More Efficient Adaptively Secure Revocable Hierarchical Identity-based Encryption with Compact Ciphertexts: Achieving Shorter Keys and Tighter Reductions

Atsushi Takayasu*

April 23, 2021

Abstract

Revocable hierarchical identity-based encryption (RHIBE) is a variant of the standard hierarchical identity-based encryption (HIBE) satisfying the key revocation functionality. Recently, the first adaptively secure RHIBE scheme with compact ciphertexts was proposed by Emura et al. by sacrificing the efficiency of the schemes for achieving adaptive security so that the secret keys are much larger than Seo and Emura's selectively secure scheme with compact ciphertexts. In this paper, we propose a more efficient adaptively secure RHIBE scheme with compact ciphertexts. Our scheme has much shorter secret keys and key updates than Emura et al.'s scheme. Moreover, our scheme has much shorter key updates than Seo and Emura's selectively secure scheme. Emura et al. proved the adaptive security of their scheme by reducing the security of the underlying HIBE schemes to that of their proposed RHIBE scheme, where the adaptive security of the HIBE scheme is inherently proven through the dual system encryption methodology. In contrast, we prove the adaptive security of the proposed RHIBE scheme directly through the dual system encryption methodology. Furthermore, our security proof achieves a tighter reduction than that of Emura et al.

*National Institute of Information and Communications Technology (NICT), Japan. takayasu@nict.go.jp

Contents

1 Introduction

1.1 Background

Identity-based encryption (IBE) [Sha84] is an extension of the traditional public key encryption. We can use an arbitrary string $\mathtt{ID}$ as the public key of IBE. The key generation center (KGC) of IBE takes the master public key MPK and $\mathtt{ID}$ as input and computes a secret key $\mathsf{sk}_{\mathtt{ID}}$. Hierarchical IBE (HIBE) is an extension of IBE. In HIBE, a vector of arbitrary strings $\mathtt{ID} = (\mathtt{id}_1, \ldots, \mathtt{id}_\ell)$ can be used as the public key of HIBE. In an HIBE system, not only the KGC, but also the user $\mathtt{ID}'$ with a secret key $\mathsf{sk}_{\mathtt{ID}'}$ can create a secret key $\mathsf{sk}_{\mathtt{ID}}$ iff $\mathtt{ID}'$ is a prefix of $\mathtt{ID}$. So far, several efficient and adaptively secure HIBE schemes have been proposed over prime-order pairing groups (e.g., [BKP14, CGW15, CG17, CW14, GCTC16, LP19, LP20a, LP20b, Lew12, OT15, RS14, Wat09]) through Waters' dual system encryption methodology [Wat09].

Despite the convenience of using an HIBE system, such systems do not have a naive way to revoke malicious users dynamically and efficiently. Boldyreva et al. [BGK08] resolved this problem by introducing *revocable* IBE (RIBE), a variant of IBE with a scalable revocation functionality. They proposed the first RIBE scheme by utilizing a subset cover framework [NNL01] such as the complete subtree (CS) method. Then, Seo and Emura [SE13b] refined the security model of RIBE by introducing a new security notion called decryption key exposure resistance (DKER). Later, Seo and Emura [SE13a] introduced revocable HIBE (RHIBE). Seo and Emura [SE15] and Katsumata et al. [KMT19] refined the security model by introducing the DKER and an insider security as the security requirements of RHIBE.

Although there are several adaptively secure RIBE schemes over prime-order pairing groups under the standard assumptions [LV09, ML19, SE13b, TW21], the first hierarchical analog was recently proposed by Emura et al. [ETW20]. Emura et al.'s scheme achieves compact ciphertexts or compact master public keys. Specifically, Emura et al. introduced several algebraic properties of pairing-based HIBE schemes and proposed a semi-generic construction of RHIBE from pairing-based HIBE, e.g., [CG17, CW14, GCTC16]. Thus, they used those HIBE schemes whose adaptive security was proved through the dual system encryption methodology as a building block and constructed adaptively secure RHBIE schemes. To achieve adaptive security, Emura et al. sacrificed the efficiency; their proposed RHIBE schemes have much larger secret keys than the existing selectively secure RHIBE schemes. Therefore, constructing more efficient RHIBE schemes with adaptive security is an interesting research topic. Recently, other adaptively secure RHIBE schemes have been proposed by Lee and Kim [LK21] and Emura et al. [ETW21] although both schemes cannot achieve compact ciphertexts.

1.2 Our Contribution

In this paper, we propose a more efficient adaptively secure RHIBE scheme with compact ciphertexts. Our RHIBE scheme is a modification of Chen and Gong's HIBE scheme with compact ciphertexts [CG17] that satisfies adaptive security under the standard k-linear assumption. We followed the design principle of Lee and Park's selectively secure RHIBE scheme [LP18] and constructed the proposed RHIBE scheme. Our proposed RHIBE scheme has much shorter secret keys and key updates than those of Emura et al.'s RHIBE scheme [ETW20], which was constructed from the same Chen and Gong's HIBE scheme. Moreover, our proposed RHIBE scheme has much shorter key update than Seo and Emura's selectively secure scheme with compact ciphertexts [SE15].

1.3 Technical Overview

We provide a brief overview of our proof technique. Similar to the schemes of Emura et al. and Lee and Park, the master secret key $\mathbf{k}$ of our scheme is split into $\mathtt{ID}$'s secret key $\mathsf{sk}_{\mathtt{ID}}$ and the parent user $\mathsf{pa}(\mathtt{ID})$'s key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ at time period $\mathtt{T}$ as two-out-of-two secret sharing. More concretely, $\mathsf{sk}_{\mathtt{ID}}$ contains several *sub*-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$, and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ contains several *sub*-key updates $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ that are associated with nodes θ in a binary tree $\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$ managed by $\mathsf{pa}(\mathtt{ID})$. Specifically, $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ are HIBE secret keys with $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ and $\mathbf{k} - \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as the master secret key-parts, respectively, where $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ is the uniformly random element to mask the master secret key $\mathbf{k}$. Given the key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$, a user $\mathtt{ID}$ can decrypt ciphertext $\mathsf{ct}_{\mathtt{ID},\mathtt{T}}$ in the same time period $\mathtt{T}$ iff there are $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ that share the same node θ. In other words, $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ associated with the same node θ can delete the random mask $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ and exploit the true master secret key $\mathbf{k}$.

Overview of Emura et al.'s Proof. Emura et al. [ETW20] proved the adaptive security by simply extending the technique of the selectively secure RHIBE scheme. First, they introduced the adaptive node division technique that divides all nodes θ in the security proof into exclusive two groups. Let $(\mathtt{ID}^\star, \mathtt{T}^\star)$ denote the tuple of the challenge identity and challenge time period. Emura et al.'s adaptive node division technique ensures that all $\mathsf{sk}_{\mathtt{ID},\theta}$ whose nodes are members of the first group satisfy $\mathtt{ID} \notin \mathsf{prefix}^+(\mathtt{ID}^\star)$, whereas all $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ whose nodes are members of the second group satisfy $\mathsf{pa}(\mathtt{ID}) \notin \mathsf{prefix}^+(\mathtt{ID}^\star) \vee \mathtt{T} \neq \mathtt{T}^\star$. Then, Emura et al. switched the positions of the master secret key $\mathbf{k}$ so that $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ associated with the node θ in the first group are HIBE secret keys with $\mathbf{k} - \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ and $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$, respectively, as the master secret key-parts. Therefore, the reduction algorithm itself can create all $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ associated with the node θ in the second and first groups, respectively, since the master secret key $\mathbf{k}$ is not required. Moreover, the reduction algorithm could interact with the HIBE challenger and receive $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ associated with the node θ in the first and second groups based on the conditions $\mathtt{ID} \notin \mathsf{prefix}^+(\mathtt{ID}^\star)$ and $\mathsf{pa}(\mathtt{ID}) \notin \mathsf{prefix}^+(\mathtt{ID}^\star) \vee \mathtt{T} \neq \mathtt{T}^\star$, respectively.

Here, the one problem to avoid is that the adversary can receive not only the decryption-purpose secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$, but also the delegation-purpose secret key $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$. In short, setting $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as the delegation-purpose secret keys is sufficient for achieving correctness; however, it means that the adversary can receive $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$. In this case, we cannot switch the positions of the master secret key $\mathbf{k}$ since the reduction algorithm cannot answer $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k} - \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ in the first group. Therefore, Emura et al. set the delegation-purpose secret keys $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as the HIBE secret keys with $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as the master secret key-part. As a result, even when the delegation-purpose secret key $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ becomes the HIBE secret key with $\mathbf{k} - \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ after switching the master secret key, the reduction algorithm can interact with the HIBE challenger and receive the corresponding HIBE secret keys $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ owing to the condition that $\mathsf{pa}(\mathtt{ID}) \notin \mathsf{prefix}^+(\mathtt{ID}^\star)$. In contrast, the modification to answer $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ results in a larger secret key.

Overview of Our Proof. Although Emura et al. proved the adaptive security by reducing the security of the underlying HIBE scheme to the security of their proposed RHBIE scheme, we prove the adaptive security of our proposed RHIBE scheme directly by Waters' dual system encryption methodology [Wat09] and its variants [CGW15, CG17, CW14]. To prove the security of HIBE through the dual system encryption methodology, we use *semi-functional* distributions for the challenge ciphertexts and secret keys in addition to the *normal* distributions in the real scheme. In brief, the semi-functional secret keys are HIBE secret keys for the same identity with $\mathbf{k} + \alpha\mathbf{a}^\perp$ as the master secret key-part, where $\mathbf{a}^\perp$ is a specific vector, and α is a uniformly random element in $\mathbb{Z}_p$. The normal secret keys can decrypt both normal and semi-functional ciphertexts. Although the semi-functional secret keys can decrypt normal ciphertexts, they cannot decrypt

semifuncitonal ciphertexts. In the proof, we first change the challenge ciphertexts from normal to semi-functional. Then, we change each secret key queried by the adversary from normal to semi-functional one by one. Once all the secret keys $\mathsf{sk}_{\mathtt{ID}}$ are changed to the semi-functional type, $\alpha\mathbf{a}^{\perp}$ masks the distribution of the master secret key $\mathbf{k}$; then, the plaintext of the semi-functional challenge ciphertext is information theoretically hidden.

Unlike in Emura et al.'s proof, we do not switch the positions of the master secret key $\mathbf{k}$ so that we set the delegation-purpose secret keys $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as the compact form. In turn, we change the position of the semi-functional randomness; this process is called a *semi-functional randomness switching* that was implicitly introduced by Takayasu and Watanabe [TW21]. When $\mathsf{pa}(\mathtt{ID}) \notin \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, we change all $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ to be semi-functional through the standard dual system argument. To prove $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ such that $\mathsf{pa}(\mathtt{ID}) \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, we employ the semi-functional randomness switching. Here, we provide an overview of the simplest form of the semi-functional randomness switching. If the adversary does not receive both the parent user $\mathsf{pa}(\mathtt{ID})$'s secret key $\mathsf{sk}_{\mathsf{pa}(\mathtt{ID})}$ and $\mathtt{ID}$'s secret key $\mathsf{sk}_{\mathtt{ID}}$ such that $\mathtt{ID} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, the reduction algorithm changes all secret keys $\mathsf{sk}_{\mathtt{ID}}$ from normal to be semi-functional through the standard dual system argument. Specifically, $\mathsf{sk}_{\mathtt{ID}}$ becomes the HIBE secret keys with $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} + \alpha_{\mathtt{ID},\theta}\mathbf{a}^{\perp}$ as the maser secret key-parts, where $\alpha_{\mathtt{ID},\theta}$ is the uniformly random element in $\mathbb{Z}_p$. Once all the secret keys $\mathsf{sk}_{\mathtt{ID}}$ are changed to be semi-functional, $\mathsf{sk}_{\mathtt{ID}}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ are the HIBE secret keys with $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} + \alpha_{\mathtt{ID},\theta}\mathbf{a}^{\perp}$ and $\mathbf{k} - \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$, respectively, as the master secret key-parts. Note that the adversary does not receive $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$. Observe that $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} + \alpha\mathbf{a}^{\perp}$ is the uniformly random element; thus, if we set $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} + \alpha\mathbf{a}^{\perp}$, $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ is properly distributed. Furthermore, $\mathsf{sk}_{\mathtt{ID}}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ become HIBE secret keys with $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} + (\alpha_{\mathtt{ID},\theta} - \alpha)\mathbf{a}^{\perp}$ and $\mathbf{k} + \alpha\mathbf{a}^{\perp} - \mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$, respectively, as the master secret key-parts. Here, $\alpha_{\mathtt{ID},\theta} - \alpha$ is a properly distributed uniformly random element in $\mathbb{Z}_p$. Thus, we successfully switch the positions of the semi-functional random $\alpha\mathbf{a}^{\perp}$ from $\mathsf{sk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ to $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ by using $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as the bridge. By using semi-functional randomness switching, we can change all required keys to be semi-functional and successfully prove the adaptive security of the proposed RHIBE scheme.

1.4 Related Work

Boneh and Franklin [BF01] pointed out the necessity of the revocation functionality for IBE. Boldyreva et al. [BGK08] introduced the concept of RIBE for achieving the scalable revocation and proposed the first RIBE scheme with selective security. The first adaptively secure RIBE scheme was proposed by Libert and Vergnaud [LV09]. Seo and Emura [SE13b] introduced a new security notion for RIBE called DKER and proposed the first RIBE scheme with DKER. All these schemes are constructed over pairing groups. Subsequently, several adaptively secure RIBE schemes with DKER were proposed over pairing groups [ISW17, Lee19, LLP17, WLXZ14, WES17], improving the efficiency and/or security. Then, RIBE schemes from the LWE assumption [CLL+12], the CDH assumption without pairing and the factoring assumption of Blum integers [HLCL18], and the code-based assumption [CCKS18] were proposed though they did not satisfy DKER. To break the varier of DKER without pairing, Takayasu and Watanabe [TW17] proposed a lattice-based RIBE scheme with bounded DKER. Their scheme, unlike other known RIBE schemes with DKER, satisfies the anonymity. Takayasu and Watanabe [TW21] also constructed a pairing-based anonymous RIBE scheme with bounded DKER. Katsumata et al. [KMT19] proposed the generic construction of RIBE with DKER by combining RIBE without DKER and 2-level HIBE. The result implies that RIBE without DKER implies RIBE with DKER based on [DG17]. Ma and Lin [ML19] proposed the generic construction of RIBE with DKER from 2-level HIBE.

RHIBE was first introduced by Seo and Emura [SE13a]. Unfortunately, it does not have a

convincing security definition since the adversary cannot receive the delegation-purpose secret keys $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ of *corrupted* parent users $\mathsf{pa}(\mathtt{ID})$. Seo and Emura [SE15] refined the security definition to resolve the above issue by introducing a new security notion called insider security; encryption schemes are regarded as RHIBE only when they satisfy insider security. Furthermore, they also defined DKER for RHIBE. In the security model, several RHIBE schemes were proposed over pairing-groups [ESY16, LP18, RLPL15, SE15]. Katsumata et al. [KMT19] further refined the security model and introduced a stronger notion of DKER. Katsumata et al. proposed a lattice-based RHIBE scheme, and Wang et al. [WZH+19] proposed a more efficient variant. None of these RHIBE schemes in the standard model satisfy adaptive security. Furthermore, most pairing-based RHIBE schemes [LP18, RLPL15, SE15] are based on nonstandard q-type assumptions. Emura et al. [ETW20] proposed the first adaptively secure RHIBE schemes in the standard model. They introduced several algebraic properties of known pairing-based HIBE schemes and proposed the generic construction of RHIBE from pairing-based HIBE. Thus, the instantiations capture the adaptively secures RHIBE schemes under the standard k-linear assumption. Recently, Lee and Kim [LK21] and Emura et al. [ETW21] proposed a generic construction of RHIBE from HIBE. These schemes inherently suffer from large ciphertexts.

1.5 Roadmap

In Section 2, we review the pairing groups and the definition of RHIBE. In Section 3, we propose our RHIBE scheme. In Section 4, we provide the main security theorem and its high level proof. In Sections 5 and 6, we prove the core lemmata for proving the main security theorem. In Section 7, we compare our proposed RHIBE scheme with the other known RHIBE schemes.

2 Preliminaries

For two non-negative integers a and b such that $a \leq b$, let $[a,b] \coloneqq \{a, a+1, \ldots, b\}$ and $[a] \coloneqq [1,a]$. Let a lowercase bold letter $\mathbf{a}$ and an uppercase bold letter $\mathbf{A}$ denote a column vector and matrix, respectively. Throughout the paper, let λ denote the security parameter. For a finite set S, let $x \leftarrow_R S$ denote sampling x from S uniformly at random. For two probability distributions P and Q with a support S, let $\frac{1}{2}\sum_{x\in S}|P(x)-Q(x)|$ denote the statistical distance. For two security games Game_A and Game_B, let $\mathsf{Game}_A \approx_c \mathsf{Game}_B$ denote that Game_A and Game_B are computationally indistinguishable from an adversary's view and let $\mathsf{Game}_A \equiv \mathsf{Game}_B$ denote that Game_A and Game_B are identically distributed from an adversary's view. We use the same notation $\approx_c$ and $\equiv$ for two probability distributions.

2.1 Bilinear Groups

Let $\mathcal{G}$ denote a prime-order pairing groups generator. Given the security parameter 1^λ as input, $\mathcal{G}$ outputs $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e)$, where p is a $\Theta(\lambda)$-bit prime number, $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ are cyclic groups of order p, g_1 and g_2 are the generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, and $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an efficiently computable non-degenerate bilinear map. Let $[a]_1 \coloneqq g_1^a \in \mathbb{G}_1$ denote a group element, where $a \in \mathbb{Z}_p$. Similarly, let $[\mathbf{a}]_1$ and $[\mathbf{A}]_1$ denote a vector and matrix of group elements. We use the same notations for the other groups $\mathbb{G}_2$ and $\mathbb{G}_T$. For two matrices $\mathbf{A} \in \mathbb{Z}_p^{\ell\times m}$ and $\mathbf{B} \in \mathbb{Z}_p^{\ell\times n}$, let $e([\mathbf{A}]_1, [\mathbf{B}]_2) = [\mathbf{A}^\top\mathbf{B}]_T$.

Next, we review the matrix decisional Diffie-Hellman (MDDH) assumption [EHK+17].

Definition 1 (Matrix Distribution)**.** *For a positive integer k, a matrix distribution $\mathcal{D}_k$ outputs a rank k matrix $\mathbf{A} \in \mathbb{Z}_p^{(k+1)\times k}$ and non-zero vector $\mathbf{a}^\perp \in \mathbb{Z}_p^{k+1}$ satisfying $\mathbf{A}^\top \mathbf{a}^\perp = \mathbf{0}$.*

Without loss of generality, we assume that the top $k \times k$ sub-matrix of $\mathbf{A}$ output by $\mathcal{D}_k$ is full-rank. Briefly speaking, for $\mathbf{A} \leftarrow \mathcal{D}_k$ the MDDH assumption states that $([\mathbf{A}]_1, [\mathbf{As}]_1) \approx_c ([\mathbf{A}]_1, [\mathbf{u}]_1)$ for uniformly random vectors $\mathbf{s} \leftarrow_R \mathbb{Z}_p^k$ and $\mathbf{u} \leftarrow_R \mathbb{Z}_p^{k+1}$.

Definition 2 (MDDH Assumption in $\mathbb{G}_1$)**.** *Let $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e) \leftarrow \mathcal{G}(1^\lambda)$ denote a description of a pairing group. The MDDH assumption in $\mathbb{G}_1$ states that the advantage function*

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}(\lambda) := \left|\Pr\left[\mathcal{A}(\mathcal{G}(1^\lambda), [\mathbf{A}]_1, [\mathbf{As}]_1) = 1\right] - \Pr\left[\mathcal{A}(\mathcal{G}(1^\lambda), [\mathbf{A}]_1, [\mathbf{u}]_1) = 1\right]\right|$$

is negligible in λ for all PPT adversary $\mathcal{A}$, where $\mathbf{A} \leftarrow \mathcal{D}_k, \mathbf{s} \leftarrow_R \mathbb{Z}_p^k, \mathbf{u} \leftarrow_R \mathbb{Z}_p^{k+1}$.

We also define the MDDH assumption in $\mathbb{G}_2$ in the same way. The k-linear assumption is a particular case of the MDDH assumption when the top $k \times k$ sub-matrix of $\mathbf{A}$ is a diagonal matrix with $a_i \leftarrow_R \mathbb{Z}_p^*$ in i-th diagonal and the bottom row vector of $\mathbf{A}$ is $(1, 1, \ldots, 1)$. In this case, we can set $\mathbf{a}^\perp = (a_1^{-1}, \ldots, a_k^{-1}, -1)^\top$. The symmetric external Diffie-Hellman (SXDH) assumption is a particular case of the k-linear assumption for $k = 1$.

2.2 RHIBE

In this section, we review the definition for RHIBE by following [KMT19].

Hierarchical Identities. Let $\mathcal{I}$ denote an identity space and let $\mathtt{id} \in \mathcal{I}$ denote an element identity. Let $\mathtt{ID} = (\mathtt{id}_1, \ldots, \mathtt{id}_\ell)$ denote an identity that is a vector of element identities and let $|\mathtt{ID}| := \ell$ denote the length of the identity. For $\mathtt{ID} = (\mathtt{id}_1, \ldots, \mathtt{id}_{|\mathtt{ID}|})$, let $\mathsf{pa}(\mathtt{ID}) := (\mathtt{id}_1, \ldots, \mathtt{id}_{|\mathtt{ID}|-1} = \mathtt{id}_{|\mathsf{pa}(\mathtt{ID})|})$ denote a parent of $\mathtt{ID}$ and let $\mathtt{ID}_{[\ell]} := (\mathtt{id}_1, \ldots, \mathtt{id}_\ell)$ denote a length ℓ prefix of $\mathtt{ID}$ for $\ell \le |\mathtt{ID}|$. Let $\mathsf{prefix}^+(\mathtt{ID}) := \left\{\mathtt{ID}_{[1]}, \mathtt{ID}_{[2]}, \ldots, \mathtt{ID}_{[|\mathtt{ID}|]} = \mathtt{ID}\right\}$ denote a set of identities that are prefix of $\mathtt{ID}$ and $\mathtt{ID}$ itself.

Syntax. An RHIBE scheme Π consists of six algorithms $(\mathsf{Setup}, \mathsf{Enc}, \mathsf{GenSK}, \mathsf{KeyUp}, \mathsf{GenDK}, \mathsf{Dec})$ defined as follows.

- $\mathsf{Setup}(1^\lambda, L) \to (\mathsf{MPK}, \mathsf{sk}_{\mathsf{kgc}})$: The *setup* algorithm takes security parameter 1^λ and the maximum depth of the hierarchy $L \in \mathbb{N}$ as input, and outputs a master public key MPK and the KGC's secret key $\mathsf{sk}_{\mathsf{kgc}}$.
- $\mathsf{Enc}(\mathsf{MPK}, \mathtt{ID}, \mathtt{T}, \mathsf{M}) \to \mathsf{ct}_{\mathtt{ID},\mathtt{T}}$: The *encryption* algorithm takes MPK, an identity $\mathtt{ID} \in \mathcal{I}^{|\mathtt{ID}|}$, time period $\mathtt{T} \in \mathcal{T}$, and a plaintext $\mathsf{M} \in \mathcal{M}$ as input, and outputs a ciphertext $\mathsf{ct}_{\mathtt{ID},\mathtt{T}}$.
- $\mathsf{GenSK}(\mathsf{MPK}, \mathsf{sk}_{\mathsf{pa}(\mathtt{ID})}, \mathtt{ID}) \to (\mathsf{sk}_{\mathtt{ID}}, \mathsf{sk}'_{\mathsf{pa}(\mathtt{ID})})$: The *secret key generation* algorithm takes MPK, a parent's secret key $\mathsf{sk}_{\mathsf{pa}(\mathtt{ID})}$, and an identity $\mathtt{ID} \in \mathcal{I}_{\mathsf{pa}(\mathtt{ID})}$ as input, and outputs $\mathsf{sk}_{\mathtt{ID}}$ for $\mathtt{ID}$ and the "updated" $\mathsf{sk}'_{\mathsf{pa}(\mathtt{ID})}$.
- $\mathsf{KeyUp}(\mathsf{MPK}, \mathtt{T}, \mathsf{sk}_{\mathtt{ID}}, \mathsf{RL}_{\mathtt{ID},\mathtt{T}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}) \to (\mathsf{ku}_{\mathtt{ID},\mathtt{T}}, \mathsf{sk}'_{\mathtt{ID}})$: The *key update information generation* algorithm takes MPK, $\mathtt{T} \in \mathcal{T}$, $\mathsf{sk}_{\mathtt{ID}}$ for $\mathtt{ID} \in \mathcal{I}^{|\mathtt{ID}|}$, revocation list $\mathsf{RL}_{\mathtt{ID},\mathtt{T}} \subseteq \mathcal{I}_{\mathtt{ID}}$, and a parent's key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ as input, and outputs $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ and the "updated" $\mathsf{sk}'_{\mathtt{ID}}$. As a special case, we define $\mathsf{ku}_{\mathsf{pa}(\mathsf{kgc}),\mathtt{T}} := \perp$ for all $\mathtt{T} \in \mathcal{T}$.
- $\mathsf{GenDK}(\mathsf{MPK}, \mathsf{sk}_{\mathtt{ID}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}) \to \mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ or $\perp$: The *decryption key generation* algorithm, which takes MPK, $\mathsf{sk}_{\mathtt{ID}}$ for $\mathtt{ID} \in \mathcal{I}^{|\mathtt{ID}|}$, and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ as input, and outputs a decryption key $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \in \mathcal{T}$ or the special symbol $\perp$, indicating that $\mathtt{ID}$ or some of its ancestors have been revoked.

- $\mathsf{Dec}(\mathsf{MPK}, \mathsf{dk}_{\mathtt{ID},\mathtt{T}}, \mathsf{ct}_{\mathtt{ID},\mathtt{T}}) \to \mathsf{M}$: The *decryption* algorithm takes MPK, $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{ct}_{\mathtt{ID},\mathtt{T}}$ as input, and outputs the decryption result M.

Correctness. We require ciphertext $\mathsf{ct}_{\mathtt{ID},\mathtt{T}}$ to be decrypted properly by a correctly-generated decryption key $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for the same $\mathtt{ID}$ and $\mathtt{T}$ when $\mathtt{ID}$ is not revoked at $\mathtt{T}$. In other words, for all $\lambda \in \mathbb{N}$, $L \in \mathbb{N}$, $(\mathsf{PP}, \mathsf{sk}_{\mathsf{kgc}}) \leftarrow \mathsf{Setup}(1^\lambda, L)$, $\ell \in [L]$, $\mathtt{ID} \in (\mathcal{I})^\ell$, $\mathtt{T} \in \mathcal{T}$, $\mathsf{M} \in \mathcal{M}$, $\mathsf{RL}_{\mathsf{kgc},\mathtt{T}} \subseteq \mathcal{I}$, $\mathsf{RL}_{\mathtt{ID}_{[1]},\mathtt{T}} \subseteq \mathcal{I}_{\mathtt{ID}_{[1]}}, \ldots, \mathsf{RL}_{\mathtt{ID}_{[\ell-1]},\mathtt{T}} \subseteq \mathcal{I}_{\mathtt{ID}_{[\ell-1]}}$, if $\mathtt{ID}' \notin \mathsf{RL}_{\mathsf{pa}(\mathtt{ID}'),\mathtt{T}}$ holds for all $\mathtt{ID}' \in \mathsf{prefix}^+(\mathtt{ID})$. Then, we require $\mathsf{M}' = \mathsf{M}$ to hold after executing the following procedures.

(1) $(\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}, \mathsf{sk}_{\mathsf{kgc}}) \leftarrow \mathsf{KeyUp}(\mathsf{PP}, \mathtt{T}, \mathsf{sk}_{\mathsf{kgc}}, \mathsf{RL}_{\mathsf{kgc},\mathtt{T}}, \bot)$.
(2) For all $\mathtt{ID}' \in \mathsf{prefix}^+(\mathtt{ID})$ (in short-to-long order), execute the following (2.1) and (2.2):
  (2.1) $(\mathsf{sk}_{\mathtt{ID}'}, \mathsf{sk}'_{\mathsf{pa}(\mathtt{ID}')}) \leftarrow \mathsf{GenSK}(\mathsf{PP}, \mathsf{sk}_{\mathsf{pa}(\mathtt{ID}')}, \mathtt{ID}')$.
  (2.2) $(\mathsf{ku}_{\mathtt{ID}',\mathtt{T}}, \mathsf{sk}'_{\mathtt{ID}'}) \leftarrow \mathsf{KeyUp}(\mathsf{PP}, \mathtt{T}, \mathsf{sk}_{\mathtt{ID}'}, \mathsf{RL}_{\mathtt{ID}',\mathtt{T}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}'),\mathtt{T}})$.[1]
(3) $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} \leftarrow \mathsf{GenDK}(\mathsf{PP}, \mathsf{sk}_{\mathtt{ID}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}})$.[2]
(4) $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{PP}, \mathtt{ID}, \mathtt{T}, \mathsf{M})$.
(5) $\mathsf{M}' \leftarrow \mathsf{Dec}(\mathsf{PP}, \mathsf{dk}_{\mathtt{ID},\mathtt{T}}, \mathsf{ct})$.

Security Definition. Let Π be an RHIBE scheme. Adaptive security of RHIBE is defined by a security game between adversary $\mathcal{A}$ and challenger $\mathcal{C}$. The game is parameterized by security parameter λ and polynomial $L = L(\lambda)$ representing the maximum hierarchical depth. Let a global counter $\mathtt{T}_{\mathsf{cu}}$ denote the current time period initialized as 1. $\mathtt{T}_{\mathsf{cu}}$ controlls $\mathcal{C}$'s responses to $\mathcal{A}$'s queries and the game terminates when $\mathtt{T}_{\mathsf{cu}} = |\mathcal{T}|$. Intuitively, $\mathcal{A}$ can receive all secret keys $\mathsf{sk}_{\mathtt{ID}}$, key updates $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, and decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ if they are insufficient to derive $\mathsf{dk}_{\mathtt{ID}^\star,\mathtt{T}^\star}$ for target tuple $(\mathtt{ID}^\star, \mathtt{T}^\star)$. The game proceeds as follows.

$\mathcal{C}$ runs $(\mathsf{MPK}, \mathsf{sk}_{\mathsf{kgc}}) \leftarrow \mathsf{Setup}(1^\lambda, L)$ and prepares $\mathtt{SKList}$, which initially contains $(\mathsf{kgc}, \mathsf{sk}_{\mathsf{kgc}})$, and into which pairs of $(\mathtt{ID}, \mathsf{sk}_{\mathtt{ID}})$ generated during the game are stored. When a new $\mathsf{sk}_{\mathtt{ID}}$ is generated or existing ones are updated by executing GenSK or KeyUp, $\mathcal{C}$ stores $(\mathtt{ID}, \mathsf{sk}_{\mathtt{ID}})$ or updates them in $\mathtt{SKList}$. Hereafter, we omit the descriptions of this addition/update for simplicity. Then, $\mathcal{C}$ executes $(\mathsf{ku}_{\mathsf{kgc},1}, \mathsf{sk}'_{\mathsf{kgc}}) \leftarrow \mathsf{KeyUp}(\mathsf{MPK}, \mathtt{T}_{\mathsf{cu}} = 1, \mathsf{sk}_{\mathsf{kgc}}, \mathsf{RL}_{\mathsf{kgc},1} = \emptyset, \bot)$ to generate a key update for the initial time period $\mathtt{T}_{\mathsf{cu}} = 1$ and gives $(\mathsf{MPK}, \mathsf{ku}_{\mathsf{kgc},1})$ to $\mathcal{A}$.

Then, $\mathcal{A}$ may adaptively make the following five types of a query to $\mathcal{C}$.

Secret Key Generation Query: Upon a query $\mathtt{ID} \in \mathcal{I}^{|\mathtt{ID}|}$ from $\mathcal{A}$, $\mathcal{C}$ checks if it holds that

– $(\mathtt{ID}, *) \notin \mathtt{SKList}$ and $(\mathsf{pa}(\mathtt{ID}), \mathsf{sk}_{\mathsf{pa}(\mathtt{ID})}) \in \mathtt{SKList}$ for some $\mathsf{sk}_{\mathsf{pa}(\mathtt{ID})}$.

This condition ensures that $\mathcal{C}$ has not still created $\mathsf{sk}_{\mathtt{ID}}$ and $\mathcal{C}$ has already created $\mathsf{sk}_{\mathsf{pa}(\mathtt{ID})}$. If the condition does not hold, $\mathcal{C}$ returns $\bot$ to $\mathcal{A}$. Otherwise, $\mathcal{C}$ executes $(\mathsf{sk}_{\mathtt{ID}}, \mathsf{sk}'_{\mathsf{pa}(\mathtt{ID})}) \leftarrow \mathsf{GenSK}(\mathsf{MPK}, \mathsf{sk}_{\mathsf{pa}(\mathtt{ID})}, \mathtt{ID})$. If $|\mathtt{ID}| = 1$, or $2 \le |\mathtt{ID}| \le L - 1$ and $\mathsf{pa}(\mathtt{ID}) \notin \mathsf{RL}_{\mathsf{pa}(\mathsf{pa}(\mathtt{ID})),\mathtt{T}_{\mathsf{cu}}}$, then $\mathcal{C}$ executes $(\mathsf{ku}_{\mathtt{ID},\mathtt{T}}, \mathsf{sk}'_{\mathtt{ID}}) \leftarrow \mathsf{KeyUp}(\mathsf{PP}, \mathtt{T}, \mathsf{sk}_{\mathtt{ID}}, \mathsf{RL}_{\mathtt{ID},\mathtt{T}} := \emptyset, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}})$ for $\mathtt{T} \in [\mathtt{T}_{\mathsf{cu}}]$ and returns $(\mathsf{ku}_{\mathtt{ID},\mathtt{T}})_{\mathtt{T} \in [\mathtt{T}_{\mathsf{cu}}]}$ to $\mathcal{A}$. If $2 \le |\mathtt{ID}| \le L$ and $\mathsf{pa}(\mathtt{ID}) \in \mathsf{RL}_{\mathsf{pa}(\mathsf{pa}(\mathtt{ID})),\mathtt{T}_{\mathsf{cu}}}$, then $\mathcal{C}$ executes $\mathsf{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}_{\mathsf{cu}}} \leftarrow \mathsf{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}_{\mathsf{cu}}} \cup \{\mathtt{ID}\}$ and returns nothing to $\mathcal{A}$.

Note that all $\mathtt{ID}$ in the following queries (except the challenge query) must be "activated", in the sense that $\mathsf{sk}_{\mathtt{ID}}$ has already been generated via this query; thus, $(\mathtt{ID}, \mathsf{sk}_{\mathtt{ID}}) \in \mathtt{SKList}$.

Secret Key Reveal Query: Until the challenge query, upon a query $\mathtt{ID} \in \mathcal{I}^{|\mathtt{ID}|}$ from $\mathcal{A}$, $\mathcal{C}$ finds $\mathsf{sk}_{\mathtt{ID}}$ from $\mathtt{SKList}$ and returns it to $\mathcal{A}$. After the challenge query, $\mathcal{C}$ checks if it holds that

– If $\mathtt{T}_{\mathsf{cu}} \ge \mathtt{T}^\star$ and $\mathtt{ID} \in \mathsf{prefix}^+(\mathtt{ID}^\star)$, then $\mathtt{ID}' \in \mathsf{RL}_{\mathsf{pa}(\mathtt{ID}'),\mathtt{T}^\star}$ for some $\mathtt{ID}' \in \mathsf{prefix}^+(\mathtt{ID})$.

This condition ensures that if $\mathtt{ID}$ is the ancestor of the challenge $\mathtt{ID}^\star$, $\mathtt{ID}$ or an ancestor of $\mathtt{ID}$ must be revoked by the challenge $\mathtt{T}^\star$. If the condition does not hold, $\mathcal{C}$ returns $\bot$ to $\mathcal{A}$;

[1] If $|\mathtt{ID}'| = L$, this step is skipped.
[2] Here, $\mathsf{sk}_{\mathtt{ID}}$ is the latest secret key, i.e., the result of Step (2).

otherwise, $\mathcal{C}$ finds $\mathsf{sk}_{\mathtt{ID}}$ from $\mathtt{SKList}$ and returns it to $\mathcal{A}$.

Revoke & Key Update Query: Until the challenge query, upon a query $\mathsf{RL}_{\mathtt{T_{cu}}+1} \subseteq \mathcal{I}^{\leq L}$ (denoting the set of identities to be revoked in the next time period $\mathtt{T_{cu}}+1$) from $\mathcal{A}$, $\mathcal{C}$ checks if the following conditions are satisfied simultaneously.

- $\mathsf{RL}_{\mathtt{ID},\mathtt{T_{cu}}} \subseteq \mathsf{RL}$ for all $\mathtt{ID} \in \mathcal{I}^{\leq L-1}$ that appear in $\mathtt{SKList}$.
- For all identities $\mathtt{ID}$ such that $(\mathtt{ID}, *) \in \mathtt{SKList}$ and $\mathtt{ID}' \in \mathsf{prefix}^{+}(\mathtt{ID})$, if $\mathtt{ID}' \in \mathsf{RL}$, then $\mathtt{ID} \in \mathsf{RL}$.

The first condition ensures that once $\mathtt{ID}$ has been revoked, the same $\mathtt{ID}$ must be continuously revoked. The second condition ensures that $\mathtt{ID}$ must be revoked if one of its ancestor $\mathtt{ID}' \in \mathsf{prefix}^{+}(\mathtt{ID})$ is revoked. After the challenge query, $\mathcal{C}$ also checks

- $\mathtt{ID} \in \mathsf{RL}$ if $\mathtt{ID} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, $\mathtt{T_{cu}} = \mathtt{T}^{\star} - 1$, and $\mathsf{sk}_{\mathtt{ID}'}$ for some $\mathtt{ID}' \in \mathsf{prefix}^{+}(\mathtt{ID})$ has been revealed previously by the secret key *reveal* query.

The condition ensures that once $\mathcal{A}$ receives $\mathsf{sk}_{\mathtt{ID}}$ for some $\mathtt{ID} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, the same $\mathtt{ID}$ must be revoked at $\mathtt{T}^{\star}$. If these conditions do not hold, then $\mathcal{C}$ returns $\perp$ to $\mathcal{A}$. Otherwise, $\mathcal{C}$ increments the current time period by $\mathtt{T_{cu}} \leftarrow \mathtt{T_{cu}}+1$ and executes the following operations (1) and (2) for all "activated" and non-revoked identities $\mathtt{ID}$, i.e., $\mathtt{ID} \in \mathcal{I}^{\leq L-1} \cup \{\mathsf{kgc}\}$, $(\mathtt{ID}, *) \in \mathtt{SKList}$ and $\mathtt{ID} \notin \mathsf{RL}$, in breadth-first order in the identity hierarchy.

(1) Set $\mathsf{RL}_{\mathtt{ID},\mathtt{T_{cu}}} \leftarrow \mathsf{RL} \cap \mathcal{I}_{\mathtt{ID}}$, where we define $\mathcal{I}_{\mathsf{kgc}} := \mathcal{I}$.

(2) Run $(\mathsf{ku}_{\mathtt{ID},\mathtt{T_{cu}}}, \mathsf{sk}'_{\mathtt{ID}}) \leftarrow \mathsf{KeyUp}(\mathsf{MPK}, \mathtt{T_{cu}}, \mathsf{sk}_{\mathtt{ID}}, \mathsf{RL}_{\mathtt{ID},\mathtt{T_{cu}}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T_{cu}}})$, where $\mathsf{ku}_{\mathsf{pa}(\mathsf{kgc}),\mathtt{T_{cu}}} := \perp$.

Finally, $\mathcal{C}$ returns all of the generated $\{\mathsf{ku}_{\mathtt{ID},\mathtt{T_{cu}}}\}_{(\mathtt{ID},*)\in\mathtt{SKList}\setminus\mathsf{RL}}$ to $\mathcal{A}$.

Decryption Key Reveal Query: Until the challenge query, upon a query $(\mathtt{ID}, \mathtt{T}) \in \mathcal{I}^{|\mathtt{ID}|} \times \mathcal{T}$ from $\mathcal{A}$, $\mathcal{C}$ checks

- If $\mathtt{T} \leq \mathtt{T_{cu}}$ holds.

After the challenge query, $\mathcal{C}$ also checks

- If $(\mathtt{ID}, \mathtt{T}) \neq (\mathtt{ID}^{\star}, \mathtt{T}^{\star})$ holds.

If these conditions are *not* satisfied, then $\mathcal{C}$ returns $\perp$ to $\mathcal{A}$. Otherwise, $\mathcal{C}$ finds $\mathsf{sk}_{\mathtt{ID}}$ from $\mathtt{SKList}$, runs $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} \leftarrow \mathsf{GenDK}(\mathsf{MPK}, \mathsf{sk}_{\mathtt{ID}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}})$, and returns $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ to $\mathcal{A}$.

