \boldsymbol{b}^*_{1,5(n+r)}) \text{ is calculated from } \{B^*_{i,j,\iota}, B'^*_{i,j,l}\},\\
&\quad \widehat{\mathbb{B}}_1 := (\boldsymbol{b}_{1,1}, .., \boldsymbol{b}_{1,n+r}, \boldsymbol{b}_{1,3(n+r)+1}, .., \boldsymbol{b}_{1,5(n+r)}),\\
&\quad \tau, \rho \xleftarrow{\mathsf{U}} \mathbb{F}^\times_q, \quad \boldsymbol{h}^*_0 := \rho \boldsymbol{b}^*_{0,3}, \quad \boldsymbol{e}_0 := \tau \boldsymbol{b}_{0,3},\\
&\quad \text{for } i = 1, \ldots, n+r; \quad \vec{e}_i := (0^{i-1}, 1, 0^{n+r-i}) \in \mathbb{F}^{n+r}_q, \quad \delta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q,\\
&\qquad\qquad \overbrace{\qquad}^{n+r} \overbrace{\qquad\qquad}^{2n+2r} \overbrace{\qquad}^{n+r} \overbrace{\qquad}^{n+r}\\
&\quad \boldsymbol{h}^*_{0,i} := (\ 0^{n+r},\ \rho\vec{e}_i,\ 0^{n+r},\ \delta_i\vec{e}_i,\ 0^{n+r}\ )_{\mathbb{B}^*_1}\\
&\quad \boldsymbol{h}^*_{1,i} := (\ 0^{n+r},\ 0^{n+r},\ \rho\vec{e}_i,\ \delta_i\vec{e}_i,\ 0^{n+r}\ )_{\mathbb{B}^*_1}\\
&\quad \boldsymbol{e}_i := \ (\ 0^{n+r},\ \tau\vec{e}_i,\ \tau\vec{e}_i,\ 0^{n+r},\ 0^{n+r}\ )_{\mathbb{B}_1},\\
&\quad \text{return } (\mathsf{param}_n, \mathbb{B}_0, \mathbb{B}^*_0, \boldsymbol{h}^*_0, \boldsymbol{e}_0, \widehat{\mathbb{B}}_1, \mathbb{B}^*_1, \{\boldsymbol{h}^*_{\beta,i}, \boldsymbol{e}_i\}_{i=1,\ldots,n+r}),
\end{aligned}
$$

*for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. For a probabilistic adversary $\mathcal{B}$, the advantage of $\mathcal{B}$ for Problem 3, $\mathsf{Adv}^{\mathsf{P3}}_{\mathcal{B}}(\lambda)$, is similarly defined as in Definition 8.*

**Lemma 24.** *For any adversary $\mathcal{B}$, there exist probabilistic machines $\mathcal{F}_1$ and $\mathcal{F}_2$, whose running times are essentially the same as that of $\mathcal{B}$, such that for any security parameter $\lambda$, $\mathsf{Adv}^{\mathsf{P3}}_{\mathcal{B}}(\lambda) \leq \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_1}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{F}_2}(\lambda) + 10/q$.*

Lemma 24 is proven in a similar manner to Lemmas 6 in [26]. □

# C Proofs of Lemmas 16, 18 and 20

## C.1 Proof of Lemma 16

*Proof.* To prove Lemma 16, we will show distribution $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}^{(j)*}_{\mathbb{S}}\}_{j=1,\ldots,\nu}, \mathsf{ct}_{\Gamma})$ in Games 3-$h$-2-$p$-1 and 3-$h$-2-$p$-2 are equivalent. For that purpose, we define an intermediate game, Game 3-$h$-2-$p$-1', as

**Game 3-$h$-2-$p$-1' $(h = 1, \ldots, \nu_1; p = 1, \ldots, \ell)$ :** Game 3-$h$-2-$p$-1' is the same as Game 3-$h$-2-$p$-1 except $\boldsymbol{k}^*_p$ in the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ and the challenge ciphertext $\boldsymbol{c}_1$ are:

$$
\begin{aligned}
&\boldsymbol{c}_1 := (\ \omega\vec{y},\ \vec{f},\ \omega'\vec{y},\ \vec{f}',\ \boxed{\vec{z}},\ 0^{n+r},\ \vec{\varphi}_1)_{\mathbb{B}_1},\\
&\text{if } \vec{v}_p \cdot \vec{y} \neq 0, \quad \boldsymbol{k}^*_p := (\ \theta_p\vec{v}_p,\ \xi M_p,\ 0^{n+r},\ \boxed{\vec{w}},\ \psi_p\vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{B}^*_1},
\end{aligned}
$$

where $\vec{v}'_p := (\vec{v}_p,\ 0^r), M'_p := (0^n,\ M_p)$ and if $\vec{v}_p \cdot \vec{y} \neq 0$, then $(\vec{w}, \vec{z}) \xleftarrow{\cup} W'_{\neq 0} := \{(\vec{w}, \vec{z}) \in \mathsf{span}\langle \vec{e}_1, \vec{v}'_p, M'_p\rangle \times \mathbb{F}_q^{n+r} \mid \vec{w} \cdot \vec{z} \neq 0\}$.

