J Cryptol (2020) 33:1787–1821
https://doi.org/10.1007/s00145-020-09356-x

Journal of CRYPTOLOGY

Tightly Secure Hierarchical Identity-Based Encryption

Roman Langrehr*[1] Jiaxin Pan*[2]

Online publication 15 September 2020

[1] ETH Zurich, Zurich, Switzerland
roman.langrehr@inf.ethz.ch
[2] Department of Mathematical Sciences
NTNU – Norwegian University of Science and Technology, Trondheim, Norway
jiaxin.pan@ntnu.no

Communicated by Masayuki Abe

Abstract

We construct the *first* tightly secure hierarchical identity-based encryption (HIBE) scheme based on standard assumptions, which solves an open problem from Blazy, Kiltz, and Pan (CRYPTO 2014). At the core of our constructions is a novel randomization technique that enables us to randomize user secret keys for identities with flexible length.

The security reductions of previous HIBEs lose at least a factor of Q, which is the number of user secret key queries. Different to that, the security loss of our schemes is only dependent on the security parameter. Our schemes are adaptively secure based on the Matrix Diffie-Hellman assumption, which is a generalization of standard Diffie-Hellman assumptions such as k-Linear. We have two tightly secure constructions, one with constant ciphertext size, and the other with tighter security at the cost of linear ciphertext size. Among other things, our schemes imply the *first* tightly secure identity-based signature scheme by a variant of the Naor transformation.

Keywords: Hierarchical identity-based encryption, tight security, affine message authentication codes.

1 Introduction

1.1 Motivation

Tight security. Reductions are useful tools for proving the security of public-key cryptographic schemes. Asymptotically, a reduction shows that if there is an efficient adversary $\mathcal{A}$ that breaks the security of a scheme, then we can have another adversary $\mathcal{R}$ that solves the underlying computationally hard problem. Concretely, a reduction provides a security bound for the scheme, $\varepsilon_{\mathcal{A}} \leq \ell \cdot \varepsilon_{\mathcal{R}}$,[1] where $\varepsilon_{\mathcal{A}}$ is the success probability of $\mathcal{A}$ and $\varepsilon_{\mathcal{R}}$ is that of $\mathcal{R}$. Ideally, it is more desirable to have ℓ as small as a constant. We say a reduction is *tight* if ℓ is a small constant and the running time of $\mathcal{A}$ is approximately the same as that of $\mathcal{R}$. Most of the current works have considered the tightness notion called "almost tight security", where ℓ may linearly (or, even better, logarithmically) depend on the security parameter, but not on the size of $\mathcal{A}$.[2] Recently, tightly secure cryptographic schemes drew a large amount of attention (e.g. [HJ12, CW13, BKP14, GHKW16, GDCC16, AHN$^+$17, GHKP18, HHK18]), since tightly secure schemes do not need to compensate for any security loss.

*Most of this work were done when both authors were at Karlsruhe Institute of Technology (KIT), Germany. In particular, J. Pan was employed at the group of Dennis Hofheinz and supported by DFG grant HO 4534/4-1.

[1] Here we ignore the additive negligible terms for simplicity.

[2] In this paper, we do not distinguish almost tight security from tight security, but we will detail the security loss in the security proof and comparison of our schemes.

(Hierarchical) identity-based encryption. The concept of identity-based encryption (IBE) was proposed by Shamir [Sha84] to simplify the management of public keys and certificates. With an IBE scheme, one can encrypt a message under a recipient's identity id (for instance, email address or ID card number), and this encrypted message can be decrypted with user id's secret key from a trusted authority. The first constructions of IBE were given in 2001 [BF01, Coc01, SOK00] in the random oracle model.

A hierarchical IBE (HIBE) scheme [HL02, GS02] generalizes the concept of IBE and provides more functionality by forming levels of a hierarchy. In an L-level HIBE, a hierarchical identity is a vector of maximal L identities, and a user at level i can delegate a secret key for its descendants at level i' (where $i < i' \leq L$). Moreover, a user at level i is not supposed to decrypt any encryption from a recipient who is not among its descendants. HIBE schemes not only are more general than IBE schemes (for instance, an IBE is simply a 1-level HIBE), but also provide numerous applications. Most famous ones are CCA-secure IBEs [CHK04] and identity-based signatures [KN09] from HIBE. Both implications are tight.

Adaptive security is a widely accepted security notion for (H)IBEs, where an adversary is allowed to adaptively choose a challenge identity id^* after it sees the (master) public key and Q-many user secret keys for adversarially chosen identities. To achieve adaptive security in the standard model, the early IBE constructions require either non-tight reductions to the hardness of the underlying assumptions [Wat05, CLL$^+$13, Lew12, JR13], or Q-type, non-static assumptions [Gen06].

In 2013, Chen and Wee constructed the first tightly secure IBE based on static assumptions in the standard model [CW13]. After that, several works have been done to improve its efficiency and achieve stronger security [BKP14, HKS15, GDCC16, HJP18]. However, constructing an L-level HIBE for $L > 1$ with a tight (i.e., independent of Q) security reduction to a standard assumption remains open.

HIBEs meet tightness: difficulties and the hope. Before analyzing the difficulties of achieving tightly secure HIBE, we consider the security loss of the current state-of-the-art HIBEs. The L-level HIBE from [Wat05] has a relatively large security loss, Q^L, which depends on both Q and L. Although the security loss of more recent HIBEs [Wat09, Lew12, CW13, BKP14, GCTC16] does not depend on the number of maximal levels L, they are still not tight and lose a factor of Q.

In general, it is harder to construct HIBEs than IBEs, since HIBEs allow public delegation of user secret keys, given the corresponding ancestor's secret key. Hence, given a tightly secure IBE, there is no (tight) black-box transformation to HIBE. The work of Lewko and Waters [LW14] shows the potential difficulty of constructing HIBE with tight reductions. More precisely, [LW14] proves that it is hard to have an HIBE scheme with security loss less than exponential in L if the HIBE has rerandomizable user secret keys (over all "functional" user secret keys).

The first attempt of constructing tightly secure HIBEs is due to Blazy, Kiltz, and Pan (cf. the proceeding version and the first full version of [BKP14]), where they tightly transform algebraic message authentication code (MAC) schemes with affine structures to (H)IBE schemes. As long as the algebraic MAC has tight security, the resulting (H)IBE is tightly secure. The first version of their paper contains a tightly secure delegatable MAC, which results in a tightly secure HIBE. The resulting HIBE has bypassed the impossibility result of [LW14] and their user secret keys are only rerandomizable over all keys generated by the user secret key generation algorithm, which is only a subspace of all "functional" keys. However, shortly after its publication, a flaw was found in a proof step of the delegatable MAC, and they remove this tightly secure delegatable MAC from their paper. The flaw is basically due to the fact that the BKP randomization technique failed to randomize MAC tags (which is an important part of user secret keys) for hierarchical identities.

The hope of achieving tight security for HIBEs lies in developing a novel method that enables randomization of user secret keys for identities with flexible level.

1.2 Our contributions

We answer the aforementioned open question affirmatively with two *tightly secure* hierarchical identity-based encryption schemes with identity space $\mathcal{ID} := (\{0,1\}^\alpha)^{\leq L}$: One with constant ciphertext size (in terms of the number of group elements) and $O(\alpha L^2)$ security loss, and the other with ciphertext size linear in L but $O(\alpha L)$ security loss. Both schemes are the *first* tightly secure HIBEs. We compare our schemes with the existing HIBE schemes in prime-order pairing groups in Table 1.

Furthermore, via the known tight transformations from [KN09] and [CHK04], our HIBEs imply the *first* tightly secure identity-based signature and tightly CCA-secure HIBEs almost for free. We note that

Scheme	$\lvert\mathsf{mpk}\rvert$	$\lvert\mathsf{usk}\rvert$	$\lvert\mathsf{C}\rvert$	Loss	Assumption
Wat05 [Wat05]	$O(\alpha L)\lvert\mathbb{G}\rvert$	$O(\alpha L)\lvert\mathbb{G}\rvert$	$(1+p)\lvert\mathbb{G}\rvert$	$O(\alpha Q)^L$	DBDH
Wat09 [Wat09]	$O(L)\lvert\mathbb{G}\rvert$	$O(p)(\lvert\mathbb{G}\rvert+\lvert\mathbb{Z}_q\rvert)$	$O(p)(\lvert\mathbb{G}\rvert+\lvert\mathbb{Z}_q\rvert)$	$O(Q)$	2-LIN
Lew12 [Lew12]	$60\lvert\mathbb{G}\rvert+2\lvert\mathbb{G}_T\rvert$	$(60+10p)\lvert\mathbb{G}\rvert$	$10p\lvert\mathbb{G}\rvert$	$O(QL)$	2-LIN
CW13 [CW13]	$O(Lk^2)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$O(Lk)\lvert\mathbb{G}_2\rvert$	$(2k+2)\lvert\mathbb{G}_1\rvert$	$O(Q)$	k-LIN
BKP14 [BKP14]	$O(Lk^2)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$O(Lk)\lvert\mathbb{G}_2\rvert$	$(2k+2)\lvert\mathbb{G}_1\rvert$	$O(Q)$	k-LIN
GCTC16 [GCTC16]	$(6k^2+12k)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$ $+(k+2)\lvert\mathbb{G}_T\rvert$	$((6k+12)\lceil p/3\rceil$ $-(k+2)p)\lvert\mathbb{G}_2\rvert$	$(3k+6)\lceil p/3\rceil\lvert\mathbb{G}_1\rvert$	$O(QL)$	k-LIN
HIBKEM_1 (Fig. 13)	$O(\alpha L^2k^2)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$O(\alpha L^2k)\lvert\mathbb{G}_2\rvert$	$(4k+1)\lvert\mathbb{G}_1\rvert$	$O(\alpha L^2k)$	k-LIN
$\mathsf{HIBKEM}_1^{\mathcal{H}}$ (Fig. 7)	$O(\gamma Lk^2)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$O(\gamma Lk)\lvert\mathbb{G}_2\rvert$	$(4k+1)\lvert\mathbb{G}_1\rvert$	$O(\gamma Lk)$	k-LIN
HIBKEM_2 (Fig. 14)	$O(\alpha L^2k^2)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$(3kp+k+1)\lvert\mathbb{G}_2\rvert$	$(3kp+k+1)\lvert\mathbb{G}_1\rvert$	$O(\alpha Lk)$	k-LIN
$\mathsf{HIBKEM}_2^{\mathcal{H}}$ (Fig. 11)	$O(\gamma Lk^2)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$(3kp+k+1)\lvert\mathbb{G}_2\rvert$	$(3kp+k+1)\lvert\mathbb{G}_1\rvert$	$O(\gamma k)$	k-LIN

Table 1: Comparison of L-level HIBEs with identity-space $\mathcal{ID} = (\{0,1\}^\alpha)^{\leq L}$ in prime-order pairing groups. Schemes in gray are new ones from this paper. In particular, $\mathsf{HIBKEM}_1^{\mathcal{H}}$ is an improvement of HIBKEM_1 with collision-resistant hash functions, namely, implementing the generic construction (cf. Figure 12) with the MAC scheme from Figure 7. Similarly, $\mathsf{HIBKEM}_2^{\mathcal{H}}$ is an improvement of HIBKEM_2 with the MAC scheme from Figure 11. '$\lvert\mathsf{mpk}\rvert$', '$\lvert\mathsf{usk}\rvert$' and '$\lvert\mathsf{C}\rvert$' stand for the size of master public key, user secret key and ciphertext. We count the number of group elements in $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_T$. For a scheme that works in symmetric pairing groups, we write $\mathbb{G} := \mathbb{G}_1 = \mathbb{G}_2$. The schemes that work in asymmetric pairing groups can be instantiated with SXDH=1-LIN. Q is the number of user secret key queries by the adversary. γ is the bit length of the range of a collision-resistant hash functions. In the '$\lvert\mathsf{usk}\rvert$' and '$\lvert\mathsf{C}\rvert$' columns p stands for the hierarchy depth of the identity vector

an $(L+1)$-level HIBE tightly implies an L-level CCA-secure HIBE via the CHK transformation [CHK04] in the single-challenge setting.

Core idea. In a nutshell, the technical novelty of our constructions is a new randomization technique that enables us to randomize user secret keys with flexible identity length. This technique is motivated by the recent tightly CCA-secure public-key encryption of Gay et al. [GHKW16].