Challenge Query: Note that $\mathcal{A}$ is permitted to make this query exactly once. Upon a query $(\mathtt{ID}^{\star}, \mathtt{T}^{\star}, \mathsf{M}_0^{\star}, \mathsf{M}_1^{\star})$ such that $|\mathsf{M}_0^{\star}| = |\mathsf{M}_1^{\star}|$ from $\mathcal{A}$, $\mathcal{C}$ determines if the following conditions are satisfied simultaneously.

- If $\mathtt{T}^{\star} \leq \mathtt{T_{cu}}$, $\mathcal{A}$ has not submitted $(\mathtt{ID}^{\star}, \mathtt{T}^{\star})$ as a decryption key reveal query.
- If $\mathtt{T}^{\star} \leq \mathtt{T_{cu}}$ and $\mathsf{sk}_{\mathtt{ID}}$ for $\mathtt{ID} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$ has been revealed to $\mathcal{A}$, then $\mathtt{ID} \in \mathsf{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}^{\star}-1}$.

If these conditions are *not* satisfied, then $\mathcal{C}$ returns $\perp$ to $\mathcal{A}$. Otherwise, $\mathcal{C}$ selects a bit $b \in \{0,1\}$ uniformly at random, runs $\mathsf{ct}^{\star} \leftarrow \mathsf{Enc}(\mathsf{MPK}, \mathtt{ID}^{\star}, \mathtt{T}^{\star}, \mathsf{M}_b^{\star})$, and returns the challenge ciphertext $\mathsf{ct}^{\star}$ to $\mathcal{A}$.

At some point, $\mathcal{A}$ outputs $b' \in \{0,1\}$ as its guess for b and terminates.

This completes the description of the game. In this game, $\mathcal{A}$'s adaptive security advantage is defined by $\mathsf{Adv}^{\mathtt{RHIBE}}_{\Pi,L,\mathcal{A}}(\lambda) := 2 \cdot |\Pr[b' = b] - 1/2|$.

Definition 3. *We say that an RHIBE scheme* Π *of depth* L *satisfies* adaptive security *if the advantage* $\mathsf{Adv}^{\mathtt{RHIBE}}_{\Pi,L,\mathcal{A}}(\lambda)$ *is negligible for all PPT adversaries* $\mathcal{A}$.

3 Proposed RHIBE Scheme

In this section, we propose an adaptively secure RHIBE scheme. First, we present the CS method in Section 3.1. Then, we present the proposed RHIBE scheme in Section 3.2. Finally, we prove the correctness of the scheme in Section 3.3.

3.1 CS Method

Before presenting the CS method, we summarize the notation of binary trees. Let $\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$ denote a binary tree with N leaves managed by a parent user $\mathsf{pa}(\mathtt{ID})$. We use θ to denote a node in a binary tree. Especially, we use η to denote a leaf node in a binary tree. For a leaf node η, let $\mathsf{Path}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta)$ denote a path in a binary tree $\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$ from the root node to the leaf node η.

In this paper, we describe the CS method as follows.

$\mathsf{CS.SetUp}(1^\lambda, \mathsf{pa}(\mathtt{ID})) \to \mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$: The setup algorithm takes the security parameter 1^λ and a parent identity $\mathsf{pa}(\mathtt{ID}) \in \mathcal{I}^{\leq L-1}$ as input, and outputs the description of a binary tree $\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$ for $\mathsf{pa}(\mathtt{ID})$.

$\mathsf{CS.Assign}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \mathcal{AL}_{\mathsf{pa}(\mathtt{ID})}, \mathtt{ID}) \to (\eta_{\mathtt{ID}}, \mathcal{AL}'_{\mathsf{pa}(\mathtt{ID})})$: The assign algorithm takes binary tree $\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$, a set of leaf nodes $\mathcal{AL}_{\mathsf{pa}(\mathtt{ID})}$, and an identity $\mathtt{ID} \in \mathcal{I}^{|\mathtt{ID}|}$, and assigns $\mathtt{ID}$ to a leaf node $\eta_{\mathtt{ID}} \in \mathcal{L}_{\mathsf{pa}(\mathtt{ID})} \setminus \mathcal{AL}_{\mathsf{pa}(\mathtt{ID})}$ and updates $\mathcal{AL}'_{\mathsf{pa}(\mathtt{ID})} \leftarrow \mathcal{AL}_{\mathsf{pa}(\mathtt{ID})} \cup \{\eta_{\mathtt{ID}}\}$. Finally, it outputs $\eta_{\mathtt{ID}}$ and $\mathcal{AL}'_{\mathsf{pa}(\mathtt{ID})}$.

$\mathsf{CS.Cover}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \mathcal{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}) \to \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$: The cover algorithm takes a binary tree $\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$ and a set of leaf nodes $\mathcal{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$, and outputs a set of nodes $\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$.

$\mathsf{CS.Match}(\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}, \eta_{\mathtt{ID}}) \to \theta$ or $\perp$: The matching algorithm takes a set of nodes $\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ output by $\mathsf{CS.Cover}$ and a leaf node $\eta_{\mathtt{ID}}$ as input, and outputs $\theta \in \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} \cap \mathsf{Path}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})$ if such a node exists; otherwise, it outputs an invalid symbol $\perp$.

The CS method satisfies the following properties:

Correctness: For any leaf node $\eta_{\mathtt{ID}} \in \mathcal{AL}_{\mathsf{pa}(\mathtt{ID})} \setminus \mathcal{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$, it holds that $\mathsf{Path}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}}) \cap \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} \neq \emptyset$.

Security: For any leaf node $\eta_{\mathtt{ID}} \in \mathcal{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$, it holds that $\mathsf{Path}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}}) \cap \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} = \emptyset$.

Scalability: It holds that $|\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}| = O(|\mathcal{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}| \log(N/|\mathcal{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}|))$.

Remark 1. *In this paper, we did not define how CS.Assign algorithm samples the leaf node $\eta_{\mathtt{ID}}$ from $\mathcal{L}_{\mathsf{pa}(\mathtt{ID})} \setminus \mathcal{AL}_{\mathsf{pa}(\mathtt{ID})}$. In most R(H)IBE schemes such as adaptively secure Emura et al.'s RHIBE schemes [ETW20], $\eta_{\mathtt{ID}}$ should be sampled from $\mathcal{L}_{\mathsf{pa}(\mathtt{ID})} \setminus \mathcal{AL}_{\mathsf{pa}(\mathtt{ID})}$ uniformly at random so that their security proof works. In contrast, our security proof does not require any conditions for the distribution of $\eta_{\mathtt{ID}}$. For example, we can set $\eta_{\mathtt{ID}}$ as the leftmost leaf node in $\mathcal{L}_{\mathsf{pa}(\mathtt{ID})} \setminus \mathcal{AL}_{\mathsf{pa}(\mathtt{ID})}$.*

3.2 Construction

Here, we provide an overview of our proposed RHIBE scheme. In our RHIBE scheme, each parent user $\mathsf{pa}(\mathtt{ID})$ manages a binary tree $\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})} \leftarrow \mathsf{CS.SetUp}(1^\lambda, \mathsf{pa}(\mathtt{ID}))$ and assigns their children users $\mathtt{ID}$ to distinct leaf nodes $\eta_{\mathtt{ID}} \leftarrow \mathsf{CS.Assign}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \mathcal{AL}_{\mathsf{pa}(\mathtt{ID})}, \mathtt{ID})$. The secret key $\mathsf{sk}_{\mathtt{ID}}$ of the user $\mathsf{sk}_{\mathtt{ID}}$ contains the *sub*-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$ associated with all nodes $\theta \in \mathsf{Path}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})$. The parent user $\mathsf{pa}(\mathtt{ID})$ sets a set of leaf nodes $\mathcal{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ so that $\eta_{\mathtt{ID}} \in \mathcal{RL}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ hold iff children users $\mathtt{ID}$ are revoked at the time period $\mathtt{T}$. The key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ of a parent user $\mathsf{pa}(\mathtt{ID})$ contains *sub*-key updates $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ associated with all nodes $\theta \in \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$. We designed the proposed RHIBE scheme so that children users $\mathtt{ID}$ can produce their decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ iff their sub-secret keys

and their parent user $\mathsf{pa}(\mathtt{ID})$'s sub-key updates share the same node. Thus, the correctness of the CS method ensures that non-revoked users $\mathtt{ID}$ can produce their decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ properly, while the security of the CS method ensures that revoked users cannot produce them. Furthermore, the scalability of the CS method ensures that the size of the key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ grows logarithmically with the maximum number of children users N. For simplicity, we set $N = \lambda^{\Omega(1)}$ so that the parent user $\mathsf{pa}(\mathtt{ID})$ can register arbitrary polynomial numbers of children users in the RHIBE scheme.

We further provide an overview of how our RHIBE scheme achieves the revocation functionality by following the Lee and Park's RHIBE scheme [LP18]. In our RHIBE scheme, the KGC has the *master secret key* $\mathsf{MSK} \in \mathbb{Z}_p^{k+1}$ as a part of $\mathsf{sk}_{\mathsf{kgc}}$. When the parent user $\mathsf{pa}(\mathtt{ID})$ creates a sub-secret key $\mathsf{sk}_{\mathtt{ID},\theta}$ or a sub-key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ associated with a node $\theta \in \mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$, the $\mathsf{pa}(\mathtt{ID})$ samples a *delegation key* $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ associated with the node θ. The sub-secret key $\mathsf{sk}_{\mathtt{ID},\theta}$ is an $\mathtt{ID}$'s HIBE secret key according to Chen and Gong's scheme achieved by setting $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as a master secret key. The key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ consists of sub-key updates $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ associated with all nodes $\theta \in \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ and a *helper key update* $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ that is independent of any nodes $\theta \in \mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}$. To create a key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$, the parent user $\mathsf{pa}(\mathtt{ID})$ samples an *ephemeral delegation key* $\overline{\mathsf{delk}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$ and creates the sub-key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ as a $\mathtt{T}$'s IBE secret key of Chen and Gong's scheme by setting $-\overline{\mathsf{delk}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} - \mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as a master secret key whereas the helper key update $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ as a multiplication of $\mathsf{pa}(\mathtt{ID})$'s HIBE secret key and $\mathtt{T}$'s IBE secret key of Chen and Gong's scheme by setting $\mathsf{MSK} + \overline{\mathsf{delk}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ as a master secret key. That is, non-revoked users have the HIBE/IBE secret keys of Chen and Gong's scheme with the master secret keys $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$, $-\overline{\mathsf{delk}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} - \mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$, and $\mathsf{MSK} + \overline{\mathsf{delk}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ for the same node θ. Thus, by multiplying all the elements, the non-revoked users can produce decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ with the master secret key MSK. In contrast, non-revoked users cannot cancel the delegation keys $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ from their own sub-secret keys and the parent user $\mathsf{pa}(\mathtt{ID})$'s key update. Thus, non-revoked users cannot produce their decryption keys.

Then, we propose the following RHIBE scheme.

$\mathsf{Setup}(1^\lambda) \to (\mathsf{MPK}, \mathsf{sk}_{\mathsf{kgc}})$: Run $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e) \leftarrow \mathcal{G}(1^\lambda)$ and sample $\mathbf{A} \leftarrow \mathcal{D}_k$, uniformly random matrices $((\mathbf{V}_\ell)_{\ell\in[0,L+2]}, \mathbf{Z}) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$, and a random vector $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$. Then, output

$$\mathsf{MPK} \coloneqq \Big([\mathbf{A}]_1, ([\mathbf{V}_\ell^\top \mathbf{A}]_1)_{\ell\in[0,L+2]}, [\mathbf{Z}]_2, ([\mathbf{V}_\ell \mathbf{Z}]_2)_{\ell\in[0,L+2]}, [\mathbf{A}^\top \mathbf{k}]_T\Big)$$

and $\mathsf{sk}_{\mathsf{kgc}} \coloneqq (\mathsf{MSK} \coloneqq \mathbf{k}, \mathtt{BT}_{\mathsf{kgc}})$, where $\mathsf{MPK} \in \mathbb{G}_1^{(k+1)\times k} \times (\mathbb{G}_1^{k\times k})^{L+3} \times \mathbb{G}_2^{k\times k} \times (\mathbb{G}_2^{(k+1)\times k})^{L+3} \times \mathbb{G}_T^k$.

$\mathsf{Enc}(\mathsf{MPK}, \mathtt{ID}, \mathtt{T}, \mathsf{M}) \to \mathsf{ct}_{\mathtt{ID},\mathtt{T}}$: Sample $\mathbf{s} \leftarrow_R \mathbb{Z}_p^k$, $(v_0, v_1, \ldots, v_{|\mathtt{ID}|}, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{|\mathtt{ID}|+2}$, then output $\mathsf{ct}_{\mathtt{ID},\mathtt{T}} \coloneqq (C_0, C_1, C_1', C_2, \mathtt{tag}, \mathtt{tag}') \in \mathbb{G}_1^{k+1} \times (\mathbb{G}_1^k)^2 \times \mathbb{G}_T \times \mathbb{Z}_p^2$;

$$\begin{aligned}
&\mathtt{tag} \coloneqq v_0 + v_1\mathtt{id}_1 + \cdots + v_{|\mathtt{ID}|}\mathtt{id}_{|\mathtt{ID}|}, \qquad \mathtt{tag}' \coloneqq v_0 + v_{L+1}\mathtt{T}, \qquad C_0 \coloneqq [\mathbf{As}]_1,\\
&C_1 \coloneqq [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|} + \mathtt{tag}\mathbf{V}_{L+2}\big)^\top \mathbf{As}]_1,\\
&C_1' \coloneqq [\big(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1} + \mathtt{tag}'\mathbf{V}_{L+2}\big)^\top \mathbf{As}]_1, \qquad C_2 \coloneqq \mathsf{M} \cdot [\mathbf{s}^\top \mathbf{A}^\top \mathbf{k}]_T.
\end{aligned}$$

$\mathsf{GenSK}(\mathsf{MPK}, \mathsf{sk}_{\mathsf{pa}(\mathtt{ID})}, \mathtt{ID}) \to \mathsf{sk}_{\mathtt{ID}}$: Run $(\eta_{\mathtt{ID}}, \mathtt{BT}'_{\mathsf{pa}(\mathtt{ID})}) \leftarrow \mathsf{CS.Assign}(\mathtt{BT}_{\mathsf{pa}(\mathtt{ID})}, \mathtt{ID})$. Parse

$$\mathsf{sk}_{\mathsf{kgc}} = (\mathbf{k}, \mathtt{BT}_{\mathsf{kgc}}, (\theta, \mathsf{delk}_{\mathsf{kgc},\theta})_{\theta\in\mathcal{AN}_{\mathsf{kgc}}})$$

or

$$\mathsf{sk}_{\mathsf{pa}(\mathtt{ID})} = \Big((\theta, \mathsf{sk}_{\mathsf{pa}(\mathtt{ID}),\theta})_{\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathsf{pa}(\mathtt{ID}))}, \eta_{\mathsf{pa}(\mathtt{ID})})}, \mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, (\theta, \mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta})_{\theta \in \mathcal{AN}_{\mathsf{pa}(\mathtt{ID})}}\Big)$$

if $\mathsf{pa}(\mathtt{ID}) \neq \mathsf{kgc}$.

Delegation Key Generation: If there is a node $\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}}) \setminus \mathcal{AN}_{\mathsf{pa}(\mathtt{ID})}$, sample a delegation key $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} \coloneqq \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ and update $\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}$ by $\mathcal{AN}'_{\mathsf{pa}(\mathtt{ID})} \leftarrow \mathcal{AN}_{\mathsf{pa}(\mathtt{ID})} \cup \{\theta\}$ until $\mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}}) \subseteq \mathcal{AN}_{\mathsf{pa}(\mathtt{ID})}$.

Sub-secret Key Generation: For each $\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})$, retrieve a delegation key $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$, sample $\mathbf{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$, and compute a sub-secret key $\mathsf{sk}_{\mathtt{ID},\theta} \coloneqq (\mathsf{SK}_{\mathtt{ID},\theta,0}, \mathsf{SK}_{\mathtt{ID},\theta,1}, \mathsf{SK}_{\mathtt{ID},\theta,2}, (\widetilde{\mathsf{SK}}_{\mathtt{ID},\theta,\ell})_{\ell \in [|\mathtt{ID}|+1, L]}) \in \mathbb{G}_2^k \times (\mathbb{G}_2^{k+1})^{L-|\mathtt{ID}|+2}$:

$$\begin{aligned}
&\mathsf{SK}_{\mathtt{ID},\theta,0} \coloneqq [\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2,\\
&\mathsf{SK}_{\mathtt{ID},\theta,1} \coloneqq [\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2,\\
&\mathsf{SK}_{\mathtt{ID},\theta,2} \coloneqq [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2, \qquad \widetilde{\mathsf{SK}}_{\mathtt{ID},\theta,\ell} \coloneqq [\mathbf{V}_\ell \mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2.
\end{aligned}$$

Finally, run $\mathsf{BT}_{\mathtt{ID}} \leftarrow \mathsf{CS.SetUp}(1^\lambda, \mathtt{ID})$, and output $\mathtt{ID}$'s secret key

$$\mathsf{sk}_{\mathtt{ID}} \coloneqq ((\theta, \mathsf{sk}_{\mathtt{ID},\theta})_{\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})}, \mathsf{BT}_{\mathtt{ID}}),$$

and an updated secret key $\mathsf{sk}'_{\mathsf{pa}(\mathtt{ID})}$

$$\mathsf{sk}'_{\mathsf{kgc}} = (\mathbf{k}, \mathsf{BT}'_{\mathsf{kgc}}, (\theta, \mathsf{delk}_{\mathsf{kgc},\theta})_{\theta \in \mathcal{AN}'_{\mathsf{kgc}}})$$

if $\mathsf{pa}(\mathtt{ID}) = \mathsf{kgc}$, or

$$\mathsf{sk}'_{\mathsf{pa}(\mathtt{ID})} = \left(\begin{array}{c} (\theta, \mathsf{sk}_{\mathsf{pa}(\mathtt{ID}),\theta})_{\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathsf{pa}(\mathtt{ID}))}, \eta_{\mathsf{pa}(\mathtt{ID})})}, \mathsf{BT}'_{\mathsf{pa}(\mathtt{ID})}, \\ (\theta, \mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta})_{\theta \in \mathcal{AN}'_{\mathsf{pa}(\mathtt{ID})}} \end{array} \right)$$

otherwise.

$\mathsf{KeyUp}(\mathsf{MPK}, \mathsf{sk}_{\mathtt{ID}}, \mathtt{T}, \mathsf{RL}_{\mathtt{ID},\mathtt{T}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}) \to (\mathsf{ku}_{\mathtt{ID},\mathtt{T}}, \mathsf{sk}'_{\mathtt{ID}})$: Run $\mathcal{KUN}_{\mathtt{ID},\mathtt{T}} \leftarrow \mathsf{CS.Cover}(\mathsf{BT}_{\mathtt{ID}}, \mathcal{RL}_{\mathtt{ID},\mathtt{T}})$. Parse

$$\mathsf{sk}_{\mathsf{kgc}} = (\mathbf{k}, \mathsf{BT}_{\mathsf{kgc}}, (\theta, \mathsf{delk}_{\mathsf{kgc},\theta})_{\theta \in \mathcal{AN}_{\mathsf{kgc}}})$$

or

$$\mathsf{sk}_{\mathtt{ID}} = \Big((\theta, \mathsf{sk}_{\mathtt{ID},\theta})_{\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})}, \mathsf{BT}_{\mathtt{ID}}, (\theta, \mathsf{delk}_{\mathtt{ID},\theta})_{\theta \in \mathcal{AN}_{\mathtt{ID}}}\Big)$$

if $\mathtt{ID} \neq \mathsf{kgc}$.

Delegation Key Generation: If there is a node $\theta \in \mathcal{KUN}_{\mathtt{ID},\mathtt{T}} \setminus \mathcal{AN}_{\mathtt{ID}}$, sample a delegation key $\mathsf{delk}_{\mathtt{ID},\theta} \coloneqq \mathbf{k}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ and update $\mathsf{BT}_{\mathtt{ID}}$ by $\mathcal{AN}'_{\mathtt{ID}} \leftarrow \mathcal{AN}_{\mathtt{ID}} \cup \{\theta\}$ until $\mathcal{KUN}_{\mathtt{ID},\mathtt{T}} \subseteq \mathcal{AN}_{\mathtt{ID}}$.

Ephemeral Delegation Key Generation: If $\mathtt{ID} = \mathsf{kgc}$, skip this step. Otherwise, sample an ephemeral delegation key $\overline{\mathsf{delk}}_{\mathtt{ID},\mathtt{T}} \coloneqq \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$.

Sub-key Update Generation: For each $\theta \in \mathcal{KUN}_{\mathtt{ID},\mathtt{T}}$, retrieve a delegation key $\mathsf{delk}_{\mathtt{ID},\theta} = \mathbf{k}_{\mathtt{ID},\theta}$ and ephemeral delegation key $\overline{\mathsf{delk}}_{\mathtt{ID},\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}$, and proceed as follows:

Case of $\mathtt{ID} = \mathsf{kgc}$: Retrieve a master secret key $\mathsf{MSK} = \mathbf{k}$, sample $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ and compute a sub-key update $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta} := (\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,0}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,2}) \in \mathbb{G}_2^k \times (\mathbb{G}_2^{k+1})^2$:

$$\begin{aligned}
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,0} &:= [\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} &:= [\mathbf{k} - \mathbf{k}_{\mathsf{kgc},\theta}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,2} &:= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2.
\end{aligned}$$

Case of $\mathtt{ID} \neq \mathsf{kgc}$: Sample $\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ and compute a sub-key update $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta} := (\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,0}, \mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,1}, \mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,2}) \in \mathbb{G}_2^k \times (\mathbb{G}_2^{k+1})^2$:

$$\begin{aligned}
\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,0} &:= [\mathbf{Z}\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,1} &:= [\mathbf{k}_{\mathtt{ID},\theta} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2^{-1} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,2} &:= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta}]_2.
\end{aligned}$$

Helper Key Update Generation: If $\mathtt{ID} = \mathsf{kgc}$, skip this step. Otherwise, run $\mathsf{GenDK}(\mathsf{MPK}, \mathsf{sk}_{\mathtt{ID}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}})$ algorithm to compute a helper decryption key $\overline{\mathsf{dk}}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell \in [|\mathtt{ID}|+1,L]})$ as in (2) or (3). Retrieve an ephemeral delegation key $\overline{\mathsf{delk}}_{\mathtt{ID},\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}$, sample $\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}, \widetilde{\overline{\mathbf{t}}}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ and compute a helper key update $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}} := (\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell \in [|\mathtt{ID}|+1,L]}) \in \mathbb{G}_2^k \times (\mathbb{G}_2^{k+1})^{L-|\mathtt{ID}|+4}$:

$$\begin{aligned}
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0} &:= \mathsf{DK}_{\mathtt{ID},\mathtt{T},0} \cdot [\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2, \\
\overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0} &:= \overline{\mathsf{DK}}'_{\mathtt{ID},\mathtt{T},0} \cdot [\mathbf{Z}\widetilde{\overline{\mathbf{t}}}'_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2, \\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1} &:= [\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot \mathsf{DK}_{\mathtt{ID},\mathtt{T},1} \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2 \\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\overline{\mathbf{t}}}'_{\mathtt{ID},\mathtt{T}}]_2 \\
&= [\mathbf{k} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2 \\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2, \qquad (1) \\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2} &:= \mathsf{DK}_{\mathtt{ID},\mathtt{T},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{V}_{L+2}\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2, \\
\overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2} &:= \overline{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\overline{\mathbf{t}}}'_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{V}_{L+2}\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2, \\
\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell} &:= \widetilde{\overline{\mathsf{DK}}}_{\mathtt{ID},\mathtt{T},\ell} \cdot [\mathbf{V}_\ell\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{V}_\ell\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2,
\end{aligned}$$

where $\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}} = \mathbf{u}_{\mathtt{ID},\mathtt{T}} + \widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}$ and $\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}} = \mathbf{u}'_{\mathtt{ID},\mathtt{T}} + \widetilde{\overline{\mathbf{t}}}'_{\mathtt{ID},\mathtt{T}}$.

Finally, output a key update and updated secret key

$$\mathsf{ku}_{\mathsf{kgc},\mathtt{T}} = (\theta, \mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta})_{\theta \in \mathcal{KUN}_{\mathsf{kgc},\mathtt{T}}}, \qquad \mathsf{sk}'_{\mathsf{kgc}} = (\mathbf{k}, \mathsf{BT}'_{\mathsf{kgc}}, (\theta, \mathsf{delk}_{\mathsf{kgc},\theta})_{\theta \in \mathcal{AN}'_{\mathsf{kgc}}})$$

if $\mathtt{ID} = \mathsf{kgc}$, or

$$\begin{aligned}
\mathsf{ku}_{\mathtt{ID},\mathtt{T}} &= \big((\theta, \mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta})_{\theta \in \mathcal{KUN}_{\mathtt{ID},\mathtt{T}}}, \overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}\big), \\
\mathsf{sk}'_{\mathtt{ID}} &= \Big((\theta, \mathsf{sk}_{\mathtt{ID},\theta})_{\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})}, \mathsf{BT}'_{\mathtt{ID}}, (\theta, \mathsf{delk}_{\mathtt{ID},\theta})_{\theta \in \mathcal{AN}'_{\mathtt{ID}}}\Big)
\end{aligned}$$

otherwise.

$\mathsf{GenDK}(\mathsf{MPK}, \mathsf{sk}_{\mathtt{ID}}, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}) \to \mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ or $\perp$: Parse

$$\mathsf{sk}_{\mathtt{ID}} = \left((\theta, \mathsf{sk}_{\mathtt{ID},\theta})_{\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})}, \mathsf{BT}_{\mathtt{ID}}, (\theta, \mathsf{delk}_{\mathtt{ID},\theta})_{\theta \in \mathcal{AN}_{\mathtt{ID}}}\right)$$

and

$$\mathsf{ku}_{\mathsf{kgc},\mathtt{T}} = (\theta, \mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta})_{\theta \in \mathcal{KUN}_{\mathsf{kgc},\mathtt{T}}}$$

if $\mathsf{pa}(\mathtt{ID}) = \mathsf{kgc}$, or

$$\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} = \left((\theta, \mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta})_{\theta \in \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}}, \overline{\mathsf{ku}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}\right)$$

otherwise.

Helper Decryption Key Generation: Run $\mathsf{CS.Match}(\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}, \eta_{\mathtt{ID}})$ to find $\tilde{\theta} \in \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} \cap \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})$ and proceed as follows:

Case of $\mathsf{pa}(\mathtt{ID}) = \mathsf{kgc}$: Retrieve

$$\begin{aligned}
\mathsf{sk}_{\mathtt{ID},\tilde{\theta}} &= (\mathsf{SK}_{\mathtt{ID},\tilde{\theta},0}, \mathsf{SK}_{\mathtt{ID},\tilde{\theta},1}, \mathsf{SK}_{\mathtt{ID},\tilde{\theta},2}, (\widetilde{\mathsf{SK}}_{\mathtt{ID},\tilde{\theta},\ell})_{\ell \in [|\mathtt{ID}|+1, L]}),\\
\mathsf{ku}_{\mathsf{kgc},\tilde{\theta}} &= (\mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},0}, \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},1}, \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},2}),
\end{aligned}$$

sample $\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}, \widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, and compute a helper decryption key $\overline{\mathsf{dk}}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell \in [2,L]}) \in \mathbb{G}_2^k \times (\mathbb{G}_2^{k+1})^{L+3}$:

$$\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},0} &\coloneqq \mathsf{SK}_{\mathtt{ID},\tilde{\theta},0} \cdot [\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}'_{\mathtt{ID},\mathtt{T},0} &\coloneqq \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\tilde{\theta},0} \cdot [\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &\coloneqq \mathsf{SK}_{\mathtt{ID},\tilde{\theta},1} \cdot \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\tilde{\theta},1} \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1)\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2\\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2\\
&= [\mathbf{k}]_2 \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1)\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2 \qquad (2)\\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},2} &\coloneqq \mathsf{SK}_{\mathtt{ID},\tilde{\theta},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}'_{\mathtt{ID},\mathtt{T},2} &\coloneqq \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\tilde{\theta},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell} &\coloneqq \widetilde{\mathsf{SK}}_{\mathtt{ID},\tilde{\theta},\ell} \cdot [\mathbf{V}_\ell\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{V}_\ell\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2,
\end{aligned}$$

where $\mathbf{u}_{\mathtt{ID},\mathtt{T}} = \mathbf{r}_{\mathtt{ID},\tilde{\theta}} + \widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}$ and $\mathbf{u}'_{\mathtt{ID},\mathtt{T}} = \mathbf{t}_{\mathsf{kgc},\mathtt{T},\tilde{\theta}} + \widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}$.

Case of $\mathsf{pa}(\mathtt{ID}) \neq \mathsf{kgc}$: Retrieve

$$\begin{aligned}
\mathsf{sk}_{\mathtt{ID},\tilde{\theta}} &= (\mathsf{SK}_{\mathtt{ID},\tilde{\theta},0}, \mathsf{SK}_{\mathtt{ID},\tilde{\theta},1}, \mathsf{SK}_{\mathtt{ID},\tilde{\theta},2}, (\widetilde{\mathsf{SK}}_{\mathtt{ID},\tilde{\theta},\ell})_{\ell \in [|\mathtt{ID}|+1, L]}),\\
\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\tilde{\theta}} &= (\mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},0}, \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},1}, \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},2}),\\
\overline{\mathsf{ku}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} &= \left(\begin{array}{c} \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},0}, \overline{\mathsf{KU}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},1}, \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},2},\\ \overline{\mathsf{KU}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},2}, (\widetilde{\overline{\mathsf{KU}}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\ell})_{\ell \in [|\mathsf{pa}(\mathtt{ID})|+1, L]} \end{array}\right),
\end{aligned}$$

sample $\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}, \widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, and compute a helper decryption key $\overline{\mathsf{dk}}_{\mathtt{ID},\mathtt{T}} \coloneqq (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell\in[|\mathtt{ID}|+1,L]}) \in \mathbb{G}_2^k \times (\mathbb{G}_2^{k+1})^{L-|\mathtt{ID}|+4}$:

$$
\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},0} &\coloneqq \mathsf{SK}_{\mathtt{ID},\tilde{\theta},0} \cdot \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},0} \cdot [\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}'_{\mathtt{ID},\mathtt{T},0} &\coloneqq \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},0} \cdot \overline{\mathsf{KU}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},0} \cdot [\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &\coloneqq \mathsf{SK}_{\mathtt{ID},\tilde{\theta},1} \cdot \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},1} \cdot \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},1} \cdot \widetilde{\overline{\mathsf{KU}}}^{\mathtt{id}_{|\mathtt{ID}|}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},|\mathtt{ID}|}\\
&\quad \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2\\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2\\
&= [\mathbf{k}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2\\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},2} &\coloneqq \mathsf{SK}_{\mathtt{ID},\tilde{\theta},2} \cdot \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2\\
&= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}'_{\mathtt{ID},\mathtt{T},2} &\coloneqq \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},2} \cdot \overline{\mathsf{KU}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2\\
&= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell} &\coloneqq \widetilde{\mathsf{SK}}_{\mathtt{ID},\tilde{\theta},\ell} \cdot \widetilde{\overline{\mathsf{KU}}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\ell} \cdot [\mathbf{V}_\ell\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 = [\mathbf{V}_\ell\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2,
\end{aligned}
\tag{3}
$$

where $\mathbf{u}_{\mathtt{ID},\mathtt{T}} = \mathbf{r}_{\mathtt{ID},\tilde{\theta}} + \overline{\mathbf{t}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} + \widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}$ and $\mathbf{u}'_{\mathtt{ID},\mathtt{T}} = \mathbf{t}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta}} + \overline{\mathbf{t}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} + \widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}$.

Finally, output $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} \coloneqq (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2}) \in (\mathbb{G}_2^k)^2 \times (\mathbb{G}_2^{k+1})^3$.

$\mathsf{Dec}(\mathsf{MPK}, \mathsf{ct}_{\mathtt{ID},\mathtt{T}}, \mathsf{dk}_{\mathtt{ID},\mathtt{T}}) \to \mathsf{M}$: Parse $\mathsf{ct}_{\mathtt{ID},\mathtt{T}} \coloneqq (C_0, C_1, C'_1, C_2, \mathtt{tag}, \mathtt{tag}')$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})$. Output

$$
\mathsf{M} = C_2 \cdot \frac{e(C_1, \mathsf{DK}_{\mathtt{ID},\mathtt{T},0}) \cdot e(C'_1, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0})}{e(C_0, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1} \cdot \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}^{\mathtt{tag}} \cdot (\mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})^{\mathtt{tag}'})}.
$$

3.3 Correctness

The correctness of the CS method ensures that $\mathsf{CS.Match}(\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}, \eta_{\mathtt{ID}})$ does not output $\perp$, and there is a node $\tilde{\theta} \in \mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} \cap \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})$ for the non-revoked user $\mathtt{ID}$. Since all $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ are computed directly, it is clear that they follow the distributions as we specified above. In contrast, we have to check that all $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ created by using $\mathsf{sk}_{\mathtt{ID},\tilde{\theta}}$ and $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\tilde{\theta}}$ or $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta}}, \overline{\mathsf{ku}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ follow the aforementioned distributions. Therefore, we first check that the helper decryption key $\overline{\mathsf{dk}}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell\in[2,L]})$ follows the distribution as specified in (2) and (3). In the following part of this section, we check the distribution of $\mathsf{DK}_{\mathtt{ID},\mathtt{T},1}$; the validity of the other elements $(\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell\in[2,L]})$ can be checked similarly.

Case of $\mathsf{pa}(\mathtt{ID}) = \mathsf{kgc}$: Since $\mathsf{sk}_{\mathtt{ID},\tilde{\theta}} = (\mathsf{SK}_{\mathtt{ID},\tilde{\theta},0}, \mathsf{SK}_{\mathtt{ID},\tilde{\theta},1}, \mathsf{SK}_{\mathtt{ID},\tilde{\theta},2}, (\widetilde{\mathsf{SK}}_{\mathtt{ID},\tilde{\theta},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ and $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\tilde{\theta}} = (\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\tilde{\theta},0}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\tilde{\theta},1}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\tilde{\theta},2})$ follow the aforementioned distributions, we have

$$
\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &= \mathsf{SK}_{\mathtt{ID},\tilde{\theta},1} \cdot \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\tilde{\theta},1} \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1)\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2\\
&= [\mathbf{k}_{\mathsf{kgc},\tilde{\theta}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\tilde{\theta}}]_2 \cdot [\mathbf{k} - \mathbf{k}_{\mathsf{kgc},\tilde{\theta}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\tilde{\theta}}]_2
\end{aligned}
$$

$$\cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1)\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2$$
$$= [\mathbf{k}]_2 \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1)\mathbf{Z}\Big(\mathbf{r}_{\mathtt{ID},\tilde{\theta}} + \widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}\Big)]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\Big(\mathbf{t}_{\mathsf{kgc},\mathtt{T},\tilde{\theta}} + \widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}\Big)]_2$$
$$= [\mathbf{k}]_2 \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1)\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2$$

as we specified in (2).

Case of $\mathsf{pa}(\mathtt{ID}) \neq \mathsf{kgc}$: $\mathsf{sk}_{\mathtt{ID},\tilde{\theta}} = (\mathsf{SK}_{\mathtt{ID},\tilde{\theta},0}, \mathsf{SK}_{\mathtt{ID},\tilde{\theta},1}, \mathsf{SK}_{\mathtt{ID},\tilde{\theta},2}, (\widetilde{\mathsf{SK}}_{\mathtt{ID},\tilde{\theta},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta}} = (\mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},0}, \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},1}, \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},2})$ follow the distributions as we specified above. When we assume that $\overline{\mathsf{ku}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} = (\overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},0}, \overline{\mathsf{KU}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},1}, \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},2}, \overline{\mathsf{KU}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},2}, (\widetilde{\overline{\mathsf{KU}}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ also follows the distribution as we specified in (1), we have

$$\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &= \mathsf{SK}_{\mathtt{ID},\tilde{\theta},1} \cdot \mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta},1} \cdot \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},1} \widetilde{\overline{\mathsf{KU}}}^{\mathtt{ID}_{|\mathtt{ID}|}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},|\mathtt{ID}|} \\
&\quad \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2 \\
&= [\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\tilde{\theta}}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\tilde{\theta}}]_2 \\
&\quad \cdot [\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\tilde{\theta}} + \overline{\mathbf{k}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}]_2^{-1} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta}}]_2 \\
&\quad \cdot [\mathbf{k} + \overline{\mathbf{k}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathsf{pa}(\mathtt{ID})|}\mathbf{V}_{|\mathsf{pa}(\mathtt{ID})|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}]_2 \\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}]_2 \cdot [\mathbf{V}_{|\mathtt{ID}|}\mathbf{Z}\overline{\mathbf{t}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}]_2^{\mathtt{id}_{|\mathtt{ID}|}} \\
&\quad \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2 \\
&= [\mathbf{k}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\Big(\mathbf{r}_{\mathtt{ID},\tilde{\theta}} + \overline{\mathbf{t}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} + \widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}\Big)]_2 \\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\Big(\mathbf{t}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\tilde{\theta}} + \overline{\mathbf{t}}'_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}} + \widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}\Big)]_2 \\
&= [\mathbf{k}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2
\end{aligned}$$

as we specified in (3).

Thus, $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})$ follows the distribution as we specified above.

Next, we check that the helper key update $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}} = (\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ follows the distribution as specified in (1). In the following, we check the distribution of $\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1}$, whereas the validity of the other elements $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}} = (\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ can be checked in the same manner. When we assume that $\overline{\mathsf{dk}}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell\in[2,L]})$ follows the distribution as specified in (2) and (3), we have

$$\begin{aligned}
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1} &= [\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot \mathsf{DK}_{\mathtt{ID},\mathtt{T},1} \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\overline{\mathbf{t}}}'_{\mathtt{ID},\mathtt{T}}]_2 \\
&= [\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \\
&\quad \cdot [\mathbf{k}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2 \\
&\quad \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\overline{\mathbf{t}}}'_{\mathtt{ID},\mathtt{T}}]_2 \\
&= [\mathbf{k} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\Big(\mathbf{u}_{\mathtt{ID},\mathtt{T}} + \widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}\Big)]_2 \\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\Big(\mathbf{u}'_{\mathtt{ID},\mathtt{T}} + \widetilde{\overline{\mathbf{t}}}'_{\mathtt{ID},\mathtt{T}}\Big)]_2 \\
&= [\mathbf{k} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2
\end{aligned}$$

as we specified in (1).

Finally, we check that the decryption succeeds. Since we have

$$
\begin{aligned}
& e(C_1, \mathsf{DK}_{\mathtt{ID},\mathtt{T},0}) \\
&= e([\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)^\top \mathbf{As}]_1 \cdot [\mathbf{V}_{L+2}^\top \mathbf{As}]_1^{\mathtt{tag}}, [\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_2) \\
&= [(\mathbf{As})^\top \big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_T \cdot [(\mathbf{As})^\top \mathbf{V}_{L+2}\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_T^{\mathtt{tag}}, \\
& e(C_1', \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}) \\
&= e([(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})^\top \mathbf{As}]_1 \cdot [\mathbf{V}_{L+2}^\top \mathbf{As}]_1^{\mathtt{tag}'}, [\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_2) \\
&= [(\mathbf{As})^\top (\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_T \cdot [(\mathbf{As})^\top \mathbf{V}_{L+2}\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_T^{\mathtt{tag}'}, \\
& e(C_0, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}) \\
&= e([\mathbf{As}]_1, [\mathbf{k}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_2) \\
&\quad \cdot e([\mathbf{As}]_1, [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_2) \\
&= [(\mathbf{As})^\top \mathbf{k}]_T \cdot [(\mathbf{As})^\top \big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_T \\
&\quad \cdot [(\mathbf{As})^\top (\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_T, \\
& e(C_0, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}^{\mathtt{tag}} \cdot (\mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})^{\mathtt{tag}'}) \\
&= e([\mathbf{As}]_1, [\mathbf{V}_{L+2}\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_2^{\mathtt{tag}} \cdot [\mathbf{V}_{L+2}\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_2^{\mathtt{tag}'}) \\
&= [(\mathbf{As})^\top \mathbf{V}_{L+2}\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_T^{\mathtt{tag}} \cdot [(\mathbf{As})^\top \mathbf{V}_{L+2}\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_T^{\mathtt{tag}'},
\end{aligned}
$$

it holds that

$$
C_2 \cdot \frac{e(C_1, \mathsf{DK}_{\mathtt{ID},\mathtt{T},0}) \cdot e(C_1', \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0})}{e(C_0, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1} \cdot \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}^{\mathtt{tag}} \cdot (\mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})^{\mathtt{tag}'})} = \mathsf{M}.
$$

4 Main Theorem

The proposed RHIBE scheme in Section 3.2 achieves the adaptive security according to the following theorem.