**Claim 3** *The distribution* $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \mathsf{ct}_\Gamma)$ *in Game 3-h-2-p-1 and that in Game 3-h-2-p-1' are equivalent except with negligible probability.*

*Proof of Claim 3.* We will consider the distribution in Game 3-$h$-2-$p$-1. We define new dual orthonormal bases $(\mathbb{D}_1, \mathbb{D}_1^*)$ of $\mathbb{V}_1$. First, we generate matrix $U \xleftarrow{\cup} \mathcal{H}(n, r, \mathbb{F}_q) \cap GL(n + r, \mathbb{F}_q)$, and set $\tilde{n} := n + r$,

$$\begin{pmatrix} \boldsymbol{d}_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{d}_{1,3\tilde{n}} \end{pmatrix} := U^{\mathrm{T}} \cdot \begin{pmatrix} \boldsymbol{b}_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{b}_{1,3\tilde{n}} \end{pmatrix}, \begin{pmatrix} \boldsymbol{d}^*_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{d}^*_{1,3\tilde{n}} \end{pmatrix} := U^{-1} \cdot \begin{pmatrix} \boldsymbol{b}^*_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{b}^*_{1,3\tilde{n}} \end{pmatrix},$$
$$\mathbb{D}_1 := (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,2\tilde{n}}, \boldsymbol{d}_{1,2\tilde{n}+1}, \ldots, \boldsymbol{d}_{1,3\tilde{n}}, \boldsymbol{b}_{1,3\tilde{n}+1}, \ldots, \boldsymbol{b}_{1,5\tilde{n}}),$$
$$\mathbb{D}_1^* := (\boldsymbol{b}^*_{1,1}, \ldots, \boldsymbol{b}^*_{1,2\tilde{n}}, \boldsymbol{d}^*_{1,2\tilde{n}+1}, \ldots, \boldsymbol{d}^*_{1,3\tilde{n}}, \boldsymbol{b}^*_{1,3\tilde{n}+1}, \ldots, \boldsymbol{b}^*_{1,5\tilde{n}}).$$

Then, $\mathbb{D}_1$ and $\mathbb{D}_1^*$ are dual orthonormal bases. $\boldsymbol{k}^*_p$ in the $h$-th queried key for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix $M = (M_i)$ and the challenge ciphertext $\boldsymbol{c}_1$ is expressed as

$$\begin{aligned}
&\text{if } \vec{v}_p \cdot \vec{y} \neq 0, \\
&\quad \boldsymbol{k}^*_p = (\ \theta_p \vec{v}_p,\ \xi M_p,\ 0^{n+r},\ \theta'_p \vec{v}_p,\ \xi' M_p,\ \psi_p \vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{B}_1^*} \\
&\qquad = (\ \theta_p \vec{v}_p,\ \xi M_p,\ 0^{n+r},\ (\theta'_p \vec{v}_p,\ \xi' M_p) \cdot U,\ \psi_p \vec{v}_p,\ \eta_p M_p,\ 0^{n+r})_{\mathbb{D}_1^*}, \quad (24) \\
&\boldsymbol{c}_1 = (\ \omega \vec{y},\ \vec{f},\ \omega' \vec{y},\ \vec{f}',\ \omega' \vec{y},\ \vec{f}',\ 0^{n+r},\ \vec{\varphi}_1)_{\mathbb{B}_1} \\
&\quad = (\ \omega \vec{y},\ \vec{f},\ \omega' \vec{y},\ \vec{f}',\ (\omega' \vec{y},\ \vec{f}') \cdot Z,\ 0^{n+r},\ \vec{\varphi}_1)_{\mathbb{D}_1} \quad (25)
\end{aligned}$$

where $Z := (U^{-1})^{\mathrm{T}}$. From Lemma 3, the pair of coefficients $(\vec{w}, \vec{z}) := ((\omega' \vec{y},\ \vec{f}') \cdot Z,\ (\theta'_p \vec{v}_p,\ \xi' M_p) \cdot U)$ is uniformly distributed in

$$W_{\vec{v}'_p, M'_p, \pi} := \{(\vec{w}, \vec{z}) \in (\mathsf{span}\langle \vec{e}_1, \vec{v}'_p, M'_p\rangle \backslash \mathsf{span}\langle \vec{e}_1\rangle) \times (\mathbb{F}_q^{n+r} \backslash \mathsf{span}\langle \vec{e}_1\rangle^{\perp}) \mid \vec{w} \cdot \vec{z} = \pi\}.$$

where $\pi := \omega' \theta'_p (\vec{v}_p \cdot \vec{y}) + \xi' M_p \cdot \vec{f}',\ \vec{v}'_p := (\vec{v}_p,\ 0^r)$ and $M'_p := (0^n,\ M_p)$. In particular, if $\vec{v}_p \cdot \vec{y} \neq 0$, then $\pi$ is independently and uniformly distributed since $\theta'_p \xleftarrow{\cup} \mathbb{F}_q$, and $(\vec{w}, \vec{z})$ is independently and uniformly distributed in $W'_{\neq 0}$ (except with negligible probability).