At the core of our constructions lie two new pseudorandom message authentication code (MAC) schemes for messages with flexible length. Their pseudorandomness can be proven with tight reductions to the Matrix Decisional Diffie-Hellman (MDDH) assumption [EHK+13]. The MDDH assumption is a generalization of the known standard Diffie-Hellman assumptions, such as the k-linear (k-LIN) assumption. Our MAC schemes have algebraic structures compatible with the BKP transformation. In the end, together with a variant of the BKP framework [BKP14], we can tightly randomize user secret keys with hierarchical identities, and we have tightly secure HIBEs.

A closer look at the BKP framework. The BKP framework proposes the notion of affine MACs and transforms it to an (H)IBE scheme with pairings. Their transformation is tightness-preserving. Under the MDDH assumption, if the affine MAC is tightly secure, then the (H)IBE is also tightly secure. It is worth mentioning that the BKP transformation and its variants are widely used in constructing identity-based encryption [HJP18] with multi-challenge CCA security, predicate encryption [Wee14, CGW15], quasi-adaptive NIZK [KW15], and structure-preserving signature [KPW15, GHKP18] based on standard, static assumptions.

We recall their tightly secure MAC, $\mathsf{MAC}_{\mathsf{NR}}$, based on the Naor-Reingold pseudorandom function [NR97], which is implicitly in the Chen-Wee (CW) IBE [CW13] as well. We observe in this paper that $\mathsf{MAC}_{\mathsf{NR}}$ is tightly secure in the semi-adaptive sense, which implies a tightly semi-adaptively secure HIBE. For completeness, we provide a detailed proof for that in Appendix A. However, until now, we do not know any tight security proof for the adaptive security of $\mathsf{MAC}_{\mathsf{NR}}$. In the following, we give a more detailed analysis of that.

$\mathsf{MAC}_{\mathsf{NR}}$ is defined over an additive prime-order group $\mathbb{G}_2 := \langle P_2 \rangle$ and its message space is corresponding to the identity space of the resulting IBE. We use the implicit notation $[x]_2 := xP_2$ from [EHK+13]. $\mathsf{MAC}_{\mathsf{NR}}$ chooses $\mathbf{B} \in \mathbb{Z}_q^{(k+1)\times k}$ according to the underlying assumption. For message space $\mathcal{M} := \{0,1\}^\alpha$, its secret key is defined as

$$\mathsf{sk}_{\mathsf{MAC}} := \left((\mathbf{x}_{i,b})_{1\leq i\leq \alpha, b=0,1}, x_0' \right) \in \left(\mathbb{Z}_q^{k\cdot 2}\right)^\alpha \times \mathbb{Z}_q$$

and its MAC tag contains a message-independent vector $[\mathbf{t}]_2$ and a message-dependent value $[u]_2$ in the form of

$$\begin{aligned} \mathbf{t} &= \overline{\mathbf{B}}\mathbf{s} \in \mathbb{Z}_q^k \quad \text{for} \quad \mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k \\ u &= \sum\nolimits_i \mathbf{x}_{i,\mathsf{m}_i}^\top \mathbf{t} + x_0' \in \mathbb{Z}_q \,, \end{aligned} \tag{1}$$

where $\overline{\mathbf{B}}$ denotes the first k rows of $\mathbf{B}$. The BKP transformation requires the MAC scheme has pseudorandomness against chosen-message attacks ($\mathsf{PR\text{-}CMA}$ security), which is a decisional variant of the standard existential unforgeability against chosen-message attacks ($\mathsf{EUF\text{-}CMA}$ security). In order to provide a simpler and more intuitive discussion, we consider the standard $\mathsf{EUF\text{-}CMA}$ security of $\mathsf{MAC}_{\mathsf{NR}}$, where an adversary $\mathcal{A}$ is allowed to see many MAC tags $\tau_{\mathsf{m}} := ([\mathbf{t}_{\mathsf{m}}]_2, [u_{\mathsf{m}}]_2)$ on messages m of its choice and tries to forge a fresh and valid forgery (m^*, τ^*) which satisfies Equation (1).

Following the CW argument [CW13], by a hybrid argument on the bit length of m, one can show that the value $[u]_2$ is pseudorandom such that it is hard for an adversary to forge. By embedding the problem challenge in $\mathbf{t}$ and $\mathbf{x}_{i+1,1-b}$, the CW argument can manage to develop the following random function RF_{i+1} for $(i+1)$-bit messages from a random function RF_i for i-bit messages on-the-fly:

$$\mathsf{RF}_{i+1}(\mathsf{m}_{|i+1}) = \begin{cases} \mathsf{RF}_i(\mathsf{m}_{|i}) & (\text{if } \mathsf{m}_{i+1} = b) \\ \mathsf{RF}_i(\mathsf{m}_{|i}) + \mathsf{RF}_i'(\mathsf{m}_{|i}) & (\text{if } \mathsf{m}_{i+1} = 1-b) \end{cases}, \tag{2}$$

where b is the guess for the $(i+1)$-th bit of m^* and $\mathsf{m}_{|i}$ is the first i bits of m. Such an argument works well if messages have fixed length. For messages m with fixed length, an adversary can see the output of either RF_i (in Hybrid i) or RF_{i+1} (in Hybrid $i+1$), but not both. However, that is not the case for messages m' with flexible length.

Concretely, identities for HIBEs are messages with flexible level. If we follow the CW and BKP arguments, we first need to develop a random function at the 2-level based on that at the 1-level. The critical case happens when we switch from Hybrid α (namely, the end of randomization at the 1-level) to Hybrid $\alpha+1$ (namely, the beginning of randomization at the 2-level). If we define $\mathsf{RF}_{\alpha+1}$ (with message space $\{0,1\}^\alpha \cup \{0,1\}^{\alpha+1}$) via Equation (2) based on random functions $\mathsf{RF}_\alpha, \mathsf{RF}_\alpha'$ (with message space $\{0,1\}^\alpha$), then we have $\mathsf{RF}_{\alpha+1}(\mathsf{m}) = \mathsf{RF}_{\alpha+1}(\mathsf{m}||b)$ for a $\mathsf{m} \in \{0,1\}^\alpha$ and that means the resulting $\mathsf{RF}_{\alpha+1}$ is not a random function for messages with flexible levels.

1.3 Our approach: independent randomization

To circumvent the problem mentioned above, we propose a suitable pseudorandom MAC, which isolates the tag randomization for messages with different levels. Our strategy is to randomize tags for messages with only one level first, and then for those with two levels, and so on. By a novel use of the recent subspace randomization refined from [GHKW16], tags for messages with different levels are randomized independently.

Affine MACs with levels. We consider a new notion of affine MACs, called *affine MACs with levels*, and we give two constructions of it. This new notion considers messages with flexible levels and enables us to develop independent random functions RF_α for messages with only one level (i.e., in $\{0,1\}^\alpha$), and $\mathsf{RF}_{2\cdot\alpha}'$ for messages with only two levels (i.e., in $\{0,1\}^{2\alpha}$), and so on. For simplicity, we present an overview of our technique in terms of 2-level HIBEs (namely, the maximum level of the HIBE, $L = 2$), namely, the hierarchical identity space $\mathcal{ID} := (\{0,1\}^\alpha)^{\leq 2}$. We denote 1-level messages as $\mathsf{m} \in \{0,1\}^\alpha$ and 2-level messages as $\mathsf{m}' \in \{0,1\}^{\alpha\cdot 2}$.

Our first MAC construction MAC_1's secret keys have the form of

$$\mathsf{sk}_{\mathsf{MAC}_1} := \Big((\mathbf{x}_{i,b})_{i,b}, \boxed{(\hat{\mathbf{x}}_{j,b})_{1\leq j \leq 2\alpha, b}}, x_0' \Big) \in \left(\mathbb{Z}_q^{k\cdot 2}\right)^\alpha \times \boxed{\left(\mathbb{Z}_q^{k\cdot 2}\right)^{\alpha\cdot 2}} \times \mathbb{Z}_q \,.$$

The value u in the MAC tags for $\mathsf{m} \in \{0,1\}^\alpha$ and $\mathsf{m}' \in \{0,1\}^{2\alpha}$ has the form of

$$\begin{aligned} u_{\mathsf{m}} &:= \sum_{i=1}^{\alpha} \mathbf{x}_{i,\mathsf{m}_i}^\top \mathbf{t} + x_0' \in \mathbb{Z}_q \\ u_{\mathsf{m}'} &:= \sum_{i=1}^{\alpha} \mathbf{x}_{i,\mathsf{m}_i'}^\top \mathbf{t} + \boxed{\sum_{j=1}^{2\alpha} \hat{\mathbf{x}}_{j,\mathsf{m}_j'}^\top \mathbf{t}} + x_0' \in \mathbb{Z}_q \,. \end{aligned} \tag{3}$$

Essentially, our MAC first encodes m' as $(\mathsf{m}'_1, \ldots, \mathsf{m}'_\alpha)$ and $(\mathsf{m}'_1, \ldots, \mathsf{m}'_{2\cdot\alpha})$ and then uses $(\mathbf{x}_{i,b})_{1 \le i \le \alpha, b}$ for $(\mathsf{m}'_1, \ldots, \mathsf{m}'_\alpha)$ and $(\hat{\mathbf{x}}_{j,b})_{1 \le j \le 2\alpha, b}$ for $(\mathsf{m}'_1, \ldots, \mathsf{m}'_{2\cdot\alpha})$, while the BKP MAC, $\mathsf{MAC}_{\mathsf{NR}}$, encodes m' as $(\mathsf{m}'_1, \ldots, \mathsf{m}'_\alpha)$ and $(\mathsf{m}'_{\alpha+1}, \ldots, \mathsf{m}'_{2\cdot\alpha})$. However, combining this new encoding method and the BKP randomization strategy is not enough to achieve our goal.

By a similar argument as in BKP, we can randomize all the u_{m} for 1-level messages m and, after the first level messages randomization, u_{m} has the form

$$u_{\mathsf{m}} := \sum_{i=1}^{\alpha} \mathbf{x}_{i,\mathsf{m}_i}^\top \mathbf{t} + \mathsf{RF}_\alpha(\mathsf{m}),$$

namely, we replace x'_0 with $\mathsf{RF}_\alpha(\mathsf{m})$, but this affects the $u_{\mathsf{m}'}$ for 2-level messages m' as well. More precisely, $u_{\mathsf{m}'}$ carries the random function RF_α and has the form

$$u_{\mathsf{m}'} := \left(\sum_{i=1}^{\alpha} \mathbf{x}_{i,\mathsf{m}'_i}^\top + \sum_{j=1}^{2\alpha} \hat{\mathbf{x}}_{j,\mathsf{m}'_j}^\top \right) \mathbf{t} + \mathsf{RF}_\alpha(\mathsf{m}'_{|\alpha}).$$

If we continue to randomize $u_{\mathsf{m}'}$, we will run into the exact same problem as in the CW or BKP randomization, namely, the output of both RF_α and $\mathsf{RF}_{\alpha+1}$ will be leaked in Hybrid $\alpha + 1$.

Motivated by [GHKW16], we hide RF_α in some orthogonal space to solve the above problem. By switching $\mathbf{t}$ into the "right" span, RF_α appears in u_{m}, but gets canceled in $u_{\mathsf{m}'}$. Concretely, we choose $\mathbf{B} \xleftarrow{\$} \mathbb{Z}_q^{3k \times k}$ and $\mathbf{B}^\perp \in \mathbb{Z}_q^{3k \times 2k}$ is a kernel matrix of $\mathbf{B}$ such that $(\mathbf{B}^\perp)^\top \mathbf{B} = \mathbf{0}$. We replace $\mathbf{t} \xleftarrow{\$} \mathbb{Z}_q^k$ with larger $\mathbf{t} \xleftarrow{\$} \mathbb{Z}_q^{3k}$. Accordingly, we choose $\mathbf{x}_{i,b}$ and $\hat{\mathbf{x}}_{j,b}$ from $\mathbb{Z}_q^{3k}$. We embed the random function RF_α into the kernel of $\mathbf{B}$ and u_{y} ($\mathsf{y} \in \{\mathsf{m}, \mathsf{m}'\}$) has the form

$$u_{\mathsf{y}} := \Big(\quad \sim \quad + \quad \mathsf{RF}_\alpha(\mathsf{y}_{|\alpha})(\mathbf{B}^\perp)^\top \Big) \mathbf{t} + x'_0,$$

where "$\sim$" denotes corresponding summation terms. During the randomization for 1-level messages, if we choose $\mathbf{t} \in \mathsf{Span}(\mathbf{B}) := \{\mathbf{v} \mid \exists \mathbf{s} \in \mathbb{Z}_q^k : \mathbf{v} = \mathbf{B}\mathbf{s}\}$ for 2-level messages m', then RF_α will get canceled out; and if we choose $\mathbf{t} \notin \mathsf{Span}(\mathbf{B})$ for 1-level messages m, then RF_α will appear and u_{m} gets randomized. After the randomization for 1-level messages, $u_{\mathsf{m}'}$ for 2-level messages m' is distributed the same as in Equation (3) so that we can start 2-level randomization from a constant random function $\mathsf{RF}'_0(\varepsilon)$ multiplying with $(\mathbf{B}^\perp)^\top$, where ε denotes the empty string.