Theorem 1. *The proposed RHIBE scheme satisfies adaptive security if the* MDDH *assumption holds in* $\mathbb{G}_1$ *and* $\mathbb{G}_2$*. Specifically, for any PPT adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key generation queries, there exists a reduction algorithm* $\mathcal{B}_0$ *and* $\mathcal{B}_{\mathrm{I},j}$*,* $\mathcal{B}_{\mathrm{II},j}$ *for* $j \in [6]$ *such that*

$$
\begin{aligned}
& \mathsf{Adv}^{\mathtt{RHIBE}}_{\Pi,L,\mathcal{A}}(\lambda) \\
&\leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda) + Q_{\mathsf{gen}} \left(Q_{\mathsf{gen}} \sum_{i\in\{0,4\}} \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},i+j}}(\lambda) + \sum_{j\in[4]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j}}(\lambda) \right. \\
&\quad \left. + |\mathcal{T}| \cdot \sum_{j\in[2]} \left(\mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},2+j}}(\lambda) + \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},4+j}}(\lambda) \right) \right) + O\left(\frac{Q_{\mathsf{gen}}|\mathcal{T}|}{p} \right)
\end{aligned}
$$

and $\mathtt{T}(\mathcal{B}_0) \approx \max_{j\in[6]} \{\mathtt{T}(\mathcal{B}_{\mathrm{I},j}), \mathtt{T}(\mathcal{B}_{\mathrm{II},j})\} \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

4.1 Auxiliary Distributions

To prove theorem 1, we introduce the following *semi-functional* distributions of the challenge ciphertext $\mathsf{ct}^\star$, KGC's sub-key updates $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\mathtt{ID}$'s helper key updates $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$, and decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$.

Semi-functional Ciphertext: A *semi-functional* ciphertext for the target for $(\mathtt{ID}^\star, \mathtt{T}^\star)$ and a plaintext $\mathsf{M}^\star_{\mathsf{coin}}$ is defined as $\mathsf{ct}^\star = (C_0, C_1, C_1', C_2, \mathtt{tag}, \mathtt{tag}')$:

$$\begin{aligned}
\mathtt{tag} &\coloneqq v_0 + v_1 \mathtt{id}_1^\star + \cdots + v_{|\mathtt{ID}^\star|} \mathtt{id}^\star_{|\mathtt{ID}^\star|}, \qquad \mathtt{tag}' \coloneqq v_0 + v_{L+1} \mathtt{T}^\star, \\
C_0 &\coloneqq [\![\boxed{\mathbf{c}}]\!]_1, \\
C_1 &\coloneqq [\Big(\mathbf{V}_0 + \mathtt{id}_1^\star \mathbf{V}_1 + \cdots + \mathtt{id}^\star_{|\mathtt{ID}^\star|} \mathbf{V}_{|\mathtt{ID}^\star|} + \mathtt{tag} \mathbf{V}_{L+2} \Big)^\top \boxed{\mathbf{c}}]\!]_1, \\
C_1' &\coloneqq [\big(\mathbf{V}_0 + \mathtt{T}^\star \mathbf{V}_{L+1} + \mathtt{tag}' \mathbf{V}_{L+2} \big)^\top \boxed{\mathbf{c}}]\!]_1, \\
C_2 &\coloneqq \mathsf{M}_{\mathsf{coin}} \cdot [\![\boxed{\mathbf{c}}^\top \mathsf{MSK}]\!]_T,
\end{aligned} \tag{4}$$

where $(v_0, v_1, \ldots, v_{|\mathtt{ID}|}, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{|\mathtt{ID}|+2}$ and $\mathbf{c} \leftarrow_R \mathbb{Z}_p^{k+1}$. Here, the boxed parts denote the change from the *normal* ciphertext.

Semi-functional KGC's Key Updates: A *semi-functional* KGC's key update $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ for $\mathtt{T}$ is defined with the following sub-key updates $\mathsf{ku}_{\mathsf{kgc},\theta} = (\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,0}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,2})$:

$$\begin{aligned}
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,0} &\coloneqq [\mathbf{Z} \mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} &\coloneqq [\mathsf{MSK} + \boxed{\alpha \mathbf{a}^\perp} - \mathsf{delk}_{\mathsf{kgc},\theta}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T} \mathbf{V}_{L+1}) \mathbf{Z} \mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,2} &\coloneqq [\mathbf{V}_{L+2} \mathbf{Z} \mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2,
\end{aligned} \tag{5}$$

where $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ and $\alpha \leftarrow_R \mathbb{Z}_p^*$ is shared by all semi-functional $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ unless stated otherwise. Here, the boxed part denotes the change from the *normal* KGC's key update.

Semi-functional Helper Key Updates: A *semi-functional* helper key update $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ for $(\mathtt{ID}, \mathtt{T})$ is defined as $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}} = (\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell \in [|\mathtt{ID}|+1, L]})$:

$$\begin{aligned}
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0} &\coloneqq [\mathbf{Z} \bar{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0} \coloneqq [\mathbf{Z} \bar{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2, \\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1} &\coloneqq [\mathsf{MSK} + \boxed{\alpha \mathbf{a}^\perp} + \overline{\mathsf{delk}}_{\mathtt{ID},\mathtt{T}}]_2 \\
&\quad \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1 \mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|} \mathbf{V}_{|\mathtt{ID}|} \big) \mathbf{Z} \bar{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2 \\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T} \mathbf{V}_{L+1}) \mathbf{Z} \bar{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2, \\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2} &\coloneqq [\mathbf{V}_{L+2} \mathbf{Z} \bar{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2} \coloneqq [\mathbf{V}_{L+2} \mathbf{Z} \bar{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2, \\
\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell} &\coloneqq [\mathbf{V}_\ell \mathbf{Z} \bar{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2,
\end{aligned} \tag{6}$$

where $\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta}, \bar{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}, \bar{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, $\bar{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$, and $\alpha \leftarrow_R \mathbb{Z}_p^*$ is shared by all semi-functional $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ unless stated otherwise. Here, the boxed part denotes the change from the *normal* helper key update.

Semi-functional Decryption Keys: A *semi-functional* decryption key for $(\mathtt{ID}, \mathtt{T})$ is defined as

$\mathsf{dk}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})$:

$$
\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},0} &:= [\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0} := [\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &:= [\mathsf{MSK} + \boxed{\alpha\mathbf{a}^\perp}]_2 \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|})\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_2\\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_2, \qquad (7)\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},2} &:= [\mathbf{V}_{L+2}\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_2, \quad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2} := [\mathbf{V}_{L+2}\mathbf{Zu}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\widetilde{\mathsf{DK}}_{\mathtt{ID},\mathtt{T},\ell} &:= [\mathbf{V}_\ell\mathbf{Zu}_{\mathtt{ID},\mathtt{T}}]_2,
\end{aligned}
$$

where $\mathbf{u}_{\mathtt{ID},\mathtt{T}}, \mathbf{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ and $\alpha \leftarrow_R \mathbb{Z}_p^*$ is shared by all semi-functional $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ unless stated otherwise. Here, the boxed part denotes the change from the *normal* decryption key.

In brief, the above semi-functional ciphertext $\mathsf{ct}^\star$ is the same as the normal ciphertext when we set $\mathbf{c} = \mathbf{As}$, where $\mathbf{s} \leftarrow_R \mathbb{Z}_p^k$, while the above semi-functional $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ are the same as the normal ones when we set $\alpha = 0$. If KGC's sub-key updates $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ or $\mathtt{ID}$'s helper key updates $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ are semi-functional, the decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ computed by them with the normal sub-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$ become semi-functional. Both normal and semi-functional decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ can correctly decrypt normal ciphertexts, whereas the semi-functional decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ cannot correctly decrypt the semi-functional ciphertexts. By following the standard dual system argument [CGW15, CG17, CW14, Wat09], we first change the challenge ciphertext $\mathsf{ct}^\star$ to be semi-functional; then, we change a part of keys that $\mathcal{A}$ receives to be semi-functional. To this end, the semi-functional distributions of secret keys $\mathsf{sk}_{\mathtt{ID}}$, key updates $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, and decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ are defined so that the MSK is masked by $\alpha\mathbf{a}^\perp$. In other words, we do not define semi-functional distributions for sub-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathtt{ID}$'s sub-key updates $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ such that $|\mathtt{ID}| \geq 1$ since they do not contain MSK. If all information of the MSK that $\mathcal{A}$ receives is masked by $\alpha\mathbf{a}^\perp$, the standard dual system argument [CGW15, CG17, CW14] enables us to show that the plaintext $\mathsf{M}^\star_{\mathsf{coin}}$ is information theoretically hidden. The main technical hurdle to proving the security is to change all of KGC's sub-key updates $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\mathtt{ID}$'s helper key updates $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ that $\mathcal{A}$ receive to be semi-functional. Care should be taken that $\mathcal{A}$ can receive $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ for $\mathtt{T} = \mathtt{T}^\star$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{ID} \in \mathsf{prefix}^+(\mathtt{ID}^\star) \wedge \mathtt{T} = \mathtt{T}^\star$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{ID} \in \mathsf{prefix}^+(\mathtt{ID}^\star) \setminus \{\mathtt{ID}^\star\} \wedge \mathtt{T} = \mathtt{T}^\star$, which the standard dual system argument cannot change to be semi-functional. We will use the semi-functional randomness switching for changing them to semi-functional.

4.2 Proof of Main Theorem

We conclude this section by introducing the way we prove Theorem 1. By following previous security proofs of RHIBE (e.g., [ETW20, LP18, SE15]), we divide $\mathcal{A}$'s attack strategy into the following two types.

Type-I Adversary: $\mathcal{A}$ is called Type-I if it makes secret key *reveal* queries on some $\mathtt{ID} \in \mathsf{prefix}^+(\mathtt{ID}^\star)$.

Type-II Adversary: $\mathcal{A}$ is called Type-II if it does not make secret key *reveal* queries on any $\mathtt{ID} \in \mathsf{prefix}^+(\mathtt{ID}^\star)$.

Remark 2. *To be precise, previous security proofs of RHIBE (e.g., [ETW20, LP18, SE15]) further divides the Type-*I *adversary into L types depending on the value $\ell^\star \in [L]$ so that $\mathcal{A}$ receive $\mathsf{sk}_{\mathtt{ID}^\star_{[\ell^\star]}}$ whereas $\mathcal{A}$ does not receive $\mathsf{sk}_{\mathtt{ID}^\star_{[\ell]}}$ for any $\ell \in [\ell^\star - 1]$. Since our proof does not require the division, our proof saves the reduction loss by a factor $O(L)$.*

Note that the Type-I adversary and Type-II adversary are mutually exclusive and cover all the possible strategies of $\mathcal{A}$. We prove the adaptive security of the proposed RHIBE scheme against the Type-I adversary and Type-II adversary in distinct ways and obtain the following results.

Lemma 1 (Adaptive Security against the Type-I Adversary)**.** *The proposed RHIBE scheme satisfies adaptive security against the Type* I *adversary if the* MDDH *assumption holds in* $\mathbb{G}_1$ *and* $\mathbb{G}_2$*. Specifically, for any PPT Type-*I *adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key generation queries, there exists reduction algorithms* $\mathcal{B}_0$ *and* $\mathcal{B}_{\mathrm{II},j}$ *for* $j \in [6]$ *such that*

$$\mathsf{Adv}^{\mathtt{RHIBE}}_{\Pi,L,\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda) + Q_{\mathsf{gen}} \left(Q_{\mathsf{gen}} \sum_{i\in\{0,4\}} \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},i+j}}(\lambda) + |\mathcal{T}| \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},2+j}}(\lambda) + \frac{1}{p} \right)$$

and $\mathtt{T}(\mathcal{B}_0) \approx \max_{j\in[6]} \mathtt{T}(\mathcal{B}_{\mathrm{I},j}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

Lemma 2 (Adaptive Security against the Type-II Adversary)**.** *The proposed RHIBE scheme satisfies adaptive security against the Type* II *adversary if the* MDDH *assumption holds in* $\mathbb{G}_1$ *and* $\mathbb{G}_2$*. Specifically, for any PPT Type-*II *adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key generation queries, there exists a reduction algorithm* $\mathcal{B}_0$ *and* $\mathcal{B}_{\mathrm{II},j}$ *for* $j \in [6]$ *such that*

$$\begin{aligned} &\mathsf{Adv}^{\mathtt{RHIBE}}_{\Pi,L,\mathcal{A}}(\lambda) \\ &\leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda) + Q_{\mathsf{gen}} \left(\sum_{j\in[4]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j}}(\lambda) + |\mathcal{T}| \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},4+j}}(\lambda) \right) \\ &\quad + O\left(\frac{Q_{\mathsf{gen}}|\mathcal{T}|}{p} \right). \end{aligned}$$

and $\mathtt{T}(\mathcal{B}_0) \approx \max_{j\in[6]} \mathtt{T}(\mathcal{B}_{\mathrm{II},j}) \approx \mathtt{T}(\mathcal{A}) + Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

We omit the proof of Theorem 1 since it is clear from Lemmata 1 and 2. Since the Type-I adversary can receive $\mathsf{sk}_{\mathtt{ID}}$ for some $\mathtt{ID} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$ as oppose to the Type-II adversary, the proof against the Type-I adversary is more complicated than the proof against the Type-II adversary; thus, we first prove Lemma 2 in Section 5. Then, we prove Lemma 1 in Section 6.

5 Adaptive Security against the Type-II Adversary

Here, we repeat the definition of a Type-II adversary:

Type-II Adversary: $\mathcal{A}$ is called Type-II if it does not make secret key *reveal* queries on any $\mathtt{ID} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$.

We first provide an overview of Emura et al.'s proof for adaptively secure RHIBE schemes against Type-II Adversary [ETW20] and observe that a simple dual system translation cannot prove the adaptive security of our RHIBE scheme. Subsequently, we explain the proof of the adaptive security of our RHIBE scheme against the Type-II adversary.

Overview of Emura et al.'s Proof [ETW20]. Emura et al.'s proof is an adaptively secure adaptation of Seo–Emura's proof for selectively secure RHIBE schemes [SE15]. Specifically, Emura

et al. reduced the adaptive security of the underlying HIBE scheme to the adaptive security of their proposed RHIBE schemes. Similar to our secret key $\mathsf{sk}_{\mathtt{ID}}$, Emura et al.'s secret key $\mathsf{sk}_{\mathtt{ID}}$ consists of sub-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$. Furthermore, similar to our sub-secret key $\mathsf{sk}_{\mathtt{ID},\theta}$, Emura et al.'s sub-secret key $\mathsf{sk}_{\mathtt{ID},\theta}$ is an HIBE secret key with $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ as the master secret key. Although Emura et al.'s key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ does not have a helper key update $\overline{\mathsf{ku}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ than our KGC's key update $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$, the former consists of sub-key updates $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$. Furthermore, our KGC's sub-key update $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, Emura et al.'s sub-key update is an HIBE secret key with $\mathsf{MSK} - \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k} - \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ as the master secret key. To prove the adaptive security against Type-II Adversary, Emura et al. observed that all $\mathsf{delk}_{\mathtt{ID},\theta}$ and $\mathsf{sk}_{\mathtt{ID},\theta}$ that is revealed to $\mathcal{A}$ satisfy $\mathtt{ID} \notin \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$. Then, Emura et al. modified the creation of the delegation keys so that $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ is sampled by the reduction algorithm. Based on the modification, Emura et al. switched the position of MSK so that their sub-secret key $\mathsf{sk}_{\mathtt{ID},\theta}$ and sub-key update $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ are HIBE secret keys with $\mathsf{MSK} - \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ and $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$, respectively, as the master secret key. This switching enables the reduction algorithm to answer all $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ upon $\mathcal{A}$'s queries. Specifically, since all $\mathsf{sk}_{\mathtt{ID},\theta}$ that is revealed to $\mathcal{A}$ satisfy $\mathtt{ID} \notin \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, the reduction algorithm can interact with the HIBE challenger to receive an $\mathtt{ID}$'s HIBE secret key that is sufficient for creating $\mathsf{sk}_{\mathtt{ID},\theta}$. Since the reduction algorithm knows $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$, the algorithm can create all $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ by itself.

The last obstacle to overcome is answering $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ to $\mathcal{A}$. The security proof of RHIBE without insider security is relatively easy since we can neglect the obstacle. When the delegation key $\mathsf{delk}_{\mathtt{ID},\theta}$ follows the same distribution as our scheme, the above proof strategy fails unless the reduction algorithm knows the master secret key of the underlying HIBE scheme. To avoid the obstacle and achieve insider security, Emura et al. defined the delegation key $\mathsf{delk}_{\mathtt{ID},\theta}$ so that it is an HIBE secret key with $\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ as the master secret key. Then, based on the fact that $\mathtt{ID} \notin \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, the reduction algorithm interacts with the HIBE challenger to receive an $\mathtt{ID}$'s HIBE secret key that is sufficient for creating $\mathsf{delk}_{\mathtt{ID},\theta}$. Therefore, Emura et al.'s delegation key $\mathsf{delk}_{\mathtt{ID},\theta}$ becomes larger than that of ours by a factor $O(L - |\mathtt{ID}|)$.

Overview of Our Proof against the Type-II Adversary. As we observed in Section 4, the task of our proof is changing all $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ associated with MSK to be semi-functional. Here, we observe that Emura et al.'s technique cannot prove the adaptive security of our scheme. When we switch the position of MSK, we do not have to change $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ and $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ to be semi-functional. Instead, we have to change $\mathsf{delk}_{\mathtt{ID},\theta}$ and $\mathsf{sk}_{\mathtt{ID},\theta}$ associated with MSK to be semi-functional. Since all $\mathsf{sk}_{\mathtt{ID},\theta}$ that is revealed to $\mathcal{A}$ satisfy $\mathtt{ID} \notin \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, we can change all $\mathsf{sk}_{\mathtt{ID},\theta}$ to be semi-functional by properly introducing the semi-functional distribution of $\mathsf{sk}_{\mathtt{ID},\theta}$. On the other hand, we cannot change the distribution of $\mathsf{delk}_{\mathtt{ID},\theta} = \mathsf{MSK} - \mathbf{k}_{\mathtt{ID},\theta}$ unless we use larger delegation keys as adopted in Emura et al.'s scheme.

To avoid this obstacle, we employ the dual system encryption methodology and another theoretic trick that we call the *semi-functional randomness switching*. This switching was implicitly introduced by Takayasu and Watanabe [TW21] to construct adaptively secure anonymous (non-hierarchical) RIBE schemes. In the following texts, we explain how Takayasu and Watanabe changed $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}^{\star}}$ to be semi-functional. Initially, Takayasu and Watanabe changed all $\mathsf{sk}_{\mathtt{ID},\theta}$ to be semi-functional. Then, they guessed the value of $\mathtt{T}^{\star}$ with a polynomial reduction loss $|\mathcal{T}|$ and changed all $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ for $\mathtt{T} \neq \mathtt{T}^{\star}$ to be semi-functional. Finally, from these changes, they showed that *normal* and *semi-functional* $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}^{\star}}$ were identically distributed.

By following their argument and applying their strategy to the hierarchical case, we can prove the adaptive security of our RHIBE scheme. Before providing an overview of our proof, we introduce the following *seed* secret keys and its semi-functional distribution.

Normal Seed Secret Keys: A *normal* seed secret key is defined as $\mathsf{s.sk}_{\mathtt{ID}} := (\mathsf{s.SK}_{\mathtt{ID},0}, \mathsf{s.SK}_{\mathtt{ID},1},$

$\mathsf{s.SK}_{\mathtt{ID},2}, (\mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell})_{\ell\in[|\mathtt{ID}|+1,L]}$):

$$\begin{aligned}
&\mathsf{s.SK}_{\mathtt{ID},0} \coloneqq [\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2,\\
&\mathsf{s.SK}_{\mathtt{ID},1} \coloneqq [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2,\\
&\mathsf{s.SK}_{\mathtt{ID},2} \coloneqq [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2, \qquad \mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell} \coloneqq [\mathbf{V}_\ell\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2
\end{aligned} \tag{8}$$

where $\mathbf{r}_{\mathtt{ID}} \leftarrow_R \mathbb{Z}_p^k$.

Semi-functional Seed Secret Keys: A *semi-functional* seed secret key is defined as $\mathsf{s.sk}_{\mathtt{ID}} \coloneqq (\mathsf{s.SK}_{\mathtt{ID},0}, \mathsf{s.SK}_{\mathtt{ID},1}, \mathsf{s.SK}_{\mathtt{ID},2}, (\mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$:

$$\begin{aligned}
&\mathsf{s.SK}_{\mathtt{ID},0} \coloneqq [\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2,\\
&\mathsf{s.SK}_{\mathtt{ID},1} \coloneqq \boxed{[\alpha\mathbf{a}^\perp]_2} \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2,\\
&\mathsf{s.SK}_{\mathtt{ID},2} \coloneqq [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2, \qquad \mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell} \coloneqq [\mathbf{V}_\ell\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2
\end{aligned} \tag{9}$$

where $\mathbf{r}_{\mathtt{ID}} \leftarrow_R \mathbb{Z}_p^k$, and the *semi-functional randomness* $\alpha \leftarrow_R \mathbb{Z}_p^*$ is shared with all seed secret keys unless otherwise stated. Here, the boxed part denotes the change from the normal seed secret key.

We prove the adaptive security of our RHIBE scheme against the Type-II adversary based on the following sequence of games:

$\mathsf{Game}_{\mathrm{II},0}$: This is a real security game between the challenger $\mathcal{C}$ and adversary $\mathcal{A}$.

$\mathsf{Game}_{\mathrm{II},1}$: This game is the same as $\mathsf{Game}_{\mathrm{II},0}$ except that the challenge ciphertext $\mathsf{ct}^\star$ is *semi-functional*.

$\mathsf{Game}_{\mathrm{II},2}$: This game is the same as $\mathsf{Game}_{\mathrm{II},1}$ except that $\mathcal{C}$ modifies the method for creating secret keys $\mathsf{sk}_{\mathtt{ID}}$, key updates $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, and decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ as follows:

- Secret Key Creation: Upon $\mathcal{A}$'s secret key *generation* queries on $\mathtt{ID}$, $\mathcal{C}$ does not create sub-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$. Upon $\mathcal{A}$'s secret key *reveal* queries on $\mathtt{ID}$, $\mathcal{C}$ first creates a normal *seed* secret keys $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$. Then, $\mathcal{C}$ uses $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ to create all sub-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$.
- Key Update Creation: Upon $\mathcal{A}$'s secret key *generation* queries, $\mathcal{C}$ first creates *seed* secret keys $\mathsf{s.sk}_{\mathtt{ID}}^{(2)}$. $\mathcal{C}$ creates $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ in the same way as done in the real scheme. To create $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$, $\mathcal{C}$ creates sub-key updates $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ in the same way as done in the real scheme, while $\mathcal{C}$ uses $\mathsf{s.sk}_{\mathtt{ID}}^{(2)}$ to create all helper key updates $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$.
- Decryption Key Creation: Upon $\mathcal{A}$'s decryption key reveal queries, $\mathcal{C}$ does not use $\mathsf{sk}_{\mathtt{ID}}$ and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ to create $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$.

$\mathsf{Game}_{\mathrm{II},3}$: This game is the same as $\mathsf{Game}_{\mathrm{II},2}$ except that $\mathcal{C}$ creates *semi-functional* seed secret keys $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ upon $\mathcal{A}$'s secret key *reveal* queries.

$\mathsf{Game}_{\mathrm{II},4}$: This game is the same as $\mathsf{Game}_{\mathrm{II},3}$ except that $\mathcal{C}$ always creates *semi-functional* $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$.

$\mathsf{Game}_{\mathrm{II},5}$: This game is the same as $\mathsf{Game}_{\mathrm{II},4}$ except that $\mathcal{C}$ always creates *semi-functional* helper key updates $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ to create $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$.

$\mathsf{Game}_{\mathrm{II},6}$: This game is the same as $\mathsf{Game}_{\mathrm{II},5}$ except that $\mathcal{C}$ creates *semi-functional* $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ to answer $\mathcal{A}$'s decryption key reveal queries.

$\mathsf{Game}_{\mathrm{II},7}$: This game is the same as $\mathsf{Game}_{\mathrm{II},6}$ except that the challenge ciphertext $\mathsf{ct}^\star$ is the semi-functional encryption of a *random plaintext*.

Table 1: Distributions of $\mathsf{ct}^\star$, $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ for creating $\mathsf{sk}_{\mathtt{ID},\theta}$, and $\mathsf{sk}_{\mathtt{ID},\theta}$ in each game in the proof against the Type-II adversary. In the column $\mathsf{ct}^\star$, we specify the distribution and encrypted plaintext. In the other columns, we specify the distributions and semi-functional randomness of $\mathsf{s.sk}_{\mathtt{ID}}$ and $\mathsf{sk}_{\mathtt{ID},\theta}$.

Game	$\mathsf{ct}^\star$	$\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$	$\mathsf{sk}_{\mathtt{ID},\theta}$
$\mathsf{Game}_{\mathrm{II},0}$	normal $\mathsf{M}^\star_{\mathsf{coin}}$	normal	normal
$\mathsf{Game}_{\mathrm{II},1}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	normal	normal
$\mathsf{Game}_{\mathrm{II},2}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	normal	normal
$\mathsf{Game}_{\mathrm{II},3}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{II},4}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{II},5}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{II},6}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{II},7}$	semi-functional $\mathsf{M}^\star \leftarrow_R \mathbb{G}_T$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$

Table 2: Distributions of $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ for $|\mathtt{ID}| \geq 1$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ in each game in the proof against the Type-II adversary. We specify the distributions and semi-functional randomness of $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$.

Game	$\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$	$\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$	$\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$
$\mathsf{Game}_{\mathrm{II},0}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{II},1}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{II},2}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{II},3}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{II},4}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	normal	normal
$\mathsf{Game}_{\mathrm{II},5}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	normal
$\mathsf{Game}_{\mathrm{II},6}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$
$\mathsf{Game}_{\mathrm{II},7}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$

In Tables 1 and 2, we summarize the distributions of $\mathsf{ct}^\star$, $\mathsf{sk}_{\mathtt{ID}}$, $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ in each game. $\mathsf{Game}_{\mathrm{II},0}$ is the real security game. In $\mathsf{Game}_{\mathrm{II},1}$, we change the challenge ciphertext $\mathsf{ct}^\star$ to be semi-functional as per the standard dual system argument (Lemma 3). $\mathsf{Game}_{\mathrm{II},2}$ is the conceptual change that is useful to reduce the reduction loss. Thus, the indistinguishability $\mathsf{Game}_{\mathrm{II},1} \equiv \mathsf{Game}_{\mathrm{II},2}$ (Lemma 4) immediately holds. In $\mathsf{Game}_{\mathrm{II},2}$, $\mathcal{C}$ does not create $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ as the real scheme. In turn, $\mathcal{C}$ first creates *seed* secret keys $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ and $\mathsf{s.sk}_{\mathtt{ID}}^{(2)}$, and uses the seed secret keys to create $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, respectively. In $\mathsf{Game}_{\mathrm{II},3}$, all $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ revealed to $\mathcal{A}$ become *semi-functional*. We use the standard dual system argument to prove the indistinguishability $\mathsf{Game}_{\mathrm{II},2} \approx_c \mathsf{Game}_{\mathrm{II},3}$ (Lemma 5) by considering the fact that $\mathtt{ID} \notin \mathsf{prefix}^+(\mathtt{ID}^\star)$. Then, we can apply the *semi-functional randomness switching* for $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$. In $\mathsf{Game}_{\mathrm{II},4}$, all $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ become *semi-functional*. Thus, the proof of the indistinguishability $\mathsf{Game}_{\mathrm{II},3} \equiv \mathsf{Game}_{\mathrm{II},4}$ (Lemma 9) is the first main part of the proof. Here, we use the following two facts:

- All $\mathsf{sk}_{\mathtt{ID},\theta}$ revealed to $\mathcal{A}$ such that $\mathsf{pa}(\mathtt{ID}) = \mathsf{kgc}$ are created by *semi-functional* $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$.
- No $\mathsf{delk}_{\mathsf{kgc},\theta}$ are revealed to $\mathcal{A}$.

Based on the facts, the randomness of $\mathbf{k}_{\mathsf{kgc},\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ enables us to prove that *normal* and *semi-functional* $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ are identically distributed.

In $\mathsf{Game}_{\mathrm{II},5}$, all $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T},\theta}$ such that $|\mathtt{ID}| \geq 1$ are created by *semi-functional* $\mathsf{s.sk}_{\mathtt{ID}}^{(2)}$. In other words, all $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T},\theta}$ follow the *semi-functional* distribution in $\mathsf{Game}_{\mathrm{II},5}$. The proof of the indistinguishability $\mathsf{Game}_{\mathrm{II},4} \equiv \mathsf{Game}_{\mathrm{II},5}$ (Lemma 10) is the second main part of the proof, and it is more technical than the proof of $\mathsf{Game}_{\mathrm{II},3} \equiv \mathsf{Game}_{\mathrm{II},4}$. If $\mathtt{ID} \in \mathsf{prefix}^+(\mathtt{ID}^\star)$, we cannot apply the standard dual system argument to change $\mathsf{s.sk}_{\mathtt{ID}}^{(2)}$ to be semi-functional. In contrast, if $\mathtt{ID} \notin \mathsf{prefix}^+(\mathtt{ID}^\star)$, we cannot apply

the semi-functional randomness switching since $\mathsf{delk}_{\mathtt{ID},\theta}$ may be revealed to $\mathcal{A}$ via secret key *reveal* queries. Thus, in brief, the proof is a combination of the standard dual system argument and semi-functional randomness switching. By following the same procedure, either the standard dual system argument or semi-functional randomness switching enables us to prove that *normal* and *semi-functional* $\mathsf{sk}^{(2)}_{\mathtt{ID},\mathtt{T}}$ are identically distributed.

In $\mathsf{Game}_{\mathrm{II},6}$, we change all $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ to be *semi-functional* one by one. Here, the standard dual system argument is sufficient for proving the indistinguishability $\mathsf{Game}_{\mathrm{II},5} \approx_c \mathsf{Game}_{\mathrm{II},6}$ (Lemma 14) considering the fact that $(\mathtt{ID},\mathtt{T}) \neq (\mathtt{ID}^\star,\mathtt{T}^\star)$. Finally, in $\mathsf{Game}_{\mathrm{II},7}$, we change the challenge ciphertext $\mathsf{ct}^\star$ to be a semi-functional encryption of a random plaintext. Since all $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ are semi-functional, the standard dual system argument is sufficient for proving the indistinguishability $\mathsf{Game}_{\mathrm{II},6} \equiv \mathsf{Game}_{\mathrm{II},7}$ (Lemma 18).

5.1 Proof of Lemma 2

Now, we are ready to prove Lemma 2.

Proof of Lemma 2. Let $\mathsf{Adv}_i(\lambda)$ denote $\mathcal{A}$'s advantage in $\mathsf{Game}_{\mathrm{II},i}$. Hereafter, we prove that the difference of $\mathcal{A}$'s advantage between each game (i.e., $|\mathsf{Adv}_{i-1}(\lambda) - \mathsf{Adv}_i(\lambda)|$) is negligible. The key points to note is the transitions $\mathsf{Game}_{\mathrm{II},3} \equiv \mathsf{Game}_{\mathrm{II},4}$ and $\mathsf{Game}_{\mathrm{II},4} \approx_c \mathsf{Game}_{\mathrm{II},5}$ since we have to change $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ and $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $\mathtt{ID} \in \mathsf{prefix}^+(\mathtt{ID}^\star) \wedge \mathtt{T} = \mathtt{T}^\star$ to be semi-functional. In other words, we rely on Chen-Gong's technique [CG17] to prove the other transitions.

Lemma 3 ($\mathsf{Game}_{\mathrm{II},0} \approx_c \mathsf{Game}_{\mathrm{II},1}$)**.** *$\mathsf{Game}_{\mathrm{II},0}$ and $\mathsf{Game}_{\mathrm{II},1}$ are computationally indistinguishable under the* MDDH *assumption in $\mathbb{G}_1$. Specifically, for any PPT adversary $\mathcal{A}$ making at most Q_{gen} secret key generation queries, there exists a reduction algorithm $\mathcal{B}_0$ such that*

$$|\mathsf{Adv}_0(\lambda) - \mathsf{Adv}_1(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda)$$

and $\mathtt{T}(\mathcal{B}_0) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

The proof completely follows the same step of Chen-Gong [CG17]. For the completeness, we formally prove Lemma 3.

Proof of Lemma 3. The reduction algorithm $\mathcal{B}_0$ is given an MDDH instance in $\mathbb{G}_1$: $(\mathcal{G}(1^\lambda), [\mathbf{A}]_1, [\mathbf{c}]_1 = [\mathbf{As}+\mathbf{e}]_1)$, where $\mathbf{A} \leftarrow_R \mathcal{D}_k, \mathbf{s} \leftarrow_R \mathbb{Z}_p^k$, $\mathbf{e} = \mathbf{0}$ or $\mathbf{e} \leftarrow_R \mathbb{Z}_p^{k+1}$. Then, $\mathcal{B}_0$ samples random matrices $((\mathbf{V}_\ell)_{\ell\in[0,L+2]}, \mathbf{Z}) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$, and a random vector $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$ uniformly at random. $\mathcal{B}_0$ then returns

$$\mathsf{MPK} = \left([\mathbf{A}]_1, ([\mathbf{V}_\ell^\top \mathbf{A}]_1)_{\ell\in[0,L+2]}, [\mathbf{Z}]_2, ([\mathbf{V}_\ell \mathbf{Z}]_2)_{\ell\in[0,L+2]}, [\mathbf{A}^\top \mathbf{k}]_T\right)$$

to $\mathcal{A}$. Since $\mathcal{B}_0$ knows $\mathsf{MSK} = \mathbf{k}$, it can answer all $\mathcal{A}$'s key queries in the same way as the real scheme.

Upon $\mathcal{A}$'s challenge query on $(\mathtt{ID}^\star, \mathtt{T}^\star, \mathsf{M}^\star_0, \mathsf{M}^\star_1)$, $\mathcal{B}_0$ samples $\mathsf{coin} \leftarrow_R \{0,1\}$, $(v_0, v_1, \ldots, v_{|\mathtt{ID}^\star|}, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{|\mathtt{ID}^\star|+2}$, and returns the challenge ciphertext $\mathsf{ct}^\star = (C_0, C_1, C_1', C_2, \mathtt{tag}, \mathtt{tag}')$:

$$\mathtt{tag} = v_0 + v_1 \mathtt{id}^\star_1 + \cdots + v_{|\mathtt{ID}^\star|} \mathtt{id}^\star_{|\mathtt{ID}^\star|}, \quad \mathtt{tag}' = v_0 + v_{L+1} \mathtt{T}^\star, \quad C_0 = [\mathbf{c}]_1,$$
$$C_1 = [\left(\mathbf{V}_0 + \mathtt{id}^\star_1 \mathbf{V}_1 + \cdots + \mathtt{id}^\star_{|\mathtt{ID}^\star|} \mathbf{V}_{|\mathtt{ID}^\star|} + \mathtt{tag} \mathbf{V}_{L+2}\right)^\top \mathbf{c}]_1,$$
$$C_1' = [\left(\mathbf{V}_0 + \mathtt{T}^\star \mathbf{V}_{L+1} + \mathtt{tag}' \mathbf{V}_{L+2}\right)^\top \mathbf{c}]_1, \qquad C_2 = \mathsf{M}^\star_{\mathsf{coin}} \cdot [\mathbf{c}^\top \mathsf{MSK}]_T,$$

to $\mathcal{A}$. If $\mathbf{e} = \mathbf{0}$, $\mathsf{ct}^\star$ is a *normal* ciphertext as in $\mathsf{Game}_{\mathrm{II},0}$. Otherwise, $\mathsf{ct}^\star$ is a *semi-functional* ciphertext as in $\mathsf{Game}_{\mathrm{II},1}$. Thus, we complete the proof. □

Lemma 4 ($\mathsf{Game}_{\mathrm{II},1} \equiv \mathsf{Game}_{\mathrm{II},2}$)**.** $\mathsf{Game}_{\mathrm{II},1}$ *and* $\mathsf{Game}_{\mathrm{II},2}$ *are identically distributed from* $\mathcal{A}$*'s view. Specifically, for any PPT Type-*II *adversary* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}_{\mathrm{II},1}(\lambda) = \mathsf{Adv}_{\mathrm{II},2}(\lambda).$$

The proof is clear since the key creations of our scheme are path-oblivious.

Proof of Lemma 4. $\mathcal{C}$ creates MPK in the same way as the real scheme. Hereafter, we describe how $\mathcal{C}$ creates $\mathsf{sk}_{\mathtt{ID}}$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{ct}^\star$ in $\mathsf{Game}_{\mathrm{II},2}$.

Secret Key Creation: Upon $\mathcal{A}$'s secret key *generation* query on $\mathtt{ID}$, $\mathcal{C}$ runs $(\eta_{\mathtt{ID}}, \mathsf{BT}'_{\mathsf{pa}(\mathtt{ID})}) \leftarrow \mathsf{CS.Assign}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \mathtt{ID})$ and performs the delegation key generation in the same way as the real scheme and runs $\mathsf{BT}_{\mathtt{ID}} \leftarrow \mathsf{CS.SetUp}(1^\lambda, \mathtt{ID})$.

Upon $\mathcal{A}$'s secret key *reveal* query on $\mathtt{ID}$, $\mathcal{C}$ samples $\mathbf{r}^{(1)}_{\mathtt{ID}} \leftarrow_R \mathbb{Z}_p^k$ and creates a *normal* seed secret key $\mathsf{s.sk}_{\mathtt{ID}} = (\mathsf{s.SK}^{(1)}_{\mathtt{ID},0}, \mathsf{s.SK}^{(1)}_{\mathtt{ID},1}, \mathsf{s.SK}^{(1)}_{\mathtt{ID},2}, (\mathsf{s.}\widetilde{\mathsf{SK}}^{(1)}_{\mathtt{ID},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ by computing (8). Then, for each $\theta \in \mathsf{Path}(\mathsf{BT}_{\mathsf{pa}(\mathtt{ID})}, \eta_{\mathtt{ID}})$, $\mathcal{C}$ retrieves the delegation key $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$, samples $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$ and $\widetilde{\mathbf{r}}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$, and computes a sub-secret key $\mathsf{sk}_{\mathtt{ID},\theta} = (\mathsf{SK}_{\mathtt{ID},\theta,0}, \mathsf{SK}_{\mathtt{ID},\theta,1}, \mathsf{SK}_{\mathtt{ID},\theta,2}, (\widetilde{\mathsf{SK}}_{\mathtt{ID},\theta,\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$:

$$\begin{aligned}
\mathsf{SK}_{\mathtt{ID},\theta,0} &= (\mathsf{s.SK}^{(1)}_{\mathtt{ID},0})^{\tilde{r}_{\mathtt{ID},\theta}} \cdot [\mathbf{Z}\widetilde{\mathbf{r}}_{\mathtt{ID},\theta}]_2,\\
\mathsf{SK}_{\mathtt{ID},\theta,1} &= [\mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}]_2 \cdot (\mathsf{s.SK}^{(1)}_{\mathtt{ID},1})^{\tilde{r}_{\mathtt{ID},\theta}} \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\widetilde{\mathbf{r}}_{\mathtt{ID},\theta}]_2,\\
\mathsf{SK}_{\mathtt{ID},\theta,2} &= (\mathsf{s.SK}^{(1)}_{\mathtt{ID},2})^{\tilde{r}_{\mathtt{ID},\theta}} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{r}}_{\mathtt{ID},\theta}]_2,\\
\widetilde{\mathsf{SK}}_{\mathtt{ID},\theta,\ell} &= (\mathsf{s.}\widetilde{\mathsf{SK}}^{(1)}_{\mathtt{ID},\ell})^{\tilde{r}_{\mathtt{ID},\theta}} \cdot [\mathbf{V}_\ell\mathbf{Z}\widetilde{\mathbf{r}}_{\mathtt{ID},\theta}]_2.
\end{aligned} \tag{10}$$

The distribution is the same as in $\mathsf{Game}_{\mathrm{II},1}$ by setting $\mathbf{r}_{\mathtt{ID},\theta} = \tilde{r}_{\mathtt{ID},\theta} \cdot \mathbf{r}^{(1)}_{\mathtt{ID}} + \widetilde{\mathbf{r}}_{\mathtt{ID},\theta}$. Due to the fresh random $\widetilde{\mathbf{r}}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$, $\mathbf{r}_{\mathtt{ID},\theta}$ is distributed in $\mathbb{Z}_p^k$ uniformly at random.