When $1 \leq i \neq p \leq \ell$, the $i$-th component of the $h$-th queried key $\boldsymbol{k}^*_i$ is

$$\left.\begin{aligned}
&\text{if } i < p, \quad \boldsymbol{k}^*_i = (\ \theta_i \vec{v}_i,\ \xi M_i,\ \theta''_i,\ \theta'_i \vec{v}_i^{\geq 2},\ \xi' M_i,\ 0^{n+r},\ \ldots)_{\mathbb{B}_1^*} \\
&\qquad\qquad\quad = (\ \theta_i \vec{v}_i,\ \xi M_i,\ \theta''_i,\ \theta'_i \vec{v}_i^{\geq 2},\ \xi' M_i,\ 0^{n+r},\ \ldots)_{\mathbb{D}_1^*}, \\
&\text{if } i > p, \quad \boldsymbol{k}^*_i = (\ \theta_i \vec{v}_i,\ \xi M_i,\ \theta'_i \vec{v}_i,\ \xi' M_i,\ 0^{n+r},\ \ldots)_{\mathbb{B}_1^*} \\
&\qquad\qquad\quad = (\ \theta_i \vec{v}_i,\ \xi M_i,\ \theta'_i \vec{v}_i,\ \xi' M_i,\ 0^{n+r},\ \ldots)_{\mathbb{D}_1^*}.
\end{aligned}\right\} \quad (26)$$

In the light of the adversary's view, $(\mathbb{D}_1, \mathbb{D}_1^*)$ is consistent with public key $\mathsf{pk} := (1^\lambda, \mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1})$. Moreover, since the RHS of Eqs. (24), (25) and

(26) are in the same forms in those in Game 3-$h$-2-$p$-1', namely, Game 3-$h$-2-$p$-1 can be conceptually changed to Game 3-$h$-2-$p$-1' except with negligible probability. □

**Claim 4** *The distribution* $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \mathsf{ct}_{\Gamma})$ *in Game 3-h-2-p-2 and that in Game 3-h-2-p-1' are equivalent except with negligible probability.*

*Proof of Claim 4.* Claim 4 can be proven in a similar manner to Claim 3. Namely, we show that Game 3-$h$-2-$p$-2 can be conceptually changed to Game 3-$h$-2-$p$-1' except with negligible probability. In the proof, we consider a pair of coefficients $(\vec{w}, \vec{z}) := ((\omega'\vec{y},\ \vec{f}')\cdot Z,\ (\theta''_p, \theta'_p\vec{v}_p^{\geq 2},\ \xi' M_p)\cdot U)$ where a new randomness $\theta''_p$ is used instead of $\theta'_p v_{p,1}$ in Claim 3. Lemma 3 shows the pair is uniformly distributed in $W_{\vec{v}'_p, M'_p, \pi}$ with the inner product value $\pi := \omega'\theta''_p + \omega'\theta'_p(\vec{v}_p^{\geq 2}\cdot\vec{y}^{\geq 2}) + \xi' M'_p\cdot\vec{f}'$ since $y_1 = 1$. Since $\theta''_p$ is uniformly distributed, then $\pi$ is also uniform, and the same technique in Claim 3 is used in the rest of the proof of Claim 4. □

From Claims 3 and 4, the distribution $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \mathsf{ct}_{\Gamma})$ in Game 3-$h$-2-$p$-1 and that in Game 3-$h$-2-$p$-2 are equivalent except with negligible probability. This completes the proof of Lemma 16. □

### C.2 Proof of Lemma 18

*Proof.* To prove Lemma 18, we will show distribution $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{B}}_t\}_{t=0,1}, \{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \mathsf{ct}_{\Gamma})$ in Games 3-$h$-2-$\ell$-3 and 3-$h$-3 are equivalent. For that purpose, we define new subbases $\boldsymbol{d}_{1,n+r+1}, \ldots, \boldsymbol{d}_{1,3(n+r)}$ and $\boldsymbol{d}^*_{1,n+r+1}, \ldots, \boldsymbol{d}^*_{1,3(n+r)}$ of $\mathbb{V}_1$ as follows: The first component of the target vector $\vec{y} := (y_1, \ldots, y_n)$ in the challenge ciphertext is $y_1 = 1$. Then, we set $\mu'_1 := 1, \mu'_\iota := -y_\iota$ for $\iota = 2, \ldots, n$, $\mu'_{n+\iota} := -(\omega')^{-1} f'_\iota$ for $\iota = 1, \ldots, r$, and $Z_1 := \begin{pmatrix} 1 & \mu'_2 & \cdots & \mu'_{n+r} \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix} \in \mathbb{F}_q^{(n+r)\times(n+r)}$, and $U_2 \xleftarrow{\mathsf{U}} \mathcal{H}(n, r, \mathbb{F}_q)^\times$. Then we set $\tilde{n} := n + r$,

$$\begin{pmatrix} \boldsymbol{d}_{1,\tilde{n}+1} \\ \vdots \\ \boldsymbol{d}_{1,2\tilde{n}} \end{pmatrix} := Z_1^{-1}\cdot\begin{pmatrix} \boldsymbol{b}_{1,\tilde{n}+1} \\ \vdots \\ \boldsymbol{b}_{1,2\tilde{n}} \end{pmatrix},\ \begin{pmatrix} \boldsymbol{d}^*_{1,\tilde{n}+1} \\ \vdots \\ \boldsymbol{d}^*_{1,2\tilde{n}} \end{pmatrix} := Z_1^{\mathrm{T}}\cdot\begin{pmatrix} \boldsymbol{b}^*_{1,\tilde{n}+1} \\ \vdots \\ \boldsymbol{b}^*_{1,2\tilde{n}} \end{pmatrix}, \qquad (27)$$