The way of developing RF_α (or $\mathsf{RF}'_{2\cdot\alpha}$, respectively) from RF_0 (or RF'_0, respectively) is similar to [GHKW16]. Roughly, we choose two random matrices $\mathbf{B}_0, \mathbf{B}_1 \xleftarrow{\$} \mathbb{Z}_q^{3k \times k}$ and decompose $\mathbb{Z}_q^{3k}$ into the span of $\mathbf{B}, \mathbf{B}_0, \mathbf{B}_1$. The span of $\mathbf{B}^\perp$ is decomposed into that of $\mathbf{B}_0^* \in \mathbb{Z}_q^{3k \times k}$ and $\mathbf{B}_1^* \in \mathbb{Z}_q^{3k \times k}$. An overview of the orthogonal relations between all these matrices is given in Figure 1. After the decomposition of linear spaces, $\mathsf{RF}_i(\mathsf{m}_{|i})(\mathbf{B}^\perp)^\top = \mathsf{RF}_i^{(0)}(\mathsf{m}_{|i})(\mathbf{B}_0^*)^\top + \mathsf{RF}_i^{(1)}(\mathsf{m}_{|i})(\mathbf{B}_1^*)^\top$. By using the MDDH assumption, we can switch $[\mathbf{t}]_2$ to the right span and develop $\mathsf{RF}_{i+1}(\mathsf{m}_{|i+1})(\mathbf{B}^\perp)^\top$ from $\mathsf{RF}_i(\mathsf{m}_{|i})(\mathbf{B}^\perp)^\top$ in a tight fashion.

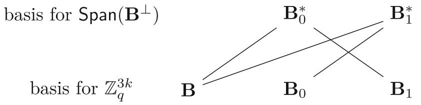
basis for $\mathsf{Span}(\mathbf{B}^\perp)$
$\mathbf{B}_0^*$
$\mathbf{B}_1^*$
basis for $\mathbb{Z}_q^{3k}$
$\mathbf{B}$
$\mathbf{B}_0$
$\mathbf{B}_1$

Figure 1: Solid lines mean orthogonal: $\mathbf{B}^\top \mathbf{B}_0^* = \mathbf{B}_1^\top \mathbf{B}_0^* = \mathbf{0} = \mathbf{B}^\top \mathbf{B}_1^* = \mathbf{B}_0^\top \mathbf{B}_1^* \in \mathbb{Z}_q^{k \times k}$

In order to have public delegation, the user secret keys at level 1 contain delegation terms $[\hat{\mathbf{x}}_{j,b}^\top \mathbf{t}]_2$. Since our randomization at different levels is isolated, the published terms will not affect our randomization strategy. Details are given in Section 3.1. In the end, our security reduction loses a factor of $O(\alpha L^2)$ due to L-many randomization loops and the fact that in each loop an additional factor of $O(\alpha L)$ is required. Applying a variant of the BKP transformation (cf. Section 4), we obtain the *first* HIBE scheme with tight security.

Achieving tighter security. Our second MAC construction (MAC_2 in Section 3.2) parallelizes the above randomization strategy and it has a scheme with security loss $O(\alpha L)$. The cost of doing this is to have different $\mathbf{t}_i$ at different levels for a message with L levels, which results in an HIBE with $O(L)$-size ciphertext via the BKP transformation.

1.4 More related work and open problems

Bader et al. [BHJ+15] use some idea from the BKP HIBE to construct digital signature schemes with corruptions, but it does not involve any randomization for messages with flexible length, and thus it does not have the same issue as the BKP.

Very recently, Hofheinz, Jia, and Pan [HJP18] extend the BKP construction with the information-theoretical Cramer-Shoup-like argument of [GHKW16] to answer multiple challenge ciphertext queries for IBE. However, we do not think that their technique and the one from [GDCC16] can work in a straightforward way here to construct tightly multi-challenge secure HIBE. To give more details, the main idea in [GHKW16, HJP18] is to have the secret keys as matrices instead of vectors in the BKP construction such that they can create subspaces to answer multiple challenge queries. Since our randomization technique is quite different from BKP, these subspace creating technique does not work here. Essentially, the information-theoretic arguments in Lemmata 3.10, 3.11 and 3.14 (for our first MAC) and Lemmata 3.25 and 3.26 (for our second MAC) will fail in the multi-challenge setting even with larger secret keys as matrices. Thus, we leave achieving tight multi-challenge security for HIBE as an open problem.

1.5 Publication Information and Acknowledgments

An extended abstract of this work appeared in the proceedings of PKC 2019 [LP19]. Shortly after the publication, we got invited to submit our full version to the Journal of Cryptology (JoC) based on the recommendation of the program chairs. We thank all our anonymous reviewers for their valuable comments.

This paper is the JoC version of [LP19]. It contains a security proof of our second MAC scheme with hash functions (cf. Theorem 3.19) and our generic HIBKEM construction (cf. Theorem 4.1), and a concrete instantiation of our generic construction based on the SXDH assumption.

In [LP19] the constant factor of the security loss has been understated. In this work this has been corrected. For details see end of Section 2.1.

2 Preliminaries

Notations. We use $x \stackrel{\$}{\leftarrow} \mathcal{S}$ to denote the process of sampling an element x from $\mathcal{S}$ uniformly at random if $\mathcal{S}$ is a set. For positive integers $k, \eta \in \mathbb{N}_+$ and a matrix $\mathbf{A} \in \mathbb{Z}_q^{(k+\eta)\times k}$, we denote the upper square matrix of $\mathbf{A}$ by $\overline{\mathbf{A}} \in \mathbb{Z}_q^{k\times k}$ and the lower η rows of $\mathbf{A}$ by $\underline{\mathbf{A}} \in \mathbb{Z}_q^{\eta\times k}$. Similarly, for a column vector $\mathbf{v} \in \mathbb{Z}_q^{k+\eta}$, we denote the upper k elements by $\overline{\mathbf{v}} \in \mathbb{Z}_q^k$ and the lower η elements of $\mathbf{v}$ by $\underline{\mathbf{v}} \in \mathbb{Z}_q^\eta$. We use $\mathbf{A}^{-\top}$ as shorthand for $\left(\mathbf{A}^{-1}\right)^\top$. For a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, we use $\mathsf{Span}(\mathbf{A}) := \left\{\mathbf{A}\mathbf{v} \mid \mathbf{v} \in \mathbb{Z}_q^m\right\}$ to denote the linear span of $\mathbf{A}$ and $\mathbf{A}^\perp$ denotes an arbitrary matrix with $\mathsf{Span}\left(\mathbf{A}^\perp\right) = \left\{\mathbf{v} \mid \mathbf{A}^\top \mathbf{v} = \mathbf{0}\right\}$.

For a set $\mathcal{S}$ and $n \in \mathbb{N}_+$, $\mathcal{S}^n$ denotes the set of all n-tuples with components in $\mathcal{S}$. For a string $\mathsf{m} \in \Sigma^n$, m_i denotes the i-th component of m $(1 \le i \le n)$ and $\mathsf{m}_{|i}$ denotes the prefix of length i of m. Furthermore for a p-tuple of bit strings $\mathsf{m} \in (\{0,1\}^n)^p$, we use $[\![\mathsf{m}]\!]$ to denote the string $\mathsf{m}_1||\ldots||\mathsf{m}_p$. Thus for $1 \le i \le np$, $[\![\mathsf{m}]\!]_i$ denotes the i-th bit of $\mathsf{m}_1||\ldots||\mathsf{m}_p$ and $[\![\mathsf{m}]\!]_{|i}$ denotes the i-bit-long prefix of $\mathsf{m}_1||\ldots||\mathsf{m}_p$.

All algorithms in this paper are probabilistic polynomial-time unless we state otherwise. If $\mathcal{A}$ is an algorithm, then we write $a \stackrel{\$}{\leftarrow} \mathcal{A}(b)$ to denote the random variable outputted by $\mathcal{A}$ on input b.

Games. Following [BKP14], we use code-based games to define and prove security. A game G contains procedures Init and Finalize, and some additional procedures $\mathrm{P}_1, \ldots, \mathrm{P}_n$, which are defined in pseudocode. Initially all variables in a game are undefined (denoted by $\perp$), all sets are empty (denote by $\emptyset$), and all partial maps (denoted by $f : A \dashrightarrow B$) are totally undefined. An adversary $\mathcal{A}$ is executed in game G (denote by $\mathsf{G}^{\mathcal{A}}$) if it first calls Init, obtaining its output. Next, it may make arbitrary queries to P_i (according to their specification), again obtaining their output. Finally, it makes one single call to Finalize$(\cdot)$ and stops. We use $\mathsf{G}^{\mathcal{A}} \Rightarrow d$ to denote that G outputs d after interacting with $\mathcal{A}$, and d is the output of Finalize. $T(\mathcal{A})$ denotes the running time of $\mathcal{A}$.

2.1 Pairing groups and matrix Diffie-Hellman assumptions

Let GGen be a probabilistic polynomial time (PPT) algorithm that on input 1^λ returns a description $\mathcal{G} := (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, P_1, P_2, e)$ of asymmetric pairing groups where $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ are cyclic groups of order q for a λ-bit prime q, P_1 and P_2 are generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, and $e : \mathbb{G}_1 \times \mathbb{G}_2$ is an efficient computable (non-degenerated) bilinear map. Define $P_T := e(P_1, P_2)$, which is a generator in $\mathbb{G}_T$. In this paper, we only consider Type III pairings, where $\mathbb{G}_1 \neq \mathbb{G}_2$ and there is no efficient homomorphism between them. All constructions in this paper can be easily instantiated with Type I pairings by setting $\mathbb{G}_1 = \mathbb{G}_2$ and defining the dimension k to be greater than 1.

We use the implicit representation of group elements as in [EHK+13]. For $s \in \{1, 2, T\}$ and $a \in \mathbb{Z}_q$ define $[a]_s = aP_s \in \mathbb{G}_s$ as the implicit representation of a in $\mathbb{G}_s$. Similarly, for a matrix $\mathbf{A} = (a_{ij}) \in \mathbb{Z}_q^{n\times m}$ we define $[\mathbf{A}]_s$ as the implicit representation of $\mathbf{A}$ in $\mathbb{G}_s$. $\mathsf{Span}(\mathbf{A}) := \{\mathbf{Ar} | \mathbf{r} \in \mathbb{Z}_q^m\} \subset \mathbb{Z}_q^n$ denotes the linear span of $\mathbf{A}$, and similarly $\mathsf{Span}([\mathbf{A}]_s) := \{[\mathbf{Ar}]_s | \mathbf{r} \in \mathbb{Z}_q^m\} \subset \mathbb{G}_s^n$. Note that it is efficient to compute $[\mathbf{AB}]_s$ given $([\mathbf{A}]_s, \mathbf{B})$ or $(\mathbf{A}, [\mathbf{B}]_s)$ with matching dimensions. We define $[\mathbf{A}]_1 \circ [\mathbf{B}]_2 := e([\mathbf{A}]_1, [\mathbf{B}]_2) = [\mathbf{AB}]_T$, which can be efficiently computed given $[\mathbf{A}]_1$ and $[\mathbf{B}]_2$.

Next we recall the definition of the matrix Diffie-Hellman (MDDH) and related assumptions [EHK+13].

Definition 2.1 (Matrix distribution). Let $k, \ell \in \mathbb{N}$ with $\ell > k$. We call $\mathcal{D}_{\ell,k}$ a *matrix distribution* if it outputs matrices in $\mathbb{Z}_q^{\ell\times k}$ of full rank k in polynomial time.