Key Update Creation: Upon $\mathcal{A}$'s secret key *generation* query on $\mathtt{ID}$, $\mathcal{C}$ samples $\mathbf{r}^{(2)}_{\mathtt{ID}} \leftarrow_R \mathbb{Z}_p^k$ and creates a *normal* seed secret key $\mathsf{s.sk}^{(2)}_{\mathtt{ID}} = (\mathsf{s.SK}^{(2)}_{\mathtt{ID},0}, \mathsf{s.SK}^{(2)}_{\mathtt{ID},1}, \mathsf{s.SK}^{(2)}_{\mathtt{ID},2}, (\mathsf{s.}\widetilde{\mathsf{SK}}^{(2)}_{\mathtt{ID},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ by computing (8). $\mathcal{C}$ runs $\mathcal{KUN}_{\mathtt{ID},\mathtt{T}} \leftarrow \mathsf{CS.Cover}(\mathsf{BT}_{\mathtt{ID}}, \mathcal{RL}_{\mathtt{ID},\mathtt{T}})$ and creates $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ by computing

$$\begin{aligned}
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,0} &= [\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2,\\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} &= [\mathbf{k} - \mathsf{delk}_{\mathsf{kgc},\theta}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2,\\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,2} &= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2.
\end{aligned} \tag{11}$$

To create $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$, $\mathcal{C}$ samples the ephemeral delegation key $\overline{\mathsf{delk}}_{\mathtt{ID},\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$ and creates the sub-key update $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ in the same way as the real scheme. Then, $\mathcal{C}$ retrieves the delegation key $\mathbf{k}_{\mathtt{ID},\theta}$ and ephemeral delegation key $\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}$, samples $\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}, \overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, and computes a helper key update $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}} = (\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$:

$$\begin{aligned}
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0} &= \mathsf{s.SK}^{(2)}_{\mathtt{ID},0} \cdot [\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0} = [\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1} &= [\mathbf{k} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot \mathsf{s.SK}^{(2)}_{\mathtt{ID},1} \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2\\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2} &= \mathsf{s.SK}^{(2)}_{\mathtt{ID},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2} = [\mathbf{V}_{L+2}\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell} &= \mathsf{s.}\widetilde{\overline{\mathsf{SK}}}^{(2)}_{\mathtt{ID},\ell} \cdot [\mathbf{V}_\ell\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2.
\end{aligned} \tag{12}$$

This is the *normal* helper key update as in $\mathsf{Game}_{\mathrm{II},1}$ by setting $\bar{\mathbf{t}}_{\mathtt{ID},\mathtt{T}} = \mathbf{r}^{(2)}_{\mathtt{ID}} + \widetilde{\bar{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}$. Due to the fresh random $\widetilde{\bar{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, $\bar{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}$ is distributed in $\mathbb{Z}_p^k$ uniformly at random.

Decryption Key Creations: To create $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$, $\mathcal{C}$ retrieves the master secret key $\mathbf{k}$, samples $\mathbf{u}_{\mathtt{ID},\mathtt{T}}, \mathbf{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, and computes $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})$:

$$\begin{aligned}
&\mathsf{DK}_{\mathtt{ID},\mathtt{T},0} = [\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0} = [\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
&\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} = [\mathbf{k}]_2 \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|})\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2, \qquad (13)\\
&\mathsf{DK}_{\mathtt{ID},\mathtt{T},2} = [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2} = [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,
\end{aligned}$$

where the distribution is the same as the real scheme.

Challenge Ciphertext Creation: Upon $\mathcal{A}$'s challenge query on $(\mathtt{ID}^\star, \mathtt{T}^\star, \mathsf{M}^\star_0, \mathsf{M}^\star_1)$, $\mathcal{C}$ retrieves $(\mathbf{V}_\ell)_{\ell\in[0,|\mathtt{ID}^\star|]\cup\{L+1\}}$ and master secret key $\mathbf{k}$, samples $\mathbf{c} \leftarrow_R \mathbb{Z}_p^{k+1}$, $(v_0, v_1, \ldots, v_{|\mathtt{ID}^\star|}, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{|\mathtt{ID}^\star|+2}$, and $\mathsf{coin} \leftarrow_R \{0,1\}$, and creates the *semi-functional* challenge ciphertext $\mathsf{ct}^\star = (\mathtt{tag}, \mathtt{tag}', C_0, C_1, C'_1, C_2)$ by computing (4).

As we observed so far, all the elements distribute in the same way as in $\mathsf{Game}_{\mathrm{II},1}$. Thus, we complete the proof of Lemma 4. □

Lemma 5 (Secret Key Invariance, $\mathsf{Game}_{\mathrm{II},2} \approx_c \mathsf{Game}_{\mathrm{II},3}$)**.** *$\mathsf{Game}_{\mathrm{II},2}$ and $\mathsf{Game}_{\mathrm{II},3}$ are computationally indistinguishable under the* MDDH *assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-II adversary $\mathcal{A}$ making at most Q_{gen} secret key* generation *queries and Q_{rev} secret key* reveal *queries, there exist reduction algorithms $\mathcal{B}_{\mathrm{II},1}$ and $\mathcal{B}_{\mathrm{II},2}$ such that*

$$|\mathsf{Adv}_{\mathrm{II},2}(\lambda) - \mathsf{Adv}_{\mathrm{II},3}(\lambda)| \le Q_{\mathsf{rev}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j}}(\lambda) + \frac{4Q_{\mathsf{rev}}}{p-1}$$

and $\max_{j\in[2]} \mathtt{T}(\mathcal{B}_{\mathrm{II},j}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

Here, we change each seed secret key $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$ on which $\mathcal{A}$ makes a secret key *reveal* query to be semi-functional one by one. The proof essentially follows the standard dual system argument [CGW15, CG17, CW14]. For the completeness, we formally prove Lemma 5.

Proof of Lemma 5. To prove Lemma 5, we further introduce the following auxiliary distributions for *seed* secret keys.

Pseudo-normal Seed Secret Keys: A *pseudo-normal* seed secret key $\mathsf{s.sk}_{\mathtt{ID}} = (\mathsf{s.SK}_{\mathtt{ID},0}, \mathsf{s.SK}_{\mathtt{ID},1}, \mathsf{s.SK}_{\mathtt{ID},2}, (\mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ is defined as follows:

$$\begin{aligned}
&\mathsf{s.SK}_{\mathtt{ID},0} \coloneqq [\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2,\\
&\mathsf{s.SK}_{\mathtt{ID},1} \coloneqq [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|})\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2 \cdot \boxed{[\hat{r}\mathbf{a}^\perp]_2^{v_0 + v_1\mathtt{id}_1 + \ldots + v_{|\mathtt{ID}|}\mathtt{id}_{|\mathtt{ID}|}}},\\
&\mathsf{s.SK}_{\mathtt{ID},2} \coloneqq [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2 \cdot \boxed{[\hat{r}\mathbf{a}^\perp]_2^{-1}}, \qquad \mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell} \coloneqq [\mathbf{V}_\ell\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2 \cdot \boxed{[\hat{r}\mathbf{a}^\perp]_2^{v_\ell}},
\end{aligned}$$

where $\mathbf{r}_{\mathtt{ID}} \leftarrow_R \mathbb{Z}_p^k$, $\hat{r} \leftarrow_R \mathbb{Z}_p^*$, and $(v_0, v_1, \ldots, v_{|\mathtt{ID}|}) \leftarrow_R \mathbb{Z}_p^{|\mathtt{ID}|+1}$ is the random coin used to create the challenge ciphertext. Here, the boxed parts denote the change from the *normal* seed secret key.

Pseudo-SF Seed Secret Keys: A *pseudo-SF* seed secret key $\mathsf{s.sk}_{\mathtt{ID}} = (\mathsf{s.SK}_{\mathtt{ID},0}, \mathsf{s.SK}_{\mathtt{ID},1}, \mathsf{s.SK}_{\mathtt{ID},2}, (\mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ is defined as follows:

$$\mathsf{s.SK}_{\mathtt{ID},0} \coloneqq [\mathbf{Z}\mathbf{r}_{\mathtt{ID}}]_2,$$

$$\mathsf{s.SK}_{\mathtt{ID},1} \coloneqq \boxed{[\alpha \mathbf{a}^\perp]_2} \cdot [(\mathbf{V}_0 + \mathtt{id}_1 \mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|} \mathbf{V}_{|\mathtt{ID}|}) \mathbf{Z} \mathbf{r}_{\mathtt{ID}}]_2 \cdot [\hat{r} \mathbf{a}^\perp]_2^{v_0 + v_1 \mathtt{id}_1 + \ldots + v_{|\mathtt{ID}|} \mathtt{id}_{|\mathtt{ID}|}},$$
$$\mathsf{s.SK}_{\mathtt{ID},2} \coloneqq [\mathbf{V}_{L+2} \mathbf{Z} \mathbf{r}_{\mathtt{ID}}]_2 \cdot [\hat{r} \mathbf{a}^\perp]_2^{-1}, \qquad \widetilde{\mathsf{s.SK}}_{\mathtt{ID},\ell} \coloneqq [\mathbf{V}_\ell \mathbf{Z} \mathbf{r}_{\mathtt{ID}}]_2 \cdot [\hat{r} \mathbf{a}^\perp]_2^{v_\ell},$$

where $\mathbf{r}_{\mathtt{ID}} \leftarrow_R \mathbb{Z}_p^k$, $\hat{r} \leftarrow_R \mathbb{Z}_p^*$, $(v_0, v_1, \ldots, v_{|\mathtt{ID}|}) \leftarrow_R \mathbb{Z}_p^{|\mathtt{ID}|+1}$ is the random coin used to create the challenge ciphertext, and $\alpha \leftarrow_R \mathbb{Z}_p^*$ is shared by all seed secret keys. Here, the boxed parts denote the change from the *pseudo-normal* seed secret key.

We further introduce the following sequence of games for $q \in [0, Q_{\mathsf{rev}}]$:

$\mathsf{Game}_{\mathrm{II},2,q,1}$: This game is the same as $\mathsf{Game}_{\mathrm{II},2}$ except that

- $\mathcal{C}$ creates *semi-functional* $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ to answer $\mathcal{A}$'s first $q-1$ secret key *reveal* queries,
- $\mathcal{C}$ creates *pseudo-normal* $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ to answer $\mathcal{A}$'s q-th secret key *reveal* query,
- $\mathcal{C}$ creates *normal* $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ to answer $\mathcal{A}$'s last $(Q_{\mathsf{rev}} - q)$ secret key *reveal* queries.

$\mathsf{Game}_{\mathrm{II},2,q,2}$: This game is the same as $\mathsf{Game}_{\mathrm{II},2,q,1}$ except that

- $\mathcal{C}$ creates *pseudo-SF* $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ to answer $\mathcal{A}$'s q-th secret key *reveal* query.

$\mathsf{Game}_{\mathrm{II},2,q,3}$: This game is the same as $\mathsf{Game}_{\mathrm{II},2,q,2}$ except that

- $\mathcal{C}$ creates *semi-functional* $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ to answer $\mathcal{A}$'s q-th secret key *reveal* query.

Table 3: Distributions of $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ in the proof of Lemma 5

Game	first $q-1$ $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$	q-th $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$	last $Q_{\mathsf{rev}} - q$ $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$
$\mathsf{Game}_{\mathrm{II},2,q,1}$	semi-functional	pseudo-normal	normal
$\mathsf{Game}_{\mathrm{II},2,q,2}$	semi-functional	pseudo-SF	normal
$\mathsf{Game}_{\mathrm{II},2,q,3}$	semi-functional	semi-functional	normal

In Table 3, we summarize the distributions of $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ in each game. By definition, $\mathsf{Game}_{\mathrm{II},2,0,3} = \mathsf{Game}_{\mathrm{II},2}$ and $\mathsf{Game}_{\mathrm{II},2,Q_{\mathsf{rev}},3} = \mathsf{Game}_{\mathrm{II},3}$. Hereafter, we prove

$$\mathsf{Game}_{\mathrm{II},2,q-1,3} \approx_c \mathsf{Game}_{\mathrm{II},2,q,1} \equiv \mathsf{Game}_{\mathrm{II},2,q,2} \approx_c \mathsf{Game}_{\mathrm{II},2,q,3},$$

where the fact implies that $\mathsf{Game}_{\mathrm{II},2} \approx_c \mathsf{Game}_{\mathrm{II},3}$.

Lemma 6 (Seed Secret Key Transition from Normal to Pseudo-normal, $\mathsf{Game}_{\mathrm{II},2,q-1,3} \approx_c \mathsf{Game}_{\mathrm{II},2,q,1}$)**.** *$\mathsf{Game}_{\mathrm{II},2,q-1,3}$ and $\mathsf{Game}_{\mathrm{II},2,q,1}$ are computationally indistinguishable under the* MDDH *assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-II adversary $\mathcal{A}$ making at most Q_{gen} secret key generation queries, there exists a reduction algorithm $\mathcal{B}_{\mathrm{II},1}$ such that*

$$|\mathsf{Adv}_{\mathrm{II},2,q-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{II},2,q,1}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},1}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{II},1}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

Proof of Lemma 6. The reduction algorithm $\mathcal{B}_{\mathrm{II},1}$ is given an MDDH instance in $\mathbb{G}_2$: $(\mathcal{G}(1^\lambda), [\mathbf{B}]_2, [\mathbf{b}]_2 = [\mathbf{Br} + \hat{r}\mathbf{e}]_2)$, where $\mathbf{B} \leftarrow_R \mathcal{D}_k, \mathbf{r} \leftarrow_R \mathbb{Z}_p^k$, $\hat{r} = 0$ or $\hat{r} \leftarrow_R \mathbb{Z}_p$, and $\mathbf{e} = (0, \dots, 0, 1)^\top \in \mathbb{Z}_p^{k+1}$. Hereafter, we assume that $\hat{r} \leftarrow_R \mathbb{Z}_p^*$ in the latter case with the statistical difference $1/p$.

We describe how $\mathcal{C}$ creates MPK, $\mathsf{sk}_{\mathtt{ID}}$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{ct}^\star$.

MPK Creation: At the beginning of the game, $\mathcal{B}_{\mathrm{II},1}$ samples $(\mathbf{A}, \mathbf{a}^\perp) \leftarrow \mathcal{D}_k$, $((\widetilde{\mathbf{V}}_\ell)_{\ell\in[0,L+2]}, \widetilde{\mathbf{Z}}) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$, $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$, $(v_0, v_1, \dots, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{L+2}$, and $\alpha \leftarrow_R \mathbb{Z}_p^*$. As the special case, we set $v_{L+2} = -1$. Let $\overline{\mathbf{B}} \in \mathbb{Z}_p^{k\times k}$ and $\underline{\mathbf{B}} \in \mathbb{Z}_p^{1\times k}$ denote a top $k \times k$ sub-matrix and bottom row vector of $\mathbf{B}$, respectively, where $\overline{\mathbf{B}}$ is full-rank. Let

$$\mathbf{M} \coloneqq \mathbf{a}^\perp(\underline{\mathbf{B}}\overline{\mathbf{B}}^{-1}) \in \mathbb{Z}_p^{(k+1)\times k}$$

denote a matrix that is not computable by $\mathcal{B}_{\mathrm{II},1}$. $\mathcal{B}_{\mathrm{II},1}$ sets

$$\mathbf{V}_\ell = \widetilde{\mathbf{V}}_\ell + v_\ell \mathbf{M}, \qquad [\mathbf{Z}]_2 = [\overline{\mathbf{B}}\widetilde{\mathbf{Z}}]_2.$$

Since $\overline{\mathbf{B}}$ is full-rank, $\mathbf{Z}$ is distributed in $\mathbb{Z}_p^{k\times k}$ uniformly at random as required. Then, $\mathcal{B}_{\mathrm{II},1}$ computes

$$\begin{aligned}
[\widetilde{\mathbf{V}}_\ell^\top \mathbf{A}]_1 &= [\mathbf{V}_\ell^\top \mathbf{A} - v_\ell(\underline{\mathbf{B}}\overline{\mathbf{B}}^{-1})^\top \cdot ({\mathbf{a}^\perp}^\top \mathbf{A})]_1 = [\mathbf{V}_\ell^\top \mathbf{A}]_1, \\
[\widetilde{\mathbf{V}}_\ell \overline{\mathbf{B}}\widetilde{\mathbf{Z}} + v_\ell \mathbf{a}^\perp \underline{\mathbf{B}}\widetilde{\mathbf{Z}}]_2 &= [\mathbf{V}_\ell \mathbf{Z} - v_\ell \mathbf{a}^\perp(\underline{\mathbf{B}}\overline{\mathbf{B}}^{-1}) \cdot \overline{\mathbf{B}}\widetilde{\mathbf{Z}} + v_\ell \mathbf{a}^\perp \underline{\mathbf{B}}\widetilde{\mathbf{Z}}]_2 = [\mathbf{V}_\ell \mathbf{Z}]_2,
\end{aligned}$$

for $\ell \in [0, L+2]$. Therefore, $\mathcal{B}_{\mathrm{II},1}$ can compute

$$\mathsf{MPK} = \left([\mathbf{A}]_1, ([\mathbf{V}_\ell^\top \mathbf{A}]_1)_{\ell\in[0,L+2]}, [\mathbf{Z}]_2, ([\mathbf{V}_\ell \mathbf{Z}]_2)_{\ell\in[0,L+2]}, [\mathbf{A}^\top \mathbf{k}]_T\right)$$

that is distributed in the same way as the real scheme.

Secret Key Creation: Upon $\mathcal{A}$'s secret key *reveal* queries on $\mathtt{ID}$, $\mathcal{B}_{\mathrm{II},1}$ creates *seed* secret keys $\mathsf{s.sk}_{\mathtt{ID}}$ as follows:

- If this is not still the q-th query, $\mathcal{B}_{\mathrm{II},1}$ retrieves a delegation key $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta} = \mathbf{k}_{\mathsf{pa}(\mathtt{ID}),\theta}$ and α, samples $\mathbf{r}_{\mathtt{ID}}^{(1)} \leftarrow_R \mathbb{Z}_p^k$, and computes a *semi-functional* seed secret key $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ by computing (9).
- If this is the q-th query, let $\overline{\mathbf{b}} \in \mathbb{Z}_p^k$ and $\underline{\mathbf{b}} \in \mathbb{Z}_p$ denote the first k-entries and last entry of $\mathbf{b}$, respectively. Then, $\mathcal{B}_{\mathrm{II},1}$ retrieves $(v_0, v_1, \dots, v_{|\mathtt{ID}|})$ and computes a seed secret key $\mathsf{s.sk}_{\mathtt{ID}}^{(1)} = (\mathsf{s.SK}_{\mathtt{ID},0}^{(1)}, \mathsf{s.SK}_{\mathtt{ID},1}^{(1)}, \mathsf{s.SK}_{\mathtt{ID},2}^{(1)}, (\mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell}^{(1)})_{\ell\in[|\mathtt{ID}|+1,L]})$:

$$\begin{aligned}
\mathsf{s.SK}_{\mathtt{ID},0} &= [\overline{\mathbf{b}}]_2 \\
\mathsf{s.SK}_{\mathtt{ID},1}^{(1)} &= [(\widetilde{\mathbf{V}}_0 + \mathtt{id}_1\widetilde{\mathbf{V}}_1 + \mathtt{id}_{|\mathtt{ID}|}\widetilde{\mathbf{V}}_{|\mathtt{ID}|})\overline{\mathbf{b}}]_2 \cdot [\mathbf{a}^\perp \underline{\mathbf{b}}]_2^{v_0+\mathtt{id}_1 v_1+\cdots+\mathtt{id}_{|\mathtt{ID}|}v_{|\mathtt{ID}|}}, \\
\mathsf{s.SK}_{\mathtt{ID},2}^{(1)} &= [\widetilde{\mathbf{V}}_{L+2}\overline{\mathbf{b}}]_2 \cdot [-\mathbf{a}^\perp \underline{\mathbf{b}}]_2, \qquad \mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell}^{(1)} = [\widetilde{\mathbf{V}}_\ell \overline{\mathbf{b}}]_2 \cdot [\mathbf{a}^\perp \underline{\mathbf{b}}]_2^{v_\ell}.
\end{aligned} \tag{14}$$

- If this is after the q-th query, $\mathcal{B}_{\mathrm{II},1}$ creates a *normal* seed secret key $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ by computing (8). Then, $\mathcal{B}_{\mathrm{II},1}$ creates sub-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$ by computing (10).

Here, we check that the q-th queried seed secret key $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ is properly distributed. By definition,

$$\overline{\mathbf{b}} = \overline{\mathbf{B}}\mathbf{r}, \qquad \underline{\mathbf{b}} = \underline{\mathbf{B}}\mathbf{r} + \hat{r},$$

where $\mathbf{r} \leftarrow_R \mathbb{Z}_p^k$, $\hat{r} = 0$ or $\hat{r} \leftarrow_R \mathbb{Z}_p^*$. At first, we show that $\mathsf{SK}_{\mathtt{ID},\theta,0}$ is properly distributed by setting

$$\mathbf{r}_{\mathtt{ID}}^{(1)} = \mathbf{Z}^{-1}\overline{\mathbf{b}} = (\widetilde{\mathbf{Z}}^{-1}\overline{\mathbf{B}}^{-1}) \cdot (\overline{\mathbf{B}}\mathbf{r}) = \widetilde{\mathbf{Z}}^{-1}\mathbf{r}.$$

Since $\widetilde{\mathbf{Z}}$ is distributed in $\mathbb{Z}_p^{k\times k}$ uniformly at random, the matrix is full-rank with probability at least $1 - 1/(p-1)$. Since $\mathbf{r}$ is distributed in $\mathbb{Z}_p^k$ uniformly at random, $\mathbf{r}_{\mathtt{ID}}^{(1)}$ also follows the same distribution if $\widetilde{\mathbf{Z}}$ is full-rank. Next, we observe that

$$\begin{aligned}\widetilde{\mathbf{V}}_\ell\overline{\mathbf{b}} + v_\ell\mathbf{a}^\perp\underline{\mathbf{b}} &= (\mathbf{V}_\ell - v_\ell\mathbf{a}^\perp(\underline{\mathbf{B}}\overline{\mathbf{B}}^{-1})) \cdot (\overline{\mathbf{B}}\mathbf{r}) + v_\ell\mathbf{a}^\perp \cdot (\underline{\mathbf{B}}\mathbf{r} + \hat{r}) \\ &= \mathbf{V}_\ell\mathbf{Z}\mathbf{r}_{\mathtt{ID}}^{(1)} + v_\ell\hat{r}\mathbf{a}^\perp\end{aligned}$$

for $\ell \in [0, L+2]$, where $v_{L+2} = -1$. Therefore, it holds that

$$\begin{aligned}\mathsf{s.SK}_{\mathtt{ID},1}^{(1)} &= [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{r}_{\mathtt{ID}}^{(1)}]_2 \cdot [\hat{r}\mathbf{a}^\perp]_2^{v_0+\mathtt{id}_1 v_1+\cdots+\mathtt{id}_{|\mathtt{ID}|}v_{|\mathtt{ID}|}}, \\ \mathsf{SK}_{\mathtt{ID},\theta,2}^{(1)} &= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{r}_{\mathtt{ID}}^{(1)}]_2 \cdot [\hat{r}\mathbf{a}^\perp]_2^{-1}, \\ \widetilde{\mathsf{SK}}_{\mathtt{ID},\theta,\ell}^{(1)} &= [\mathbf{V}\mathbf{Z}\mathbf{r}_{\mathtt{ID}}^{(1)}]_2 \cdot [\hat{r}\mathbf{a}^\perp]_2^{v_\ell}.\end{aligned} \tag{15}$$

If $\hat{r} = 0$, $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ is a *normal* seed secret key as in $\mathsf{Game}_{\mathrm{II},2,q-1,3}$. If $\hat{r} \leftarrow_R \mathbb{Z}_p^*$, $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ is a *pseudo-normal* seed secret key as in $\mathsf{Game}_{\mathrm{II},2,q,1}$.

Key Update Creation: $\mathcal{B}_{\mathrm{II},1}$ creates all $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ in the same way as in $\mathsf{Game}_{\mathrm{II},2}$.

Decryption Key Creation: $\mathcal{B}_{\mathrm{II},1}$ creates all $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ in the same way as in $\mathsf{Game}_{\mathrm{II},2}$.

Challenge Ciphertext Creation: Upon $\mathcal{A}$'s challenge query on $(\mathtt{ID}^\star, \mathtt{T}^\star, \mathsf{M}_0^\star, \mathsf{M}_1^\star)$, $\mathcal{B}_{\mathrm{II},1}$ retrieves $(\widetilde{\mathbf{V}}_\ell)_{\ell\in[0,L+2]}$ and $(v_0, v_1, \ldots, v_{|\mathtt{ID}^\star|}, v_{L+1})$, samples $\mathbf{c} \leftarrow_R \mathbb{Z}_p^{k+1}$ and $\mathsf{coin} \leftarrow_R \{0,1\}$, and creates the challenge ciphertext $\mathsf{ct}^\star = (\mathtt{tag}, \mathtt{tag}', C_0, C_1, C_1', C_2)$:

$$\begin{aligned}\mathtt{tag} &= v_0 + v_1\mathtt{id}_1^\star + \cdots + v_{|\mathtt{ID}^\star|}\mathtt{id}_{|\mathtt{ID}^\star|}^\star, \qquad \mathtt{tag}' = v_0 + v_{L+1}\mathtt{T}^\star, \\ C_0 &= [\mathbf{c}]_1, \\ C_1 &= [(\widetilde{\mathbf{V}}_0 + \mathtt{id}_1^\star\widetilde{\mathbf{V}}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}^\star|}^\star\widetilde{\mathbf{V}}_{|\mathtt{ID}^\star|} + \mathtt{tag}\widetilde{\mathbf{V}}_{L+2})^\top\mathbf{c}]_1, \\ C_1' &= [(\widetilde{\mathbf{V}}_0 + \mathtt{T}^\star\widetilde{\mathbf{V}}_{L+1} + \mathtt{tag}'\widetilde{\mathbf{V}}_{L+2})^\top\mathbf{c}]_1, \qquad C_2 = \mathsf{M}_{\mathsf{coin}}^\star \cdot [\mathbf{c}^\top\mathbf{k}]_T.\end{aligned} \tag{16}$$

It is clear that $\mathtt{tag}, \mathtt{tag}', C_0, C_2$ are properly distributed. We check that C_1 and C_1' are also properly distributed as follows:

$$\begin{aligned}C_1 &= [(\widetilde{\mathbf{V}}_0 + \mathtt{id}_1^\star\widetilde{\mathbf{V}}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}^\star|}^\star\widetilde{\mathbf{V}}_{|\mathtt{ID}^\star|} + \mathtt{tag}\widetilde{\mathbf{V}}_{L+2})^\top\mathbf{c}]_1 \\ &= [\Big((\mathbf{V}_0 - v_0\mathbf{M}) + \mathtt{id}_1^\star(\mathbf{V}_1 - v_1\mathbf{M}) + \cdots + \mathtt{id}_{|\mathtt{ID}^\star|}^\star(\mathbf{V}_{|\mathtt{ID}^\star|} - v_{|\mathtt{ID}^\star|}\mathbf{M})\Big)^\top\mathbf{c}]_1 \\ &\quad \cdot [(\mathbf{V}_{L+2} + \mathbf{M})^\top\mathbf{c}]_1^{v_0+v_1\mathtt{id}_1^\star+\cdots+v_{|\mathtt{ID}^\star|}\mathtt{id}_{|\mathtt{ID}^\star|}^\star} \\ &= [(\mathbf{V}_0 + \mathtt{id}_1^\star\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}^\star|}^\star\mathbf{V}_{|\mathtt{ID}^\star|} + \mathtt{tag}\mathbf{V}_{L+2})^\top\mathbf{c}]_1, \\ C_1' &= [(\widetilde{\mathbf{V}}_0 + \mathtt{T}^\star\widetilde{\mathbf{V}}_{L+1} + \mathtt{tag}'\widetilde{\mathbf{V}}_{L+2})^\top\mathbf{c}]_1 \\ &= [((\mathbf{V}_0 - v_0\mathbf{M}) + \mathtt{T}^\star(\mathbf{V}_{L+1} - v_{L+1}\mathbf{M}))^\top\mathbf{c}]_1 \cdot [(\mathbf{V}_{L+2} + \mathbf{M})^\top\mathbf{c}]_1^{v_0+v_{L+1}\mathtt{T}^\star} \\ &= [(\mathbf{V}_0 + \mathtt{T}^\star\mathbf{V}_{L+1} + \mathtt{tag}'\mathbf{V}_{L+2})^\top\mathbf{c}]_1.\end{aligned}$$

Therefore, $\mathsf{ct}^\star$ is properly distributed *semi-functional* ciphertext.

Thus, we complete the proof of Lemma 6. □

Lemma 7 (Seed Secret Key Transition from Pseudo-normal to Pseudo-SF, $\mathsf{Game}_{\mathrm{II},2,q,1} \equiv \mathsf{Game}_{\mathrm{II},2,q,2}$)**.** $\mathsf{Game}_{\mathrm{II},2,q,1}$ *and* $\mathsf{Game}_{\mathrm{II},2,q,2}$ *are identically distributed. Specifically, for any Type-*II *adversary* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}_{\mathrm{II},2,q,1}(\lambda) = \mathsf{Adv}_{\mathrm{II},2,q,2}(\lambda).$$

Proof of Lemma 7. Here, we prove a stronger claim that $\mathsf{Game}_{\mathrm{II},2,q,1}$ and $\mathsf{Game}_{\mathrm{II},2,q,2}$ are identically distributed for any fixed

- $(\mathbf{A}, \mathbf{a}) \leftarrow_R \mathcal{D}_k$,
- $\left((\mathbf{V}_\ell)_{\ell\in[0,L+2]}, \mathbf{Z}\right) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$,
- master secret key $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$,
- $\mathbf{c} \leftarrow_R \mathbb{Z}_p^{k+1}$ for creating the challenge ciphertext,
- $(\mathtt{ID}^\star, \mathsf{T}^\star, \mathsf{M}_0^\star, \mathsf{M}_1^\star) \in \mathbb{Z}_p^2 \times \mathcal{M}^2$ and random coin $\mathsf{coin} \leftarrow_R \{0,1\}$,
- delegation keys $\mathbf{k}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$,
- $\mathbf{r}_{\mathtt{ID}}^{(1)} \leftarrow_R \mathbb{Z}_p^k$ and $\hat{r} \leftarrow_R \mathbb{Z}_p$ for creating q-th queried $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$,
- semi-functional randomness $\alpha \leftarrow_R \mathbb{Z}_p^*$.

Specifically, the randomness of $(v_0, v_1, \ldots, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{L+2}$ enables us to prove the claim. Since $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ which is not q-th queried ones, $\mathsf{s.sk}_{\mathtt{ID}}^{(2)}$, and $\mathsf{dk}_{\mathtt{ID},\mathsf{T}}$ are created in the same way in both $\mathsf{Game}_{\mathrm{II},2,q,1}$ and $\mathsf{Game}_{\mathrm{II},2,q,2}$, and all the other elements have been already fixed, it is sufficient to show that

$$\left\{ \begin{array}{c} v_0 + v_1\mathtt{id}_1^\star + \cdots + v_{|\mathtt{ID}^\star|}\mathtt{id}_{|\mathtt{ID}^\star|}^\star,\ v_0 + v_{L+1}\mathsf{T}^\star \\ v_0 + v_1\mathtt{id}_1 + \cdots + v_{|\mathtt{ID}|}\mathtt{id}_{|\mathtt{ID}|}, (v_\ell)_{\ell\in[|\mathtt{ID}|+1,L]} \end{array} \right\}$$
$$\equiv \left\{ \begin{array}{c} v_0 + v_1\mathtt{id}_1^\star + \cdots + v_{|\mathtt{ID}^\star|}\mathtt{id}_{|\mathtt{ID}^\star|}^\star,\ v_0 + v_{L+1}\mathsf{T}^\star, \\ \alpha/\hat{r} + v_0 + v_1\mathtt{id}_1 + \cdots + v_{|\mathtt{ID}|}\mathtt{id}_{|\mathtt{ID}|}^\star, (v_\ell)_{\ell\in[|\mathtt{ID}|+1,L]} \end{array} \right\},$$

where $(v_0, v_1, \ldots, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{L+2}$. Here, the first and second elements are $\mathtt{tag}, \mathtt{tag}'$, and last element is the exponent of $[\hat{r}\mathbf{a}^\perp]_2$ of q-th queried $(\mathsf{SK}_{\mathtt{ID},\theta,1}, (\widetilde{\mathsf{SK}}_{\mathtt{ID},\theta,\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ in $\mathsf{Game}_{\mathrm{II},2,q,1}$ and $\mathsf{Game}_{\mathrm{II},2,q,2}$, respectively. Due to the randomness of $v_{L+1} \leftarrow_R \mathbb{Z}_p$, the second element is distributed in $\mathbb{Z}_p$ uniformly at random. Since $\mathtt{ID} \notin \mathsf{prefix}^+(\mathtt{ID}^\star)$ holds due to the definition of the Type-II adversary, the first and last elements are distributed in $\mathbb{Z}_p^2$ uniformly at random due to the randomness of $(v_0, v_1, \ldots, v_L) \leftarrow_R \mathbb{Z}_p^{L+1}$. Thus, we complete the proof of Lemma 7. □

Lemma 8 (Secret Key Transition from Pseudo-SF to Semi-functional, $\mathsf{Game}_{\mathrm{II},2,q,2} \approx_c \mathsf{Game}_{\mathrm{II},2,q,3}$)**.** $\mathsf{Game}_{\mathrm{II},2,q,2}$ *and* $\mathsf{Game}_{\mathrm{II},2,q,3}$ *are computationally indistinguishable under the* MDDH *assumption in* $\mathbb{G}_2$*. Specifically, for any PPT Type-*II *adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key generation queries, there exists a reduction algorithm* $\mathcal{B}_{\mathrm{II},2}$ *such that*

$$|\mathsf{Adv}_{\mathrm{II},2,q,2}(\lambda) - \mathsf{Adv}_{\mathrm{II},2,q,3}(\lambda)| \le \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},2}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{II},2}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

We omit the detailed proof of Lemma 8 since it is almost the same as the proof of Lemma 6. The only difference is that $\mathcal{B}_{\mathrm{II},2}$ creates $\mathsf{s.SK}_{\mathtt{ID},1}$ upon $\mathcal{A}$'s q-th secret key *reveal* query by

$$\mathsf{s.SK}_{\mathtt{ID},1}^{(1)} = \boxed{[\alpha\mathbf{a}^\perp]_2} \cdot [(\widetilde{\mathbf{V}}_0 + \mathtt{id}_1\widetilde{\mathbf{V}}_1 + \mathtt{id}_{|\mathtt{ID}|}\widetilde{\mathbf{V}}_{|\mathtt{ID}|})\overline{\mathbf{b}}]_2 \cdot [\mathbf{a}^\perp\underline{\mathbf{b}}]_2^{v_0+\mathtt{id}_1 v_1+\cdots+\mathtt{id}_{|\mathtt{ID}|}v_{|\mathtt{ID}|}},$$

$$= \boxed{[\alpha \mathbf{a}^\perp]_2} \cdot [(\mathbf{V}_0 + \mathtt{id}_1 \mathbf{V}_1 + \mathtt{id}_{|\mathtt{ID}|} \mathbf{V}_{|\mathtt{ID}|}) \mathbf{r}^{(1)}_{\mathtt{ID}}]_2 \cdot [\hat{r} \mathbf{a}^\perp]_2^{v_0 + \mathtt{id}_1 v_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|} v_{|\mathtt{ID}|}},$$

where the boxed parts denote the changes from (14). If $\hat{r} \leftarrow_R \mathbb{Z}_p^*$, $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$ is a *pseudo-SF* seed secret key as in $\mathsf{Game}_{\mathrm{II},2,q,2}$. If $\hat{r} = 0$, $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$ is a *semi-functional* seed secret key as in $\mathsf{Game}_{\mathrm{II},2,q,3}$.

By combining Lemmata 6–8, we have

$$\begin{aligned}
&|\mathsf{Adv}_{\mathrm{II},2}(\lambda) - \mathsf{Adv}_{\mathrm{II},3}(\lambda)| \\
\leq &\sum_{q \in [Q_{\mathsf{rev}}]} |\mathsf{Adv}_{\mathrm{II},2,q-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{II},2,q,1}(\lambda)| + \sum_{q \in [Q_{\mathsf{rev}}]} |\mathsf{Adv}_{\mathrm{II},2,q,1}(\lambda) - \mathsf{Adv}_{\mathrm{II},2,q,2}(\lambda)| \\
&+ \sum_{q \in [Q_{\mathsf{rev}}]} |\mathsf{Adv}_{\mathrm{II},2,q,2}(\lambda) - \mathsf{Adv}_{\mathrm{II},2,q,3}(\lambda)| \\
\leq &\, Q_{\mathsf{rev}} \cdot \sum_{j \in [2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j}}(\lambda) + \frac{4Q_{\mathsf{rev}}}{p-1}.
\end{aligned}$$

Thus, we complete the proof of Lemma 5. □

Lemma 9 (Semi-functional Randomness Switching for KGC's Key Updates, $\mathsf{Game}_{\mathrm{II},3} \equiv \mathsf{Game}_{\mathrm{II},4}$)**.** *$\mathsf{Game}_{\mathrm{II},3}$ and $\mathsf{Game}_{\mathrm{II},4}$ are identically distributed from $\mathcal{A}$'s view. Specifically, for any Type-II adversary $\mathcal{A}$, it holds that*

$$\mathsf{Adv}_{\mathrm{II},3}(\lambda) = \mathsf{Adv}_{\mathrm{II},4}(\lambda).$$

The proof is the first core part of the proof against the Type-II adversary since we have to change $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}^\star}$ to be semi-functional. We can prove Lemma 9 based on the fact that all $\mathsf{delk}_{\mathsf{kgc},\theta}$ are never revealed to $\mathcal{A}$ and all $\mathsf{sk}_{\mathtt{ID}}$ such that $|\mathtt{ID}| = 1$ which $\mathcal{A}$ receives via secret key *reveal* queries are *semi-functional* due to the modification in $\mathsf{Game}_{\mathrm{II},3}$.

Proof of Lemma 9. Here, we prove a stronger claim that $\mathsf{Game}_{\mathrm{II},3}$ and $\mathsf{Game}_{\mathrm{II},4}$ are identically distributed from $\mathcal{A}$'s view for any fixed

- $(\mathbf{A}, \mathbf{a}) \leftarrow_R \mathcal{D}_k$,
- $\big((\mathbf{V}_\ell)_{\ell \in [0,L+2]}, \mathbf{Z}\big) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$,
- master secret key $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$,
- $\mathbf{r}^{(1)}_{\mathtt{ID}} \leftarrow_R \mathbb{Z}_p^k$ for creating all $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$ such that $|\mathtt{ID}| = 1$,
- $\mathbf{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$ for creating all $\mathsf{sk}_{\mathtt{ID},\theta}$ such that $|\mathtt{ID}| = 1$,
- $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ for creating all $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$,
- $\alpha \leftarrow_R \mathbb{Z}_p^*$.