$$\begin{pmatrix} \boldsymbol{d}_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{d}_{1,3\tilde{n}} \end{pmatrix} := U_2^{\mathrm{T}}\cdot\begin{pmatrix} \boldsymbol{b}_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{b}_{1,3\tilde{n}} \end{pmatrix},\ \begin{pmatrix} \boldsymbol{d}^*_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{d}^*_{1,3\tilde{n}} \end{pmatrix} := U_2^{-1}\cdot\begin{pmatrix} \boldsymbol{b}^*_{1,2\tilde{n}+1} \\ \vdots \\ \boldsymbol{b}^*_{1,3\tilde{n}} \end{pmatrix},$$

$\mathbb{D}_1 := (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,\tilde{n}}, \boldsymbol{d}_{1,\tilde{n}+1}, \ldots, \boldsymbol{d}_{1,3\tilde{n}}, \boldsymbol{b}_{1,3\tilde{n}+1}, \ldots, \boldsymbol{b}_{1,5\tilde{n}})$, $\mathbb{D}^*_1 := (\boldsymbol{b}^*_{1,1}, \ldots, \boldsymbol{b}^*_{1,\tilde{n}}, \boldsymbol{d}^*_{1,\tilde{n}+1}, \ldots, \boldsymbol{d}^*_{1,3\tilde{n}}, \boldsymbol{b}^*_{1,3\tilde{n}+1}, \ldots, \boldsymbol{b}^*_{1,5\tilde{n}})$. We then easily verify that $\mathbb{D}_1$ and $\mathbb{D}^*_1$ are dual orthonormal, and are distributed the same as the original bases, $\mathbb{B}_1$ and $\mathbb{B}^*_1$.

We have $(\omega'\vec{y}, \vec{f'}) \cdot Z_1 = (\omega', 0^{n+r-1})$, and since $Z_1$ is regular,

$$(\theta'_i\vec{v}_i,\ \xi' M_i) \cdot (Z_1^{-1})^{\mathrm{T}} = (\frac{\xi'}{\omega'} M_i \cdot \vec{f'},\ \theta'_i\vec{v}_i^{\geq 2},\ \xi' M_i) \ \text{ if } \vec{v}_i \cdot \vec{y} = 0. \tag{28}$$

$$(\theta''_i,\ \theta'_i\vec{v}_i^{\geq 2},\ \xi' M_i) \cdot (Z_1^{-1})^{\mathrm{T}} = (\theta'''_i,\ \theta'_i\vec{v}_i^{\geq 2},\ \xi' M_i) \ \text{ if } \vec{v}_i \cdot \vec{y} \neq 0, \tag{29}$$

where $\theta'''_i := \theta''_i + \theta'_i\vec{y}^{\geq 2} \cdot \vec{v}_i^{\geq 2} + \frac{\xi'}{\omega'} M_i \cdot \vec{f'}$ and $\theta'''_i$ is uniformly random and independent from other variables since $\theta''_i \xleftarrow{\mathsf{U}} \mathbb{F}_q$. Here, the first entry of the right hand side of Eq. (28) is determined by ratio of the inner product $(\omega'\vec{y}, \vec{f'}) \cdot (\theta'_i\vec{v}_i, \xi' M_i) = \xi' M_i \cdot \vec{f'}$ and $\omega'$, and the rest of entries are the same in both sides by definition.

Clearly, $\vec{z} := (\omega'\vec{y}, \vec{f'}) \cdot (U_2^{-1})^{\mathrm{T}}$ is independently and uniformly distributed in $\mathbb{F}_q^{n+r}$ since $U_2 \xleftarrow{\mathsf{U}} \mathcal{H}(n, r, \mathbb{F}_q)^{\times}$ and all the corresponding coefficients in keys are zero.

The challenge ciphertext in Game 3-$h$-2-$\ell$-3 is expressed over bases $\mathbb{B}_1$ and $\mathbb{D}_1$ as follows.

$$\begin{aligned}
\boldsymbol{c}_1 &= (\ \omega\vec{y},\ \vec{f},\ \omega'\vec{y},\ \vec{f'},\ \omega'\vec{y},\ \vec{f'},\ 0^{n+r},\ \vec{\varphi}_1\ )_{\mathbb{B}_1} \\
&= (\ \omega\vec{y},\ \vec{f},\ (\omega'\vec{y},\ \vec{f'}) \cdot Z_1,\ (\omega'\vec{y},\ \vec{f'}) \cdot (U_2^{-1})^{\mathrm{T}},\ 0^{n+r},\ \vec{\varphi}_1\ )_{\mathbb{D}_1} \\
&= (\ \omega\vec{y},\ \vec{f},\ \omega',\ 0^{n+r-1},\ \vec{z},\ 0^{n+r},\ \vec{\varphi}_1\ )_{\mathbb{D}_1}, \quad \text{where } \vec{z} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n+r}.
\end{aligned}$$