Without loss of generality, we assume the first k rows of $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}$ form an invertible matrix. The $\mathcal{D}_{\ell,k}$-matrix Diffie-Hellman problem is to distinguish the two distributions $([\mathbf{A}], [\mathbf{Aw}])$ and $([\mathbf{A}], [\mathbf{u}])$ where $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}$, $\mathbf{w} \xleftarrow{\$} \mathbb{Z}_q^k$ and $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^\ell$.

Definition 2.2 ($\mathcal{D}_{\ell,k}$-matrix Diffie-Hellman assumption). Let $\mathcal{D}_{\ell,k}$ be a matrix distribution and $s \in \{1, 2, T\}$. We say that the *$\mathcal{D}_{\ell,k}$-matrix Diffie-Hellman* ($\mathcal{D}_{\ell,k}$-MDDH) *assumption* holds relative to GGen in group $\mathbb{G}_s$ if for all PPT adversaries $\mathcal{A}$, it holds that

$$\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{D}_{\ell,k},\mathsf{PGGen},s}(\mathcal{A}) := |\Pr[\mathcal{A}(\mathcal{G}, [\mathbf{A}]_s, [\mathbf{Aw}]_s) = 1] - \Pr[\mathcal{A}(\mathcal{G}, [\mathbf{A}]_s, [\mathbf{u}]_s) = 1]|$$

is negligible where the probability is taken over $\mathcal{G} \xleftarrow{\$} \mathsf{GGen}(1^\lambda)$, $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}, \mathbf{w} \xleftarrow{\$} \mathbb{Z}_q^k$ and $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^\ell$.

The uniform distribution is a particular matrix distribution that deserves special attention, as an adversary breaking the $\mathcal{U}_{\ell,k}$ assumption can also distinguish between real MDDH tuples and random tuples for all other possible matrix distributions. For uniform distributions, they stated in [GHKW16] that $\mathcal{U}_k$-MDDH and $\mathcal{U}_{\ell,k}$-MDDH assumptions are equivalent.

Definition 2.3 (Uniform distribution). Let $k, \ell \in \mathbb{N}_+$ with $\ell > k$. We call $\mathcal{U}_{\ell,k}$ a *uniform distribution* if it outputs uniformly random matrices in $\mathbb{Z}_q^{\ell\times k}$ of rank k in polynomial time. Let $\mathcal{U}_k := \mathcal{U}_{k+1,k}$.

Lemma 2.4 ($\mathcal{U}_{\ell,k}$-$\mathsf{MDDH} \Leftrightarrow \mathcal{U}_k$-$\mathsf{MDDH}$ [GHKW16]). *Let $\ell, k \in \mathbb{N}_+$ with $\ell > k$. An $\mathcal{U}_{\ell,k}$-MDDH instance is as hard as an $\mathcal{U}_k$-MDDH instance. More precisely, for each adversary $\mathcal{A}$ there exists an adversary $\mathcal{B}$ and vice versa with*

$$\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_{\ell,k},\mathsf{PGGen},s}(\mathcal{A}) = \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},s}(\mathcal{B})$$

and $T(\mathcal{A}) \approx T(\mathcal{B})$.

Lemma 2.5 ($\mathcal{D}_{\ell,k}$-$\mathsf{MDDH} \Rightarrow \mathcal{U}_k$-$\mathsf{MDDH}$ [EHK+13]). *Let $\ell, k \in \mathbb{N}_+$ with $\ell > k$ and let $\mathcal{D}_{\ell,k}$ be a matrix distribution. A $\mathcal{U}_k$-MDDH instance is at least as hard as an $\mathcal{D}_{\ell,k}$ instance. More precisely, for each adversary $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},s}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{D}_{\ell,k},\mathsf{PGGen},s}(\mathcal{B})$$

and $T(\mathcal{A}) \approx T(\mathcal{B})$.

For $Q \in \mathbb{N}$, $\mathbf{W} \xleftarrow{\$} \mathbb{Z}_q^{k\times Q}, \mathbf{U} \xleftarrow{\$} \mathbb{Z}_q^{\ell\times Q}$, consider the Q-fold $\mathcal{D}_{\ell,k}$-MDDH problem which is distinguishing the distributions $([\mathbf{A}], [\mathbf{AW}])$ and $([\mathbf{A}], [\mathbf{U}])$. That is, the Q-fold $\mathcal{D}_{\ell,k}$-MDDH problem contains Q independent instances of the $\mathcal{D}_{\ell,k}$-MDDH problem (with the same $\mathbf{A}$ but different $\mathbf{w}_i$). By a hybrid argument, one can show that the two problems are equivalent, where the reduction loses a factor Q. The following lemma gives a tight reduction.

Lemma 2.6 (Random self-reducibility [EHK+13]). *For $\ell > k$ and any matrix distribution $\mathcal{D}_{\ell,k}$, the $\mathcal{D}_{\ell,k}$-MDDH assumption is random self-reducible. In particular, for any $Q \geq 1$ and any adversary $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$(\ell - k)\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{D}_{\ell,k},\mathsf{PGGen},s}(\mathcal{A}) + \frac{1}{q-1} \geq \mathsf{Adv}^{Q\text{-}\mathsf{mddh}}_{\mathcal{D}_{\ell,k},\mathsf{PGGen},s}(\mathcal{B}) := |\Pr[\mathcal{B}(\mathcal{G},[\mathbf{A}],[\mathbf{AW}] \Rightarrow 1)] - \Pr[\mathcal{B}(\mathcal{G},[\mathbf{A}],[\mathbf{U}] \Rightarrow 1)]|\,,$$

where $\mathcal{G} \xleftarrow{\$} \mathsf{GGen}(1^\lambda)$, $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}$, $\mathbf{W} \xleftarrow{\$} \mathbb{Z}_q^{k\times Q}$, $\mathbf{U} \xleftarrow{\$} \mathbb{Z}_q^{(k+1)\times Q}$, and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ for a polynomial poly independent of $\mathcal{A}$.

In [LP19] we mistakenly assumed that the Q-fold $\mathcal{U}_{\ell,k}$-MDDH assumption is tightly equivalent to the Q-fold $\mathcal{U}_k$-MDDH assumption. However, Lemma 2.4 is only applicable for standard MDDH, not for Q-fold MDDH. So when reducing Q-fold $\mathcal{U}_{\ell,k}$-MDDH to $\mathcal{U}_k$-MDDH we have to apply Lemma 2.6 to get from Q-fold $\mathcal{U}_{\ell,k}$-MDDH to standard $\mathcal{U}_{\ell,k}$-MDDH and then Lemma 2.4 to get from $\mathcal{U}_{\ell,k}$-MDDH to $\mathcal{U}_k$-MDDH. Thus for every adversary $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with

$$\mathsf{Adv}^{Q\text{-}\mathsf{mddh}}_{\mathcal{U}_{\ell,k},\mathsf{PGGen},s}(\mathcal{A}) \leq (\ell - k)\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},s}(\mathcal{B}) + \frac{1}{q-1}\,.$$

In our previous publication [LP19] (and various other publications, e.g. [GHKW16]) the factor $\ell - k$ is missing, because we mistakenly applied Lemma 2.4 to Q-fold MDDH instances. However, the proof of Lemma 2.4 given in [GHKW16] fails here: Suppose $(\mathcal{PG},[\mathbf{A}],[\mathbf{U}])$ is a uniform Q-fold $\mathcal{U}_k$-MDDH instance, then $(\mathcal{PG},[\mathbf{T}'\mathbf{A}],[\mathbf{T}'\mathbf{U}])$ (for uniformly random full-rank $\mathbf{T}' \in \mathbb{Z}_q^{\ell\times(k+1)}$) is not a uniform Q-fold $\mathcal{U}_{\ell,k}$-MDDH instance. The $\mathcal{U}_k$-MDDH instance contains $(Q+k)(k+1)$ random elements (from $\mathbb{Z}_q$), so the transformed instance contains at most $(Q+\ell+k)(k+1)$ random elements. However, a uniform Q-fold $\mathcal{U}_{\ell,k}$-MDDH instance requires $(Q+k)\ell$ random elements.

2.2 Hierarchical identity-based key encapsulation

We recall syntax and security of a hierarchical identity-based key encapsulation mechanism (HIBKEM). We only consider HIBKEM in this paper. By adapting the transformation for public-key encryption in [HK07] to the HIBE setting, one can easily prove that every HIBKEM can be transformed (tightly) into an HIBE scheme with a (one-time secure) symmetric cipher.

Definition 2.7 (Hierarchical identity-based key encapsulation mechanism). A *hierarchical identity-based key encapsulation mechanism* (HIBKEM) HIBKEM consists of five polynomial time algorithms $\mathsf{HIBKEM} := (\mathsf{Gen}, \mathsf{Del}, \mathsf{Ext}, \mathsf{Enc}, \mathsf{Dec})$ with the following properties.

- The probabilistic key generation algorithm $\mathsf{Gen}(1^\lambda)$ returns the (master) public/delegation/secret key $(\mathsf{pk}, \mathsf{dk}, \mathsf{sk})$. Note that for some constructions dk is empty. We assume that pk implicitly defines a hierarchical identity space $\mathcal{ID} = \mathcal{S}^{\leq L}$, for some base identity set $\mathcal{S}$, a key space $\mathcal{K}$ and a ciphertext space $\mathcal{C}$.
- The probabilistic user secret key generation algorithm $\mathsf{Ext}(\mathsf{sk}, \mathsf{id})$ returns a secret key $\mathsf{usk}[\mathsf{id}]$ and a delegation key $\mathsf{udk}[\mathsf{id}]$ for a hierarchical identity $\mathsf{id} \in \mathcal{ID}$. Note that for some constructions $\mathsf{udk}[\mathsf{id}]$ is empty.
- The probabilistic key delegation algorithm $\mathsf{Del}(\mathsf{dk}, \mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}], \mathsf{id} \in \mathcal{S}^p, \mathsf{id}_{p+1} \in \mathcal{S})$ returns a user secret key $\mathsf{usk}[\mathsf{id}|\mathsf{id}_{p+1}]$ for the hierarchical identity $\mathsf{id}' = \mathsf{id} \mid \mathsf{id}_{p+1} \in \mathcal{S}^{p+1}$ and the user delegation key $\mathsf{udk}[\mathsf{id}']$. We require $1 \leq |\mathsf{id}| \leq L - 1$.
- The probabilistic encapsulation algorithm $\mathsf{Enc}(\mathsf{pk}, \mathsf{id})$ returns a symmetric key $\mathsf{K} \in \mathcal{K}$ together with a ciphertext C with respect to the hierarchical identity $\mathsf{id} \in \mathcal{ID}$.
- The deterministic decapsulation algorithm $\mathsf{Dec}(\mathsf{usk}[\mathsf{id}], \mathsf{id}, \mathsf{C})$ returns a decapsulated key $\mathsf{K} \in \mathcal{K}$ or the reject symbol $\perp$.

In the HIBKEM definition we make the delegation key dk explicit to make our constructions more readable.

Definition 2.8 (Delegation Invariance). An HIBKEM $\mathsf{HIBKEM} := (\mathsf{Gen}, \mathsf{Del}, \mathsf{Ext}, \mathsf{Enc}, \mathsf{Dec})$ is *delegation invariant*, if the distribution of $\mathsf{usk}[\mathsf{id}|\mathsf{id}_{p+1}]$ generated by $\mathsf{Del}(\mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}], \mathsf{id}, \mathsf{id}_{p+1})$ for any valid user secret key $\mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}]$ for id is independent of $\mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}]$ and identical to the distribution of keys generated by $\mathsf{Ext}(\mathsf{sk}, \mathsf{id}|\mathsf{id}_{p+1})$.

We focus in this paper only on HIBKEM schemes with delegation invariance. The following definitions of correctness and security are only suitable for delegation invariant schemes. For general HIBKEMs, a more involved definition that takes the Del algorithm into account is necessary (see [SW08]).