Specifically, the randomnesses of all $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$ such that $|\mathtt{ID}| = 1$ and all $\mathsf{delk}_{\mathsf{kgc},\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ enable us to prove the claim. We note that $\mathsf{sk}_{\mathtt{ID}}$ such that $|\mathtt{ID}| \geq 2$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ are created in the same way in both $\mathsf{Game}_{\mathrm{II},3}$ and $\mathsf{Game}_{\mathrm{II},4}$. We further note that even when $\mathbf{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$ are fixed, $(\mathsf{SK}_{\mathtt{ID},\theta,0}, \mathsf{SK}_{\mathtt{ID},\theta,2})$ do not reveal the quantities of $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^*$ since they are masked by $\tilde{\mathbf{r}}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$. Since $\mathbf{r}_{\mathtt{ID},\theta}$ and $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}$ are fixed, $\mathsf{sk}_{\mathtt{ID},\theta}$ such that $|\mathtt{ID}| = 1$ and $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ are distributed in the same way in both $\mathsf{Game}_{\mathrm{II},3}$ and $\mathsf{Game}_{\mathrm{II},4}$ except $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1}$. In $\mathsf{Game}_{\mathrm{II},3}$, $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1}$ are distributed as follows:

$$\mathsf{SK}_{\mathtt{ID},\theta,1} = [\mathbf{k}_{\mathsf{kgc},\theta} + \tilde{r}_{\mathtt{ID},\theta} \alpha \mathbf{a}^\perp]_2 \cdot [(\mathbf{V}_0 + \mathtt{ID}\mathbf{V}_1)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2,$$

$$\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} = [\mathbf{k} - \mathbf{k}_{\mathsf{kgc},\theta}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Zt}_{\mathsf{kgc},\mathtt{T},\theta}]_2,$$

where $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$ and $\mathsf{delk}_{\mathsf{kgc},\theta} = \mathbf{k}_{\mathsf{kgc},\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$. In contrast, the above distribution can be written as follows:

$$\mathsf{SK}_{\mathtt{ID},\theta,1} = [(\mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp) + (\tilde{r}_{\mathtt{ID},\theta} - 1)\alpha\mathbf{a}^\perp]_2 \cdot [(\mathbf{V}_0 + \mathtt{ID}\mathbf{V}_1)\mathbf{Zr}_{\mathtt{ID},\theta}]_2,$$
$$\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} = [(\mathbf{k} + \alpha\mathbf{a}^\perp) - (\mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp)]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Zt}_{\mathsf{kgc},\mathtt{T},\theta}]_2,$$

where $\tilde{r}_{\mathtt{ID},\theta} - 1$ is distributed in $\mathbb{Z}_p$ uniformly at random and $\mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp$ is distributed in $\mathbb{Z}_p^{k+1}$ uniformly at random. Therefore, the above distribution is the same as the distribution in $\mathsf{Game}_{\mathrm{II},4}$ by setting $\tilde{r}_{\mathtt{ID},\theta} - 1$ as the randomnesses in (10) and $\mathsf{delk}_{\mathsf{kgc},\theta} = \mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp$. We note that the claim holds for all $\mathtt{ID}$ such that $|\mathtt{ID}| = 1$ and all nodes $\theta \in \mathtt{BT}_{\mathsf{kgc}}$, simultaneously. Thus, we complete the proof of Lemma 9. □

Lemma 10 (Helper Key Update Invariance, $\mathsf{Game}_{\mathrm{II},4} \approx_c \mathsf{Game}_{\mathrm{II},5}$)**.** *$\mathsf{Game}_{\mathrm{II},4}$ and $\mathsf{Game}_{\mathrm{II},5}$ are computationally indistinguishable under the* MDDH *assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-*II *adversary $\mathcal{A}$ making at most Q_{gen} secret key* generation *queries, there exist reduction algorithms $\mathcal{B}_{\mathrm{II},3}$ and $\mathcal{B}_{\mathrm{II},4}$ such that*

$$|\mathsf{Adv}_{\mathrm{II},4}(\lambda) - \mathsf{Adv}_{\mathrm{II},5}(\lambda)| \leq Q_{\mathsf{gen}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j+2}}(\lambda) + \frac{4Q_{\mathsf{gen}}}{p-1}$$

and $\max_{j\in[2]} \mathtt{T}(\mathcal{B}_{\mathrm{II},j+2}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

Let $\mathtt{ID}_q$ denote an identity on which $\mathcal{A}$ makes q-th secret key *generation* query. The structure of the proof may not look similar to the proof of Lemma 5, the spirit is almost the same.

Proof of Lemma 10. We further introduce the following sequence of games for $q \in [0, Q_{\mathsf{gen}}]$:

$\mathsf{Game}_{\mathrm{II},4,q,1}$: This game is the same as $\mathsf{Game}_{\mathrm{II},4}$ except that

- If $m < q$, $\mathcal{C}$ always creates *semi-functional* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_m}$,
- If $m = q$, $\mathcal{C}$ creates *pseudo-normal* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$,
- If $m > q$, $\mathcal{C}$ always creates *normal* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_m}$.

$\mathsf{Game}_{\mathrm{II},4,q,2}$: This game is the same as $\mathsf{Game}_{\mathrm{II},4,q,1}$ except that

- If $m = q$, $\mathcal{C}$ creates *pseudo-SF* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$,

$\mathsf{Game}_{\mathrm{II},4,q,3}$: This game is the same as $\mathsf{Game}_{\mathrm{II},4,q,2}$ except that

- If $m = q$, $\mathcal{C}$ creates *semi-functional* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$,

By definition, $\mathsf{Game}_{\mathrm{II},4,0,3} = \mathsf{Game}_{\mathrm{II},4}$ and $\mathsf{Game}_{\mathrm{II},4,Q_{\mathsf{gen}},3} = \mathsf{Game}_{\mathrm{II},5}$. Hereafter, we prove

$$\mathsf{Game}_{\mathrm{II},4,q-1,3} \approx_c \mathsf{Game}_{\mathrm{II},4,q,1} \equiv \mathsf{Game}_{\mathrm{II},4,q,2} \approx_c \mathsf{Game}_{\mathrm{II},4,q,3},$$

where the fact implies that $\mathsf{Game}_{\mathrm{II},4} \approx_c \mathsf{Game}_{\mathrm{II},5}$.

Lemma 11 (Sub-secret Key Transition from Normal to Pseudo-normal, $\mathsf{Game}_{\mathrm{II},4,q-1,3} \approx_c \mathsf{Game}_{\mathrm{II},4,q,1}$)**.** *$\mathsf{Game}_{\mathrm{II},4,q-1,3}$ and $\mathsf{Game}_{\mathrm{II},4,q,1}$ are computationally indistinguishable under the* MDDH *assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-*II *adversary $\mathcal{A}$ making at most Q_{gen} secret key generation queries, there exists a reduction algorithm $\mathcal{B}_{\mathrm{II},3}$ such that*

$$|\mathsf{Adv}_{\mathrm{II},4,q-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{II},4,q,1}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},3}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{II},3}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

We omit the proof of Lemma 11 since it is essentially the same as the proof of Lemma 6.

Lemma 12 (Sub-secret Key Transition from Pseudo-normal to Pseudo-SF, $\mathsf{Game}_{\mathrm{II},4,q,1} \equiv \mathsf{Game}_{\mathrm{II},4,q,2}$)**.** $\mathsf{Game}_{\mathrm{II},4,q,1}$ *and* $\mathsf{Game}_{\mathrm{II},4,q,2}$ *are identically distributed from* $\mathcal{A}$*'s view. Specifically, for any adversary Type-*II $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}_{\mathrm{II},4,q,1}(\lambda) = \mathsf{Adv}_{\mathrm{II},4,q,2}(\lambda).$$

The proof of Lemma 12 is the second core part of the proof against the Type-II adversary since we have to change all $\mathsf{s.sk}^{(2)}_{\mathtt{ID}}$ such that $\mathtt{ID} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$ to be semi-functional. Here, we use $\mathsf{pa}(\mathtt{ID})$ to denote the q-th queried identity. When $\mathsf{pa}(\mathtt{ID}) \notin \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, we prove Lemma 12 in the same way as the proof of Lemma 7 by showing that *pseudo-normal* $\mathsf{s.sk}^{(2)}_{\mathsf{pa}(\mathtt{ID})}$ and *pseudo-SF* $\mathsf{s.sk}^{(2)}_{\mathsf{pa}(\mathtt{ID})}$ are identically distributed. On the other hand, when $\mathsf{pa}(\mathtt{ID}) \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, we cannot follow the dual system argument. Observe that what $\mathcal{A}$ receives is not $\mathsf{s.sk}^{(2)}_{\mathsf{pa}(\mathtt{ID})}$ itself, but $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ created by using $\mathsf{s.sk}^{(2)}_{\mathsf{pa}(\mathtt{ID})}$. As the proof of Lemma 9, we use the fact that all $\mathsf{delk}_{\mathsf{pa}(\mathtt{ID}),\theta}$ are not revealed to $\mathcal{A}$ and all $\mathsf{sk}_{\mathtt{ID}}$ are *semi-functional*, and apply the semi-functional randomness switching to prove Lemma 12.

Proof of Lemma 12. If $\mathtt{ID}_q = (\mathtt{id}_{q,1}, \ldots, \mathtt{id}_{q,|\mathtt{ID}_q|}) \notin \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, we can show that *pseudo-normal* and *pseudo-SF* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ are identically distributed by following the same argument as in the proof of Lemma 7. Then, in $\mathsf{Game}_{\mathrm{II},4,q,1}$, $\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}$ is distributed as follows:

$$\begin{aligned}\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1} = [\mathbf{k} + \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}}]_2 \cdot [\alpha \mathbf{a}^{\perp}]_2 \cdot \big[\big(\mathbf{V}_0 + \mathtt{id}_{q,1}\mathbf{V}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}_q|}\mathbf{V}_{|\mathtt{ID}_q|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}\big]_2 \\ \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}}]_2 \cdot [\hat{r}\mathbf{a}^{\perp}]_2^{v_0 + v_1 \mathtt{id}_{q,1} + \ldots + v_{|\mathtt{ID}_q|}\mathtt{id}_{q,|\mathtt{ID}_q|}}.\end{aligned}$$

The distribution is the same as the distribution in $\mathsf{Game}_{\mathrm{II},4,q,2}$.

If $\mathtt{ID}_q = (\mathtt{id}_{q,1}, \ldots, \mathtt{id}_{q,|\mathtt{ID}_q|}) \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$, we prove a stronger claim that $\mathsf{Game}_{\mathrm{II},4,q,1}$ and $\mathsf{Game}_{\mathrm{II},4,q,2}$ are identically distributed from $\mathcal{A}$'s view for any fixed

- $(\mathbf{A}, \mathbf{a}) \leftarrow_R \mathcal{D}_k$,
- $\big((\mathbf{V}_\ell)_{\ell \in [0,L+2]}, \mathbf{Z}\big) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$,
- master secret key $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$,
- $\hat{r} \leftarrow_R \mathbb{Z}_p^*$ for creating $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$,
- $\mathbf{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$ for creating $\mathsf{sk}_{\mathtt{ID},\theta}$ such that $\mathsf{pa}(\mathtt{ID}) = \mathtt{ID}_q$,
- $\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta}, \overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}, \overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ for creating $\mathsf{ku}_{\mathtt{ID}_q,\mathtt{T}}$,
- $\alpha \leftarrow_R \mathbb{Z}_p^*$ and $\alpha \leftarrow_R \mathbb{Z}_p^*$.

Specifically, the randomnesses of $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$ and $\mathsf{delk}_{\mathtt{ID}_q,\theta}, \overline{\mathsf{delk}}_{\mathtt{ID}_q,\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$ enable us to prove the claim. We note that all $\mathsf{sk}_{\mathtt{ID}}$ such that $\mathsf{pa}(\mathtt{ID}) \neq \mathtt{ID}_q$, all $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $\mathtt{ID} \neq \mathtt{ID}_q$, and all $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ are created in the same way in both $\mathsf{Game}_{\mathrm{II},4,q,1}$ and $\mathsf{Game}_{\mathrm{II},4,q,2}$. We further note that even when $\mathbf{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$ are fixed, $(\mathsf{SK}_{\mathtt{ID},\theta,0}, \mathsf{SK}_{\mathtt{ID},\theta,2})$ do not reveal the quantities of $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$ in (10) since they are masked by $\widetilde{\mathbf{r}}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$. Since $\mathbf{r}_{\mathtt{ID},\theta}$ and $\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta}, \overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}, \overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}}$ are fixed, $\mathsf{sk}_{\mathtt{ID}}$ and $\mathsf{ku}_{\mathtt{ID}_q,\mathtt{T}}$ are distributed in the same way in both $\mathsf{Game}_{\mathrm{II},4,q,1}$ and $\mathsf{Game}_{\mathrm{II},4,q,2}$ except $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1}, \overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}$. In $\mathsf{Game}_{\mathrm{II},4,q,1}$, $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta,1}, \overline{\mathsf{KU}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},1}$ are distributed as follows:

$$\mathsf{SK}_{\mathtt{ID},\theta,1} = [\mathbf{k}_{\mathtt{ID}_q,\theta} + \tilde{r}_{\mathtt{ID},\theta}\alpha\mathbf{a}^{\perp}]_2 \cdot \big[\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}\big]_2,$$

$$\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1} = [\mathbf{k}_{\mathtt{ID}_q,\theta} + \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}}]_2^{-1} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Zt}_{\mathtt{ID}_q,\mathtt{T},\theta}]_2,$$
$$\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1} = [\mathbf{k} + \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_{q,1}\mathbf{V}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}_q|}\mathbf{V}_{|\mathtt{ID}_q|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}]_2$$
$$\cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}}]_2 \cdot [\hat{r}\mathbf{a}^\perp]_2^{v_0+v_1\mathtt{id}_{q,1}+\ldots+v_{|\mathtt{ID}_q|}\mathtt{id}_{q,|\mathtt{ID}_q|}},$$

where $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$, $\mathsf{delk}_{\mathtt{ID}_q,\theta} = \mathbf{k}_{\mathtt{ID}_q,\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$, and $\overline{\mathsf{delk}}_{\mathtt{ID}_q,\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$. In contrast, the above distribution can be written as follows:

$$\mathsf{SK}_{\mathtt{ID},\theta,1} = [(\mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp) + (\tilde{r}_{\mathtt{ID},\theta} - 1)\alpha\mathbf{a}^\perp]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Zr}_{\mathtt{ID},\theta}]_2,$$
$$\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1} = [(\mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp) + (\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - \alpha\mathbf{a}^\perp)]_2^{-1} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Zt}_{\mathtt{ID}_q,\mathtt{T},\theta}]_2,$$
$$\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1} = [\mathbf{k} + \alpha\mathbf{a}^\perp + (\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - \alpha\mathbf{a}^\perp)]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_{q,1}\mathbf{V}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}_q|}\mathbf{V}_{|\mathtt{ID}_q|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}]_2$$
$$\cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}}]_2 \cdot [\hat{r}\mathbf{a}^\perp]_2^{v_0+v_1\mathtt{id}_{q,1}+\ldots+v_{|\mathtt{ID}_q|}\mathtt{id}_{q,|\mathtt{ID}_q|}},$$

where $\tilde{r}_{\mathtt{ID},\theta} - 1$ is distributed in $\mathbb{Z}_p$ uniformly at random and $\mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp, \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - \alpha\mathbf{a}^\perp$ are distributed in $\mathbb{Z}_p^{k+1}$ uniformly at random. Therefore, the above distribution is the same as the distribution in $\mathsf{Game}_{\mathrm{II},4,q,2}$ by setting $\tilde{r}_{\mathtt{ID},\theta} - 1$ as the randomnesses in (10) and $\mathsf{delk}_{\mathtt{ID}_q,\theta} = \mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp$, $\overline{\mathsf{delk}}_{\mathtt{ID}_q,\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - \alpha\mathbf{a}^\perp$. We note that the claim holds for all $\mathtt{ID}$ such that $\mathsf{pa}(\mathtt{ID}) = \mathtt{ID}_q$ and all nodes $\theta \in \mathtt{BT}_{\mathtt{ID}_q}$, simultaneously. Thus, we complete the proof of Lemma 12. □

Lemma 13 (Sub-secret Key Transition from Pseudo-SF to Semi-functional, $\mathsf{Game}_{\mathrm{II},4,q,2} \approx_c \mathsf{Game}_{\mathrm{II},4,q,3}$). *$\mathsf{Game}_{\mathrm{II},4,q,2}$ and $\mathsf{Game}_{\mathrm{II},4,q,3}$ are computationally indistinguishable under the* MDDH *assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-*II *adversary $\mathcal{A}$ making at most Q_{gen} secret key generation queries, there exists a reduction algorithm $\mathcal{B}_{\mathrm{II},4}$ such that*

$$|\mathsf{Adv}_{\mathrm{II},4,q,2}(\lambda) - \mathsf{Adv}_{\mathrm{II},4,q,3}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},4}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{II},4}) \approx \mathtt{T}(\mathcal{A}) + Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

We omit the detailed proof of Lemma 13 since it is almost the same as the proof of Lemma 11.

By combining Lemmata 11–13, we have

$$\begin{aligned}
&|\mathsf{Adv}_{\mathrm{II},4}(\lambda) - \mathsf{Adv}_{\mathrm{II},5}(\lambda)| \\
\leq &\sum_{q\in[Q_{\mathsf{gen}}]} |\mathsf{Adv}_{\mathrm{II},4,q-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{II},4,q,1}(\lambda)| + \sum_{q\in[Q_{\mathsf{gen}}]} |\mathsf{Adv}_{\mathrm{II},4,q,1}(\lambda) - \mathsf{Adv}_{\mathrm{II},4,q,2}(\lambda)| \\
&+ \sum_{q\in[Q_{\mathsf{gen}}]} |\mathsf{Adv}_{\mathrm{II},4,q,2}(\lambda) - \mathsf{Adv}_{\mathrm{II},4,q,3}(\lambda)| \\
\leq &\, Q_{\mathsf{gen}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j+2}}(\lambda) + \frac{4Q_{\mathsf{gen}}}{p-1}.
\end{aligned}$$

Thus, we complete the proof of Lemma 10. □

Lemma 14 (Decryption Key Invariance, $\mathsf{Game}_{\mathrm{II},5} \approx_c \mathsf{Game}_{\mathrm{II},6}$). *$\mathsf{Game}_{\mathrm{II},5}$ and $\mathsf{Game}_{\mathrm{II},6}$ are computationally indistinguishable under the* MDDH *assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-*I I *adversary $\mathcal{A}$ making at most Q_{gen} secret key generation queries and Q_{dk} decryption key reveal queries, there exists reduction algorithms $\mathcal{B}_{\mathrm{II},5}$ and $\mathcal{B}_{\mathrm{II},6}$ such that*

$$|\mathsf{Adv}_{\mathrm{II},5}(\lambda) - \mathsf{Adv}_{\mathrm{II},6}(\lambda)| \leq Q_{\mathsf{dk}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j+4}}(\lambda) + \frac{4Q_{\mathsf{dk}}}{p-1}$$

and $\max_{j\in[2]} \mathtt{T}(\mathcal{B}_{\mathrm{II},j+4}) \approx \mathtt{T}(\mathcal{A}) + Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

The structure of the proof is the same as the proof of Lemma 5. However, the transition from pseudo-normal to pseudo-SF is a little more complicated since we have to change $\mathsf{dk}_{\mathtt{ID},\mathtt{T}^\star}$ for $\mathtt{ID} \in \mathsf{prefix}^+(\mathtt{ID}^\star) \setminus \{\mathtt{ID}^\star\}$ to be semi-functional.

Proof of Lemma 14. To prove Lemma 14, we further introduce the following auxiliary distributions.

Pseudo-normal Decryption Keys: A *pseudo-normal* decryption key $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} := (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})$ is defined as follows:

$$\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},0} &:= [\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0} := [\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &:= [\mathbf{k}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2\\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2 \cdot \boxed{[\hat{u}\mathbf{a}^\perp]^{v_0+v_1\mathtt{id}_1+\cdots+v_{|\mathtt{ID}|}\mathtt{id}_{|\mathtt{ID}|}+v_{L+1}\mathtt{T}}},\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},2} &:= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2 \cdot \boxed{[\hat{u}\mathbf{a}^\perp]^{-1}},\\
\mathsf{DK}'_{\mathtt{ID},\mathtt{T},2} &:= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2 \cdot \boxed{[\hat{u}\mathbf{a}^\perp]^{-1}},
\end{aligned}$$

where $\mathbf{u}_{\mathtt{ID},\mathtt{T}}, \mathbf{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ and $\hat{u} \leftarrow_R \mathbb{Z}_p^*$. Here, the boxed part denotes the change from the *normal* decryption key.

Pseudo-SF Decryption Keys: A *pseudo-SF* decryption key $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} := (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})$ is defined as follows:

$$\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},0} &:= [\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0} := [\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &:= [\mathbf{k} + \boxed{\alpha\mathbf{a}^\perp}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2\\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2 \cdot [\hat{u}\mathbf{a}^\perp]^{v_0+v_1\mathtt{id}_1+\cdots+v_{|\mathtt{ID}|}\mathtt{id}_{|\mathtt{ID}|}+v_0+v_{L+1}\mathtt{T}},\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},2} &:= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [\hat{u}\mathbf{a}^\perp]^{-1}, \quad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2} := [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2 \cdot [\hat{u}\mathbf{a}^\perp]^{-1},
\end{aligned}$$

where $\mathbf{u}_{\mathtt{ID},\mathtt{T}}, \mathbf{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, $\hat{u} \leftarrow_R \mathbb{Z}_p^*$, and $\alpha \leftarrow_R \mathbb{Z}_p^*$. Here, the boxed part denotes the change from the *pseudo-normal* decryption key.

Let $(\mathtt{ID}_q, \mathtt{T}_q)$ denote the tuple on which $\mathcal{A}$ makes the q-th decryption key reveal query. We further introduce the following sequence of games for $q \in [Q_{\mathsf{dk}}]$:

$\mathsf{Game}_{\mathrm{II},5,q,1}$: This game is the same as $\mathsf{Game}_{\mathrm{II},5}$ except that

- $\mathcal{C}$ creates *semi-functional* $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ upon $\mathcal{A}$'s first $q-1$ decryption key reveal queries,
- $\mathcal{C}$ creates *pseudo-normal* $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$ upon $\mathcal{A}$'s q-th decryption key reveal query,
- $\mathcal{C}$ creates *normal* $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ upon $\mathcal{A}$'s last $Q_{\mathsf{dk}} - q$ decryption key reveal queries.

$\mathsf{Game}_{\mathrm{II},5,q,2}$: This game is the same as $\mathsf{Game}_{\mathrm{II},5,q,1}$ except that

- $\mathcal{C}$ creates *pseudo-SF* $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$ upon $\mathcal{A}$'s q-th decryption key reveal query.

$\mathsf{Game}_{\mathrm{II},5,q,3}$: This game is the same as $\mathsf{Game}_{\mathrm{II},5,q,2}$ except that

- $\mathcal{C}$ creates *semi-functional* $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$ upon $\mathcal{A}$'s q-th decryption key reveal query.

We use the notation $\mathsf{Game}_{\mathrm{II},5,0,3} = \mathsf{Game}_{\mathrm{II},5}$. By definition, $\mathsf{Game}_{\mathrm{II},5,Q_{\mathsf{dk}},3} = \mathsf{Game}_{\mathrm{II},6}$. Hereafter, we prove

$$\mathsf{Game}_{\mathrm{II},5,q-1,3} \approx_c \mathsf{Game}_{\mathrm{II},5,q,1} \equiv \mathsf{Game}_{\mathrm{II},5,q,2} \approx_c \mathsf{Game}_{\mathrm{II},5,q,3},$$

where the fact implies that $\mathsf{Game}_{\mathrm{II},5} \approx_c \mathsf{Game}_{\mathrm{II},6}$.

Lemma 15 (Decryption Key Transition from Normal to Pseudo-normal, $\mathsf{Game}_{\mathrm{II},5,q-1,3} \approx_c \mathsf{Game}_{\mathrm{II},5,q,1}$). *$\mathsf{Game}_{\mathrm{II},5,q-1,3}$ and $\mathsf{Game}_{\mathrm{II},5,q,1}$ are computationally indistinguishable under the MDDH assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-II adversary $\mathcal{A}$ making at most Q_{gen} secret key generation queries, there exists a reduction algorithm $\mathcal{B}_{\mathrm{II},5}$ such that*

$$|\mathsf{Adv}_{\mathrm{II},5,q-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{II},5,q,1}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},5}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{II},5}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

Proof of Lemma 15. The reduction algorithm $\mathcal{B}_{\mathrm{II},5}$ is given a MDDH instance in $\mathbb{G}_2$: $(\mathcal{G}(1^\lambda), [\mathbf{B}]_2, [\mathbf{b}]_2 = [\mathbf{Bu} + \hat{u}\mathbf{e}]_2)$, where $\mathbf{B} \leftarrow_R \mathcal{D}_k, \mathbf{u} \leftarrow_R \mathbb{Z}_p^k$, $\hat{u} = 0$ or $\hat{u} \leftarrow_R \mathbb{Z}_p$, and $\mathbf{e} = (0, \ldots, 0, 1)^\top \in \mathbb{Z}_p^{k+1}$. Hereafter, we assume that $\hat{u} \leftarrow_R \mathbb{Z}_p^*$ in the latter case with the statistical difference $1/p$.

$\mathcal{B}_{\mathrm{II},5}$ creates MPK and $\mathsf{ct}^\star$ in the same way as the proof of Lemma 6. $\mathcal{B}_{\mathrm{II},5}$ creates $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ by computing (5) and $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ such that $|\mathtt{ID}| \geq 1$ in the same way as the real scheme. $\mathcal{B}_{\mathrm{II},5}$ creates $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$ and $\mathsf{s.sk}^{(2)}_{\mathtt{ID}}$ by computing (9) and computes $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T}}$ by computing (10) and (12), respectively.

We describe how $\mathcal{B}_{\mathrm{II},5}$ creates $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$.

Decryption Key Creations: Upon $\mathcal{A}$' m-th decryption key reveal query on $(\mathtt{ID}_m, \mathtt{T}_m)$, $\mathcal{B}_{\mathrm{II},5}$ creates $\mathsf{dk}_{\mathtt{ID}_m,\mathtt{T}_m}$ as follows:

- If $m < q$, $\mathcal{B}_{\mathrm{II},5}$ creates *semi-functional* $\mathsf{dk}_{\mathtt{ID}_m,\mathtt{T}_m}$ by computing (7).
- If $m = q$, $\mathcal{B}_{\mathrm{II},5}$ retrieves the master secret key $\mathbf{k}$ and $(v_0, v_1, \ldots, v_{|\mathtt{ID}_q|}, v_{L+1})$, samples $\widetilde{\mathbf{u}}_{\mathtt{ID}_q,\mathtt{T}_q}, \widetilde{\mathbf{u}}'_{\mathtt{ID}_q,\mathtt{T}_q} \leftarrow_R \mathbb{Z}_p^{k+1}$, and computes $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q} = (\mathsf{DK}_{\mathtt{ID}_q,\mathtt{T}_q,0}, \mathsf{DK}'_{\mathtt{ID}_q,\mathtt{T}_q,0}, \mathsf{DK}_{\mathtt{ID}_q,\mathtt{T}_q,1}, \mathsf{DK}_{\mathtt{ID}_q,\mathtt{T}_q,2}, \mathsf{DK}'_{\mathtt{ID}_q,\mathtt{T}_q,2})$:

$$\begin{aligned}
\mathsf{DK}_{\mathtt{ID}_q,\mathtt{T}_q,0} &= [\overline{\mathbf{b}}]_2 \cdot [\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID}_q,\mathtt{T}_q}]_2, \qquad \mathsf{DK}'_{\mathtt{ID}_q,\mathtt{T}_q,0} = [\overline{\mathbf{b}}]_2 \cdot [\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID}_q,\mathtt{T}_q}]_2, \\
\mathsf{DK}_{\mathtt{ID}_q,\mathtt{T}_q,1} &= [\mathbf{k}]_2 \cdot [(\widetilde{\mathbf{V}}_0 + \mathtt{id}_{q,1}\widetilde{\mathbf{V}}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}|}\widetilde{\mathbf{V}}_{|\mathtt{ID}|})\overline{\mathbf{b}}]_2 \\
&\quad \cdot [(\widetilde{\mathbf{V}}_0 + \mathtt{T}_q\widetilde{\mathbf{V}}_{L+1})\overline{\mathbf{b}}]_2 \cdot [\mathbf{a}^\perp \underline{\mathbf{b}}]_2^{v_0 + v_1\mathtt{id}_{q,1} + \cdots + v_{q,|\mathtt{ID}_q|}\mathtt{id}_{|\mathtt{ID}_q|} + v_{L+1}\mathtt{T}_q} \\
&\quad \cdot [\big(\mathbf{V}_0 + \mathtt{id}_{q,1}\mathbf{V}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}_q|}\big)\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID}_q,\mathtt{T}_q}]_2 \\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}_q\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID}_q,\mathtt{T}_q}]_2, \\
\mathsf{DK}_{\mathtt{ID}_q,\mathtt{T}_q,2} &= [\widetilde{\mathbf{V}}_{L+2}\overline{\mathbf{b}}]_2 \cdot [-\mathbf{a}^\perp\underline{\mathbf{b}}]_2 \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID}_q,\mathtt{T}_q}]_2, \\
\mathsf{DK}'_{\mathtt{ID}_q,\mathtt{T}_q,2} &= [\widetilde{\mathbf{V}}_{L+2}\overline{\mathbf{b}}]_2 \cdot [-\mathbf{a}^\perp\underline{\mathbf{b}}]_2 \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID}_q,\mathtt{T}_q}]_2.
\end{aligned} \tag{17}$$

  By following the same argument in the proof of Lemma 6, $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$ is a *normal* decryption key as in $\mathsf{Game}_{\mathrm{II},5,q-1,3}$ if $\hat{u} = 0$, and *pseudo-normal* decryption key as in $\mathsf{Game}_{\mathrm{II},5,q,1}$ if $\hat{u} \leftarrow_R \mathbb{Z}_p^*$, by setting $\mathbf{u}_{\mathtt{ID}_q,\mathtt{T}_q} = \widetilde{\mathbf{Z}}^{-1}\mathbf{u} + \widetilde{\mathbf{u}}_{\mathtt{ID}_q,\mathtt{T}_q}$ and $\mathbf{u}'_{\mathtt{ID}_q,\mathtt{T}_q} = \widetilde{\mathbf{Z}}^{-1}\mathbf{u} + \widetilde{\mathbf{u}}'_{\mathtt{ID}_q,\mathtt{T}_q}$.
- If $m > q$, $\mathcal{B}_{\mathrm{II},5}$ creates *normal* $\mathsf{dk}_{\mathtt{ID}_m,\mathtt{T}_m}$ by computing (13).

Thus, we complete the proof of Lemma 15. □

Lemma 16 (Decryption Key Transition from Pseudo-normal to Pseudo-SF, $\mathsf{Game}_{\mathrm{II},5,q,1} \equiv \mathsf{Game}_{\mathrm{II},5,q,2}$). *$\mathsf{Game}_{\mathrm{II},5,q,1}$ and $\mathsf{Game}_{\mathrm{II},5,q,2}$ are identically distributed from $\mathcal{A}$'s view. Specifically, for any Type-II adversary $\mathcal{A}$, it holds that*

$$\mathsf{Adv}_{\mathrm{II},5,q,1}(\lambda) = \mathsf{Adv}_{\mathrm{II},5,q,2}(\lambda).$$

Proof of Lemma 16. Here, we prove a stronger claim that $\mathsf{Game}_{\mathrm{II},5,q,1}$ and $\mathsf{Game}_{\mathrm{II},5,q,2}$ are identically distributed for any fixed

- $(\mathbf{A}, \mathbf{a}) \leftarrow_R \mathcal{D}_k$,
- $\left((\mathbf{V}_\ell)_{\ell\in[0,L+2]}, \mathbf{Z}\right) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$,
- master secret key $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$,
- $\hat{u} \leftarrow_R \mathbb{Z}_p^*$ for creating $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$,
- $\mathbf{c} \leftarrow_R \mathbb{Z}_p^{k+1}$ for creating the challenge ciphertext,
- $(\mathtt{ID}^\star, \mathtt{T}^\star, \mathsf{M}_0^\star, \mathsf{M}_1^\star) \in \mathbb{Z}_p^2 \times \mathcal{M}^2$ and random coin $\mathsf{coin} \leftarrow_R \{0,1\}$,
- $\mathbf{u}_{\mathtt{ID}_q,\mathtt{T}_q}, \mathbf{u}'_{\mathtt{ID}_q,\mathtt{T}_q} \leftarrow_R \mathbb{Z}_p^k$ and $\hat{u} \leftarrow_R \mathbb{Z}_p^*$ for creating q-th queried $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$.

Specifically, the randomness of $(v_0, v_1, \ldots, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{L+2}$ enables us to prove the claim. Since all $\mathsf{sk}_{\mathtt{ID}}$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ except $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$ are created in the same way in both $\mathsf{Game}_{\mathrm{II},5,q,1}$ and $\mathsf{Game}_{\mathrm{II},5,q,2}$, and all the other elements have been already fixed, it is sufficient to show that

$$\begin{aligned}&\left\{\begin{array}{c} v_0 + \mathtt{id}_1^\star v_1 + \cdots + \mathtt{ID}_{|\mathtt{ID}^\star|} v_{|\mathtt{ID}^\star|}, v_0 + \mathtt{T}^\star v_{L+1}, \\ v_0 + \mathtt{id}_{q,1} v_1 + \cdots + \mathtt{ID}_{q,|\mathtt{ID}_q|} v_{|\mathtt{ID}_q|} + \mathtt{T}_q v_{L+1} \end{array}\right\} \\ \equiv &\left\{\begin{array}{c} v_0 + \mathtt{id}_1^\star v_1 + \cdots + \mathtt{ID}_{|\mathtt{ID}^\star|} v_{|\mathtt{ID}^\star|}, v_0 + \mathtt{T}^\star v_{L+1}, \\ \alpha/\hat{u} + v_0 + \mathtt{id}_{q,1} v_1 + \cdots + \mathtt{ID}_{q,|\mathtt{ID}_q|} v_{|\mathtt{ID}_q|} + \mathtt{T}_q v_{L+1} \end{array}\right\},\end{aligned} \tag{18}$$

where $(v_0, v_1, \ldots, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{L+2}$. Here, the first two elements are $\mathtt{tag}, \mathtt{tag}'$ and last element is the exponent of $[\hat{u}\mathbf{a}^\perp]_2$ of $\mathsf{DK}_{\mathtt{ID}_q,\mathtt{T}_q,1}$ in $\mathsf{Game}_{\mathrm{II},5,q,1}$ and $\mathsf{Game}_{\mathrm{II},5,q,2}$, respectively. If $\mathtt{ID}_q \notin \mathsf{prefix}^+(\mathtt{ID}^\star)$ holds, $\left\{v_0 + \mathtt{id}_1^\star v_1 + \cdots + \mathtt{ID}_{|\mathtt{ID}^\star|} v_{|\mathtt{ID}^\star|}, v_0 + \mathtt{id}_{q,1} v_1 + \cdots + \mathtt{ID}_{q,|\mathtt{ID}_q|} v_{|\mathtt{ID}_q|}\right\}$ is distributed in $\mathbb{Z}_p^2$ uniformly at random by following the standard argument for proving HIBE. If $\mathtt{T}_q \neq \mathtt{T}^\star$ holds, $\{v_0 + \mathtt{T}^\star v_{L+1}, v_0 + \mathtt{T}_q v_{L+1}\}$ is distributed in $\mathbb{Z}_p^2$ uniformly at random by following the standard argument for proving IBE. If $\mathtt{ID}_q \in \mathsf{prefix}^+(\mathtt{ID}^\star) \setminus \{\mathtt{ID}^\star\}$, $\left\{v_0 + \mathtt{id}_1^\star v_1 + \cdots + \mathtt{ID}_{|\mathtt{ID}^\star|} v_{|\mathtt{ID}^\star|}, v_0 + \mathtt{id}_{q,1} v_1 + \cdots + \mathtt{ID}_{q,|\mathtt{ID}_q|} v_{|\mathtt{ID}_q|}\right\}$ is distributed in $\mathbb{Z}_p^2$ uniformly at random due to the random $\mathtt{ID}_{q,|\mathtt{ID}_q|+1} v_{|\mathtt{ID}_q|+1} + \mathtt{ID}_{|\mathtt{ID}^\star|} v_{|\mathtt{ID}^\star|}$. Since $(\mathtt{ID}_q, \mathtt{T}_q) \neq (\mathtt{ID}^\star, \mathtt{T}^\star)$ holds due to the security definition of RHIBE, we have proved the claim. Thus, we complete the proof of Lemma 16. □

Lemma 17 (Decryption Key Transition from Pseudo-SF to Semi-functional, $\mathsf{Game}_{\mathrm{II},5,q,2} \approx_c \mathsf{Game}_{\mathrm{II},5,q,3}$)**.** $\mathsf{Game}_{\mathrm{II},5,q,2}$ *and* $\mathsf{Game}_{\mathrm{II},5,q,3}$ *are computationally indistinguishable under the* MDDH *assumption in* $\mathbb{G}_2$*. Specifically, for any PPT Type-*II *adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key generation queries, there exists a reduction algorithm* $\mathcal{B}_{\mathrm{II},6}$ *such that*

$$|\mathsf{Adv}_{\mathrm{II},5,q,2}(\lambda) - \mathsf{Adv}_{\mathrm{II},5,q,3}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},6}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{II},6}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

We omit the detailed proof of Lemma 17 since it is almost the same as the proof of Lemma 15. The only difference is that $\mathcal{B}_{\mathrm{II},6}$ creates $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$ upon $\mathcal{A}$'s q-th decryption key reveal query by

$$\begin{aligned}\mathsf{DK}_{\mathtt{ID}_q,\mathtt{T}_q,1} = &[\mathbf{k} + \boxed{\alpha\mathbf{a}^\perp}]_2 \cdot [(\widetilde{\mathbf{V}}_0 + \mathtt{id}_{q,1}\widetilde{\mathbf{V}}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}|}\widetilde{\mathbf{V}}_{|\mathtt{ID}|})\overline{\mathbf{b}}]_2 \\ &\cdot [(\widetilde{\mathbf{V}}_0 + \mathtt{T}_q\widetilde{\mathbf{V}}_{L+1})\overline{\mathbf{b}}]_2 \cdot [\mathbf{a}^\perp \underline{\mathbf{b}}]_2^{v_0 + v_1\mathtt{id}_{q,1} + \cdots + v_{q,|\mathtt{ID}_q|}\mathtt{id}_{|\mathtt{ID}_q|} + v_{L+1}\mathtt{T}_q} \\ &\cdot [\left(\mathbf{V}_0 + \mathtt{id}_{q,1}\mathbf{V}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}_q|}\right)\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID}_q,\mathtt{T}_q}]_2 \\ &\cdot [(\mathbf{V}_0 + \mathtt{T}_q\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID}_q,\mathtt{T}_q}]_2,\end{aligned}$$

where the boxed parts denote the changes from (17). If $\hat{u} \leftarrow_R \mathbb{Z}_p^*$, $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$ is a *pseudo-SF* decryption key as in $\mathsf{Game}_{\mathrm{II},5,q,2}$. If $\hat{u} = 0$, $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}_q}$ is a *semi-functional* decryption key as in $\mathsf{Game}_{\mathrm{II},5,q,3}$.

By combining Lemmata 15–17, we have

$$\begin{aligned}
&|\mathsf{Adv}_{\mathrm{II},5}(\lambda) - \mathsf{Adv}_{\mathrm{II},6}(\lambda)| \\
\leq &\sum_{q\in[Q_{\mathsf{dk}}]} |\mathsf{Adv}_{\mathrm{II},5,q-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{II},5,q,1}(\lambda)| + \sum_{q\in[Q_{\mathsf{dk}}]} |\mathsf{Adv}_{\mathrm{II},5,q,1}(\lambda) - \mathsf{Adv}_{\mathrm{II},5,q,2}(\lambda)| \\
&+ \sum_{q\in[Q_{\mathsf{dk}}]} |\mathsf{Adv}_{\mathrm{II},5,q,2}(\lambda) - \mathsf{Adv}_{\mathrm{II},5,q,3}(\lambda)| \\
\leq &\, Q_{\mathsf{dk}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j+4}}(\lambda) + \frac{4Q_{\mathsf{dk}}}{p-1}.
\end{aligned}$$

Thus, we complete the proof of Lemma 14. □

Lemma 18 (Final Transition, $\mathsf{Game}_{\mathrm{II},6} \equiv \mathsf{Game}_{\mathrm{II},7}$)**.** $\mathsf{Game}_{\mathrm{II},6}$ *and* $\mathsf{Game}_{\mathrm{II},7}$ *are identically distributed. Specifically, for any Type-*II *adversary* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}_{\mathrm{II},6}(\lambda) = \mathsf{Adv}_{\mathrm{II},7}(\lambda).$$

Proof of Lemma 18. Run $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e) \leftarrow \mathcal{G}(1^\lambda)$ and sample $(\mathbf{A}, \mathbf{a}^\perp) \leftarrow \mathcal{D}_k$, $\big((\mathbf{V}_\ell)_{\ell\in[0,L+2]}, \mathbf{Z}\big) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$, $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$, and $\alpha \leftarrow_R \mathbb{Z}_p^*$. We sets $\mathsf{MSK} = \mathbf{k} - \alpha\mathbf{a}^\perp$ and returns

$$\mathsf{MPK} = \Big([\mathbf{A}]_1, ([\mathbf{V}_\ell^\top \mathbf{A}]_1)_{\ell\in[0,L+2]}, [\mathbf{Z}]_2, ([\mathbf{V}_\ell \mathbf{Z}]_2)_{\ell\in[0,L+2]}, [\mathbf{A}^\top \mathbf{k}]_T\Big)$$

to $\mathcal{A}$. Since it holds that

$$\begin{aligned}
[\mathbf{A}^\top \mathbf{k}]_T &= e([\mathbf{A}]_1, [\mathbf{k}]_2) = e([\mathbf{A}]_1, [\mathbf{k}]_2) \cdot e([\mathbf{A}]_1, [\mathbf{a}^\perp]_2^{-\alpha}) = e([\mathbf{A}]_1, [\mathbf{k} - \alpha\mathbf{a}^\perp]_2) \\
&= [\mathbf{A}^\top \mathsf{MSK}]_T,
\end{aligned}$$

MPK follows the same distribution as the real scheme. Furthermore, MPK does not reveal the quantity of α in both $\mathsf{Game}_{\mathrm{II},6}$ and $\mathsf{Game}_{\mathrm{II},7}$. We create $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$ by computing (9) and create $\mathsf{sk}_{\mathtt{ID},\theta}$ by computing (10). We create $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ such that $|\mathtt{ID}| \geq 1$ in the same way as the real scheme. In $\mathsf{Game}_{\mathrm{II},6}$, $\mathcal{C}$ uses MSK only for computing *semi-functional* $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ (5), *semi-functional* $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ (6), and *semi-functional* $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ (7). In this proof, since $[\mathbf{k}]_2 = \mathsf{MSK} + \alpha\mathbf{a}^\perp$, we use $[\mathbf{k}]_2$ and create $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ by computing (11), (12), and (13), and they follow *semi-functional* distribution as in (5), (6), and (7), respectively.