The $i$-th component of the $h$-th queried key $\{\boldsymbol{k}_i^*\}_{i=1}^{\ell}$ in Game 3-$h$-2-$\ell$-3 is expressed over bases $\mathbb{B}_1^*$ and $\mathbb{D}_1^*$ as follows. Using Eqs. (28), (29) and (27), we have, for $i = 1, \ldots, \ell$,

$$\begin{aligned}
\text{if } \vec{y} \cdot \vec{v}_i = 0, \quad \boldsymbol{k}_i^* &= (\ \theta_i\vec{v}_i,\ \xi M_i,\ \theta'_i\vec{v}_i,\ \xi' M_i,\ 0^{n+r},\ \psi_i\vec{v}_i,\ \eta_i M_i,\ 0^{n+r})_{\mathbb{B}_1^*} \\
&= (\ \theta_i\vec{v}_i,\ \xi M_i,\ \boxed{\tfrac{\xi'}{\omega'} M_i \cdot \vec{f'}},\ \theta'_i\vec{v}_i^{\geq 2},\ \xi' M_i,\ 0^{n+r},\ \psi_i\vec{v}_i,\ \eta_i M_i,\ 0^{n+r})_{\mathbb{D}_1^*}, \\
\text{if } \vec{y} \cdot \vec{v}_i \neq 0, \quad \boldsymbol{k}_i^* &= (\ \theta_i\vec{v}_i,\ \xi M_i,\ \theta''_i,\ \theta'_i\vec{v}_i^{\geq 2},\ \xi' M_i,\ 0^{n+r},\ \psi_i\vec{v}_i,\ \eta_i M_i,\ 0^{n+r})_{\mathbb{B}_1^*} \\
&= (\ \theta_i\vec{v}_i,\ \xi M_i,\ \boxed{\theta'''_i},\ \theta'_i\vec{v}_i^{\geq 2},\ \xi' M_i,\ 0^{n+r},\ \psi_i\vec{v}_i,\ \eta_i M_i,\ 0^{n+r})_{\mathbb{D}_1^*},
\end{aligned} \tag{30}$$

where $\theta'''_i$ is defined after Eq. (29). Therefore, we have Eq. (20) if $\vec{y} \cdot \vec{v}_i = 0$. And, the right hand side of Eq. (30) is distributed equivalently to Eq. (17).

We have only $M_i \cdot \vec{f'}$ for $i$ when $\vec{v}_i \cdot \vec{y} = 0$. From the security of liner secret sharing, the central secret $s_0 := \vec{1} \cdot \vec{f'}$ is (uniformly distributed and) independent from information $\{M_i \cdot \vec{f'}$ for $i$ when $\vec{v}_i \cdot \vec{y} = 0\}$. Since $s_0$ is the third coefficient of $\boldsymbol{c}_0$, the corresponding coefficient of $\boldsymbol{k}_0^*$ also becomes uniformly distributed and independent from all the other variables by the one-dimensional coordinate change. That is, we have Eq. (19).

Therefore, the distribution $(\mathsf{param}_{(n,r)}, \{\widehat{\mathbb{D}}_t\}_{t=0,1}, \{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \mathsf{ct}_{\Gamma})$ is equivalent to that in Game 3-$h$-3. This completes the proof of Lemma 18. □

### C.3 Proof of Lemma 20

*Proof.* Lemma 20 is proven in a similar manner to Lemma 7 in [20]. In Game 3-$\nu$-4, the 0-th components of all keys, $\boldsymbol{k}_0^{(h)*}$ for $h = 1, \ldots, \nu$, are given by $\boldsymbol{k}_0^{(h)*} = (1, \xi^{(h)}, \xi''^{(h)}, \eta_0^{(h)}, 0)_{\mathbb{B}_0^*}$ with independent randomness $\xi''^{(h)}$, and the 0-th component of the challenge ciphertext is given by $\boldsymbol{c}_0 = (\zeta, \vec{1}\cdot\vec{f}, \vec{1}\cdot\vec{f'}, 0, \varphi_0)_{\mathbb{B}_0}$. By setting $\boldsymbol{d}_{0,1}^* := \boldsymbol{b}_{0,1}^* + \theta \boldsymbol{b}_{0,3}^*$, $\boldsymbol{d}_{0,3} := \boldsymbol{b}_{0,3} - \theta \boldsymbol{b}_{0,1}$ and $\mathbb{D} := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,2}, \boldsymbol{d}_{0,3}, \boldsymbol{b}_{0,4}, \boldsymbol{b}_{0,5})$, $\mathbb{D}^* := (\boldsymbol{d}_{0,1}^*, \boldsymbol{b}_{0,2}^*, \ldots, \boldsymbol{b}_{0,5}^*)$, we obtain $\boldsymbol{k}_0^{(h)*} = (1, \xi^{(h)}, \xi''^{(h)} - \theta, \eta_0^{(h)}, 0)_{\mathbb{D}_0^*}$ and $\boldsymbol{c}_0 = (\zeta + \theta\vec{1}\cdot\vec{f'}, \vec{1}\cdot\vec{f}, \vec{1}\cdot\vec{f'}, 0, \varphi_0)_{\mathbb{D}_0}$. Since $\zeta' := \zeta + \theta\vec{1}\cdot\vec{f'}$ is uniformly random and independent from all the others, we obtain the distributions in Game 4. □