Definition 2.9 (Correctness). A delegation invariant HIBKEM $\mathsf{HIBKEM} := (\mathsf{Gen}, \mathsf{Del}, \mathsf{Ext}, \mathsf{Enc}, \mathsf{Dec})$ is *correct*, if for all $\lambda \in \mathbb{N}_+$, all pairs $(\mathsf{pk}, \mathsf{sk})$ generated by $\mathsf{Gen}(\lambda)$, all $\mathsf{id} \in \mathcal{ID}$, all $\mathsf{usk}[\mathsf{id}]$ generated by $\mathsf{Ext}(\mathsf{sk}, \mathsf{id})$ and all (K, c) generated by $\mathsf{Enc}(\mathsf{pk}, \mathsf{id})$:

$$\Pr[\mathsf{Dec}(\mathsf{usk}[\mathsf{id}], \mathsf{id}, \mathsf{C}) = \mathsf{K}] = 1.$$

We define indistinguishability ($\mathsf{IND\text{-}HID\text{-}CPA}$) against adaptively chosen identity and plaintext attacks for a HIBKEM via games $\mathsf{IND\text{-}HID\text{-}CPA}_{\mathsf{real}}$ and $\mathsf{IND\text{-}HID\text{-}CPA}_{\mathsf{rand}}$ from Figure 2.

```
INIT:
$(\mathsf{pk}, \mathsf{sk}, \mathsf{dk}) \xleftarrow{\$} \mathsf{Gen}(\lambda)$
return $(\mathsf{pk}, \mathsf{dk})$

EXT(id):
$\mathcal{Q}_{\mathcal{ID}} \leftarrow \mathcal{Q}_{\mathcal{ID}} \cup \{\mathsf{id}\}$
return $(\mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}]) \xleftarrow{\$} \mathsf{Ext}(\mathsf{sk}, \mathsf{id})$

ENC(id*):                                    //one query
$(\mathsf{K}^\star, \mathsf{C}^\star) \xleftarrow{\$} \mathsf{Enc}(\mathsf{pk}, \mathsf{id}^\star)$
$\boxed{\mathsf{K}^\star \xleftarrow{\$} \mathcal{K}}$
return $(\mathsf{K}^\star, \mathsf{C}^\star)$

FINALIZE($\beta \in \{0, 1\}$):
return $(\mathsf{Prefix}(\mathsf{id}^*) \cap \mathcal{Q}_{\mathcal{ID}} \overset{?}{=} \emptyset \wedge \beta$
```

Figure 2: Games $\mathsf{IND\text{-}HID\text{-}CPA}_{\mathsf{real}}$ and $\boxed{\mathsf{IND\text{-}HID\text{-}CPA}_{\mathsf{rand}}}$ for defining $\mathsf{IND\text{-}HID\text{-}CPA}$ security. For any identity $\mathsf{id} \in \mathcal{S}^p$, $\mathsf{Prefix}(\mathsf{id})$ denotes the set of all prefixes of id

Definition 2.10 ($\mathsf{IND\text{-}HID\text{-}CPA}$ security). A hierarchical identity-based key encapsulation scheme HIBKEM is $\mathsf{IND\text{-}HID\text{-}CPA}$-*secure* if for all PPT $\mathcal{A}$,

$$\mathsf{Adv}^{\mathsf{ind\text{-}hid\text{-}cpa}}_{\mathsf{HIBKEM}}(\mathcal{A}) := |\Pr[\mathsf{IND\text{-}HID\text{-}CPA}^{\mathcal{A}}_{\mathsf{real}} \Rightarrow 1] - \Pr[\mathsf{IND\text{-}HID\text{-}CPA}^{\mathcal{A}}_{\mathsf{rand}}]|$$

is negligible.

2.3 Collision resistant hash functions

Some constructions in this paper make use of collision resistant hash functions.

Definition 2.11 (Hash function). A *family of hash functions* is a tuple $\mathcal{H} := (\mathsf{HGen}, \mathsf{HEval})$ of polynomial time algorithms with:

- HGen is a probabilistic algorithm that gets the security parameter 1^λ and returns a (public) hash key K.
- HEval is a deterministic algorithm that gets a hash key K and an input $X \in \mathcal{D}_K$ and outputs a hash $\mathsf{HEval}_K(X) \in \mathcal{R}_K$, where $\mathcal{D}_K$ is the domain set and $\mathcal{R}_K$ is the finite range set.

The security notion we require for the hash functions is collision resistance.

Definition 2.12 (Collision resistance). A family of hash functions $\mathcal{H} := (\mathsf{HGen}, \mathsf{HEval})$ is *collision-resistant* if for all PPT adversaries $\mathcal{A}$,

$$\mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H}}(\mathcal{A}) := \Pr\big[X_1 \neq X_2 \wedge \mathsf{HEval}_K(K, X_1) = \mathsf{HEval}_K(K, X_2) \mid (X_1, X_2) \xleftarrow{\$} \mathcal{A}\big(1^\lambda, K\big), K \xleftarrow{\$} \mathsf{HGen}\big(1^\lambda\big)\big]$$

is negligible in λ.

3 Affine MAC with levels

The core of our HIBE constructions is a Message Authentication Code with suitable algebraic structures, and we call it affine MAC with levels. This is a generalization of the delegatable, affine MAC used in [BKP14], namely, a delegatable, affine MAC is affine MAC with levels with $\ell(p) = 1$ for all $p \in \{1, \ldots L\}$.

Definition 3.1 (Affine MAC with levels). An *affine MAC with levels* MAC consists of three PPT algorithms $(\mathsf{Gen}_{\mathsf{MAC}}, \mathsf{Tag}, \mathsf{Ver}_{\mathsf{MAC}})$ with the following properties:

```
Init:
$\mathsf{sk}_{\mathsf{MAC}} \xleftarrow{\$} \mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2)$
Parse $\mathsf{sk}_{\mathsf{MAC}} =: \left(\mathbf{B}, (\mathbf{x}_{l,i,j})_{1\le l\le \ell(L), 1\le i\le L, 1\le j\le \ell'(l,i)}, x'_0\right)$
$\mathsf{dk} := \left(\left[\mathbf{x}_{l,i,j}^\top \mathbf{B}\right]_2\right)_{1\le l\le \ell(L), 1\le i\le L, 1\le j\le \ell'(l,i)}$
return $([\mathbf{B}]_2, \mathsf{dk})$

Eval($\mathsf{m} \in \mathcal{S}^p$):
$\mathcal{Q}_{\mathcal{M}} = \mathcal{Q}_{\mathcal{M}} \cup \{\mathsf{m}\}$
$\left(([\mathbf{t}_l]_2)_{1\le l\le \ell(p)}, [u]_2\right) \xleftarrow{\$} \mathsf{Tag}(\mathsf{sk}, \mathsf{m})$
for $l \in \{1, \dots, \ell(p)\}$, $i \in \{p+1, \dots, L\}$, $j \in \{1, \dots$
$\dots, \ell'(l,i)\}$ do $d_{l,i,j} = \mathbf{x}_{l,i,j}^\top \mathbf{t}_l$
$\mathsf{tdk} := \left([d_{l,i,j}]_2\right)_{1\le l\le \ell(p), p+1\le i\le L, 1\le j\le \ell'(l,i)}$
return $\left(([\mathbf{t}_l]_2)_{1\le l\le \ell(p)}, [u]_2, \mathsf{tdk}\right)$

Chal($\mathsf{m}^\star \in \mathcal{S}^p$):                // one query
$h \xleftarrow{\$} \mathbb{Z}_q$
for $l \in \{1, \dots, \ell(p)\}$ do
    $\mathbf{h}_{0,l} := \left(\sum_{i=1}^{L} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}\left(\mathsf{m}^\star_{|i}\right)\mathbf{x}_{l,i,j}\right) h$
$h_1 = x'_0 \cdot h \in \mathbb{Z}_q$
$\boxed{h_1 \xleftarrow{\$} \mathbb{Z}_q}$
return $\left([h]_1, ([\mathbf{h}_{0,l}]_1)_{1\le l\le \ell(p)}, [h_1]_T\right)$

Finalize($\beta \in \{0,1\}$):
return $\beta \wedge (\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} \overset{?}{=} \emptyset$
```

Figure 3: Games $\mathsf{HPR_0\text{-}CMA_{real}}$, and $\boxed{\mathsf{HPR_0\text{-}CMA_{rand}}}$ for defining $\mathsf{HPR_0\text{-}CMA}$ security for affine MACs with levels

- $\mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2)$ gets a description of a prime-order group $(\mathbb{G}_2, q, P_2)$ and returns a secret key $\mathsf{sk}_{\mathsf{MAC}} := \left(\mathbf{B}, (\mathbf{x}_{l,i,j})_{1\le l\le \ell(L), 1\le i\le L, 1\le j\le \ell'(l,i)}, x'_0\right)$ where $\mathbf{B} \in \mathbb{Z}_q^{n\times n'}$, $\mathbf{x}_{l,i,j} \in \mathbb{Z}_q^n$ for $l \in \{1, \dots, \ell(L)\}$, $i \in \{1, \dots, L\}$, and $j \in \{0, \dots, \ell'(l,i)\}$ and $x'_0 \in \mathbb{Z}_q$.
- $\mathsf{Tag}\left(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^{p\le L}\right)$ returns a tag $\tau := \left(([\mathbf{t}_l]_2)_{1\le l\le \ell(p)}, [u]_2\right)$ where

$$\mathbf{t}_l := \mathbf{B}\mathbf{s}_l \quad \text{for } \mathbf{s}_l \xleftarrow{\$} \mathbb{Z}_q^{n'} \quad (1 \le l \le \ell(p))$$
$$u := \sum_{l=1}^{\ell(p)} \left(\sum_{i=1}^{p} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}(\mathsf{m}_{|i})\mathbf{x}_{l,i,j}^\top\right) \mathbf{t}_l + x'_0 \,. \tag{4}$$

- $\mathsf{Ver}_{\mathsf{MAC}}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m}, \tau = ([\mathbf{t}]_2, [u]_2))$ checks, whether Equation (4) holds.

The messages of MAC have the form $\mathsf{m} = (\mathsf{m}_1, \dots, \mathsf{m}_p)$ where $p \le L$ and $\mathsf{m}_i \in \mathcal{S}$. After the transformation to an HIBE, $\mathcal{S}$ will be the base set of the identity space and L will be the maximum number of levels. The functions $f_{l,i,j} : \mathcal{S}^i \to \mathbb{Z}_q$ must be public, efficiently computable functions. The parameters $\ell : \{1, \dots, p\} \to \mathbb{N}_+$, $n, n' \in \mathbb{N}_+$ and $\ell' : \{1, \dots, p\} \times \{1, \dots, L\} \to \mathbb{N}_+$ $(1 \le i \le L)$ are arbitrary, scheme-depending parameters. The function ℓ must be monotonous increasing.

Security. We require hierarchical pseudorandomness against chosen-message attacks ($\mathsf{HPR_0\text{-}CMA}$-security) for affine MACs with levels. This is a generalization of the $\mathsf{HPR_0\text{-}CMA}$-security for delegatable affine MACs defined in [BKP14].The security is defined by games in Figure 3.

Definition 3.2 ($\mathsf{HPR_0\text{-}CMA}$ security). An affine MAC with levels is $\mathsf{HPR_0\text{-}CMA}$-*secure* in $\mathbb{G}_2$ if for all PPT adversaries $\mathcal{A}$ the function

$$\mathsf{Adv}^{\mathsf{hpr_0\text{-}cma}}_{\mathsf{MAC},\mathbb{G}_2}(\mathcal{A}) := \left|\Pr\left[\mathsf{HPR_0\text{-}CMA}^{\mathcal{A}}_{\mathsf{real}} \Rightarrow 1\right] - \Pr\left[\mathsf{HPR_0\text{-}CMA}^{\mathcal{A}}_{\mathsf{rand}} \Rightarrow 1\right]\right|$$

is negligible.

3.1 The first construction

Let $(\mathbb{G}_2, q, P_2)$ be a group of prime order q. The first affine MAC with levels $\mathsf{MAC}_1[\mathcal{U}_{3k,k}] := (\mathsf{Gen}_{\mathsf{MAC}}, \mathsf{Tag}, \mathsf{Ver}_{\mathsf{MAC}})$ with message space $\mathcal{ID} := \mathcal{S}^{\le L} := (\{0,1\}^\alpha)^{\le L}$ is defined in Figure 4. The identity vectors bit-length α and the maximum length L of the identity vectors can be chosen freely.[3] The resulting HIBE from this MAC has constant ciphertext length.