Summarizing the creations so far, we do not use MSK for creating all MPK, $\mathsf{sk}_{\mathtt{ID},\theta}$, $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$. In other words, the quantity of α is not revealed to $\mathcal{A}$ so far. In both $\mathsf{Game}_{\mathrm{II},6}$ and $\mathsf{Game}_{\mathrm{II},7}$, $\mathsf{ct}^\star$ follows the same distribution except C_2. In $\mathsf{Game}_{\mathrm{II},6}$, C_2 is distributed as follows:

$$C_2 = \mathsf{M}_{\mathsf{coin}} \cdot [\mathbf{c}^\top \mathsf{MSK}]_T = \Big(\mathsf{M}_{\mathsf{coin}} \cdot [-\alpha\mathbf{c}^\top \mathbf{a}^\perp]_T\Big) \cdot [\mathbf{c}^\top \mathbf{k}]_T.$$

Since $\alpha \in \mathbb{Z}_p^*$, $-\alpha\mathbf{c}\mathbf{a}^\perp = 0$ holds only when $\mathbf{c}^\top \mathbf{a}^\perp = 0$. Since $\mathbf{c}$ is distributed in $\mathbb{Z}_p^{k+1}$ uniformly at random, is holds that $\mathbf{c}^\top \mathbf{a}^\perp = 0$ with probability $1/p$. In contrast, when $\mathbf{c}^\top \mathbf{a}^\perp \neq 0$, $-\alpha\mathbf{c}\mathbf{a}^\perp$ for $\alpha \leftarrow_R \mathbb{Z}_p^*$ is distributed in $\mathbb{Z}_p^*$ uniformly at random. Then, $-\alpha\mathbf{c}\mathbf{a}^\perp$ becomes each non-zero value

with probability $\left(1-\frac{1}{p}\right)\cdot\frac{1}{p-1}=\frac{1}{p}$. Therefore, $\mathsf{M}_{\mathsf{coin}}\cdot[-\alpha\mathbf{c}^{\top}\mathbf{a}^{\perp}]_T$ is distributed in $\mathbb{G}_T$ uniformly at random. Thus, we complete the proof of Lemma 18. □

By combining with Lemmata 3, 4, 5, 9, 10, 14, and 18, we have

$$\begin{aligned}
&\mathsf{Adv}^{\mathtt{RHIBE}}_{\Pi,L,\mathcal{A}}(\lambda)\\
&\leq \sum_{i\in[7]}|\mathsf{Adv}_{\mathbb{II},i-1}(\lambda)-\mathsf{Adv}_{\mathbb{II},i}(\lambda)|+\mathsf{Adv}_{\mathbb{II},7}(\lambda)\\
&\leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda)+Q_{\mathsf{rev}}\cdot\sum_{j\in[2]}\mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathbb{II},j}}(\lambda)+Q_{\mathsf{gen}}\cdot\sum_{j\in[2]}\mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathbb{II},j+2}}(\lambda)\\
&\quad+Q_{\mathsf{dk}}\cdot\sum_{j\in[2]}\mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathbb{II},j+4}}(\lambda)+\frac{4(Q_{\mathsf{rev}}+Q_{\mathsf{gen}}+Q_{\mathsf{dk}})}{p}.
\end{aligned}$$

By definition, $Q_{\mathsf{rev}}\leq Q_{\mathsf{gen}}$ and $Q_{\mathsf{rev}}\leq Q_{\mathsf{gen}}|\mathcal{T}|$ hold. Therefore, it holds that

$$\begin{aligned}
&\mathsf{Adv}^{\mathtt{RHIBE}}_{\Pi,L,\mathcal{A}}(\lambda)\\
&\leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda)+Q_{\mathsf{gen}}\left(\sum_{j\in[4]}\mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathbb{II},j}}(\lambda)+|\mathcal{T}|\cdot\sum_{j\in[2]}\mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathbb{II},4+j}}(\lambda)\right)\\
&\quad+O\left(\frac{Q_{\mathsf{gen}}|\mathcal{T}|}{p}\right).
\end{aligned}$$

Thus, we complete the proof of Lemma 2. □

6 Adaptive Security against the Type-I Adversary

We repost the definition of the Type-I adversary:

Type-I Adversary: $\mathcal{A}$ is called Type-I if it makes the secret key *reveal* queries on some $\mathtt{ID}\in\mathsf{prefix}^{+}(\mathtt{ID}^{\star})$.

It is clear that our proof strategy against the Type-II adversary is insufficient for proving the adaptive security against the Type-I adversary since $\mathcal{A}$ receives $\mathsf{sk}_{\mathtt{ID}}$ such that $\mathtt{ID}\in\mathsf{prefix}^{+}(\mathtt{ID}^{\star})$. Although we do not perform a detailed analysis of this problem, we believe that by combining the proof technique of Emura et al. [ETW20] against the Type-I adversary and the semi-functional randomness switching, we may be able to prove the adaptive security of our RHIBE scheme against the Type-I adversary. As we claimed in Remark 2, Emura et al. divided the Type-I adversary into the Type-I-$\ell^{\star}$ adversary for $\ell^{\star}\in[L]$ such that $\mathcal{A}$ makes the secret key *reveal* a query on $\mathtt{ID}^{\star}_{[\ell^{\star}]}$, while $\mathcal{A}$ does not make the secret key *reveal* queries on any $\mathtt{ID}^{\star}_{[\ell]}$ for $\ell\in[\ell^{\star}-1]$. Thus, Emura et al.'s proof technique inherently suffers from $O(L)$ reduction loss. This reduction loss is unavoidable for their proof technique since they used the value $\ell^{\star}$ to define the way in which the reduction algorithm answers $\mathcal{A}$'s key queries.

We adopted another approach for achieving tighter reduction. Let $\mathtt{ID}_q$ denote the identity on which $\mathcal{A}$ makes q-th secret key *generation* query. First, we determine the number $Q^{\star}\in[Q_{\mathsf{gen}}]$ such that $\mathtt{ID}_{Q^{\star}}=\mathtt{ID}^{\star}_{[\ell^{\star}]}$ with Q_{gen} reduction loss. Although we also use the value $\ell^{\star}$ to design the manner in which the reduction algorithm answers $\mathcal{A}$'s key queries, as done by Emura et al., our proof does not suffer from $O(L)$ reduction loss since the reduction algorithm answers all the key queries of $\mathcal{A}$ in the same manner until $\mathcal{A}$'s $Q^{\star}$-th secret key *generation* query. By definition of $\mathtt{ID}^{\star}_{[\ell^{\star}]}$,

all $\mathtt{ID}$ on which $\mathcal{A}$ makes the secret key *reveal* queries satisfy $\mathtt{ID} \notin \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$ until $\mathcal{A}$'s $Q^{\star}$-th secret key *generation* query. After $\mathcal{A}$'s $Q^{\star}$-th secret key *generation* query, we can detect whether $\mathtt{ID}^{\star}_{[\ell^{\star}]} \notin \mathsf{prefix}^{+}(\mathtt{ID})$ holds for any $\mathtt{ID}$. Thus, upon $\mathcal{A}$'s secret key *reveal* queries on $\mathtt{ID}$, we change $\mathsf{sk}_{\mathtt{ID}}$ to be semi-functional only when it holds that $\mathtt{ID}^{\star}_{[\ell^{\star}]} \notin \mathsf{prefix}^{+}(\mathtt{ID})$.

Next, we explain how to change all $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ to be semi-functional. In this case, our proof technique against the Type-II adversary is still insufficient for proving adaptive security against the Type-I adversary. By definition of $\mathtt{ID}^{\star}_{[\ell^{\star}]}$, $\mathsf{sk}_{\mathtt{ID}^{\star}_{[\ell^{\star}-1]}}$ that includes the delegation keys $\mathsf{delk}_{\mathtt{ID}^{\star}_{[\ell^{\star}-1]},\theta}$ is not revealed to $\mathcal{A}$. Nevertheless, we cannot apply the semi-functional randomness switching to change $\mathsf{ku}_{\mathtt{ID}^{\star}_{[\ell^{\star}-1]},\mathtt{T}}$ to be semi-functional since we cannot change $\mathsf{sk}_{\mathtt{ID}^{\star}_{[\ell^{\star}]}}$ to be semi-functional. To overcome this problem, we use the information derived from the guess of $Q^{\star}$. After $\mathcal{A}$'s $Q^{\star}$-th secret key *generation* query, we can detect the time period $\mathtt{T}_{\mathsf{RL}}$ when $\mathtt{ID}^{\star}_{[\ell^{\star}]}$ is revoked. From the security definition of RHIBE, since $\mathtt{ID}^{\star}_{[\ell^{\star}]}$ has to be revoked by the challenge time period $\mathtt{T}^{\star}$, $\mathtt{T}_{\mathsf{RL}} \leq \mathtt{T}^{\star}$ holds. In other words, we can use the fact $\mathtt{T} \neq \mathtt{T}^{\star}$ to change all $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ to be semi-functional before the time period $\mathtt{T}_{\mathsf{RL}}$. From the above discussion, $\mathcal{A}$ receives normal $\mathsf{sk}_{\mathtt{ID}^{\star}_{[\ell^{\star}]}}$. On the other hand, after time period $\mathtt{T}_{\mathsf{RL}}$, all $\mathsf{ku}_{\mathtt{ID}^{\star}_{[\ell^{\star}-1]},\mathtt{T},\theta}$ and $\mathsf{sk}_{\mathtt{ID}^{\star}_{[\ell^{\star}]},\theta}$ do not share the same node since $\mathtt{ID}^{\star}_{[\ell^{\star}]}$ is already revoked. Based on this fact, we can apply semi-functional randomness switching to change all $\mathsf{ku}_{\mathtt{ID}^{\star}_{[\ell^{\star}-1]},\mathtt{T}}$ to be semi-functional.

Following this argument, we prove the adaptive security of our RHIBE scheme. Before providing an overview of our proof, we introduce the following *seed* key update and its semi-functional distribution.

Normal Seed Key Updates: A *normal* seed key update is defined as $\mathsf{s.ku}_{\mathtt{T}} \coloneqq (\mathsf{s.KU}_{\mathtt{T},0}, \mathsf{s.KU}_{\mathsf{KU},1}, \mathsf{s.KU}_{\mathsf{KU},2})$:

$$\begin{aligned} &\mathsf{s.KU}_{\mathtt{T},0} \coloneqq [\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2, \qquad \mathsf{s.KU}_{\mathtt{T},1} \coloneqq [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2, \\ &\mathsf{s.KU}_{\mathtt{T},2} \coloneqq [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2, \end{aligned} \tag{19}$$

where $\mathbf{t}_{\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$.

Semi-functional Seed Key Updates: A *semi-functional* seed key update is defined as $\mathsf{s.ku}_{\mathtt{T}} \coloneqq (\mathsf{s.KU}_{\mathtt{T},0}, \mathsf{s.KU}_{\mathsf{KU},1}, \mathsf{s.KU}_{\mathsf{KU},2})$:

$$\begin{aligned} &\mathsf{s.KU}_{\mathtt{T},0} \coloneqq [\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2, \qquad \mathsf{s.KU}_{\mathtt{T},1} \coloneqq \boxed{[\alpha\mathbf{a}^{\perp}]_2} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2, \\ &\mathsf{s.KU}_{\mathtt{T},2} \coloneqq [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2, \end{aligned} \tag{20}$$

where $\mathbf{t}_{\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ and $\alpha \leftarrow_R \mathbb{Z}_p^*$. Here, the term in the box denotes the change from the *normal* seed key update.

We use the following sequence of games to prove the adaptive security against the Type-I adversary:

$\mathsf{Game}_{\mathrm{I},0}$: This is the real security game between the challenger $\mathcal{C}$ and adversary $\mathcal{A}$.

$\mathsf{Game}_{\mathrm{I},1}$: This game is the same as $\mathsf{Game}_{\mathrm{II},0}$ except that the challenge ciphertext $\mathsf{ct}^{\star}$ is *semi-functional*.

$\mathsf{Game}_{\mathrm{I},2}$: Let $\mathtt{ID}^{\star}_{[\ell^{\star}]} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star})$ denote an identity such that $\mathcal{A}$ makes the secret key r*eveal* queries on $\mathtt{ID}^{\star}_{[\ell^{\star}]}$, while $\mathcal{A}$ does not make the secret key reveal any query on any $(\mathtt{ID}^{\star}_{[\ell]})_{\ell \in [\ell^{\star}-1]}$. Let $\mathtt{ID}_q$ denote the identity on which $\mathcal{A}$ makes q-th secret key *generation* query. This game is the same as $\mathsf{Game}_{\mathrm{I},1}$ except that $\mathcal{C}$ guesses the number $Q^{\star}$ such that $\mathtt{ID}_{Q^{\star}} = \mathtt{ID}^{\star}_{[\ell^{\star}]}$. If the guess is not correct, $\mathcal{C}$ aborts the game and outputs a random bit $\widehat{\mathsf{coin}} \leftarrow_R \{0,1\}$. Hereafter, let

T_{RL} denote the first time period such that $\mathtt{ID}_{Q^\star} \in \mathsf{RL}_{\mathsf{T}_{\mathsf{RL}}}$. From the definition of the Type-I adversary, it holds that $\mathsf{T}_{\mathsf{RL}} \leq \mathsf{T}^\star$ if the guess is correct. Hereafter, we describe the case only when the guess is correct.

$\mathsf{Game}_{\mathrm{I},3}$: This game is the same as $\mathsf{Game}_{\mathrm{I},2}$ except that $\mathcal{C}$ modifies the method of creating secret keys $\mathsf{sk}_{\mathtt{ID}}$, key updates $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, and decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ as follows:

Table 4: Distributions of $\mathsf{ct}^\star$, $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ for creating $\mathsf{sk}_{\mathtt{ID},\theta}$, and $\mathsf{sk}_{\mathtt{ID},\theta}$ in each game in the proof against the Type-I adversary. In the column $\mathsf{ct}^\star$, we specify the distribution and encrypted plaintext. In the other columns, we specify the distributions and semi-functional randomness of $\mathsf{s.sk}_{\mathtt{ID}}$ and $\mathsf{sk}_{\mathtt{ID},\theta}$.

Game	$\mathsf{ct}^\star$	$\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ for $\mathtt{ID}^\star_{[\ell^\star]} \notin \mathsf{prefix}^+(\mathtt{ID})$	$\mathsf{sk}_{\mathtt{ID},\theta}$ for $\mathtt{ID}^\star_{[\ell^\star]} \notin \mathsf{prefix}^+(\mathtt{ID})$
$\mathsf{Game}_{\mathrm{I},0}$	normal $\mathsf{M}^\star_{\mathsf{coin}}$	normal	normal
$\mathsf{Game}_{\mathrm{I},1}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	normal	normal
$\mathsf{Game}_{\mathrm{I},2}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	normal	normal
$\mathsf{Game}_{\mathrm{I},3}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{I},4}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{I},5}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{I},6}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{I},7}$	semi-functional $\mathsf{M}^\star_{\mathsf{coin}}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{I},8}$	semi-functional $\mathsf{M}^\star \leftarrow_R \mathbb{G}_T$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\tilde{r}_{\mathtt{ID},\theta}\alpha;\ \tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$

Secret Key Creation: Upon $\mathcal{A}$'s secret key *generation* queries on $\mathtt{ID}$, $\mathcal{C}$ does not create sub-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$. Upon $\mathcal{A}$'s secret key *reveal* queries on $\mathtt{ID}$, $\mathcal{C}$ first creates normal *seed* secret keys $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$. Then, $\mathcal{C}$ uses $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ to create all sub-secret keys $\mathsf{sk}_{\mathtt{ID},\theta}$.

Key Update and Decryption Key Creation: $\mathcal{C}$ proceeds as follows.

- For each time period $\mathtt{T} < \mathsf{T}_{\mathsf{RL}}$ upon the setup and $\mathcal{A}$'s revoke & key update queries, $\mathcal{C}$ first creates normal *seed* key updates $\mathsf{s.ku}_{\mathtt{T}}$. To create $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ (including $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$) for $\mathtt{T} < \mathsf{T}_{\mathsf{RL}}$, $\mathcal{C}$ uses the seed key updates $\mathsf{s.ku}_{\mathtt{T}}$ for computing the sub-key updates $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$. It also creates the helper key updates $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$ in the same way as in

the real scheme. To create $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} < \mathtt{T}_{\mathsf{RL}}$, $\mathcal{C}$ uses the seed key updates $\mathsf{s.ku}_{\mathtt{T}}$ for computing the decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$.

– Upon $\mathcal{A}$'s secret key *generation* queries on $\mathtt{ID}$, $\mathcal{C}$ first creates normal *seed* secret keys $\mathsf{s.sk}^{(2)}_{\mathtt{ID}}$. $\mathcal{C}$ creates $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ in the same manner as in the real scheme. To create $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for all $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$, $\mathcal{C}$ uses the seed secret keys $\mathsf{s.sk}^{(2)}_{\mathtt{ID}}$ for computing the helper decryption keys $\overline{\mathsf{dk}}_{\mathtt{ID},\mathtt{T}}$ and creating the sub-key updates $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ in the same way as in the real scheme.

$\mathsf{Game}_{\mathrm{I},4}$: This game is the same as $\mathsf{Game}_{\mathrm{I},3}$ except that $\mathcal{C}$ creates *semi-functional* seed secret keys $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$ upon $\mathcal{A}$'s secret key *reveal* queries if $\mathtt{ID}_{Q^\star} \notin \mathsf{prefix}^{+}(\mathtt{ID})$ holds.

$\mathsf{Game}_{\mathrm{I},5}$: This game is the same as $\mathsf{Game}_{\mathrm{I},4}$ except that $\mathcal{C}$ creates *semi-functional* seed key updates $\mathsf{s.ku}_{\mathtt{T}}$. Furthermore, each helper key update $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} < \mathtt{T}_{\mathsf{RL}}$ is also *semi-functional*.

$\mathsf{Game}_{\mathrm{I},6}$: This game is the same as $\mathsf{Game}_{\mathrm{I},5}$ except that $\mathcal{C}$ creates *semi-functional* $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$.

$\mathsf{Game}_{\mathrm{I},7}$: This game is the same as $\mathsf{Game}_{\mathrm{I},6}$ except that $\mathcal{C}$ creates *semi-functional* helper key updates $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ and decryption keys $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$.

Table 5: Distributions of $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ for $|\mathtt{ID}| \geq 1$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} < \mathsf{T}_{\mathsf{RL}}$ in each game in the proof against the Type-I adversary. We specify the distributions and semi-functional randomness of $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$.

Game	$\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ for $\mathtt{T} < \mathsf{T}_{\mathsf{RL}}$	$\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} < \mathsf{T}_{\mathsf{RL}}$	$\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} < \mathsf{T}_{\mathsf{RL}}$
$\mathsf{Game}_{\mathrm{I},0}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},1}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},2}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},3}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},4}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},5}$	semi-functional $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta}\alpha$; $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$	semi-functional $\tilde{\tilde{t}}_{\mathtt{ID},\mathtt{T}}\alpha$; $\tilde{\tilde{t}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$	semi-functional $\tilde{u}'_{\mathtt{ID},\mathtt{T}}\alpha$; $\tilde{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{I},6}$	semi-functional $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta}\alpha$; $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$	semi-functional $\tilde{\tilde{t}}_{\mathtt{ID},\mathtt{T}}\alpha$; $\tilde{\tilde{t}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$	semi-functional $\tilde{u}'_{\mathtt{ID},\mathtt{T}}\alpha$; $\tilde{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{I},7}$	semi-functional $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta}\alpha$; $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$	semi-functional $\tilde{\tilde{t}}_{\mathtt{ID},\mathtt{T}}\alpha$; $\tilde{\tilde{t}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$	semi-functional $\tilde{u}'_{\mathtt{ID},\mathtt{T}}\alpha$; $\tilde{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$
$\mathsf{Game}_{\mathrm{I},8}$	semi-functional $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta}\alpha$; $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$	semi-functional $\tilde{\tilde{t}}_{\mathtt{ID},\mathtt{T}}\alpha$; $\tilde{\tilde{t}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$	semi-functional $\tilde{u}'_{\mathtt{ID},\mathtt{T}}\alpha$; $\tilde{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$

Table 6: Distributions of $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ for $|\mathtt{ID}| \geq 1$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ in each game in the proof against the Type-I adversary. In the columns, we specify the distributions and semi-functional randomness of $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$, $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$.

Game	$\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$	$\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$	$\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$
$\mathsf{Game}_{\mathrm{I},0}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},1}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},2}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},3}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},4}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},5}$	normal	normal	normal
$\mathsf{Game}_{\mathrm{I},6}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	normal	normal
$\mathsf{Game}_{\mathrm{I},7}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$
$\mathsf{Game}_{\mathrm{I},8}$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$	semi-functional $\alpha \leftarrow_R \mathbb{Z}_p^*$

$\mathsf{Game}_{\mathrm{I},8}$: This game is the same as $\mathsf{Game}_{\mathrm{I},7}$ except that the challenge ciphertext $\mathsf{ct}^\star$ is the semi-functional encryption of a *random plaintext*.

In Tables 4–6, we summarize the distributions of $\mathsf{ct}^\star$, $\mathsf{sk}_{\mathtt{ID}}$, $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ in each game. The definitions of $\mathsf{Game}_{\mathrm{I},0}$ and $\mathsf{Game}_{\mathrm{I},1}$ are identical to those of $\mathsf{Game}_{\mathrm{II},0}$ and $\mathsf{Game}_{\mathrm{II},1}$, respectively. Thus, we can prove the indistinguishability $\mathsf{Game}_{\mathrm{I},0} \approx_c \mathsf{Game}_{\mathrm{I},1}$ by Lemma 3. In $\mathsf{Game}_{\mathrm{I},2}$, we guess the value $Q^\star$ with Q_{gen} reduction loss. $\mathsf{Game}_{\mathrm{I},3}$ is the conceptual change that is useful to reduce the reduction loss. In $\mathsf{Game}_{\mathrm{I},3}$, $\mathcal{C}$ does not create $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ unlike in the real scheme. In turn, $\mathcal{C}$ first creates *seed* secret keys $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ and $\mathsf{s.sk}_{\mathtt{ID}}^{(2)}$, and uses the seed secret keys to create $\mathsf{sk}_{\mathtt{ID},\theta}$ and $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}, \mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$, respectively. For each time period $\mathtt{T} < \mathtt{T}_{\mathsf{RL}}$, $\mathcal{C}$ creates *seed* key update $\mathsf{s.ku}_{\mathtt{T}}$ and uses the seed key update to create $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}, \mathsf{dk}_{\mathtt{ID},\mathtt{T}}$. In $\mathsf{Game}_{\mathrm{I},4}$, $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ revealed to $\mathcal{A}$ become *semi-functional* when $\mathtt{ID}^\star_{[\ell^\star]} \notin \mathtt{ID}$. We use the standard dual system argument to prove the indistinguishability $\mathsf{Game}_{\mathrm{I},3} \approx_c \mathsf{Game}_{\mathrm{I},4}$ (Lemma 21) by considering the fact that $\mathtt{ID} \notin \mathsf{prefix}^+(\mathtt{ID}^\star)$. In $\mathsf{Game}_{\mathrm{I},5}$, $\mathsf{s.ku}_{\mathtt{T}}$ revealed to $\mathcal{A}$ becomes *semi-functional*. We use the standard dual system argument to prove that all $\mathsf{s.ku}_{\mathtt{T}}$ are semi-functional by considering the fact that $\mathtt{T} \neq \mathtt{T}^\star$. Although $\mathsf{s.ku}_{\mathtt{T}}$ are used to create $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$, we define $\mathsf{Game}_{\mathrm{I},5}$ so that $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ are semi-functional. Hence, we apply *semi-functional randomness switching* and prove the indistinguishability $\mathsf{Game}_{\mathrm{I},4} \approx_c \mathsf{Game}_{\mathrm{I},5}$ (Lemma 22). In $\mathsf{Game}_{\mathrm{I},6}$ and $\mathsf{Game}_{\mathrm{I},7}$, we change $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ and $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}, \mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ to be semi-functional by applying *semi-functional randomness switching*. Finally, in $\mathsf{Game}_{\mathrm{I},8}$, we change the challenge ciphertext $\mathsf{ct}^\star$ to be a semi-functional encryption of a random plaintext as done in the proof against the Type-II adversary.

6.1 Proof of Lemma 1

Now, we are ready to prove Lemma 1.

Proof of Lemma 1. Let $\mathsf{Adv}_i(\lambda)$ denote $\mathcal{A}$'s advantage in $\mathsf{Game}_{\mathrm{II},i}$. Hereafter, we prove that the difference of $\mathcal{A}$'s advantage between each game (i.e., $|\mathsf{Adv}_{i-1}(\lambda) - \mathsf{Adv}_i(\lambda)|$) is negligible. The indistinguishability $\mathsf{Game}_{\mathrm{I},0} \approx_c \mathsf{Game}_{\mathrm{I},1}$ is proven as Lemma 3. The key points to note is the transitions $\mathsf{Game}_{\mathrm{II},3} \equiv \mathsf{Game}_{\mathrm{II},4}$ and $\mathsf{Game}_{\mathrm{II},4} \approx_c \mathsf{Game}_{\mathrm{II},5}$ since we have to change $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ and $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $\mathtt{ID} \in \mathsf{prefix}^{+}(\mathtt{ID}^{\star}) \wedge \mathtt{T} = \mathtt{T}^{\star}$ to be semi-functional. In other words, we rely on standard dual system proof [CGW15, CG17, CW14] to prove most of the other transitions.

Lemma 19 ($\mathsf{Game}_{\mathrm{I},1} \equiv \mathsf{Game}_{\mathrm{I},2}$)**.** $\mathsf{Game}_{\mathrm{II},1}$ *and* $\mathsf{Game}_{\mathrm{I},2}$ *are identically distributed from* $\mathcal{A}$*'s view with non-negligible probability. Specifically, for any Type-*I *adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key* generation *queries, it holds that*

$$\mathsf{Adv}_{\mathrm{I},1}(\lambda) = Q_{\mathsf{gen}} \cdot \mathsf{Adv}_{\mathrm{I},2}(\lambda).$$

Proof of Lemma 19. Let $\mathsf{E}_{\mathrm{I},1}$ and $\mathsf{E}_{\mathrm{I},2}$ denote the event that $\mathcal{A}$ wins in $\mathsf{Game}_{\mathrm{I},1}$ and $\mathsf{Game}_{\mathrm{I},2}$, respectively. Let F denote the event that $\mathcal{C}$'s guess is correct in $\mathsf{Game}_{\mathrm{I},2}$. By definition, it holds that $\Pr[\mathsf{F}] = 1/Q_{\mathsf{gen}}$ and $\mathsf{Game}_{\mathrm{I},1}$ and $\mathsf{Game}_{\mathrm{I},2}$ are identically distributed if F happens since all the behavior of $\mathcal{C}$ is the same. Thus, it holds that

$$\Pr[\mathsf{E}_{\mathrm{I},1}] = \Pr[\mathsf{E}_{\mathrm{I},2} \mid \mathsf{F}]. \tag{21}$$

If F does not happen, $\mathcal{C}$ outputs a random bit and aborts the game. Thus, it holds that

$$\Pr[\mathsf{E}_{\mathrm{I},2} \mid \neg\mathsf{F}] = \frac{1}{2}. \tag{22}$$

Observe that

$$\begin{aligned}
\mathsf{Adv}_{\mathrm{I},2}(\lambda) &= \left|\Pr[\mathsf{E}_{\mathrm{I},2}] - \frac{1}{2}\right| \\
&= \left|\Pr[\mathsf{E}_{\mathrm{I},2} \wedge \mathsf{F}] + \Pr[\mathsf{E}_{\mathrm{I},2} \wedge \neg\mathsf{F}] - \frac{1}{2}\right| \\
&= \left|\Pr[\mathsf{E}_{\mathrm{I},2} \mid \mathsf{F}] \cdot \Pr[\mathsf{F}] + \Pr[\mathsf{E}_{\mathrm{I},2} \mid \neg\mathsf{F}] \cdot \Pr[\neg\mathsf{F}] - \frac{1}{2}\right|.
\end{aligned}$$

From the equation (21) and (22), we have

$$\begin{aligned}
\mathsf{Adv}_{\mathrm{I},2}(\lambda) &= \left|\Pr[\mathsf{E}_{\mathrm{I},1}] \cdot \Pr[\mathsf{F}] - \frac{1}{2}(1 - \Pr[\neg\mathsf{F}])\right| \\
&= \left|\Pr[\mathsf{E}_{\mathrm{I},1}] \cdot \Pr[\mathsf{F}] - \frac{1}{2} \cdot \Pr[\mathsf{F}]\right| \\
&= \frac{1}{Q_{\mathsf{gen}}}\left|\Pr[\mathsf{E}_{\mathrm{I},1}] - \frac{1}{2}\right| \\
&= \frac{1}{Q_{\mathsf{gen}}} \cdot \mathsf{Adv}_{\mathrm{I},1}(\lambda).
\end{aligned}$$

Thus, we complete the proof. □

Lemma 20 ($\mathsf{Game}_{\mathrm{I},2} \equiv \mathsf{Game}_{\mathrm{I},3}$)**.** $\mathsf{Game}_{\mathrm{I},2}$ *and* $\mathsf{Game}_{\mathrm{I},3}$ *are identically distributed from* $\mathcal{A}$*'s view. Specifically, for any PPT Type-*I *adversary* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}_{\mathrm{I},2}(\lambda) = \mathsf{Adv}_{\mathrm{I},3}(\lambda).$$

Proof of Lemma 20. We describe how $\mathcal{C}$ creates $\mathsf{sk}_{\mathtt{ID}}$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ in $\mathsf{Game}_{\mathrm{I},3}$. $\mathcal{C}$ creates MPK, $\mathsf{sk}_{\mathtt{ID}}$, and $\mathsf{ct}^\star$ in the same way as $\mathsf{Game}_{\mathrm{II},2}$.

Key Update and Decryption Key Creation for $\mathtt{T} < \mathtt{T}_{\mathsf{RL}}$: For each time period $\mathtt{T}$ upon the setup and $\mathcal{A}$'s revoke & key update queries, $\mathcal{C}$ samples $\mathbf{t}_{\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ and creates a *seed* key update $\mathsf{s.ku}_{\mathtt{T}} = (\mathsf{s.KU}_{\mathtt{T},0}, \mathsf{s.KU}_{\mathsf{KU},1}, \mathsf{s.KU}_{\mathsf{KU},2})$ by computing (19).

For each $\theta \in \mathcal{KUN}_{\mathsf{kgc},\mathtt{T}}$, $\mathcal{C}$ retrieves a delegation key $\mathbf{k}_{\mathsf{kgc},\theta}$, samples $\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$ and $\widetilde{\mathbf{t}}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$, and computes a sub-key update $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta} := (\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,0}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,2})$:

$$\begin{aligned}
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,0} &= (\mathsf{s.KU}_{\mathtt{T},0})^{\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta}} \cdot [\mathbf{Z}\widetilde{\mathbf{t}}_{\mathsf{kgc},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} &= [\mathbf{k} - \mathbf{k}_{\mathsf{kgc},\theta}]_2 \cdot (\mathsf{s.KU}_{\mathtt{T},1})^{\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta}} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{t}}_{\mathsf{kgc},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,2} &= (\mathsf{s.KU}_{\mathtt{T},2})^{\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta}} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{t}}_{\mathsf{kgc},\mathtt{T},\theta}]_2.
\end{aligned} \tag{23}$$

This is the *normal* sub-key update by setting $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta} = \tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} \cdot \mathbf{t}_{\mathtt{T}} + \widetilde{\mathbf{t}}_{\mathsf{kgc},\mathtt{T},\theta}$. Due to the fresh random $\widetilde{\mathbf{t}}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$, $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}$ is distributed in $\mathbb{Z}_p^k$ uniformly at random.

For each $\theta \in \mathcal{KUN}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$, $\mathcal{C}$ retrieves a delegation key $\mathbf{k}_{\mathtt{ID},\theta}$, samples the ephemeral delegation key $\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$, $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$, and $\widetilde{\mathbf{t}}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$, and computes a sub-key update $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta} := (\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,0}, \mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,1}, \mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,2})$:

$$\begin{aligned}
\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,0} &= (\mathsf{s.KU}_{\mathtt{T},0})^{\tilde{t}_{\mathtt{ID},\mathtt{T},\theta}} \cdot [\mathbf{Z}\widetilde{\mathbf{t}}_{\mathtt{ID},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,1} &= [\mathbf{k}_{\mathtt{ID},\theta} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2^{-1} \cdot (\mathsf{s.KU}_{\mathtt{T},1})^{\tilde{t}_{\mathtt{ID},\mathtt{T},\theta}} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{t}}_{\mathtt{ID},\mathtt{T},\theta}]_2, \\
\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,2} &= (\mathsf{s.KU}_{\mathtt{T},2})^{\tilde{t}_{\mathtt{ID},\mathtt{T},\theta}} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{t}}_{\mathtt{ID},\mathtt{T},\theta}]_2.
\end{aligned} \tag{24}$$

This is the *normal* sub-key update by setting $\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta} = \tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \cdot \mathbf{t}_{\mathtt{T}} + \widetilde{\mathbf{t}}_{\mathtt{ID},\mathtt{T},\theta}$. Due to the fresh random $\widetilde{\mathbf{t}}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$, $\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta}$ is distributed in $\mathbb{Z}_p^k$ uniformly at random. $\mathcal{C}$ creates the helper key update $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ in the same way as the real scheme.

$\mathcal{C}$ retrieves the master secret key $\mathbf{k}$, samples $\tilde{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p$ and $\mathbf{u}_{\mathtt{ID},\mathtt{T}}, \widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, and computes $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})$:

$$\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},0} &= [\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0} = \mathsf{s.KU}_{\mathtt{T},0}^{\tilde{u}'_{\mathtt{ID},\mathtt{T}}} \cdot [\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2, \\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &= [\mathbf{k}]_2 \cdot \mathsf{s.KU}_{\mathtt{T},1}^{\tilde{u}'_{\mathtt{ID},\mathtt{T}}} \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2 \\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2 \\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},2} &= [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2} = \mathsf{s.KU}_{\mathtt{T},2}^{\tilde{u}'_{\mathtt{ID},\mathtt{T}}} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}]_2.
\end{aligned} \tag{25}$$

This is the *normal* decryption key by setting $\mathbf{u}'_{\mathtt{ID},\mathtt{T}} = \tilde{u}'_{\mathtt{ID},\mathtt{T}} \cdot \mathbf{t}_{\mathtt{T}} + \widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}}$. Due to the fresh random $\widetilde{\mathbf{u}}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, $\mathbf{u}'_{\mathtt{ID},\mathtt{T}}$ is distributed in $\mathbb{Z}_p^k$ uniformly at random.

Key Update and Decryption Key Creation for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$: Upon $\mathcal{A}$'s secret key *generation* query on $\mathtt{ID}$, $\mathcal{C}$ samples $\mathbf{r}_{\mathtt{ID}}^{(2)} \leftarrow_R \mathbb{Z}_p^k$ and creates a *seed* secret key $\mathsf{s.sk}_{\mathtt{ID}}^{(2)} = (\mathsf{s.SK}_{\mathtt{ID},0}^{(2)}, \mathsf{s.SK}_{\mathtt{ID},1}^{(2)}, \mathsf{s.SK}_{\mathtt{ID},2}^{(2)}, (\mathsf{s.}\widetilde{\mathsf{SK}}_{\mathtt{ID},\ell}^{(2)})_{\ell \in [|\mathtt{ID}|+1, L]})$ by computing (8). $\mathcal{C}$ creates $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ in the same way as the real scheme.

To create $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$, $\mathcal{C}$ samples the ephemeral delegation key $\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$ and creates the sub-key update $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ in the same way as the real scheme. Then, $\mathcal{C}$ retrieves the

delegation key $\mathbf{k}_{\mathtt{ID},\theta}$ and ephemeral delegation key $\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}$, samples $\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}, \overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, and computes a helper key update $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}} = (\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2}, \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2}, (\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$:

$$\begin{aligned}
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0} &= \mathsf{s.SK}^{(2)}_{\mathtt{ID},0} \cdot [\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},0} = [\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1} &= [\mathbf{k} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot \mathsf{s.SK}^{(2)}_{\mathtt{ID},1} \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|})\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2\\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2} &= \mathsf{s.SK}^{(2)}_{\mathtt{ID},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \overline{\mathsf{KU}}'_{\mathtt{ID},\mathtt{T},2} = [\mathbf{V}_{L+2}\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\widetilde{\overline{\mathsf{KU}}}_{\mathtt{ID},\mathtt{T},\ell} &= \mathsf{s.}\overline{\mathsf{SK}}^{(2)}_{\mathtt{ID},\ell} \cdot [\mathbf{V}_\ell\mathbf{Z}\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}]_2.
\end{aligned} \tag{26}$$

This is the *normal* helper key update as in $\mathsf{Game}_{\mathrm{I},2}$ by setting $\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}} = \mathbf{r}^{(2)}_{\mathtt{ID}} + \widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}}$. Due to the fresh random $\widetilde{\overline{\mathbf{t}}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, $\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}$ is distributed in $\mathbb{Z}_p^k$ uniformly at random.

$\mathcal{C}$ retrieves the master secret key $\mathbf{k}$, samples $\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}, \mathbf{u}'_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ and computes $\mathsf{dk}_{\mathtt{ID},\mathtt{T}} = (\mathsf{DK}_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},1}, \mathsf{DK}_{\mathtt{ID},\mathtt{T},2}, \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2})$:

$$\begin{aligned}
\mathsf{DK}_{\mathtt{ID},\mathtt{T},0} &= \mathsf{s.SK}^{(2)}_{\mathtt{ID},0} \cdot [\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},0} = [\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2,\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},1} &= [\mathbf{k}]_2 \cdot \mathsf{s.SK}^{(2)}_{\mathtt{ID},1} \cdot [(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|})\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2\\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2\\
\mathsf{DK}_{\mathtt{ID},\mathtt{T},2} &= \mathsf{s.SK}^{(2)}_{\mathtt{ID},2} \cdot [\mathbf{V}_{L+2}\mathbf{Z}\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}]_2, \qquad \mathsf{DK}'_{\mathtt{ID},\mathtt{T},2} = [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{u}'_{\mathtt{ID},\mathtt{T}}]_2.
\end{aligned} \tag{27}$$

This is the *normal* decryption key by setting $\mathbf{u}_{\mathtt{ID},\mathtt{T}} = \mathbf{r}_{\mathtt{ID}} + \widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}}$. Due to the fresh random $\widetilde{\mathbf{u}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, $\mathbf{u}_{\mathtt{ID},\mathtt{T}}$ is distributed in $\mathbb{Z}_p^k$ uniformly at random.

As we observed so far, all the elements distribute in the same way as in $\mathsf{Game}_{\mathrm{I},2}$. Thus, we complete the proof of Lemma 20. □

Lemma 21 ($\mathsf{Game}_{\mathrm{I},3} \approx_c \mathsf{Game}_{\mathrm{I},4}$)**.** $\mathsf{Game}_{\mathrm{I},3}$ *and* $\mathsf{Game}_{\mathrm{I},4}$ *are computationally indistinguishable under the* MDDH *assumption in* $\mathbb{G}_2$*. Specifically, for any PPT Type-I adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key* generation *queries and* Q_{rev} *secret key* reveal *queries, there exist reduction algorithms* $\mathcal{B}_{\mathrm{I},1}$ *and* $\mathcal{B}_{\mathrm{I},2}$ *such that*

$$|\mathsf{Adv}_{\mathrm{I},3}(\lambda) - \mathsf{Adv}_{\mathrm{I},4}(\lambda)| \le Q_{\mathsf{rev}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},j}}(\lambda) + \frac{4Q_{\mathsf{rev}}}{p-1}$$

and $\max_{j\in[2]} \mathtt{T}(\mathcal{B}_{\mathrm{I},j}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

We omit the proof of Lemma 21 since it is essentially the same as the proof of Lemma 5. The only essential difference is that $\mathsf{s.sk}_{\mathtt{ID}}$ such that $\mathtt{ID}_{Q^\star} \in \mathsf{prefix}^+(\mathtt{ID})$ are always normal by computing (8). Since $\mathcal{A}$ makes secret key reveal queries on $\mathtt{ID}$ such that $\mathtt{ID}_{Q^\star} \in \mathsf{prefix}^+(\mathtt{ID})$ only after $\mathcal{A}$'s secret key generation query on $\mathtt{ID}_{Q^\star}$, the reduction algorithm can detect whether $\mathtt{ID}_{Q^\star} \in \mathsf{prefix}^+(\mathtt{ID})$ holds.