## D Adaptively Secure Multi-Use CP-ABE Scheme with Short Secret Keys

### D.1 Definition of CP-ABE

**Definition 11 (Ciphertext-Policy Attribute-Based Encryption : CP-ABE).** *A ciphertext-policy attribute-based encryption scheme consists of four algorithms.*

- $\mathsf{Setup}$ *takes as input security parameter. It outputs the public parameters* $\mathsf{pk}$ *and a master key* $\mathsf{sk}$.
- $\mathsf{KeyGen}$ *takes as input a set of attributes,* $\Gamma := \{x_j\}_{1 \le j \le n'}$, $\mathsf{pk}$ *and* $\mathsf{sk}$. *It outputs a decryption key.*
- $\mathsf{Enc}$ *takes as input public parameters* $\mathsf{pk}$, *message* $m$ *in some associated message space* $\mathsf{msg}$, *and access structure* $\mathbb{S} := (M, \rho)$. *It outputs the ciphertext.*
- $\mathsf{Dec}$ *takes as input public parameters* $\mathsf{pk}$, *decryption key* $\mathsf{sk}_\Gamma$ *for a set of attributes* $\Gamma$, *and ciphertext* $\mathsf{ct}_\mathbb{S}$ *that was encrypted under access structure* $\mathbb{S}$. *It outputs either* $m' \in \mathsf{msg}$ *or the distinguished symbol* $\perp$.

A CP-ABE scheme should have the following correctness property: for all $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda)$, all attribute sets $\Gamma$, all decryption keys $\mathsf{sk}_\Gamma \xleftarrow{\mathsf{R}} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \Gamma)$, all messages $m$, all access structures $\mathbb{S}$, all ciphertexts $\mathsf{ct}_\mathbb{S} \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m, \mathbb{S})$, it holds that $m = \mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_\Gamma, \mathsf{ct}_\mathbb{S})$ with overwhelming probability, if $\mathbb{S}$ accepts $\Gamma$.

**Definition 12.** *The model for proving the adaptively payload-hiding security of CP-ABE under chosen plaintext attack is:*

**Setup** *The challenger runs the setup algorithm,* $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda)$, *and gives the public parameters* $\mathsf{pk}$ *to the adversary.*

**Phase 1** *The adversary is allowed to issue a polynomial number of queries,* $\Gamma$, *to the challenger or oracle* $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \cdot)$ *for private keys,* $\mathsf{sk}_\Gamma$ *associated with* $\Gamma$.

**Challenge** *The adversary submits two messages* $m^{(0)}, m^{(1)}$ *and an access structure,* $\mathbb{S} := (M, \rho)$*, provided that the* $\mathbb{S}$ *does not accept any* $\Gamma$ *sent to the challenger in Phase 1. The challenger flips a random coin* $b \xleftarrow{\mathsf{U}} \{0,1\}$*, and computes* $\mathsf{ct}_{\mathbb{S}}^{(b)} \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m^{(b)}, \mathbb{S})$*. It gives* $\mathsf{ct}_{\mathbb{S}}^{(b)}$ *to the adversary.*

**Phase 2** *The adversary is allowed to issue a polynomial number of queries,* $\Gamma$*, to the challenger or oracle* $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \cdot)$ *for private keys,* $\mathsf{sk}_\Gamma$ *associated with* $\Gamma$*, provided that* $\mathbb{S}$ *does not accept* $\Gamma$*.*

**Guess** *The adversary outputs a guess* $b'$ *of* $b$*.*

*The advantage of an adversary* $\mathcal{A}$ *in the above game is defined as* $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CP\text{-}ABE},\mathsf{PH}}(\lambda) := \Pr[b' = b] - 1/2$ *for any security parameter* $\lambda$*. A CP-FE scheme is adaptively payload-hiding secure if all polynomial time adversaries have at most a negligible advantage in the above game.*

### D.2 Dual Orthonormal Basis Generator

We describe random dual orthonormal basis generator $\mathcal{G}_{\mathsf{ob}}^{\mathsf{CP}}$ below, which is used as a subroutine in the proposed CP-ABE scheme, where $\mathcal{G}_{\mathsf{ob}}^{\mathsf{KP}}$ is defined in Section 5.2.

$\mathcal{G}_{\mathsf{ob}}^{\mathsf{CP}}(1^\lambda, 5, (n,r)) :$

$(\mathsf{param}_{(n,r)}, \mathbb{D}_0, \mathbb{D}_0^*, \mathbb{D}_1, \{D_{i,j,\iota}^*, D_{i,j,l}'^*\}_{l=1,\ldots,n+r}^{i,j=1,\ldots,5;\iota=1,2}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{KP}}(1^\lambda, 5, (n,r)),$

$\mathbb{B}_0 := \mathbb{D}_0^*,\ \mathbb{B}_0^* := \mathbb{D}_0,\ \mathbb{B}_1^* := \mathbb{D}_1,\ B_{i,j,\iota} := D_{i,j,\iota}^*,\ B_{i,j,l}' := D_{i,j,l}'^*$ for all $i, j, l, \iota,$

return $(\mathsf{param}_{(n,r)}, \mathbb{B}_0, \mathbb{B}_0^*, \mathbb{B}_1^*, \{B_{i,j,\iota}, B_{i,j,l}'\}_{l=1,\ldots,n+r}^{i,j=1,\ldots,5;\iota=1,2})$.