$\mathsf{MAC}_1[\mathcal{U}_{3k,k}]$ has $n := 3k$ and $n' := k$ where $k \in \mathbb{N}_+$ can be chosen arbitrary. To match the formal definition, $\mathbf{x}_{i,j,b}$ should be renamed to $\mathbf{x}_{i,2j-b}$ and $f_{i,2j-b}(\mathsf{m}_{|i}) := \left(\left[\!\left[\mathsf{m}_{|i}\right]\!\right]_j \overset{?}{=} b\right)$. Then we get $\ell(p) = 1$ and $\ell'(1,i) = 2i\alpha$.

[3] A different bit-length on each level is possible as well, but we assume it is α on each level to ease notation.

```
$\underline{\mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2):}$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
for $i \in \{1, \dots, L\},\ j \in \{1, \dots, i\alpha\},\ b \in \{0,1\}$ do $\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$x'_0 \xleftarrow{\$} \mathbb{Z}_q$
return $\mathsf{sk}_{\mathsf{MAC}} := \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le i\alpha, b \in \{0,1\}}, x'_0\big)$

$\underline{\mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p):}$
Parse $\mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le i\alpha, b \in \{0,1\}}, x'_0\big)$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{B}\mathbf{s}$
$u := \Big(\sum_{i=1}^{p} \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top\Big)\mathbf{t} + x'_0$
return $\tau := \big([\mathbf{t}]_2, [u]_2\big)$

$\underline{\mathsf{Ver}_{\mathsf{MAC}}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p, \tau):}$
Parse $\mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le i\alpha, b \in \{0,1\}}, x'_0\big)$
Parse $\mathsf{m} =: (\mathsf{m}_1, \dots, \mathsf{m}_p)$
Parse $\tau =: \big([\mathbf{t}]_2, [u]_2\big)$
return $u \stackrel{?}{=} \Big(\sum_{i=1}^{p} \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top\Big)\mathbf{t} + x'_0$
```

Figure 4: The first affine MAC

Theorem 3.3 (Security of $\mathsf{MAC}_1[\mathcal{U}_{3k,k}]$). $\mathsf{MAC}_1[\mathcal{U}_{3k,k}]$ *is tightly* HPR_0-CMA *secure in* $\mathbb{G}_2$ *under the* $\mathcal{U}_k$-MDDH *assumption for* $\mathbb{G}_2$*. More precisely, for all adversaries* $\mathcal{A}$ *there exists an adversary* $\mathcal{B}$ *with*

$$\mathsf{Adv}^{\mathsf{hpr}_0\text{-}\mathsf{cma}}_{\mathsf{MAC}_1[\mathcal{U}_{3k,k}],\mathbb{G}_2}(\mathcal{A}) \le \big(8k(\alpha+1)L + 8k\alpha L^2\big)\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{4(\alpha+1)L + 4\alpha L^2}{q-1} + \frac{2Q}{q^{2k}}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$*, where* Q *denotes the number of* Eval *queries of* $\mathcal{A}$ *and* poly *is a polynomial independent of* $\mathcal{A}$*.*

Proof. The proof uses a hybrid argument with the hybrids G_0 (the HPR_0-$\mathsf{CMA}_{\mathsf{real}}$ game), G_1, for $\hat{\imath} \in \{1, \dots \dots, L\}$ $\mathsf{G}_{2,\hat{\imath},0}$, $\mathsf{G}_{2,\hat{\imath},1}$, $\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},1}$ for $\hat{\jmath} \in \{0, \dots, \hat{\imath}\alpha\}$, $\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},1}$–$\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},3}$ for $\hat{\jmath} \in \{0, \dots, \hat{\imath}\alpha - 1\}$, $\mathsf{G}_{2,\hat{\imath},3}$, $\mathsf{G}_{2,\hat{\imath},4}$, $\mathsf{G}_{2,\hat{\imath},5}$ and finally G_3. The hybrids are given in Figure 5 and 6. A summary can be found in Table 2. They make use of random functions $\mathsf{RF}_{\hat{\imath},\hat{\jmath}} : \{0,1\}^{\hat{\jmath}} \to \mathbb{Z}_q^{1 \times 2k}$, $\mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}} : \{0,1\}^{\hat{\jmath}} \to \mathbb{Z}_q^{1 \times k}$, and $\mathsf{RF}^{(1)}_{\hat{\imath},\hat{\jmath}} : \{0,1\}^{\hat{\jmath}} \to \mathbb{Z}_q^{1 \times k}$, defined on-the-fly.

Lemma 3.4 ($\mathsf{G}_0 \rightsquigarrow \mathsf{G}_1$).

$$\Pr\big[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1\big] = \Pr\big[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\big]$$

Proof. In game G_1 each time the adversary queries a tag for a message m where he queried a tag for m before, the adversary will get a rerandomized version of the first tag he queried. The rerandomized tag is identically distributed to a fresh tag: $\mathbf{t}' := \mathbf{t} + \mathbf{B}\mathbf{s}'$ is uniformly random in $\mathsf{Span}(\mathbf{B})$, when $\mathbf{s}'$ is uniform random in $\mathbb{Z}_q^k$. Together with $u' := u + \sum_{i=1}^{p}\Big(\sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top \mathbf{B}\mathbf{s}'\Big)$ we get a valid message tag for m, when $([\mathbf{t}]_2, [u]_2)$ is a valid tag for m.

Note that the rerandomization uses only the "public key" returned by the Init oracle, so it could actually be carried out by the adversary herself. To put it in a nutshell, repeated Eval queries for a message m will leak no information, that is not already leaked by the first Eval query for m or by the "public key".[4] □

Lemma 3.5 ($\mathsf{G}_1 \rightsquigarrow \mathsf{G}_{2,1,0}$).

$$\Pr\big[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\big] = \Pr\big[\mathsf{G}_{2,1,0}^{\mathcal{A}} \Rightarrow 1\big]$$

Proof. These two games are equivalent. □

[4] The same technique can be used to prove the IBE of [BKP14] secure with duplicated Eval-queries. Thus they work without a pseudorandom function.

```
$\mathsf{G}_0$  $\mathsf{G}_1$  $\mathsf{G}_{2,\hat\imath,0}$  $\mathsf{G}_{2,\hat\imath,1}$  $\mathsf{G}_{2,\hat\imath,3}$  $\mathsf{G}_{2,\hat\imath,4}$  $\mathsf{G}_{2,\hat\imath,5}$  $\mathsf{G}_3$

INIT:
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
for $i \in \{1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$
do $\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$x'_0 \xleftarrow{\$} \mathbb{Z}_q$
$K : \mathcal{S}^{\le L} \dashrightarrow \mathbb{G}_2^{3k+1} \times \mathbb{G}_2^{\le\alpha(L^2+L)}$
$\mathsf{dk} = \left(\left[\mathbf{x}_{i,j,b}^\top \mathbf{B}\right]_2\right)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}$
return $([\mathbf{B}]_2, \mathsf{dk})$

CHAL($\mathsf{m}^\star \in \mathcal{S}^p$):                    // one query
$h \xleftarrow{\$} \mathbb{Z}_q$
$\mathbf{h}_0 := \left(\sum_{i=1}^{L}\sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}^\star]\!]_j}\right) h$
if $p \ge \hat\imath$ then $\mathbf{h}_0 := \mathbf{h}_0 + \mathbf{B}^\perp \mathsf{RF}_{\hat\imath,\hat\jmath}\left(\mathsf{m}^\star_{|\hat\imath}\right)^\top h$
$h_1 := x'_0 \cdot h$
$h_1 \xleftarrow{\$} \mathbb{Z}_q$
return $([h]_1, [\mathbf{h}_0]_1, [h_1]_T)$

FINALIZE($\beta \in \{0,1\}$):
return $\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} \overset{?}{=} \emptyset \wedge \beta$

EVAL($\mathsf{m} \in \mathcal{S}^p$):
if $\mathsf{m} \in \mathcal{Q}_{\mathcal{M}}$ then
    return $\mathsf{RerandTag}(\mathsf{m}, K(\mathsf{m}))$
$\mathcal{Q}_{\mathcal{M}} := \mathcal{Q}_{\mathcal{M}} \cup \{\mathsf{m}\}$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t} := \mathbf{B}\mathbf{s}$
if $p = \hat\imath$ then $\mathbf{t} \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$u := \left(\sum_{i=1}^{p}\sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top\right)\mathbf{t} + x'_0$
if $p < \hat\imath$ then $u \xleftarrow{\$} \mathbb{Z}_q$
if $p = \hat\imath$ then $u \xleftarrow{\$} \mathbb{Z}_q$
$u \xleftarrow{\$} \mathbb{Z}_q$
for $i \in \{p+1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$
do $d_{i,j,b} := \mathbf{x}_{i,j,b}^\top \mathbf{t}$
$\mathsf{tdk} := \left([d_{i,j,b}]_2\right)_{p+1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}$
$K(\mathsf{m}) := ([\mathbf{t}]_2, [u]_2, \mathsf{tdk})$
return $K(\mathsf{m})$

RerandTag($\mathsf{m} \in \mathcal{S}^p, \tau$):
Parse $\tau =: ([\mathbf{t}]_2, [u]_2, \mathsf{tdk})$
Parse $\mathsf{tdk} =: \left([d_{i,j,b}]_2\right)_{p+1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}$
$\mathbf{s}' \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t}' := \mathbf{t} + \mathbf{B}\mathbf{s}'$
$u' := u + \sum_{i=1}^{p}\left(\sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top \mathbf{B}\mathbf{s}'\right)$
for $i \in \{p+1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$
do $d'_{i,j,b} := d_{i,j,b} + \mathbf{x}_{i,j,b}^\top \mathbf{B}\mathbf{s}'$
$\mathsf{tdk}' := \left(\left[d'_{i,j,b}\right]_2\right)_{p+1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}$
return $([\mathbf{t}']_2, [u']_2, \mathsf{tdk}')$
```

Figure 5: Hybrids for the security proof of $\mathsf{MAC}_1[\mathcal{U}_{3k,k}]$. The algorithm RerandTag is only helper function and not an oracle for the adversary. The partial map K is initially totally undefined

Lemma 3.6 ($\mathsf{G}_{2,\hat\imath,0} \rightsquigarrow \mathsf{G}_{2,\hat\imath,1}$). *For all $\hat\imath \in \{1,\dots,L\}$ and all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_{2,\hat\imath,0}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_{2,\hat\imath,1}^{\mathcal{A}} \Rightarrow 1\right]\right| \le 2k\mathsf{Adv}_{\mathcal{U}_k,\mathsf{PGGen},2}^{\mathsf{mddh}}(\mathcal{B}) + \frac{1}{q-1}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. These two games are equivalent except that in EVAL-queries with $p = \hat\imath$ the value $\mathbf{t}$ is chosen uniformly random from $\mathsf{Span}(\mathbf{B})$ in $\mathsf{G}_{2,\hat\imath,0}$ and uniformly random from $\mathbb{Z}_q^{3k}$ in game $\mathsf{G}_{2,\hat\imath,1}$. Since for all computed values it is enough to have $[\mathbf{B}]_2$ instead of $\mathbf{B}$, this leads to a straightforward reduction to the Q-fold $\mathcal{U}_{3k,k}$-MDDH assumption. Remember that by Lemma 2.4, the $\mathcal{U}_{3k,k}$-MDDH assumption is equivalent to the $\mathcal{U}_k$-MDDH assumption.

The running time of $\mathcal{B}$ is dominated by the running time of $\mathcal{A}$ plus some (polynomial) overhead that is independent of $T(\mathcal{A})$ for the group operations in each oracle query. □

Lemma 3.7 ($\mathsf{G}_{2,\hat\imath,1} \rightsquigarrow \mathsf{G}_{2,\hat\imath,3}$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_{2,\hat\imath,1}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_{2,\hat\imath,3}^{\mathcal{A}} \Rightarrow 1\right]\right| \le 8k\hat\imath\alpha\mathsf{Adv}_{\mathcal{U}_k,\mathsf{PGGen},2}^{\mathsf{mddh}}(\mathcal{B}) + \frac{4\hat\imath\alpha}{q-1} + \frac{Q_{\hat\imath}}{q^{2k}},$$

where $Q_{\hat\imath}$ denotes the number of EVAL queries with $p = \hat\imath$ and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. To prove this transition, we introduce new hybrids $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,0}$ for $\hat\jmath \in \{0,\dots,\hat\imath\alpha\}$ and $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,1}$–$\mathsf{G}_{2,\hat\imath,2,\hat\jmath,3}$ for $\hat\jmath \in \{0,\dots,\hat\imath\alpha - 1\}$. The hybrids are given in Figure 6.