Lemma 22 (Key Update and Decryption Key Invariance for $\mathtt{T} < \mathtt{T}_{\mathsf{RL}}$, $\mathsf{Game}_{\mathrm{I},4} \approx_c \mathsf{Game}_{\mathrm{I},5}$)**.** $\mathsf{Game}_{\mathrm{I},4}$ *and* $\mathsf{Game}_{\mathrm{I},5}$ *are computationally indistinguishable under the* MDDH *assumption in* $\mathbb{G}_2$*. Specifically, for any PPT Type-I adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key* generation *queries, there exist reduction algorithms* $\mathcal{B}_{\mathrm{I},3}$ *and* $\mathcal{B}_{\mathrm{I},4}$ *such that*

$$|\mathsf{Adv}_{\mathrm{I},4}(\lambda) - \mathsf{Adv}_{\mathrm{I},5}(\lambda)| \le \mathtt{T}_{\mathsf{RL}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},j+2}}(\lambda) + \frac{4\mathtt{T}_{\mathsf{RL}}}{p-1}$$

and $\max_{j\in[2]} \mathtt{T}(\mathcal{B}_{\mathrm{I},j+2}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

Proof of Lemma 20. To prove Lemma 20, we further introduce the following auxiliary distributions.

Pseudo-normal Seed Key Updates: A *pseudo-normal* seed key update is defined as $\mathsf{s.ku}_{\mathtt{T}} := (\mathsf{s.KU}_{\mathtt{T},0}, \mathsf{s.KU}_{\mathtt{T},1}, \mathsf{s.KU}_{\mathtt{T},2})$:

$$\mathsf{s.KU}_{\mathtt{T},0} := [\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2, \qquad \mathsf{s.KU}_{\mathtt{T},1} := [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2 \cdot \boxed{[\hat{t}\mathbf{a}^{\perp}]^{v_0 + v_{L+1}\mathtt{T}}},$$
$$\mathsf{s.KU}_{\mathtt{T},2} := [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2 \cdot \boxed{[\hat{t}\mathbf{a}^{\perp}]^{-1}},$$

where $\mathbf{t}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, $\hat{t} \leftarrow_R \mathbb{Z}_p^*$, and $(v_0, v_{L+1}) \leftarrow_R \mathbb{Z}_p^2$ is the randomness for creating the challenge ciphertext. Here, the boxed parts denote the changes from the *normal* seed key update.

Pseudo-SF Seed Key Updates: A *pseudo-SF* seed key update is defined as $\mathsf{s.ku}_{\mathtt{T}} := (\mathsf{s.KU}_{\mathtt{T},0}, \mathsf{s.KU}_{\mathtt{T},1}, \mathsf{s.KU}_{\mathtt{T},2})$:

$$\mathsf{s.KU}_{\mathtt{T},0} := [\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2, \quad \mathsf{s.KU}_{\mathtt{T},1} := \boxed{[\alpha\mathbf{a}^{\perp}]_2} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2 \cdot [\hat{t}\mathbf{a}^{\perp}]^{v_0 + v_{L+1}\mathtt{T}},$$
$$\mathsf{s.KU}_{\mathtt{T},2} := [\mathbf{V}_{L+2}\mathbf{Z}\mathbf{t}_{\mathtt{T}}]_2 \cdot [\hat{t}\mathbf{a}^{\perp}]^{-1},$$

where $\mathbf{t}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$, $\hat{t} \leftarrow_R \mathbb{Z}_p^*$, $(v_0, v_{L+1}) \leftarrow_R \mathbb{Z}_p^2$ is the randomness for creating the challenge ciphertext, and $\alpha \leftarrow_R \mathbb{Z}_p^*$ is the semi-functional randomness shared by all $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ and $\mathsf{s.ku}_{\mathtt{T}}$. Here, the boxed part denotes the change from the *pseudo-normal* seed key update.

We further introduce the following sequence of games for $T \in [0, \mathtt{T}_{\mathsf{RL}} - 1]$:

$\mathsf{Game}_{\mathrm{I},4,T,1}$: This game is the same as $\mathsf{Game}_{\mathrm{I},3}$ except that
- If $\mathtt{T} < T$, $\mathcal{C}$ creates *semi-functional* $\mathsf{s.ku}_{\mathtt{T}}$ upon $\mathcal{A}$'s secret key *generation* queries,
- If $\mathtt{T} = T$, $\mathcal{C}$ creates *pseudo-normal* $\mathsf{s.ku}_{\mathtt{T}}$ upon $\mathcal{A}$'s secret key *generation* queries,
- If $\mathtt{T} > T$, $\mathcal{C}$ always creates *normal* $\mathsf{s.ku}_{\mathtt{T}}$ upon $\mathcal{A}$'s secret key *generation* queries, secret key *generation* queries.

$\mathsf{Game}_{\mathrm{I},4,T,2}$: This game is the same as $\mathsf{Game}_{\mathrm{I},4,T,1}$ except that
- If $\mathtt{T} = T$, $\mathcal{C}$ creates *pseudo-SF* $\mathsf{s.ku}_{\mathtt{T}}$ upon $\mathcal{A}$'s secret key *generation* queries,

$\mathsf{Game}_{\mathrm{I},4,T,3}$: This game is the same as $\mathsf{Game}_{\mathrm{I},4,T,2}$ except that
- If $\mathtt{T} = T$, $\mathcal{C}$ creates *semi-functional* $\mathsf{s.ku}_{\mathtt{T}}$ upon $\mathcal{A}$'s secret key *generation* queries,

By definition, $\mathsf{Game}_{\mathrm{I},4,0,3} = \mathsf{Game}_{\mathrm{I},4}$. Hereafter, we prove

$$\mathsf{Game}_{\mathrm{I},4,T-1,3} \approx_c \mathsf{Game}_{\mathrm{I},4,T,1} \equiv \mathsf{Game}_{\mathrm{I},4,T,2} \approx_c \mathsf{Game}_{\mathrm{I},4,T,3},$$

where the fact implies that $\mathsf{Game}_{\mathrm{I},4} \approx_c \mathsf{Game}_{\mathrm{I},4,\mathtt{T}_{\mathsf{RL}}-1,3}$. We note that $\mathsf{Game}_{\mathrm{I},4,\mathtt{T}_{\mathsf{RL}}-1,3} \equiv \mathsf{Game}_{\mathrm{I},5}$ will be proved later.

Lemma 23 (Seed Key Updates Transition from Normal to Pseudo-normal, $\mathsf{Game}_{\mathrm{I},4,T-1,3} \approx_c \mathsf{Game}_{\mathrm{I},4,T,1}$)**.** $\mathsf{Game}_{\mathrm{I},4,T-1,3}$ *and* $\mathsf{Game}_{\mathrm{I},4,T,1}$ *are computationally indistinguishable under the* MDDH *assumption in* $\mathbb{G}_2$*. Specifically, for any PPT Type-*I *adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key* generation *queries, there exists a reduction algorithm* $\mathcal{B}_{\mathrm{I},3}$ *such that*

$$|\mathsf{Adv}_{\mathrm{I},4,T-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{I},4,T,1}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},3}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{I},3}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

Proof of Lemma 23. The reduction algorithm $\mathcal{B}_{\mathrm{I},3}$ is given a MDDH instance in $\mathbb{G}_2$: $(\mathcal{G}(1^\lambda), [\mathbf{B}]_2, [\mathbf{b}]_2 = [\mathbf{Bt} + \hat{t}\mathbf{e}]_2)$, where $\mathbf{B} \leftarrow_R \mathcal{D}_k, \mathbf{t} \leftarrow_R \mathbb{Z}_p^k$, $\hat{t} = 0$ or $\hat{t} \leftarrow_R \mathbb{Z}_p$, and $\mathbf{e} = (0, \ldots, 0, 1)^\top \in \mathbb{Z}_p^{k+1}$. Hereafter, we assume that $\hat{t} \leftarrow_R \mathbb{Z}_p^*$ in the latter case with the statistical difference $1/p$.

$\mathcal{B}_{\mathrm{I},3}$ creates MPK and $\mathsf{ct}^\star$ in the same way as the proof of Lemma 6. $\mathcal{B}_{\mathrm{I},3}$ creates *semi-functional* $\mathsf{s.sk}_{\mathtt{ID}}^{(1)}$ by computing (9) and creates $\mathsf{sk}_{\mathtt{ID}}$ by computing (10). $\mathcal{B}_{\mathrm{I},3}$ creates $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ in the same way as the proof of Lemma 20.

After creating $\mathsf{s.ku}_{\mathtt{T}}$, $\mathcal{B}_{\mathrm{I},3}$ creates $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ in the same way as the proof of Lemma 20. We describe how $\mathcal{B}_{\mathrm{I},3}$ creates $\mathsf{s.ku}_{\mathtt{T}} = (\mathsf{s.KU}_{\mathtt{T},0}, \mathsf{s.KU}_{\mathtt{T},1}, \mathsf{s.KU}_{\mathtt{T},2})$.

- If $\mathtt{T} < T$, $\mathcal{B}_{\mathrm{I},3}$ creates *semi-functional* $\mathsf{s.ku}_{\mathtt{T}}$ by computing (20).
- If $\mathtt{T} = T$, $\mathcal{B}_{\mathrm{I},3}$ retrieves (v_0, v_{L+1}) and computes $\mathsf{s.ku}_{\mathtt{T}} = (\mathsf{s.KU}_{\mathtt{T},0}, \mathsf{s.KU}_{\mathtt{T},1}, \mathsf{s.KU}_{\mathtt{T},2})$:

$$\begin{aligned} \mathsf{s.KU}_{\mathtt{T},0} &= [\overline{\mathbf{b}}]_2, \quad \mathsf{s.KU}_{\mathtt{T},1} = [(\widetilde{\mathbf{V}}_0 + \mathtt{T}\widetilde{\mathbf{V}}_{L+1})\overline{\mathbf{b}}]_2 \cdot [\mathbf{a}^\perp \underline{\mathbf{b}}]^{v_0 + v_{L+1}\mathtt{T}}, \\ \mathsf{s.KU}_{\mathtt{T},2} &= [\widetilde{\mathbf{V}}_{L+2}\overline{\mathbf{b}}]_2 \cdot [-\mathbf{a}^\perp \underline{\mathbf{b}}]. \end{aligned} \tag{28}$$

  By following the same argument in the proof of Lemma 6, $\mathsf{s.ku}_{\mathtt{T}}$ is a *normal* seed key update as in $\mathsf{Game}_{\mathrm{I},4,T-1,3}$ if $\hat{t} = 0$, and *pseudo-normal* seed key update as in $\mathsf{Game}_{\mathrm{I},4,T,1}$ if $\hat{t} \leftarrow_R \mathbb{Z}_p^*$, by setting $\mathbf{t}_{\mathtt{T}} = \widetilde{\mathbf{Z}}^{-1}\mathbf{t}$.
- If $\mathtt{T} > T$, $\mathcal{B}_{\mathrm{I},3}$ creates *normal* $\mathsf{s.ku}_{\mathtt{T}}$ by computing (19).

Thus, we complete the proof of Lemma 23. □

Lemma 24 (Seed Key Update Transition from Pseudo-normal to Pseudo-SF, $\mathsf{Game}_{\mathrm{I},4,T,1} \equiv \mathsf{Game}_{\mathrm{I},4,T,2}$)**.** *$\mathsf{Game}_{\mathrm{I},4,T,1}$ and $\mathsf{Game}_{\mathrm{I},4,T,2}$ are identically distributed from $\mathcal{A}$'s view. Specifically, for any Type-I adversary $\mathcal{A}$, it holds that*

$$\mathsf{Adv}_{\mathrm{I},4,T,1}(\lambda) = \mathsf{Adv}_{\mathrm{I},4,T,2}(\lambda).$$

We can prove that $\mathsf{s.ku}_{\mathtt{T}}$ follows the same distribution in $\mathsf{Game}_{\mathrm{I},4,T,1}$ and $\mathsf{Game}_{\mathrm{I},4,T,2}$ by following the same argument as in the proof of Lemma 7 based on the fact that $\mathtt{T} \neq \mathtt{T}^\star$ for all $\mathtt{T} < \mathtt{T}_{\mathsf{RL}}$.

Lemma 25 (Seed Key Update Transition from Pseudo-SF to Semi-functional, $\mathsf{Game}_{\mathrm{I},4,T,2} \approx_c \mathsf{Game}_{\mathrm{I},4,T,3}$)**.** *$\mathsf{Game}_{\mathrm{I},4,T,2}$ and $\mathsf{Game}_{\mathrm{I},4,T,3}$ are computationally indistinguishable under the MDDH assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-I adversary $\mathcal{A}$ making at most Q_{gen} secret key generation queries, there exists a reduction algorithm $\mathcal{B}_{\mathrm{I},4}$ such that*

$$|\mathsf{Adv}_{\mathrm{I},4,T,2}(\lambda) - \mathsf{Adv}_{\mathrm{I},4,T,3}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},4}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{I},4}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

We omit the detailed proof of Lemma 25 since it is almost the same as the proof of Lemma 23. The only difference is that $\mathcal{B}_{\mathrm{I},4}$ creates T-th $\mathsf{s.ku}_{\mathtt{T}}$ by computing (28) except that

$$\mathsf{s.KU}_{\mathtt{T},1} = \boxed{[\alpha \mathbf{a}^\perp]} \cdot [(\widetilde{\mathbf{V}}_0 + \mathtt{T}\widetilde{\mathbf{V}}_{L+1})\overline{\mathbf{b}}]_2 \cdot [\mathbf{a}^\perp \underline{\mathbf{b}}]^{v_0 + v_{L+1}\mathtt{T}},$$

where the boxed parts denote the changes from (28). If $\hat{t} \leftarrow_R \mathbb{Z}_p^*$, $\mathsf{s.ku}_{\mathtt{T}}$ is a *pseudo-SF* seed key update as in $\mathsf{Game}_{\mathrm{I},4,T,2}$. If $\hat{t} = 0$, $\mathsf{s.ku}_{\mathtt{T}}$ is a *semi-functional* seed key update as in $\mathsf{Game}_{\mathrm{I},4,T,3}$.

Lemma 26 (Semi-functional Randomness Switching for Helper Key Updates for $\mathtt{T} < \mathsf{T_{RL}}$, $\mathsf{Game}_{\mathrm{I},4,\mathsf{T_{RL}}-1,3} = \mathsf{Game}_{\mathrm{I},5}$)**.** $\mathsf{Game}_{\mathrm{I},4,\mathsf{T_{RL}}-1,3}$ *and* $\mathsf{Game}_{\mathrm{I},5}$ *are identically distributed from* $\mathcal{A}$*'s view. Specifically, for any Type-*I *adversary* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}_{\mathrm{I},4,\mathsf{T_{RL}}-1,3}(\lambda) = \mathsf{Adv}_{\mathrm{I},5}(\lambda).$$

The proof is the first core part of the proof against the Type-I adversary. In $\mathsf{Game}_{\mathrm{I},4,\mathsf{T_{RL}}-1,3}$, all the seed key updates $\mathsf{s.ku}_{\mathtt{T}}$ for $\mathtt{T} < \mathsf{T_{RL}}$ become semi-functional. Although we use $\mathsf{s.ku}_{\mathtt{T}}$ to create $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$, we defined $\mathsf{Game}_{\mathrm{I},5}$ so that all $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ for $\mathtt{T} < \mathsf{T_{RL}}$ to be semi-functional. For this purpose, we apply the semi-functional randomness switching to show that $\mathsf{Game}_{\mathrm{I},4,\mathsf{T_{RL}}-1,3} \equiv \mathsf{Game}_{\mathrm{I},5}$.

Proof of Lemma 26. Here, we prove a stronger claim that $\mathsf{Game}_{\mathrm{I},4,\mathsf{T_{RL}}-1,3}$ and $\mathsf{Game}_{\mathrm{I},5}$ are identically distributed from $\mathcal{A}$'s view for any fixed

- $(\mathbf{A}, \mathbf{a}) \leftarrow_R \mathcal{D}_k$,
- $\big((\mathbf{V}_\ell)_{\ell\in[0,L+2]}, \mathbf{Z}\big) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$,
- master secret key $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$,
- $\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ for creating $\mathsf{ku}_{\mathtt{ID},\mathtt{T},\theta}$,
- $\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ for creating $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$,
- $\alpha \leftarrow_R \mathbb{Z}_p^*$.

Specifically, the randomnesses of $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$ in (24) and $\overline{\mathsf{delk}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$ enable us to prove the claim. Note that even when $\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ are fixed, $(\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,0}, \mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,2})$ do not reveal the quantities of $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$ since they are masked by $\widetilde{\mathbf{t}}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$. Since $\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ and $\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ are fixed, $(\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,0}, \mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,2})$ and $(\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},0}, \overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},2})$ follow the same distribution in both $\mathsf{Game}_{\mathrm{I},4,\mathsf{T_{RL}}-1,3}$ and $\mathsf{Game}_{\mathrm{I},5}$. In $\mathsf{Game}_{\mathrm{I},4,\mathsf{T_{RL}}-1,3}$, $\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,1}$ and $\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1}$ for $\mathtt{T} < \mathsf{T_{RL}}$ such that $|\mathtt{ID}| \geq 1$ are distributed as follows:

$$\begin{aligned}
\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,1} &= [\mathbf{k}_{\mathtt{ID},\theta} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} - \tilde{t}_{\mathtt{ID},\mathtt{T},\theta}\alpha\mathbf{a}^\perp]_2^{-1} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta}]_2,\\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1} &= [\mathbf{k} + \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2\\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2,
\end{aligned}$$

where $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$ and $\overline{\mathsf{delk}}_{\mathtt{ID},\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$. In contrast, the above distribution can be written as follows:

$$\begin{aligned}
\mathsf{KU}_{\mathtt{ID},\mathtt{T},\theta,1} &= [\mathbf{k}_{\mathtt{ID},\theta} + (\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} - \alpha\mathbf{a}^\perp) - (\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} - 1)\alpha\mathbf{a}^\perp]_2^{-1}\\
&\qquad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{ID},\mathtt{T},\theta}]_2,\\
\overline{\mathsf{KU}}_{\mathtt{ID},\mathtt{T},1} &= [(\mathbf{k} + \alpha\mathbf{a}^\perp) + (\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} - \alpha\mathbf{a}^\perp)]_2\\
&\qquad \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID},\mathtt{T}}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID},\mathtt{T}}]_2,
\end{aligned}$$

where $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} - 1$ is distributed in $\mathbb{Z}_p$ uniformly at random and $\overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} - \alpha\mathbf{a}^\perp$ is distributed in $\mathbb{Z}_p^{k+1}$ uniformly at random. Therefore, the above distribution is the same as the distribution in $\mathsf{Game}_{\mathrm{I},5}$ by setting $\tilde{t}_{\mathtt{ID},\mathtt{T},\theta} - 1$ as the randomnesses in (10) and $\overline{\mathsf{delk}}_{\mathtt{ID},\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID},\mathtt{T}} + \alpha\mathbf{a}^\perp$. We note that the claim holds for all $\mathtt{ID}$ such that $|\mathtt{ID}| \geq 1$, all $\mathtt{T} < \mathsf{T_{RL}}$, and all nodes $\theta \in \mathtt{BT}_{\mathtt{ID}}$, simultaneously. Thus, we complete the proof of Lemma 26. □

By combining Lemmata 23–26, we have

$$|\mathsf{Adv}_{\mathrm{I},4}(\lambda) - \mathsf{Adv}_{\mathrm{I},5}(\lambda)|$$

$$
\begin{aligned}
&\leq \sum_{T\in[\mathsf{T}_{\mathsf{RL}}]} |\mathsf{Adv}_{\mathrm{I},4,T-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{I},4,T,1}(\lambda)| + \sum_{T\in[\mathsf{T}_{\mathsf{RL}}]} |\mathsf{Adv}_{\mathrm{I},4,T,1}(\lambda) - \mathsf{Adv}_{\mathrm{I},4,T,2}(\lambda)| \\
&\quad + \sum_{T\in[\mathsf{T}_{\mathsf{RL}}]} |\mathsf{Adv}_{\mathrm{I},4,T,2}(\lambda) - \mathsf{Adv}_{\mathrm{I},4,T,3}(\lambda)| + |\mathsf{Adv}_{\mathrm{I},4,\mathsf{T}_{\mathsf{RL}}-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{I},5}(\lambda)| \\
&\leq \mathsf{T}_{\mathsf{RL}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},j+2}}(\lambda) + \frac{4\mathsf{T}_{\mathsf{RL}}}{p-1}.
\end{aligned}
$$

Thus, we complete the proof of Lemma 22. □

Lemma 27 (Semi-functional Randomness Switching for KGC's Key Updates for $\mathtt{T} \geq \mathsf{T}_{\mathsf{RL}}$, $\mathsf{Game}_{\mathrm{I},5} \equiv \mathsf{Game}_{\mathrm{I},6}$)**.** $\mathsf{Game}_{\mathrm{I},5}$ *and* $\mathsf{Game}_{\mathrm{I},6}$ *are identically distributed from* $\mathcal{A}$*'s view. Specifically, for any Type-*I *adversary* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}_{\mathrm{I},5}(\lambda) = \mathsf{Adv}_{\mathrm{I},6}(\lambda).$$

The proof is the second core part of the proof against the Type-I adversary since we have to change $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}^\star}$ to be semi-functional. We want to discuss the difference from the proof of Lemma 9. In the proof of Lemma 9, all $\mathsf{sk}_{\mathtt{ID}}$ such that $|\mathtt{ID}| = 1$ which $\mathcal{A}$ receives via secret key *reveal* queries are *semi-functional*. In contrast, in the proof of Lemma 27, $\mathcal{A}$ may receive $\mathsf{sk}_{\mathtt{ID}^\star_{[\ell^\star]}} = \mathsf{sk}_{\mathtt{ID}^\star_{[1]}}$ when $\ell^\star = 1$. However, once $\mathcal{A}$ receives $\mathsf{sk}_{\mathtt{ID}^\star_{[1]}}$, $\mathtt{ID}^\star_{[1]}$ must be revoked at T_{RL}. In other words, $\mathsf{sk}_{\mathtt{ID}^\star_{[1]},\theta}$ and $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ for $\mathtt{T} \geq \mathsf{T}_{\mathsf{RL}}$ do not share the same nodes $\theta \in \mathsf{BT}_{\mathsf{kgc}}$. The fact is sufficient for proving Lemma 27 by combining with the modifications so far.

Proof of Lemma 27. Here, we prove a stronger claim that $\mathsf{Game}_{\mathrm{I},5}$ and $\mathsf{Game}_{\mathrm{I},6}$ are identically distributed from $\mathcal{A}$'s view for any fixed

- $(\mathbf{A}, \mathbf{a}) \leftarrow_R \mathcal{D}_k$,
- $\big((\mathbf{V}_\ell)_{\ell\in[0,L+2]}, \mathbf{Z}\big) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$,
- master secret key $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$,
- $\mathbf{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$ for creating $\mathsf{sk}_{\mathtt{ID},\theta}$ such that $|\mathtt{ID}| = 1$,
- $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ for creating $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$,
- $\alpha \leftarrow_R \mathbb{Z}_p^*$ that is the semi-functional randomness of $\mathsf{ku}_{\mathsf{kgc},\mathtt{T}}$ in $\mathsf{Game}_{\mathrm{I},6}$.

Specifically, the randomnesses of $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$ in (10), $\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$ in (23), and $\mathsf{delk}_{\mathsf{kgc},\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$ enable us to prove the claim. Note that $\mathsf{sk}_{\mathtt{ID}}$ such that $|\mathtt{ID}| \geq 2$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ such that $|\mathtt{ID}| \geq 1$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ are created in the same way in both $\mathsf{Game}_{\mathrm{I},5}$ and $\mathsf{Game}_{\mathrm{I},6}$. Since $\mathbf{r}_{\mathtt{ID},\theta}$ and $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}$ are fixed, $\mathsf{sk}_{\mathtt{ID},\theta}$ such that $|\mathtt{ID}| = 1$ and $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ are distributed in the same way in both $\mathsf{Game}_{\mathrm{I},5}$ and $\mathsf{Game}_{\mathrm{I},6}$ except $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1}$. Note that even when $\mathbf{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$ and $\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$ are fixed, $(\mathsf{SK}_{\mathtt{ID},\theta,0}, \mathsf{SK}_{\mathtt{ID},\theta,2}, (\widetilde{\mathsf{SK}}_{\mathtt{ID},\theta,\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ and $(\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,0}, \mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,2})$ do not reveal the quantities of $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$ and $\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$ since they are masked by $\widetilde{\mathbf{r}}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$ and $\widetilde{\mathbf{t}}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p^k$, respectively. In $\mathsf{Game}_{\mathrm{I},5}$, for all nodes $\theta \in \mathsf{BT}_{\mathsf{kgc}}$ that correspond to $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ for $\mathtt{T} \geq \mathsf{T}_{\mathsf{RL}}$, $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1}$ are distributed as follows:

$$
\begin{aligned}
\mathsf{SK}_{\mathtt{ID},\theta,1} &= [\mathbf{k}_{\mathsf{kgc},\theta} + \tilde{r}_{\mathtt{ID},\theta}\alpha\mathbf{a}^\perp]_2 \cdot [(\mathbf{V}_0 + \mathtt{ID}\mathbf{V}_1)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} &= [\mathbf{k} - \mathbf{k}_{\mathsf{kgc},\theta} + \tilde{t}_{\mathsf{kgc},\mathtt{T},\theta}\alpha\mathbf{a}^\perp]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2 \quad \text{for } \mathtt{T} < \mathsf{T}_{\mathsf{RL}}, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} &= [\mathbf{k} - \mathbf{k}_{\mathsf{kgc},\theta}]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2 \quad \text{for } \mathtt{T} \geq \mathsf{T}_{\mathsf{RL}},
\end{aligned}
$$

where $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$, $\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$, and $\mathsf{delk}_{\mathsf{kgc},\theta} = \mathbf{k}_{\mathsf{kgc},\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$. As we observed above, the quantities of $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$ and $\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$ are revealed to $\mathcal{A}$ only via $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1}$.

Although $\mathsf{sk}_{\mathtt{ID}^\star_{[1]},\theta}$, which is created by the *normal* $\mathsf{s.sk}^{(1)}_{\mathtt{ID}^\star_{[1]}}$, may be revealed to $\mathcal{A}$, they do not share the same nodes with $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ since $\mathtt{ID}^\star_{[1]}$ is revoked by $\mathtt{T}_{\mathsf{RL}}$ and the property of the CS method ensures the fact. Thus, all $\mathsf{sk}_{\mathtt{ID},\theta}$ that share the same nodes with $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ are created by the *semi-functional* $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$ as we specified above.

In contrast, the above distribution can be written as follows:

$$\begin{aligned}
\mathsf{SK}_{\mathtt{ID},\theta,1} &= [(\mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp) + (\tilde{r}_{\mathtt{ID},\theta} - 1)\alpha\mathbf{a}^\perp]_2 \cdot [(\mathbf{V}_0 + \mathtt{ID}\mathbf{V}_1)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} &= [\mathbf{k} - (\mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp) + (\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} + 1)\alpha\mathbf{a}^\perp]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2 \quad \text{for } \mathtt{T} < \mathtt{T}_{\mathsf{RL}}, \\
\mathsf{KU}_{\mathsf{kgc},\mathtt{T},\theta,1} &= [(\mathbf{k} + \alpha\mathbf{a}^\perp) - (\mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp)]_2 \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathsf{kgc},\mathtt{T},\theta}]_2 \quad \text{for } \mathtt{T} \geq \mathtt{T}_{\mathsf{RL}},
\end{aligned}$$

where $\tilde{r}_{\mathtt{ID},\theta} - 1$ and $\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} + 1$ are distributed in $\mathbb{Z}_p$ uniformly at random and $\mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp$ is distributed in $\mathbb{Z}_p^{k+1}$ uniformly at random. Therefore, the above distribution is the same as the distribution in $\mathsf{Game}_{\mathrm{I},6}$ by setting $\tilde{r}_{\mathtt{ID},\theta} - 1$ and $\tilde{t}_{\mathsf{kgc},\mathtt{T},\theta} + 1$ as the randomnesses in (10) and (23), respectively, and $\mathsf{delk}_{\mathsf{kgc},\theta} = \mathbf{k}_{\mathsf{kgc},\theta} + \alpha\mathbf{a}^\perp$. We note that the claim holds for all $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ and all nodes θ that correspond to $\mathsf{ku}_{\mathsf{kgc},\mathtt{T},\theta}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$, simultaneously. Thus, we complete the proof of Lemma 27. □

Lemma 28 (Key Update and Decryption Key Invariance for $|\mathtt{ID}| \geq 1$ and $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$, $\mathsf{Game}_{\mathrm{I},6} \approx_c \mathsf{Game}_{\mathrm{I},7}$)**.** $\mathsf{Game}_{\mathrm{I},6}$ *and* $\mathsf{Game}_{\mathrm{I},7}$ *are computationally indistinguishable under the* MDDH *assumption in* $\mathbb{G}_2$*. Specifically, for any PPT Type-*I *adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key* generation *queries, there exists reduction algorithms* $\mathcal{B}_{\mathrm{I},5}$ *and* $\mathcal{B}_{\mathrm{I},6}$ *such that*

$$|\mathsf{Adv}_{\mathrm{I},6}(\lambda) - \mathsf{Adv}_{\mathrm{I},7}(\lambda)| \leq Q_{\mathsf{gen}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{II},j+4}}(\lambda) + \frac{4Q_{\mathsf{gen}}}{p-1}$$

and $\max_{j\in[2]} \mathtt{T}(\mathcal{B}_{\mathrm{II},j+4}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

The structure of the proof is the same as the proof of Lemma 5 although the transition from pseudo-normal to pseudo-SF is more technical.

Proof of Lemma 28. Let $\mathtt{ID}_q$ denote an identity on which $\mathcal{A}$ makes q-th secret key *generation* query. We further introduce the following sequence of games for $q \in [0, Q_{\mathsf{gen}}]$:

$\mathsf{Game}_{\mathrm{I},6,q,1}$: This game is the same as $\mathsf{Game}_{\mathrm{I},6}$ except that

- If $m < q$, $\mathcal{C}$ creates *semi-functional* $\overline{\mathsf{ku}}_{\mathtt{ID}_m,\mathtt{T}}$ and $\mathsf{dk}_{\mathtt{ID}_m,\mathtt{T}}$,
- If $m = q$, $\mathcal{C}$ creates *pseudo-normal* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ upon $\mathcal{A}$'s q-th secret key *generation* query,
- If $m > q$, $\mathcal{C}$ creates *normal* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_m}$ upon $\mathcal{A}$'s last $Q_{\mathsf{gen}} - q$ secret key *generation* queries.

$\mathsf{Game}_{\mathrm{I},6,q,2}$: This game is the same as $\mathsf{Game}_{\mathrm{I},6,q,1}$ except that

- If $m = q$, $\mathcal{C}$ creates *pseudo-SF* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ upon $\mathcal{A}$'s q-th secret key *generation* query,

$\mathsf{Game}_{\mathrm{I},6,q,3}$: This game is the same as $\mathsf{Game}_{\mathrm{I},6,q,2}$ except that

- If $m = q$, $\mathcal{C}$ creates *semi-functional* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ upon $\mathcal{A}$'s q-th secret key *generation* query,

By definition, $\mathsf{Game}_{\mathrm{I},6,0,3} = \mathsf{Game}_{\mathrm{I},6}$ and $\mathsf{Game}_{\mathrm{I},6,Q_{\mathsf{gen}},3} = \mathsf{Game}_{\mathrm{I},7}$. Hereafter, we prove

$$\mathsf{Game}_{\mathrm{I},6,q-1,3} \approx_c \mathsf{Game}_{\mathrm{I},6,q,1} \equiv \mathsf{Game}_{\mathrm{I},6,q,2} \approx_c \mathsf{Game}_{\mathrm{I},6,q,3},$$

where the fact implies that $\mathsf{Game}_{\mathrm{I},6} \approx_c \mathsf{Game}_{\mathrm{I},7}$.

Lemma 29 (Sub-secret Key Transition from Normal to Pseudo-normal, $\mathsf{Game}_{\mathrm{I},6,q-1,3} \approx_c \mathsf{Game}_{\mathrm{I},6,q,1}$). *$\mathsf{Game}_{\mathrm{I},6,q-1,3}$ and $\mathsf{Game}_{\mathrm{I},6,q,1}$ are computationally indistinguishable under the* MDDH *assumption in $\mathbb{G}_2$. Specifically, for any PPT Type-*I *adversary $\mathcal{A}$ making at most Q_{gen} secret key generation queries, there exists a reduction algorithm $\mathcal{B}_{\mathrm{I},5}$ such that*

$$|\mathsf{Adv}_{\mathrm{I},6,q-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{I},6,q,1}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},5}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{I},5}) \approx \mathtt{T}(\mathcal{A}) + k^2 Q_{\mathsf{gen}} |\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$, where $\mathsf{poly}(\lambda, L)$ is independent of $\mathtt{T}(\mathcal{A})$.

We omit the proof since it is almost the same as the proof of Lemma 6.

Lemma 30 (Sub-secret Key Transition from Pseudo-normal to Pseudo-SF, $\mathsf{Game}_{\mathrm{I},6,q,1} \equiv \mathsf{Game}_{\mathrm{I},6,q,2}$). *$\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$ are identically distributed from $\mathcal{A}$'s view. Specifically, for any Type-*I *adversary $\mathcal{A}$, it holds that*

$$\mathsf{Adv}_{\mathrm{I},6,q,1}(\lambda) = \mathsf{Adv}_{\mathrm{I},6,q,2}(\lambda).$$

The proof of Lemma 30 is the final core part of the proof against the Type-I adversary since we have to change all $\overline{\mathsf{ku}}_{\mathtt{ID}_q,\mathtt{T}}$ and $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}}$ such that $\mathtt{ID}_q \in \mathsf{prefix}^+(\mathtt{ID}^\star_{[\ell^\star-1]})$ to be semi-functional. We note that since $\mathtt{ID}^\star_{[\ell^\star]}$ is revoked at the time period $\mathtt{T}_{\mathsf{RL}}$ we do not have to change $\overline{\mathsf{ku}}_{\mathtt{ID},\mathtt{T}}$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ such that $\mathtt{ID}^\star_{[\ell^\star]} \in \mathsf{prefix}^+(\mathtt{ID})$ to be semi-functional. When it holds that $\mathtt{ID}_q \notin \mathsf{prefix}^+(\mathtt{ID}^\star_{[\ell^\star-1]})$, we prove Lemma 30 in the same way as the proof of Lemma 12 by showing that *pseudo-normal* and *pseudo-SF* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ are identically distributed. When $\mathtt{ID}_q \in \mathsf{prefix}^+(\mathtt{ID}^\star_{[\ell^\star-1]})$, we also follow the same argument as in the proof of Lemma 12 by applying the semi-functional randomness switching. In addition, as the proof of Lemma 27, we use the fact that although all $\mathsf{sk}_{\mathtt{ID},\theta}$ such that $\mathsf{pa}(\mathtt{ID}) = \mathtt{ID}_q$ may be created by the *normal* $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$, all $\mathsf{sk}_{\mathtt{ID},\theta}$ that share the same nodes with $\mathsf{ku}_{\mathtt{ID}_q,\mathtt{T},\theta}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ are created by the *semi-functional* $\mathsf{s.sk}^{(1)}_{\mathtt{ID}}$.

Proof of Lemma 30. If $\mathtt{ID}_q = (\mathtt{id}_{q,1}, \ldots, \mathtt{id}_{q,|\mathtt{ID}_q|}) \notin \mathsf{prefix}^+(\mathtt{ID}^\star_{[\ell^\star-1]})$, we can show that *pseudo-normal* and *pseudo-SF* $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ are identically distributed by following the same argument as in the proof of Lemma 7.

If $\mathtt{ID}_q = (\mathtt{id}_{q,1}, \ldots, \mathtt{id}_{q,|\mathtt{ID}_q|}) \in \mathsf{prefix}^+(\mathtt{ID}^\star_{[\ell^\star-1]})$, we prove a stronger claim that $\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$ are identically distributed from $\mathcal{A}$'s view for any fixed

- $(\mathbf{A}, \mathbf{a}) \leftarrow_R \mathcal{D}_k$,
- $\left((\mathbf{V}_\ell)_{\ell \in [0,L+2]}, \mathbf{Z}\right) \leftarrow_R (\mathbb{Z}_p^{(k+1)\times k})^{L+3} \times \mathbb{Z}_p^{k\times k}$,
- master secret key $\mathbf{k} \leftarrow_R \mathbb{Z}_p^{k+1}$,
- $\mathbf{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p^k$ for creating $\mathsf{sk}_{\mathtt{ID},\theta}$ such that $\mathsf{pa}(\mathtt{ID}) = \mathtt{ID}_q$,
- $\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ for creating $\overline{\mathsf{ku}}_{\mathtt{ID}_q,\mathtt{T}}$ for $\mathtt{T} < \mathtt{T}_{\mathsf{RL}}$,
- $\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta}, \overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}, \overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ for creating $\mathsf{ku}_{\mathtt{ID}_q,\mathtt{T}}$,
- $\mathbf{u}_{\mathtt{ID}_q,\mathtt{T}}, \mathbf{u}'_{\mathtt{ID}_q,\mathtt{T}} \leftarrow_R \mathbb{Z}_p^k$ for creating $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}}$,
- $\alpha \leftarrow_R \mathbb{Z}_p^*$.

Note that $\mathsf{sk}_{\mathtt{ID}}$ such that $\mathsf{pa}(\mathtt{ID}) \neq \mathtt{ID}_q$, $\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ such that $\mathtt{ID} \neq \mathtt{ID}_q$ are created in the same way in both $\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$.

At first, we show that the randomness of $(v_0, v_1, \ldots, v_{|\mathtt{ID}^\star|}, v_{L+1}) \leftarrow_R \mathbb{Z}_p^{|\mathtt{ID}^\star|+2}$ enables us to prove that all $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}}$ created by using $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ follow the same distribution in $\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$.

For this purpose, it is sufficient to show that $(\mathsf{s.SK}^{(2)}_{\mathtt{ID}_q,\theta,0}, \mathsf{s.SK}^{(2)}_{\mathtt{ID}_q,\theta,1}, \mathsf{s.SK}^{(2)}_{\mathtt{ID}_q,\theta,2})$ follows the same distribution in $\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$ since $(\widetilde{\mathsf{s.SK}}^{(2)}_{\mathtt{ID}_q,\theta,\ell})_{\ell\in[|\mathtt{ID}_q|+1,L]}$ is not used for creating $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}}$. By following the same argument in the proof of Lemma 7, what we have to show is

$$\begin{aligned}&\left\{\begin{array}{c} v_0+v_1\mathtt{id}^\star_1+\cdots+v_{|\mathtt{ID}^\star|}\mathtt{id}^\star_{|\mathtt{ID}^\star|},\ v_0+v_{L+1}\mathtt{T}^\star, \\ v_0+v_1\mathtt{id}_{q,1}+\cdots+v_{|\mathtt{ID}_q|}\mathtt{id}^\star_{|\mathtt{ID}_q|}\end{array}\right\}\\ \equiv&\left\{\begin{array}{c} v_0+v_1\mathtt{id}^\star_1+\cdots+v_{|\mathtt{ID}^\star|}\mathtt{id}^\star_{|\mathtt{ID}^\star|},\ v_0+v_{L+1}\mathtt{T}^\star, \\ \alpha/\hat{r}+v_0+v_1\mathtt{id}_{q,1}+\cdots+v_{|\mathtt{ID}_q|}\mathtt{id}^\star_{|\mathtt{ID}_q|}\end{array}\right\},\end{aligned} \tag{29}$$

where $(v_0,v_1,\ldots,v_{|\mathtt{ID}^\star|},v_{L+2})\leftarrow_R\mathbb{Z}_p^{|\mathtt{ID}^\star|+2}$. Here, the first and second elements are $\mathtt{tag}$ and $\mathtt{tag}'$ and the last element is the exponent of $[\hat{r}\mathbf{a}^\perp]_2$ of $\mathsf{s.SK}^{(2)}_{\mathtt{ID}_q,\theta,1}$ in $\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$, respectively. Since the only second element depends on $v_{L+1}\leftarrow_R\mathbb{Z}_p$, the second element is distributed in $\mathbb{Z}_p$ uniformly at random. As we observed above, it holds that $|\mathtt{ID}_q|<\ell^\star\le|\mathtt{ID}^\star|$. Thus, since the only first element depends on $(v_{|\mathtt{ID}_q|+1},\ldots,v_{|\mathtt{ID}^\star|})\leftarrow_R\mathbb{Z}_p^{|\mathtt{ID}^\star|-|\mathtt{ID}_q|+1}$, the first element is distributed in $\mathbb{Z}_p$ uniformly at random. As a result, the last element is also distributed in $\mathbb{Z}_p$ uniformly at random. Summarizing the discussion so far, both hand sides of (29) are distributed in $\mathbb{Z}_p^3$ uniformly at random. Thus, we complete the proof of the claim that all $\mathsf{dk}_{\mathtt{ID}_q,\mathtt{T}}$ created by using $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ follow the same distribution in $\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$.