### D.3 Construction

$\mathsf{Setup}(1^\lambda,\ (n,r)) : /*\ N_0 := 5,\ N_1 := 5(n+r)\ */$

$(\mathsf{param}_{(n,r)}, \mathbb{B}_0, \mathbb{B}_0^*, \mathbb{B}_1^*, \{B_{i,j,\iota}, B_{i,j,l}'\}_{l=1,\ldots,n+r}^{i,j=1,\ldots,5;\iota=1,2}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{CP}}(1^\lambda, 5, (n,r)),$

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,2}, \boldsymbol{b}_{0,4}),\quad \widehat{\mathbb{B}}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,2}^*, \boldsymbol{b}_{0,5}^*),$

$\widehat{\mathbb{B}}_1^* := (\boldsymbol{b}_{1,1}^*, \ldots, \boldsymbol{b}_{1,n+r}^*, \boldsymbol{b}_{1,3(n+r)+1}^*, \ldots, \boldsymbol{b}_{1,4(n+r)}^*),$

return $\mathsf{pk} := (1^\lambda, \mathsf{param}_{(n,r)}, \widehat{\mathbb{B}}_0, \{B_{i,j,\iota}, B_{i,j,l}'\}_{\iota=1,2;\, l=1,\ldots,n+r}^{i=1,4;j=1,\ldots,5}),\ \mathsf{sk} := \{\widehat{\mathbb{B}}_t^*\}_{t=0,1}.$

$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \Gamma := \{x_1, \ldots, x_{n'} \mid x_j \in \mathbb{F}_q^\times, n' \le n-1\}) :$

$\vec{y} := (y_1, \ldots, y_n)$ such that $\sum_{j=0}^{n-1} y_{n-j} z^j = z^{n-1-n'} \prod_{j=1}^{n'} (z - x_j),$

$\vec{f} \xleftarrow{\mathsf{U}} \mathbb{F}_q^r,\ \omega, \varphi_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q,\quad \vec{\varphi}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n+r},$

$\boldsymbol{k}_0^* := (1,\ \vec{1} \cdot \vec{f},\ 0,\ \varphi_0)_{\mathbb{B}_0^*},$

$$\boldsymbol{k}_1^* := (\overbrace{\quad \omega\vec{y},\ \vec{f},\quad}^{n+r}\ \overbrace{\quad 0^{2n+2r},\quad}^{2n+2r}\ \overbrace{\quad 0^{n+r},\quad}^{n+r}\ \overbrace{\quad \vec{\varphi}_1 \quad}^{n+r})_{\mathbb{B}_1^*}$$

$\mathsf{sk}_\Gamma := (\Gamma, \boldsymbol{k}_0^*,\ \boldsymbol{k}_1^*)$. return $\mathsf{sk}_\Gamma$.

$\mathsf{Enc}(\mathsf{pk},\ m,\ \mathbb{S} := (M, \rho)):\quad \zeta, \xi, \eta_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q, \qquad \boldsymbol{c}_0 := (\zeta,\ \xi,\ 0,\ \eta_0,\ 0)_{\mathbb{B}_0},$

for $i = 1, \ldots, \ell$, if $\rho(i) = v_i$, $\vec{v}_i := (v_i^{n-1}, \ldots, v_i, 1)$, $\theta_i, \psi_i, \eta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q$,

for $j = 1, \ldots, 5$,

$C_{i,1,j} := \sum_{l=1}^{n} v_{i,l}(\theta_i B'_{1,j,l} + \psi_i B'_{4,j,l}) + \sum_{l=1}^{r} M_{i,l}(\xi B'_{1,j,n+l} + \eta_i B'_{4,j,n+l})$,

$C_{i,2,j} := \theta_i B_{1,j,1} + \psi_i B_{4,j,1}$, $C_{i,3,j} := \xi B_{1,j,2} + \eta_i B_{4,j,2}$,

$c_T := g_T^{\zeta} m$, return $\mathsf{ct}_{\mathbb{S}} := (\mathbb{S}, \boldsymbol{c}_0, \{C_{i,1,j}, C_{i,2,j}, C_{i,3,j}\}_{j=1,\ldots,5}^{i=1,\ldots,\ell}, c_T)$.

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_\Gamma := (\Gamma, \boldsymbol{k}_0^*,\ \boldsymbol{k}_1^*),\ \mathsf{ct}_{\mathbb{S}} := (\mathbb{S}, \boldsymbol{c}_0, \{C_{i,1,j}, C_{i,2,j}, C_{i,3,j}\}_{j=1,\ldots,5}^{i=1,\ldots,\ell}, c_T)):$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma$, then compute $I$ and $\{\alpha_i\}_{i \in I}$ such that

$\vec{1} = \sum_{i \in I} \alpha_i M_i$, where $M_i$ is the $i$-th row of $M$, and

$I \subseteq \{i \in \{1, \ldots, \ell\} \mid \rho(i) \in \Gamma\}$.