Lemma 3.7 follows directly from Lemma 3.8–3.13. □

<table>
<tr><th>Hybrid</th><th>$\mathbf{t}$ uniform in</th><th>$r_u(\mathsf{m})$</th><th>$r_{\mathbf{h}_0}(\mathsf{m})$</th><th>Transition</th></tr>
<tr><td>G_0</td><td>$\mathsf{Span}(\mathbf{B})$</td><td colspan="2">$0$</td><td>Original game</td></tr>
<tr><td>G_1</td><td>$\mathsf{Span}(\mathbf{B})$</td><td colspan="2">$0$</td><td>Identical</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,0}$</td><td>$\mathsf{Span}(\mathbf{B})$</td><td colspan="2">$0$</td><td>Identical</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,1}$</td><td>$\mathbb{Z}_q^{3k}$</td><td colspan="2">$0$</td><td>$\mathcal{U}_k$-MDDH</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,2,\hat\jmath,0}$</td><td>$\mathbb{Z}_q^{3k}$</td><td colspan="2">$\mathsf{RF}_{\hat\imath,\hat\jmath}\left([\![\mathsf{m}]\!]_{\vert\hat\jmath}\right)\left(\mathbf{B}^{\perp}\right)^{\top}$</td><td>Identical</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,2,\hat\jmath,1}$</td><td rowspan="3"><b>if</b> $[\![\mathsf{m}]\!]_{\hat\jmath+1} = 0$ <b>then</b>
$\vert\,\mathsf{Span}(\mathbf{B}\vert\mathbf{B}_0)$
<b>else</b>
$\lfloor\mathsf{Span}(\mathbf{B}\vert\mathbf{B}_1)$</td><td colspan="2">$\mathsf{RF}_{\hat\imath,\hat\jmath}\left([\![\mathsf{m}]\!]_{\vert\hat\jmath}\right)\left(\mathbf{B}^{\perp}\right)^{\top}$</td><td>$\mathcal{U}_k$-MDDH</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,2,\hat\jmath,2}$</td><td colspan="2">$\left(\mathsf{RF}^{(0)}_{\hat\imath,\hat\jmath+1}\left([\![\mathsf{m}]\!]_{\vert\hat\jmath+1}\right)\left(\mathbf{B}_0^{*}\right)^{\top} + \mathsf{RF}^{(1)}_{\hat\imath,\hat\jmath}\left([\![\mathsf{m}]\!]_{\vert\hat\jmath}\right)\left(\mathbf{B}_1^{*}\right)^{\top}\right)$</td><td>Identical</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,2,\hat\jmath,3}$</td><td colspan="2">$\left(\mathsf{RF}^{(0)}_{\hat\imath,\hat\jmath+1}\left([\![\mathsf{m}]\!]_{\vert\hat\jmath+1}\right)\left(\mathbf{B}_0^{*}\right)^{\top} + \mathsf{RF}^{(1)}_{\hat\imath,\hat\jmath+1}\left([\![\mathsf{m}]\!]_{\vert\hat\jmath+1}\right)\left(\mathbf{B}_1^{*}\right)^{\top}\right)$</td><td>Identical</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,2,\hat\jmath+1,0}$</td><td>$\mathbb{Z}_q^{3k}$</td><td colspan="2">$\mathsf{RF}_{\hat\imath,\hat\jmath+1}\left([\![\mathsf{m}]\!]_{\vert\hat\jmath+1}\right)\left(\mathbf{B}^{\perp}\right)^{\top}$</td><td>$\mathcal{U}_k$-MDDH</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,3}$</td><td>$\mathbb{Z}_q^{3k}$</td><td>uniform random</td><td>$\mathsf{RF}_{\hat\imath}\left(\mathsf{m}_{\vert\hat\imath}\right)\left(\mathbf{B}^{\perp}\right)^{\top}$</td><td>Statistically close</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,4}$</td><td>$\mathbb{Z}_q^{3k}$</td><td>uniform random</td><td>0</td><td>Identical</td></tr>
<tr><td>$\mathsf{G}_{2,\hat\imath,5}$</td><td>$\mathsf{Span}(\mathbf{B})$</td><td>uniform random</td><td>0</td><td>$\mathcal{U}_k$-MDDH</td></tr>
<tr><td>G_3</td><td>$\mathsf{Span}(\mathbf{B})$</td><td>uniform random</td><td>0</td><td>Statistically close</td></tr>
</table>

Table 2: Summary of the hybrids in Figure 5 and 6. Non-duplicated Eval queries with $p = \hat\imath$ draw $\mathbf{t}$ from the set described by the second column and add the randomness $r_u(\mathsf{m})\mathbf{t}$ to u or choose u uniform random. The Chal query adds the term $r_{\mathbf{h}_0}(\mathsf{m}^{\star})^{\top} h$ to $\mathbf{h}_0$ if $\mathsf{m}^{\star}$ has length $\geq \hat\imath$. The column "Transition" displays how we can switch to this hybrid from the previous one. The background colors indicate repeated transitions

Lemma 3.8 ($\mathsf{G}_{2,\hat\imath,1} \rightsquigarrow \mathsf{G}_{2,\hat\imath,2,0,0}$).

$$\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,1} \Rightarrow 1\right] = \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,2,0,0} \Rightarrow 1\right]$$

Proof. These two games are equivalent. When changing in $\mathsf{G}_{2,\hat\imath,1}$ the secret values $\mathbf{x}_{\hat\imath,1,b}$ to $\mathbf{x}_{\hat\imath,1,b} + \mathbf{B}^{\perp}(\mathsf{RF}_{\hat\imath,0}(\varepsilon))^{\top}$ (for $b \in \{0,1\}$), we get game $\mathsf{G}_{2,\hat\imath,2,0,0}$. The distribution of $\mathbf{x}_{\hat\imath,1,b}$ and $\mathbf{x}_{\hat\imath,1,b} + \mathbf{B}^{\perp}(\mathsf{RF}_{\hat\imath,0}(\varepsilon))^{\top}$ is identical. Note that the term $\mathbf{B}^{\perp}(\mathsf{RF}_{\hat\imath,0}(\varepsilon))^{\top}$ cancels out in the master public key and in the user delegation keys of Eval-queries with $p < \hat\imath$. □

Lemma 3.9 ($\mathsf{G}_{2,\hat\imath,2,\hat\jmath,0} \rightsquigarrow \mathsf{G}_{2,\hat\imath,2,\hat\jmath,1}$). *For all $\hat\jmath \in \{0,\ldots,\hat\imath\alpha\}$ and all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,2,\hat\jmath,0} \Rightarrow 1\right] - \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,2,\hat\jmath,1} \Rightarrow 1\right]\right| \leq 4k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{2}{q-1}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. These two games are equivalent except that the value $\mathbf{t}$ is generated uniformly random from $\mathbb{Z}_q^{3k}$ in game $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,0}$ and from either $\mathsf{Span}(\mathbf{B}|\mathbf{B}_0)$ or $\mathsf{Span}(\mathbf{B}|\mathbf{B}_1)$ depending on the bit $[\![\mathsf{m}]\!]_{\hat\jmath+1}$ in game $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,1}$. We can switch from $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,0}$ to $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,1}$ with two Q-fold $\mathcal{U}_{3k,k}$-MDDH challenges. Remember that the $\mathcal{U}_{3k,k}$-MDDH assumption is equivalent to the $\mathcal{U}_k$-MDDH assumption by Lemma 2.4.

To achieve that, we first switch $\mathbf{t}$ for $[\![\mathsf{m}]\!]_{\hat\jmath+1} = 0$ from a random vector in $\mathbb{Z}_q^{3k}$ to $\mathbf{t} := \mathbf{B}\mathbf{s}_1 + \mathbf{s}_2$ where $\mathbf{s}_1 \overset{\$}{\leftarrow} \mathbb{Z}_q^k$ and $\mathbf{s}_2 \overset{\$}{\leftarrow} \mathbb{Z}_q^{3k}$. This change is only conceptual. Then we change $\mathbf{s}_2$ from a random vector in $\mathbb{Z}_q^{3k}$ to a random vector in the span of $\mathbf{B}_0$ via the MDDH assumption. More precisely, let $([\mathbf{B}_0]_2, [\mathbf{Z}]_2) \in \mathbb{G}_2^{3k\times(k+Q)}$ be a Q-fold $\mathcal{U}_{3k,k}$-MDDH challenge. For the i-th Eval query with $[\![\mathsf{m}]\!]_{\hat\jmath+1} = 0$, the reduction $\mathcal{B}$ computes $[\mathbf{t}]_2 := [\mathbf{B}\mathbf{s}_1 + \mathbf{Z}[i]]_2$, where $\mathbf{s}_1 \overset{\$}{\leftarrow} \mathbb{Z}_q^k$ and $\mathbf{Z}[i]$ is the i-th column vector of $\mathbf{Z}$.

To ensure that the column vectors of $(\mathbf{B}|\mathbf{B}_0|\mathbf{B}_1)$ form a basis of $\mathbb{Z}_q^{3k}$, the reduction chooses $\mathbf{B}$, $\mathbf{B}_1 \overset{\$}{\leftarrow} \mathcal{U}_{3k,k}$ such that $(\mathbf{B}|\mathbf{B}_1)$ has rank $2k$ and checks whether the kernels of $\mathbf{B}_0^{\top}$ and $(\mathbf{B}|\mathbf{B}_1)^{\top}$ are disjoint. This is equivalent to $(\mathbf{B}|\mathbf{B}_0|\mathbf{B}_1)$ forming a basis of $\mathbb{Z}_q^{3k}$ and can be done over the group by testing for all column vectors $\mathbf{b}$ of $(\mathbf{B}|\mathbf{B}_1)^{\perp}$ whether $\mathbf{B}_0^{\top}\mathbf{b} \neq \mathbf{0}$. By generating new matrices $\mathbf{B}, \mathbf{B}_1 \overset{\$}{\leftarrow} \mathcal{U}_{3k,k}$ until this is satisfied, we can ensure that $\mathbf{B}, \mathbf{B}_0, \mathbf{B}_1$ is a basis of $\mathbb{Z}_q^{3k}$.

```
$\mathsf{G}_{2,\hat{\imath},1}$  $\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},0}$  $\boxed{\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},1}\ \mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},2}\ \mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},3}}$  $\boxed{\boxed{\mathsf{G}_{2,\hat{\imath},3}}}$

INIT:
$\mathbf{B}$ $\boxed{\mathbf{B}_0, \mathbf{B}_1}$ $\mathbf{B}_0^*, \mathbf{B}_1^*$ $\overset{\$}{\leftarrow} \mathcal{U}_{3k,k}$
with the constrains:
    • $\mathbf{B}, \mathbf{B}_0, \mathbf{B}_1$ is a basis for $\mathbb{Z}_q^{3k}$
    • $\mathbf{B}_0^*, \mathbf{B}_1^*$ is a basis for $\mathsf{Span}(\mathbf{B}^\perp)$
    • $\mathbf{B}_1^\top \mathbf{B}_0^* = \mathbf{B}_0^\top \mathbf{B}_1^* = \mathbf{0}$
for $i \in \{1, \ldots, L\},\ j \in \{1, \ldots, i\alpha\},\ b \in \{0,1\}$ do
$\mathbf{x}_{i,j,b} \overset{\$}{\leftarrow} \mathbb{Z}_q^{3k}$
$x_0' \overset{\$}{\leftarrow} \mathbb{Z}_q$
$K : \mathcal{S}^{\leq L} \dashrightarrow \mathbb{G}_2^{3k+1} \times \mathbb{G}_2^{\leq \alpha(L^2+L)}$
return $\left([\mathbf{B}]_2, \left(\left[\mathbf{x}_{i,j,b}^\top \mathbf{B}\right]_2\right)_{1 \leq i \leq L, 1 \leq j \leq i\alpha, b \in \{0,1\}}\right)$

CHAL($\mathsf{m}^\star \in \mathcal{S}^p$):                      // one query
$h \overset{\$}{\leftarrow} \mathbb{Z}_q$
$\mathbf{h}_0 := \sum_{i=1}^{p} \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,\llbracket \mathsf{m}^\star \rrbracket_j} h$
if $p \geq \hat{\imath}$ then
    $\mathbf{h}_0 := \mathbf{h}_0$
        $+ \mathbf{B}^\perp \mathsf{RF}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}}\right)^\top h$
        $+ \left(\mathbf{B}_0^* \mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}+1}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}+1}\right)^\top + \mathbf{B}_1^* \mathsf{RF}^{(1)}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}}\right)^\top\right) h$
        $+ \left(\mathbf{B}_0^* \mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}+1}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}+1}\right)^\top + \mathbf{B}_1^* \mathsf{RF}^{(1)}_{\hat{\imath},\hat{\jmath}+1}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}+1}\right)^\top\right) h$
        $\boxed{\boxed{+ \mathbf{B}^\perp \mathsf{RF}_{\hat{\imath},\hat{\jmath}}\left(\mathsf{m}^\star_{|\hat{\imath}}\right)^\top h}}$
$h_1 := x_0' \cdot h$
return $([h]_1, [\mathbf{h}_0]_1, [h_1]_T)$

FINALIZE($\beta \in \{0,1\}$):
return $\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} \overset{?}{=} \emptyset \wedge \beta$

EVAL($\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p$):
if $\mathsf{m} \in \mathcal{Q}_{\mathcal{M}}$ then
    return $\mathsf{RerandTag}(\mathsf{m}, K(\mathsf{m}))$
$\mathcal{Q}_{\mathcal{M}} := \mathcal{Q}_{\mathcal{M}} \cup \{\mathsf{m}\}$
$\mathbf{s} \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{B}\mathbf{s}$
if $p = \hat{\imath}$ then
    $\mathbf{t} \overset{\$}{\leftarrow} \mathbb{Z}_q^{3k}$
    $\boxed{\begin{array}{l} \mathbf{s} \overset{\$}{\leftarrow} \mathbb{Z}_q^{2k} \\ \textbf{if } \llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0 \textbf{ then} \\ \quad \mathbf{t} := (\mathbf{B}|\mathbf{B}_0)\mathbf{s} \\ \textbf{else} \\ \quad \mathbf{t} := (\mathbf{B}|\mathbf{B}_1)\mathbf{s} \end{array}}$
if $p < \hat{\imath}$ then
    $u \overset{\$}{\leftarrow} \mathbb{Z}_q$
else
    $u := \left(\sum_{i=1}^{p} \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,\llbracket \mathsf{m} \rrbracket_j}^\top\right)\mathbf{t} + x_0'$
if $p = \hat{\imath}$ then
    $u := u$
        $+ \mathsf{RF}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right)\left(\mathbf{B}^\perp\right)^\top \mathbf{t}$
        $+ \left(\mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\right)(\mathbf{B}_0^*)^\top + \mathsf{RF}^{(1)}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right)(\mathbf{B}_1^*)^\top\right)\mathbf{t}$
        $+ \left(\mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\right)(\mathbf{B}_0^*)^\top + \mathsf{RF}^{(1)}_{\hat{\imath},\hat{\jmath}+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\right)(\mathbf{B}_1^*)^\top\right)\mathbf{t}$
        $\boxed{\boxed{u \overset{\$}{\leftarrow} \mathbb{Z}_q}}$
for $i \in \{p+1, \ldots, L\},\ j \in \{1, \ldots, i\alpha\},\ b \in \{0,1\}$
do $d_{i,j,b} := \mathbf{x}_{i,j,b}^\top \mathbf{t}$
$\mathsf{tdk} := \left([d_{i,j,b}]_2\right)_{p+1 \leq i \leq L, 1 \leq j \leq i\alpha, b \in \{0,1\}}$
$K(\mathsf{m}) := ([\mathbf{t}]_2, [u]_2, \mathsf{tdk})$
return $K(\mathsf{m})$
```