Finally, we show that for any fixed

- $(v_0,v_1,\ldots,v_{L+1})\leftarrow_R\mathbb{Z}_p^{L+2}$,

the randomnesses of $\tilde{r}_{\mathtt{ID},\theta}\leftarrow_R\mathbb{Z}_p$ such that $\mathsf{pa}(\mathtt{ID})=\mathtt{ID}_q$ in (10), $\tilde{t}_{\mathtt{ID}_q,\mathtt{T},\theta}\leftarrow_R\mathbb{Z}_p$ in (23), and $\mathsf{delk}_{\mathtt{ID}_q,\theta},\overline{\mathsf{delk}}_{\mathtt{ID}_q,\mathtt{T}}\leftarrow_R\mathbb{Z}_p^{k+1}$ for $\mathtt{T}\ge\mathtt{T}_{\mathsf{RL}}$ enable us to prove that all $\mathsf{ku}_{\mathtt{ID}_q,\mathtt{T}}$ created by using $\mathsf{s.sk}^{(2)}_{\mathtt{ID}_q}$ follow the same distribution in $\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$. Since $\mathbf{r}_{\mathtt{ID},\theta}$ and $\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta},\overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}$ are fixed, $\mathsf{sk}_{\mathtt{ID}}$ such that $\mathsf{pa}(\mathtt{ID})=\mathtt{ID}_q$ and $\mathsf{ku}_{\mathtt{ID}_q,\mathtt{T}}$ for $\mathtt{T}\ge\mathtt{T}_{\mathsf{RL}}$ are distributed in the same way in both $\mathsf{Game}_{\mathrm{I},6,q,1}$ and $\mathsf{Game}_{\mathrm{I},6,q,2}$ except $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1},\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}$. Note that even when $\mathbf{r}_{\mathtt{ID},\theta}\leftarrow_R\mathbb{Z}_p^k$ and $\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta}$ are fixed, $(\mathsf{SK}_{\mathtt{ID},\theta,0},\mathsf{SK}_{\mathtt{ID},\theta,2},(\widetilde{\mathsf{SK}}_{\mathtt{ID},\theta,\ell})_{\ell\in[|\mathtt{ID}|+1,L]})$ and $(\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,0},\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,2})$ do not reveal the quantities of $\tilde{r}_{\mathtt{ID},\theta}\leftarrow_R\mathbb{Z}_p$ and $\tilde{t}_{\mathtt{ID}_q,\mathtt{T},\theta}\leftarrow_R\mathbb{Z}_p$ since they are masked by $\widetilde{\mathbf{r}}_{\mathtt{ID},\theta}\leftarrow_R\mathbb{Z}_p^k$ and $\widetilde{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T},\theta}\leftarrow_R\mathbb{Z}_p^k$, respectively. In $\mathsf{Game}_{\mathrm{I},6,q,1}$, for all nodes $\theta\in\mathtt{BT}_{\mathtt{ID}_q}$ that correspond to $\mathsf{ku}_{\mathtt{ID}_q,\mathtt{T},\theta}$ for $\mathtt{T}\ge\mathtt{T}_{\mathsf{RL}}$, $\mathsf{SK}_{\mathtt{ID},\theta,1}$ and $\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1},\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}$ with the same nodes are distributed as follows:

- $\mathsf{SK}_{\mathtt{ID},\theta,1}$:

$$\mathsf{SK}_{\mathtt{ID},\theta,1}=[\mathbf{k}_{\mathtt{ID},\theta}+\tilde{r}_{\mathtt{ID},\theta}\alpha\mathbf{a}^\perp]_2\cdot[\big(\mathbf{V}_0+\mathtt{id}_1\mathbf{V}_1+\cdots+\mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2,$$

- $\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1}$ and $\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}$ for $\mathtt{T}<\mathtt{T}_{\mathsf{RL}}$:

$$\begin{aligned}\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1}&=[\mathbf{k}_{\mathtt{ID}_q,\theta}+\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}}-\tilde{t}_{\mathtt{ID}_q,\mathtt{T},\theta}\alpha\mathbf{a}^\perp]_2^{-1}\cdot[(\mathbf{V}_0+\mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta}]_2,\\ \overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}&=[\mathbf{k}+\alpha\mathbf{a}^\perp+\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}}]_2\cdot[\big(\mathbf{V}_0+\mathtt{id}_{q,1}\mathbf{V}_1+\cdots+\mathtt{id}_{q,|\mathtt{ID}_q|}\mathbf{V}_{|\mathtt{ID}_q|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}]_2\\ &\quad\cdot[(\mathbf{V}_0+\mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}}]_2,\end{aligned}$$

- $\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1}$ and $\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}$ for $\mathtt{T}\ge\mathtt{T}_{\mathsf{RL}}$:

$$\begin{aligned}\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1}&=[\mathbf{k}_{\mathtt{ID}_q,\theta}+\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}}]_2^{-1}\cdot[(\mathbf{V}_0+\mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta}]_2\\ \overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}&=[\mathbf{k}+\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}}]_2\cdot[\big(\mathbf{V}_0+\mathtt{id}_{q,1}\mathbf{V}_1+\cdots+\mathtt{id}_{q,|\mathtt{ID}_q|}\mathbf{V}_{|\mathtt{ID}_q|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}]_2\\ &\quad\cdot[(\mathbf{V}_0+\mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}}]_2\cdot[\hat{r}\mathbf{a}^\perp]_2^{v_0+v_1\mathtt{id}_{q,1}+\cdots+v_{|\mathtt{ID}_q|}\mathtt{id}_{q,|\mathtt{ID}_q|}},\end{aligned}$$

where $\tilde{r}_{\mathtt{ID},\theta} \leftarrow_R \mathbb{Z}_p$, $\tilde{t}_{\mathtt{ID}_q,\mathtt{T},\theta} \leftarrow_R \mathbb{Z}_p$, $\mathsf{delk}_{\mathtt{ID}_q,\theta} = \mathbf{k}_{\mathtt{ID}_q,\theta} \leftarrow_R \mathbb{Z}_p^{k+1}$, and $\overline{\mathsf{delk}}_{\mathtt{ID}_q,\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} \leftarrow_R \mathbb{Z}_p^{k+1}$. In contrast, the above distribution can be written as follows:

- $\mathsf{SK}_{\mathtt{ID},\theta,1}$:

$$\mathsf{SK}_{\mathtt{ID},\theta,1} = [(\mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp) + (\tilde{r}_{\mathtt{ID},\theta} - 1)\alpha\mathbf{a}^\perp]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_1\mathbf{V}_1 + \cdots + \mathtt{id}_{|\mathtt{ID}|}\mathbf{V}_{|\mathtt{ID}|}\big)\mathbf{Z}\mathbf{r}_{\mathtt{ID},\theta}]_2,$$

- $\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1}$ and $\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}$ for $\mathtt{T} < \mathtt{T}_{\mathsf{RL}}$:

$$\begin{aligned}
\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1} &= [(\mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp) + \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - (\tilde{t}_{\mathtt{ID}_q,\mathtt{T},\theta} + 1)\mathbf{a}^\perp]_2^{-1} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta}]_2, \\
\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1} &= [\mathbf{k} + \alpha\mathbf{a}^\perp + \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}}]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_{q,1}\mathbf{V}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}_q|}\mathbf{V}_{|\mathtt{ID}_q|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}]_2 \\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}}]_2,
\end{aligned}$$

- $\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1}$ and $\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$:

$$\begin{aligned}
\mathsf{KU}_{\mathtt{ID}_q,\mathtt{T},\theta,1} &= [(\mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp) + (\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - \alpha\mathbf{a}^\perp)]_2^{-1} \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\mathbf{t}_{\mathtt{ID}_q,\mathtt{T},\theta}]_2 \\
\overline{\mathsf{KU}}_{\mathtt{ID}_q,\mathtt{T},1} &= [(\mathbf{k} + \alpha\mathbf{a}^\perp) + (\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - \alpha\mathbf{a}^\perp)]_2 \cdot [\big(\mathbf{V}_0 + \mathtt{id}_{q,1}\mathbf{V}_1 + \cdots + \mathtt{id}_{q,|\mathtt{ID}_q|}\mathbf{V}_{|\mathtt{ID}_q|}\big)\mathbf{Z}\overline{\mathbf{t}}_{\mathtt{ID}_q,\mathtt{T}}]_2 \\
&\quad \cdot [(\mathbf{V}_0 + \mathtt{T}\mathbf{V}_{L+1})\mathbf{Z}\overline{\mathbf{t}}'_{\mathtt{ID}_q,\mathtt{T}}]_2 \cdot [\hat{r}\mathbf{a}^\perp]_2^{v_0 + v_1\mathtt{id}_{q,1} + \cdots + v_{|\mathtt{ID}_q|}\mathtt{id}_{q,|\mathtt{ID}_q|}},
\end{aligned}$$

where each $\tilde{r}_{\mathtt{ID},\theta} - 1$ and $\tilde{t}_{\mathtt{ID}_q,\mathtt{T},\theta} + 1$ is distributed in $\mathbb{Z}_p$ uniformly at random, and each $\mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp$ and $\overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - \alpha\mathbf{a}^\perp$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ is distributed in $\mathbb{Z}_p^{k+1}$ uniformly at random. Therefore, the above distribution is the same as the distribution in $\mathsf{Game}_{\mathrm{I},6,q,2}$ by setting $\tilde{r}_{\mathtt{ID},\theta} - 1$ and $\tilde{t}_{\mathtt{ID}_q,\mathtt{T},\theta} + 1$ as the randomnesses in (10) and (23), respectively, and $\mathsf{delk}_{\mathtt{ID}_q,\theta} = \mathbf{k}_{\mathtt{ID}_q,\theta} + \alpha\mathbf{a}^\perp$, $\overline{\mathsf{delk}}_{\mathtt{ID}_q,\mathtt{T}} = \overline{\mathbf{k}}_{\mathtt{ID}_q,\mathtt{T}} - \alpha\mathbf{a}^\perp$. We note that the claim holds for all $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$ and all nodes θ that correspond to $\mathsf{ku}_{\mathtt{ID}_q,\mathtt{T},\theta}$ for $\mathtt{T} \geq \mathtt{T}_{\mathsf{RL}}$, simultaneously. Thus, we complete the proof of Lemma 30. □

Lemma 31 (Sub-secret Key Transition from Pseudo-SF to Semi-functional, $\mathsf{Game}_{\mathrm{I},6,q,2} \approx_c \mathsf{Game}_{\mathrm{I},6,q,3}$)**.** $\mathsf{Game}_{\mathrm{I},6,q,2}$ *and* $\mathsf{Game}_{\mathrm{I},6,q,3}$ *are computationally indistinguishable under the* MDDH *assumption in* $\mathbb{G}_2$*. Specifically, for any PPT Type-*I *adversary* $\mathcal{A}$ *making at most* Q_{gen} *secret key generation queries, there exists a reduction algorithm* $\mathcal{B}_{\mathrm{I},6}$ *such that*

$$|\mathsf{Adv}_{\mathrm{I},6,q,2}(\lambda) - \mathsf{Adv}_{\mathrm{I},6,q,3}(\lambda)| \leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},6}}(\lambda) + \frac{2}{p-1}$$

and $\mathtt{T}(\mathcal{B}_{\mathrm{I},6}) \approx \mathtt{T}(\mathcal{A}) + Q_{\mathsf{gen}}|\mathcal{T}| \cdot \mathsf{poly}(\lambda, L)$*, where* $\mathsf{poly}(\lambda, L)$ *is independent of* $\mathtt{T}(\mathcal{A})$*.*

We omit the detailed proof of Lemma 31 since it is almost the same as the proof of Lemma 13.

By combining Lemmata 29–31, we have

$$\begin{aligned}
|\mathsf{Adv}_{\mathrm{I},6}(\lambda) - \mathsf{Adv}_{\mathrm{I},7}(\lambda)| \leq & \sum_{q \in [Q_{\mathsf{gen}}]} |\mathsf{Adv}_{\mathrm{I},6,q-1,3}(\lambda) - \mathsf{Adv}_{\mathrm{I},6,q,1}(\lambda)| \\
& + \sum_{q \in [Q_{\mathsf{gen}}]} |\mathsf{Adv}_{\mathrm{I},6,q,1}(\lambda) - \mathsf{Adv}_{\mathrm{I},6,q,2}(\lambda)| \\
& + \sum_{q \in [Q_{\mathsf{gen}}]} |\mathsf{Adv}_{\mathrm{I},6,q,2}(\lambda) - \mathsf{Adv}_{\mathrm{I},6,q,3}(\lambda)| \\
\leq & \, Q_{\mathsf{gen}} \cdot \sum_{j \in [2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},j+4}}(\lambda) + \frac{4Q_{\mathsf{gen}}}{p-1}.
\end{aligned}$$

Thus, we complete the proof of Lemma 28. □

Lemma 32 (Final Transition, $\mathsf{Game}_{\mathrm{I},7} \equiv \mathsf{Game}_{\mathrm{I},8}$)**.** $\mathsf{Game}_{\mathrm{I},7}$ *and* $\mathsf{Game}_{\mathrm{I},8}$ *are identically distributed with probability* $1 - 1/p$*. Specifically, for any Type-*I *adversary* $\mathcal{A}$*, it holds that*

$$|\mathsf{Adv}_{\mathrm{I},7}(\lambda) - \mathsf{Adv}_{\mathrm{I},8}(\lambda)| = \frac{1}{p}.$$

We omit the proof of Lemma 32 since it is almost the same as the proof of Lemma 18.

By combining with Lemmata 3, 19, 20, 21, 22, 27, 28, and 32, against the Type-I adversary we have

$$\begin{aligned}
&\mathsf{Adv}^{\mathtt{RHIBE}}_{\Pi,L,\mathcal{A}}(\lambda) \\
&\leq |\mathsf{Adv}_{\mathrm{I},0}(\lambda) - \mathsf{Adv}_{\mathrm{I},1}(\lambda)| + \mathsf{Adv}_{\mathrm{I},1}(\lambda) \\
&= \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda) + Q_{\mathsf{gen}} \cdot \mathsf{Adv}_{\mathrm{I},2}(\lambda) \\
&\leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda) + Q_{\mathsf{gen}} \cdot \sum_{i\in[3,8]} |\mathsf{Adv}_{\mathrm{I},i-1}(\lambda) - \mathsf{Adv}_{\mathrm{I},i}(\lambda)| + \mathsf{Adv}_{\mathrm{I},8}(\lambda) \\
&\leq \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_1}_{\mathcal{B}_0}(\lambda) + Q_{\mathsf{gen}} \left(Q_{\mathsf{rev}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},j}}(\lambda) + \mathsf{T}_{\mathsf{RL}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},j+2}}(\lambda) \right. \\
&\quad \left. + Q_{\mathsf{gen}} \cdot \sum_{j\in[2]} \mathsf{Adv}^{\mathsf{MDDH}\text{-}\mathbb{G}_2}_{\mathcal{B}_{\mathrm{I},j+4}}(\lambda) + \frac{1}{p} \right).
\end{aligned}$$

By definition, $Q_{\mathsf{rev}} \leq Q_{\mathsf{gen}}$ and $\mathsf{T}_{\mathsf{RL}} \leq |\mathcal{T}|$ hold. Therefore, we obtain the ineqaulity of Lemma 1. □

7 Comparison

In this section, we compare our proposed RHIBE schemes with other known RHIBE schemes achieving the same property. We use the SXDH assumption for instantiating the schemes that are secure under the k-linear assumption. Columns $|\mathsf{MPK}|$, $|\mathsf{ct}_{\mathtt{ID},\mathtt{T}}|$, $|\mathsf{sk}_{\mathtt{ID}}|$, $|\mathsf{dk}_{\mathtt{ID},\mathtt{T}}|$, and $|\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}|$ present a comparison of the size of MPK, $\mathsf{ct}_{\mathtt{ID},\mathtt{T}}$, $\mathsf{sk}_{\mathtt{ID}}$, $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$, respectively. In the column #pairing, the number of pairing computations for the Dec algorithm are compared.

7.1 Comparison among RHIBE Schemes with Compact Ciphertexts

Table 7: Comparison of RHIBE schemes with compact ciphertexts

Scheme	security	$\lvert\mathsf{MPK}\rvert$	$\lvert\mathsf{ct}_{\mathtt{ID},\mathtt{T}}\rvert$	$\lvert\mathsf{dk}_{\mathtt{ID},\mathtt{T}}\rvert$
SE15 [SE15]	selective	$(L+6)\lvert\mathbb{G}\rvert$	$3\lvert\mathbb{G}\rvert + \lvert\mathbb{G}_T\rvert$	$3\lvert\mathbb{G}\rvert$
ETW20 [ETW20]	adaptive	$(L+5)\lvert\mathbb{G}_1\rvert + 2\lvert\mathbb{G}_T\rvert$	$3\lvert\mathbb{G}_1\rvert + \lvert\mathbb{Z}_p\rvert + \lvert\mathbb{G}_T\rvert$	$5\lvert\mathbb{G}_2\rvert$
Our Scheme	adaptive	$(L+5)\lvert\mathbb{G}_1\rvert + 2\lvert\mathbb{G}_T\rvert$	$4\lvert\mathbb{G}_1\rvert + 2\lvert\mathbb{Z}_p\rvert + \lvert\mathbb{G}_T\rvert$	$8\lvert\mathbb{G}_2\rvert$

Scheme	$\lvert\mathsf{sk}_{\mathtt{ID}}\rvert$	#pairing
SE15 [SE15]	$\lvert\mathsf{PRF}\rvert + (L - \lvert\mathtt{ID}\rvert + 2)\#\mathsf{sk}_{\mathtt{ID},\theta}\lvert\mathbb{G}\rvert$	3
ETW20 [ETW20]	$(2(L - \lvert\mathtt{ID}\rvert) + 7)(\#\mathsf{delk}_{\mathtt{ID},\theta} + \#\mathsf{sk}_{\mathtt{ID},\theta})\,\lvert\mathbb{G}_2\rvert$	3
Our Scheme	$2\#\mathsf{delk}_{\mathtt{ID},\theta}\lvert\mathbb{Z}_p\rvert + (2(L - \lvert\mathtt{ID}\rvert) + 5)\#\mathsf{sk}_{\mathtt{ID},\theta}\lvert\mathbb{G}_2\rvert$	4

Scheme	$\lvert\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}\rvert$	assump.	reduction loss
SE15 [SE15]	$(L - \lvert\mathsf{pa}(\mathtt{ID})\rvert + 3)\#\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}\lvert\mathbb{G}\rvert$	q-type	$O(L)$
ETW20 [ETW20]	$(2(L - \lvert\mathsf{pa}(\mathtt{ID})\rvert) + 5)\#\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}\lvert\mathbb{G}_2\rvert$	SXDH	$O(LQ_{\mathsf{gen}}^2\lvert\mathcal{T}\rvert)$
Our Scheme	$(2(L - \lvert\mathtt{ID}\rvert) + 9 + 5\#\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta})\lvert\mathbb{G}_2\rvert$	SXDH	$O(Q_{\mathsf{gen}}(Q_{\mathsf{gen}} + \lvert\mathcal{T}\rvert))$

Table 7 compares our proposed RHIBE scheme with the other RHIBE schemes with compact ciphertexts [SE15, ETW20], i.e., Seo and Emura's selectively secure scheme (SE15) and Emura et al.'s scheme (ETW20). Since we modify Chen and Gong's HIBE scheme [CG17] for constructing the proposed RHIBE scheme, we use the same Chen and Gong's HIBE scheme to instantiate Emura et al.'s semi-generic construction. We note that $\#\mathsf{delk}_{\mathtt{ID},\theta}$, $\#\mathsf{sk}_{\mathtt{ID},\theta}$, and $\#\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ are the same among all the schemes except that SE15 does not depend on $\#\mathsf{delk}_{\mathtt{ID},\theta}$. All schemes have similar sizes of MPK, $\mathsf{ct}_{\mathtt{ID},\mathtt{T}}$, and $\mathsf{dk}_{\mathtt{ID},\mathtt{T}}$ and almost the same #pairing. Although $\lvert\mathsf{sk}_{\mathtt{ID}}\rvert$ is much larger than that of the selectively secure SE15, it is much shorter than that of the adaptively secure ETW20. We achieve the parameter saving since a delegation key $\mathsf{delk}_{\mathtt{ID},\theta}$ of SE15 consists of $2(L - \lvert\mathtt{ID}\rvert) + 7$ $\mathbb{G}_2$ elements, while that of ours consists of two $\mathbb{Z}_p$ elements. Moreover, $\lvert\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}\rvert$ of our scheme is much shorter than those of both SE15 and ETW20. We achieve the parameter saving due to the existence of helper key update $\overline{\mathsf{ku}}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}$ that consists of $2(L - \lvert\mathtt{ID}\rvert) + 9$ $\mathbb{G}_2$ elements. Specifically, sub-key updates $\mathsf{ku}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T},\theta}$ of SE15 and ETW20 consists of $L - \lvert\mathsf{pa}(\mathtt{ID})\rvert + 3$ and $2(L - \lvert\mathsf{pa}(\mathtt{ID})\rvert) + 5$ $\mathbb{G}_2$ elements, while that of ours consists of five $\mathbb{G}_2$ elements. Although the security of SE15 is based on the non-standard q-type assumption, the security of ETW20 and ours are based on the same k-linear assumption. Unlike SE15 and ETW20, the reduction loss of our scheme does not depend on L, while that of SE15 is tighter than ours. We achieve strictly tighter reduction than ETW20.

7.2 Comparison among RHIBE Schemes with Adaptive Security

Table 8: Comparison of RHIBE schemes with adaptive security

Scheme	$\lvert\mathsf{MPK}\rvert$	$\lvert\mathsf{ct}_{\mathtt{ID},\mathtt{T}}\rvert$	$\lvert\mathsf{sk}_{\mathtt{ID}}\rvert$	$\lvert\mathsf{ku}_{\mathtt{ID},\mathtt{T}}\rvert$	$\lvert\mathsf{dk}_{\mathtt{ID},\mathtt{T}}\rvert$
ETW20 [ETW20]	$O(L)$	$O(1)$	$O((L-\ell)\#\mathsf{delk}_{\mathtt{ID},\theta})$ $+O((L-\ell)\log\lambda)$	$O(R(L-\ell)\log\lambda)$	$O(1)$
LK21 [LK21] (basic)	$O(L)$	$O(\ell\lambda)$	$O(L-\ell)$	$O(R\lambda+\ell)$	$O(\ell)$
LK21 [LK21] (shorter ct)	$O(L+\lambda)$	$O(\ell)$	$O(L+\lambda-\ell)$	$O(R\lambda^2+\ell)$	$O(\ell)$
ETW21 [ETW21] (basic)	$O(L)$	$O(\ell\lambda)$	$O(L-\ell)$	$O(R\lambda+\ell)$	$O(\ell)$
ETW21 [ETW21] (shorter ct)	$O(L+M)$	$O\left(\ell\frac{\lambda}{M}\right)$	$O(L+M-\ell)$	$O(RM\lambda+\ell)$	$O(\ell)$
ETW21 [ETW21] (shorter ku)	$O(L+M)$	$O(\ell M\lambda)$	$O(L+M-\ell)$	$O\left(\frac{R\lambda}{M}+\ell\right)$	$O(\ell)$
Ours	$O(L)$	$O(1)$	$O(\#\mathsf{delk}_{\mathtt{ID},\theta})$ $+O((L-\ell)\log\lambda)$	$O(R\log\lambda)$ $+O(L-\ell)$	$O(1)$

Table 8 compares the asymptotic space efficiency of adaptively secure RHIBE schemes [ETW20, LK21, ETW21], i.e., Emura et al.'s semi-generic construction (ETW20), Lee and Kim's generic construction (LK21), and Emura et al.'s generic construction (ETW21). Since we modify Chen and Gong's HIBE scheme [CG17] for constructing the proposed RHIBE scheme, we use the same Chen and Gong's HIBE scheme to instantiate all the (semi-)generic constructions. We use a notation $|\mathtt{ID}| = \ell$ for simplicity. Here, we assume that ETW20 and our scheme use binary trees with $N = \lambda^{\omega(1)}$ leaves as claimed in Section 3.1, while KL21 and ETW21 use binary trees with $N = 2^{O(\lambda)}$ leaves since the latter use collision resistant hash functions to assign every $\mathtt{ID}$. Let R denote the number of users in $\mathsf{RL}_{\mathtt{ID},\mathtt{T}}$. As we claimed in Section 3.1, it holds that $|\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}| = O(R\log(N/R))$. Here, we set $|\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}| \approx O(R\log\lambda)$ in the cases of ETW20 and our scheme, while we set $|\mathcal{KUN}_{\mathsf{pa}(\mathtt{ID}),\mathtt{T}}| \approx O(R\lambda)$ in the cases of LK21 and ETW21 for simplicity. The parameter M used in shorter ct variant of ETW21 is an integer such that $1 \leq M \leq \lambda$, while M used in shorter ku variant of ETW21 is a non-negative integer.

Since we compare our scheme with ETW20 in Section 7.1, we here compare our scheme with LK21 and ETW21. At first, we compare our scheme with the basic schemes of LK21 and ETW21 that have the same asymptotic efficiency. All LK21, ETW21, and our scheme have the same size of MPK. The main bottleneck of our scheme is a large $|\mathsf{sk}_{\mathtt{ID}}|$ that depends on $\#\mathsf{delk}_{\mathtt{ID},\theta}$ and $\log\lambda$, while those of LK21 and ETW21 do not depend on $\#\mathsf{delk}_{\mathtt{ID},\theta}$ and $\log\lambda$. In contrast, we achieve constant-size of $|\mathsf{ct}_{\mathtt{ID},\mathtt{T}}|$ and $|\mathsf{dk}_{\mathtt{ID},\mathtt{T}}|$, while those of LK21 and ETW21 depend on $\ell\lambda$ and ℓ, respectively. Moreover, $|\mathsf{ku}_{\mathtt{ID},\mathtt{T}}$ of our scheme tends to be smaller than those of LK21 and ETW21 since we can use binary trees with less leaves N than LK21 and ETW21.

Next, we compare our scheme with shorter ct variants of LK21 and ETW21. When we set $M = \Theta(\lambda)$, the shorter ct variants of LK21 and ETW21 have the same asymptotic efficiency. Although $|\mathsf{sk}_{\mathtt{ID}}|$ of shorter ct variants of LK21 and ETW21 become larger than their basic schemes, they are still much shorter than that of our scheme. In contrast, all the other $|\mathsf{MPK}|$, $|\mathsf{ct}_{\mathtt{ID},\mathtt{T}}|$, $|\mathsf{ku}_{\mathtt{ID},\mathtt{T}}|$, and $|\mathsf{dk}_{\mathtt{ID},\mathtt{T}}|$ of our schemes are smaller than those of the shorter ct variants of LK21 and ETW21.

Finally, we compare our scheme with shorter ku variant of ETW21. Although $|\mathsf{sk}_{\mathtt{ID}}|$ of the shorter ku variant of ETW21 becomes larger than their basic schemes, they are still much shorter than that of our scheme. In contrast, $|\mathsf{MPK}|$, $|\mathsf{ct}_{\mathtt{ID},\mathtt{T}}|$, and $|\mathsf{dk}_{\mathtt{ID},\mathtt{T}}|$ of our scheme are smaller than those of the shorter ku variant of ETW21 regardless of the selections of parameter M. When we set a parameter $M = o(\lambda/\log\lambda)$, $|\mathsf{ku}_{\mathtt{ID},\mathtt{T}}|$ of our scheme is also smaller than that of the shorter ku variant of ETW21. In other words, $|\mathsf{ku}_{\mathtt{ID},\mathtt{T}}|$ of the shorter ku variant of ETW21 is smaller than that of our scheme only when $M = \Omega(\lambda/\log\lambda)$.

8 Conclusion

We propose an adaptively secure RHIBE scheme with compact ciphertexts under the standard k-linear assumption. The adaptive security of the previous scheme proposed by Emura et al. [ETW20] was proved by reducing the adaptive security of the underlying HIBE scheme to the adaptive security of their RHIBE scheme. In contrast, we proved the adaptive security of the proposed scheme directly by the dual system encryption methodology. Thus, we achieved a tighter reduction than that of Emura et al.'s scheme. Moreover, our scheme has much shorter secret keys and key updates than that of Emura et al. with ciphertexts made compact by a factor $O(L - |\mathtt{ID}|)$. Since each parent user of the current adaptively secure RHIBE scheme has to store delegation keys whose number grows at least linearly with the number of children users, reducing the size of secret keys may pose a major problem by maintaining compact ciphertexts.

References

[BF01] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. In Joe Kilian, editor, *Advances in Cryptology - CRYPTO 2001, 21st Annual International Cryptology Conference, Proceedings*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2001.

[BGK08] Alexandra Boldyreva, Vipul Goyal, and Virendra Kumar. Identity-based encryption with efficient revocation. In Peng Ning, Paul F. Syverson, and Somesh Jha, editors, *Proceedings of the 2008 ACM Conference on Computer and Communications Security, CCS 2008*, pages 417–426. ACM, 2008.

[BKP14] Olivier Blazy, Eike Kiltz, and Jiaxin Pan. (hierarchical) identity-based encryption from affine message authentication. In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Proceedings, Part I*, volume 8616 of *Lecture Notes in Computer Science*, pages 408–425. Springer, 2014.

[CCKS18] Donghoon Chang, Amit Kumar Chauhan, Sandeep Kumar, and Somitra Kumar Sanadhya. Revocable identity-based encryption from codes with rank metric. In Nigel P. Smart, editor, *Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018*, volume 10808 of *Lecture Notes in Computer Science*, pages 435–451. Springer, 2018.

[CG17] Jie Chen and Junqing Gong. ABE with tag made easy - concise framework and new instantiations in prime-order groups. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the*

Theory and Applications of Cryptology and Information Security. Proceedings, Part II, volume 10625 of *Lecture Notes in Computer Science*, pages 35–65. Springer, 2017.

[CGW15] Jie Chen, Romain Gay, and Hoeteck Wee. Improved dual system ABE in prime-order groups via predicate encodings. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, volume 9057 of *Lecture Notes in Computer Science*, pages 595–624. Springer, 2015.

[CLL+12] Jie Chen, Hoon Wei Lim, San Ling, Huaxiong Wang, and Khoa Nguyen. Revocable identity-based encryption from lattices. In Willy Susilo, Yi Mu, and Jennifer Seberry, editors, *Information Security and Privacy - 17th Australasian Conference, ACISP 2012*, volume 7372 of *Lecture Notes in Computer Science*, pages 390–403. Springer, 2012.

[CW14] Jie Chen and Hoeteck Wee. Dual system groups and its applications - compact HIBE and more. *IACR Cryptology ePrint Archive*, 2014:265, 2014.

[DG17] Nico Döttling and Sanjam Garg. From selective IBE to full IBE and selective HIBE. In Yael Kalai and Leonid Reyzin, editors, *Theory of Cryptography - 15th International Conference, TCC 2017*, volume 10677 of *Lecture Notes in Computer Science*, pages 372–408. Springer, 2017.

[EHK+17] Alex Escala, Gottfried Herold, Eike Kiltz, Carla Ràfols, and Jorge Luis Villar. An algebraic framework for Diffie-Hellman assumptions. *J. Cryptology*, 30(1):242–288, 2017.

[ESY16] Keita Emura, Jae Hong Seo, and Taek-Young Youn. Semi-generic transformation of revocable hierarchical identity-based encryption and its DBDH instantiation. *IEICE Transactions*, 99-A(1):83–91, 2016.

[ETW20] Keita Emura, Atsushi Takayasu, and Yohei Watanabe. Adaptively secure revocable hierarchical IBE from k-linear assumption. *IACR Cryptol. ePrint Arch.*, 2020:886, 2020.

[ETW21] Keita Emura, Atsushi Takayasu, and Yohei Watanabe. Generic constructions of revocable hierarchical identity-based encryption. *IACR Cryptol. ePrint Arch.*, 2021:515, 2021.

[GCTC16] Junqing Gong, Zhenfu Cao, Shaohua Tang, and Jie Chen. Extended dual system group and shorter unbounded hierarchical identity based encryption. *Des. Codes Cryptography*, 80(3):525–559, 2016.

[HLCL18] Ziyuan Hu, Shengli Liu, Kefei Chen, and Joseph K. Liu. Revocable identity-based encryption from the computational Diffie-Hellman problem. In Willy Susilo and Guomin Yang, editors, *Information Security and Privacy - 23rd Australasian Conference, ACISP 2018, Proceedings*, volume 10946 of *Lecture Notes in Computer Science*, pages 265–283. Springer, 2018.

[ISW17] Yuu Ishida, Junji Shikata, and Yohei Watanabe. CCA-secure revocable identity-based encryption schemes with decryption key exposure resistance. *IJACT*, 3(3):288–311, 2017.

[KMT19] Shuichi Katsumata, Takahiro Matsuda, and Atsushi Takayasu. Lattice-based revocable (hierarchical) IBE with decryption key exposure resistance. In Dongdai Lin and Kazue Sako, editors, *Public-Key Cryptography - PKC 2019 - 22nd IACR International Conference on Practice and Theory of Public-Key Cryptography, Proceedings, Part II*, volume 11443 of *Lecture Notes in Computer Science*, pages 441–471. Springer, 2019.

[Lee19] Kwangsu Lee. A generic construction for revocable identity-based encryption with subset difference methods. *IACR Cryptology ePrint Archive*, 2019:798, 2019.

[Lew12] Allison B. Lewko. Tools for simulating features of composite order bilinear groups in the prime order setting. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques. Proceedings*, volume 7237 of *Lecture Notes in Computer Science*, pages 318–335. Springer, 2012.

[LK21] Kwangsu Lee and Joon Sik Kim. A generic approach to build revocable hierarchical identity-based encryption. *IACR Cryptology ePrint Archive*, 2021:502, 2021.

[LLP17] Kwangsu Lee, Dong Hoon Lee, and Jong Hwan Park. Efficient revocable identity-based encryption via subset difference methods. *Des. Codes Cryptography*, 85(1):39–76, 2017.

[LP18] Kwangsu Lee and Seunghwan Park. Revocable hierarchical identity-based encryption with shorter private keys and update keys. *Des. Codes Cryptography*, 86(10):2407–2440, 2018.

[LP19] Roman Langrehr and Jiaxin Pan. Tightly secure hierarchical identity-based encryption. In Dongdai Lin and Kazue Sako, editors, *Public-Key Cryptography - PKC 2019 - 22nd IACR International Conference on Practice and Theory of Public-Key Cryptography, Proceedings, Part I*, volume 11442 of *Lecture Notes in Computer Science*, pages 436–465. Springer, 2019.

[LP20a] Roman Langrehr and Jiaxin Pan. Hierarchical identity-based encryption with tight multi-challenge security. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *Public-Key Cryptography - PKC 2020 - 23rd IACR International Conference on Practice and Theory of Public-Key Cryptography, Proceedings, Part I*, volume 12110 of *Lecture Notes in Computer Science*, pages 153–183. Springer, 2020.

[LP20b] Roman Langrehr and Jiaxin Pan. Unbounded HIBE with tight security. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Proceedings, Part II*, volume 12492 of *Lecture Notes in Computer Science*, pages 129–159. Springer, 2020.

[LV09] Benoît Libert and Damien Vergnaud. Adaptive-ID secure revocable identity-based encryption. In Marc Fischlin, editor, *Topics in Cryptology - CT-RSA 2009, The Cryptographers' Track at the RSA Conference 2009. Proceedings*, volume 5473 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2009.

[ML19] Xuecheng Ma and Dongdai Lin. Generic constructions of revocable identity-based encryption. In Zhe Liu and Moti Yung, editors, *Information Security and Cryptology - 15th International Conference, Inscrypt 2019, Revised Selected Papers*, volume 12020 of *Lecture Notes in Computer Science*, pages 381–396. Springer, 2019.

[NNL01] Dalit Naor, Moni Naor, and Jeffery Lotspiech. Revocation and tracing schemes for stateless receivers. In Joe Kilian, editor, *Advances in Cryptology - CRYPTO 2001, 21st Annual International Cryptology Conference. Proceedings*, volume 2139 of *Lecture Notes in Computer Science*, pages 41–62. Springer, 2001.

[OT15] Tatsuaki Okamoto and Katsuyuki Takashima. Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption. *Des. Codes Cryptography*, 77(2-3):725–771, 2015.

[RLPL15] Geumsook Ryu, Kwangsu Lee, Seunghwan Park, and Dong Hoon Lee. Unbounded hierarchical identity-based encryption with efficient revocation. In Howon Kim and Dooho Choi, editors, *Information Security Applications - 16th International Workshop, WISA 2015*, volume 9503 of *Lecture Notes in Computer Science*, pages 122–133. Springer, 2015.

[RS14] Somindu C. Ramanna and Palash Sarkar. Efficient (anonymous) compact HIBE from standard assumptions. In Sherman S. M. Chow, Joseph K. Liu, Lucas Chi Kwong Hui, and Siu-Ming Yiu, editors, *Provable Security - 8th International Conference, ProvSec 2014. Proceedings*, volume 8782 of *Lecture Notes in Computer Science*, pages 243–258. Springer, 2014.

[SE13a] Jae Hong Seo and Keita Emura. Efficient delegation of key generation and revocation functionalities in identity-based encryption. In Ed Dawson, editor, *Topics in Cryptology - CT-RSA 2013 - The Cryptographers' Track at the RSA Conference 2013*, volume 7779 of *Lecture Notes in Computer Science*, pages 343–358. Springer, 2013.

[SE13b] Jae Hong Seo and Keita Emura. Revocable identity-based encryption revisited: Security model and construction. In Kaoru Kurosawa and Goichiro Hanaoka, editors, *Public-Key Cryptography - PKC 2013 - 16th International Conference on Practice and Theory in Public-Key Cryptography. Proceedings*, volume 7778 of *Lecture Notes in Computer Science*, pages 216–234. Springer, 2013.

[SE15] Jae Hong Seo and Keita Emura. Revocable hierarchical identity-based encryption: History-free update, security against insiders, and short ciphertexts. In Kaisa Nyberg, editor, *Topics in Cryptology - CT-RSA 2015, The Cryptographer's Track at the RSA Conference 2015*, volume 9048 of *Lecture Notes in Computer Science*, pages 106–123. Springer, 2015.

[Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology, Proceedings of CRYPTO '84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer, 1984.

[TW17] Atsushi Takayasu and Yohei Watanabe. Lattice-based revocable identity-based encryption with bounded decryption key exposure resistance. In Josef Pieprzyk and Suriadi Suriadi, editors, *Information Security and Privacy - 22nd Australasian Conference, ACISP 2017, Proceedings, Part I*, volume 10342 of *Lecture Notes in Computer Science*, pages 184–204. Springer, 2017.

[TW21] Atsushi Takayasu and Yohei Watanabe. Revocable identity-based encryption with bounded decryption key exposure resistance: Lattice-based construction and more. *Theor. Comput. Sci.*, 849:64–98, 2021.

[Wat09] Brent Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference. Proceedings*, volume 5677 of *Lecture Notes in Computer Science*, pages 619–636. Springer, 2009.

[WES17] Yohei Watanabe, Keita Emura, and Jae Hong Seo. New revocable IBE in prime-order groups: Adaptively secure, decryption key exposure resistant, and with short public parameters. In Helena Handschuh, editor, *Topics in Cryptology - CT-RSA 2017 - The Cryptographers' Track at the RSA Conference 2017. Proceedings*, volume 10159 of *Lecture Notes in Computer Science*, pages 432–449. Springer, 2017.

[WLXZ14] Changji Wang, Yuan Li, Xiaonan Xia, and Kangjia Zheng. An efficient and provable secure revocable identity-based encryption scheme. *PLoS ONE*, 9(9):e106925, 2014.

[WZH+19] Shixiong Wang, Juanyang Zhang, Jingnan He, Huaxiong Wang, and Chao Li. Simplified revocable hierarchical identity-based encryption from lattices. In Yi Mu, Robert H. Deng, and Xinyi Huang, editors, *Cryptology and Network Security - 18th International Conference, CANS 2019, Proceedings*, volume 11829 of *Lecture Notes in Computer Science*, pages 99–119. Springer, 2019.