for $i \in I$, if $\rho(i) = v_i$, $\vec{v}_i := (v_i^{n-1}, \ldots, v_i, 1)$,

$$\boldsymbol{c}_i := (\overbrace{\begin{array}{l} C_{i,1,1},\ v_{i,2}C_{i,2,1}, .., v_{i,n}C_{i,2,1},\ M_{i,1}C_{i,3,1}, .., M_{i,r}C_{i,3,1}, \\ C_{i,1,5},\ v_{i,2}C_{i,2,5}, .., v_{i,n}C_{i,2,5},\ M_{i,1}C_{i,3,5}, .., M_{i,r}C_{i,3,5} \end{array}}^{n+r} \cdots),$$

that is, $\boldsymbol{c}_i := (\overbrace{\theta_i \vec{v}_i}^{n+r},\ \overbrace{\xi M_i}^{2n+2r},\ \overbrace{0^{2n+2r}}^{n+r},\ \overbrace{\psi_i \vec{v}_i}^{n+r},\ \overbrace{\eta_i M_i}^{n+r},\ 0^{n+r})_{\mathbb{B}_1},$

$\boldsymbol{c}' := \sum_{i \in I} \alpha_i \boldsymbol{c}_i$, $K := e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot e(\boldsymbol{c}', \boldsymbol{k}_1^*)$, return $m' := c_T / K$.

[**Correctness**] If $\Gamma$ satisfies $\mathbb{S}$, $K = e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot e(\boldsymbol{c}', \boldsymbol{k}_1^*) = g_T^{-\xi s_0 + \zeta} g_T^{\xi \sum_{i \in I} \alpha_i s_i} = g_T^{\zeta}$ where $s_0 := \vec{1} \cdot \vec{f}$, $s_i := M_i \cdot \vec{f}$ for $i = 1, \ldots, \ell$.

### D.4 Security

**Theorem 4.** *The proposed multi-use CP-ABE scheme is adaptively payload-hiding against chosen plaintext attacks under the DLIN assumption.*

Theorem 4 is similarly proven to Theorem 2.

## E Comparison with the Existing Multi-Use ABE

We compare our KP-ABE scheme with existing pairing-based schemes (Table 1) and our CP-ABE scheme with existing pairing-based ones (Table 2). In particular, Table 2 shows that even considering selectively secure CP-ABE, our scheme is the first to realize multi-use compact secret keys from a static assumption.

Attrapadung-Yamada [4] propose a generic conversion between KP-ABE and CP-ABE, using the conversion, we have a dual ABE (e.g., CP-ABE) from some primal ABE (e.g., KP-ABE) with some properties. However, in both KP- and CP-ABE schemes, our schemes are the first to break one-use barrier, so, the above conversion made a step orthogonal to our contribution.

## F Definitions for NI-VC

**Definition 13 (Adaptive soundness).** *Let a publicly verifiable computation scheme* $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{ProbGen}, \mathsf{Compute}, \mathsf{Verify})$ *be for a class of functions* $\mathcal{F}$*, and let* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ *be a stateful adversary. Consider the experiment* $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{PubVC}}[\lambda]$ *below:*

$$
\begin{aligned}
&\mathbf{Exp}_{\mathcal{A}}^{\mathsf{PubVC}}[\lambda]: \ (\mathsf{PK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda),\\
&\quad (f^*, \mathsf{state}_1) \leftarrow \mathcal{A}_1(\mathsf{PK}), \quad \mathsf{EK}_{f^*} \leftarrow \mathsf{KeyGen}(\mathsf{MSK}, f^*),\\
&\quad (x^*, \mathsf{state}_2) \leftarrow \mathcal{A}_2(\mathsf{state}_1, \mathsf{PK}, \mathsf{EK}_{f^*}), \quad (\sigma_{x^*}, \mathsf{VK}_{x^*}) \leftarrow \mathsf{ProbGen}(\mathsf{PK}, x^*),\\
&\quad \sigma^*_{\mathsf{out}} \leftarrow \mathcal{A}_3(\mathsf{state}_2, \sigma_{x^*}, \mathsf{VK}_{x^*}), \quad \mathsf{Verify}(\mathsf{VK}_{x^*}, \sigma^*_{\mathsf{out}}) \rightarrow y^*,\\
&\quad \text{if } y^* \neq \perp \ \wedge \ y^* \neq f^*(x^*), \text{ output } 1, \quad \text{otherwise, output } 0.
\end{aligned}
$$

*The advantage of an adversary* $\mathcal{A}$ *is defined to be* $\Pr[\mathbf{Exp}_{\mathcal{A}}^{\mathsf{PubVC}}[\lambda] = 1]$*. A publicly verifiable computation protocol is* adaptively secure *for a class of functions* $\mathcal{F}$ *if all ppt adversaries* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ *achieve at most a negligible advantage in the above security game.*

**Efficiency** A VC protocol needs to compute two functions $\mathsf{ProbGen}$ and $\mathsf{Verify}$ *(asymptotically) faster than the function* $f$ *itself.* More precisely, Chen-Wee [9] defines the following efficiency requirement.

**Definition 14 (Efficiency).** *A publicly verifiable computation protocol is efficient for a class of functions* $\mathcal{F}$ *that act on* $n = n(\lambda)$ *bits if there is a polynomial* $p$ *such that*

- *the running time of* $\mathsf{ProbGen}$ *and* $\mathsf{Verify}$ *together is at most* $p(n, \lambda)$*, the rest of the algorithms are probabilistic polynomial-time, and*
- *there exists a function* $f \in \mathcal{F}$ *whose running time is* $\omega(p(n, \lambda))$*.*