Figure 6: Hybrids for the transition from $\mathsf{G}_{2,\hat{\imath},1}$ to $\mathsf{G}_{2,\hat{\imath},3}$. The algorithm $\mathsf{RerandTag}$ is defined in Figure 5

If $\mathbf{Z}$ is uniform, $\mathcal{B}$ simulates the game $\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},0}$. If $\mathbf{Z}$ is from $\mathsf{Span}(\mathbf{B}_0)$ then $\mathcal{B}$ simulates the game $\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},1}$ for messages with $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0$.

By using the same argument, we can switch $\mathbf{t}$ for $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 1$ from a random vector in $\mathbb{Z}_q^{3k}$ to a random vector in $\mathsf{Span}(\mathbf{B}|\mathbf{B}_1)$.

The running time of $\mathcal{B}$ is dominated by the running time of $\mathcal{A}$ plus some (polynomial) overhead that is independent of $T(\mathcal{A})$ for the group operations in each oracle query. □

Lemma 3.10 ($\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},1} \rightsquigarrow \mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},2}$).

$$\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat{\imath},2,\hat{\jmath},1} \Rightarrow 1\right] = \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat{\imath},2,\hat{\jmath},2} \Rightarrow 1\right]$$

Proof. First of all, we replace in game $\mathsf{G}_{2,\hat{\imath},2,\hat{\jmath},1}$ the term $\mathsf{RF}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right)\left(\mathbf{B}^\perp\right)^\top$ with $\mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right)(\mathbf{B}_0^*)^\top + \mathsf{RF}^{(1)}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right)(\mathbf{B}_1^*)^\top$. This does not change the distribution, since $\mathbf{B}_0^*, \mathbf{B}_1^*$ is a basis for $\mathsf{Span}\left(\mathbf{B}^\perp\right)$.

We define

$$\mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\right) := \begin{cases} \mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right) & \text{if } \llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0 \\ \mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right) + \mathsf{RF}'^{(0)}_{\hat{\imath},\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right) & \text{if } \llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 1 \end{cases},$$

where $\mathsf{RF}'^{(0)}_{\hat\imath,\hat\jmath} : \{0,1\}^{\hat\jmath} \to \mathbb{Z}_q^{1\times k}$ is another independent random function. Since $\mathsf{RF}^{(0)}_{\hat\imath,\hat\jmath}$ does not appear in game $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,2}$ anymore, $\mathsf{RF}^{(0)}_{\hat\imath,\hat\jmath+1}$ is a random function.

Eval queries with $p \neq \hat\imath$ use the same code in both games and Eval queries with $p = \hat\imath$ and $[\![\mathsf{m}]\!]_{\hat\jmath+1} = 0$ are distributed identically in both games, by definition of $\mathsf{RF}^{(0)}_{\hat\imath,\hat\jmath+1}$.

Eval queries with $p = \hat\imath$ and $[\![\mathsf{m}]\!]_{\hat\jmath+1} = 1$ are distributed identically in both games, since for those queries $\mathbf{t} \in \mathsf{Span}(\mathbf{B}|\mathbf{B}_1)$ and both $\mathbf{B}$ and $\mathbf{B}_1$ are orthogonal to $\mathbf{B}_0^*$ and thus $\mathsf{RF}^{(0)}_{\hat\imath,\hat\jmath+1}\left([\![\mathsf{m}]\!]_{|\hat\jmath+1}\right)(\mathbf{B}_0^*)^\top\mathbf{t} = 0$.

The Chal query uses the same code if $p < \hat\imath$ and otherwise it is distributed identically if $[\![\mathsf{m}^\star]\!]_{\hat\jmath+1} = 0$. For the case $[\![\mathsf{m}^\star]\!]_{\hat\jmath+1} = 1$ note that $\mathbf{x}_{\hat\imath,\hat\jmath+1,1}$ is identically distributed as $\mathbf{x}_{\hat\imath,\hat\jmath+1,1} + \mathbf{B}_0^*\mathbf{w}$ for $\mathbf{w} \overset{\$}{\leftarrow} \mathbb{Z}_q^k$ and $\mathbf{w}$ is hidden from the adversary except for the Chal query: In all Eval queries with $p \neq \hat\imath$ only $\mathbf{x}_{\hat\imath,\hat\jmath+1,1}\mathbf{B}$ is used and thus the $\mathbf{B}_0^*$-part cancels out. In the Eval queries with $p = \hat\imath$ there is either $[\![\mathsf{m}]\!]_{\hat\jmath+1} = 0$ which means that $\mathbf{x}_{\hat\imath,\hat\jmath+1,1}$ is not used to compute the tag or there is $[\![\mathsf{m}]\!]_{\hat\jmath+1} = 1$ which means that $\mathbf{t} \in \mathsf{Span}(\mathbf{B}|\mathbf{B}_1)$ and thus the $\mathbf{B}_0^*$ -part of $\mathbf{x}_{\hat\imath,\hat\jmath+1,1}$ cancels out. All in all this means that the value $\mathbf{h}_0$ is the only one in the game that depends on $\mathbf{w}$ and thus the $\mathbf{B}_0^*$-part of $\mathbf{h}_0$ is uniformly random to the adversary. Especially $\mathbf{h}_0$ is distributed identically in both games. □

Lemma 3.11 ($\mathsf{G}_{2,\hat\imath,2,\hat\jmath,2} \rightsquigarrow \mathsf{G}_{2,\hat\imath,2,\hat\jmath,3}$).

$$\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,2,\hat\jmath,2} \Rightarrow 1\right] = \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,2,\hat\jmath,3} \Rightarrow 1\right]$$

Proof. We define

$$\mathsf{RF}^{(1)}_{\hat\imath,\hat\jmath+1}\left([\![\mathsf{m}]\!]_{|\hat\jmath+1}\right) := \begin{cases} \mathsf{RF}^{(1)}_{\hat\imath,\hat\jmath}\left([\![\mathsf{m}]\!]_{|\hat\jmath}\right) + \mathsf{RF}'^{(1)}_{\hat\imath,\hat\jmath}\left([\![\mathsf{m}]\!]_{|\hat\jmath}\right) & \text{if } [\![\mathsf{m}]\!]_{\hat\jmath+1} = 0 \\ \mathsf{RF}^{(1)}_{\hat\imath,\hat\jmath}\left([\![\mathsf{m}]\!]_{|\hat\jmath}\right) & \text{if } [\![\mathsf{m}]\!]_{\hat\jmath+1} = 1 \end{cases},$$

where $\mathsf{RF}'^{(1)}_{\hat\imath,\hat\jmath} : \{0,1\}^{\hat\jmath} \to \mathbb{Z}_q^{1\times k}$ is another independent random function. Since $\mathsf{RF}^{(1)}_{\hat\imath,\hat\jmath}$ in not used in game $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,3}$, $\mathsf{RF}^{(1)}_{\hat\imath,\hat\jmath+1}$ is a random function.

The argument, that the games $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,2}$ and $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,3}$ are identically distributed, is the same as in Lemma 3.10, just with the roles of 0 and 1 swapped. □

Lemma 3.12 ($\mathsf{G}_{2,\hat\imath,2,\hat\jmath,3} \rightsquigarrow \mathsf{G}_{2,\hat\imath,2,\hat\jmath+1,0}$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,2,\hat\jmath,3} \Rightarrow 1\right] - \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,\hat\jmath+1} \Rightarrow 1\right]\right| \leq 4k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{2}{q-1}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. In game $\mathsf{G}_{2,\hat\imath,2,\hat\jmath,3}$, replace the term $\mathsf{RF}^{(0)}_{\hat\imath,\hat\jmath+1}\left([\![\mathsf{m}]\!]_{|\hat\jmath+1}\right)(\mathbf{B}_0^*)^\top + \mathsf{RF}^{(1)}_{\hat\imath,\hat\jmath+1}\left([\![\mathsf{m}]\!]_{|\hat\jmath+1}\right)(\mathbf{B}_1^*)^\top$ with $\mathsf{RF}_{\hat\imath,\hat\jmath}\left([\![\mathsf{m}]\!]_{|\hat\jmath}\right)\left(\mathbf{B}^\perp\right)^\top$ to avoid computing $\mathbf{B}_0^*$ and $\mathbf{B}_1^*$. This does not change the distribution, since $\mathbf{B}_0^*, \mathbf{B}_1^*$ is a basis for $\mathsf{Span}\left(\mathbf{B}^\perp\right)$. The remaining transition is the reverse of Lemma 3.9. □

Lemma 3.13 ($\mathsf{G}_{2,\hat\imath,2,\hat\imath\alpha,0} \rightsquigarrow \mathsf{G}_{2,\hat\imath,3}$). *Let $Q_{\hat\imath}$ denote the number of* Eval *queries with $p = \hat\imath$.*

$$\left|\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,2,\hat\imath\alpha,0} \Rightarrow 1\right] - \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,3} \Rightarrow 1\right]\right| \leq \frac{Q_{\hat\imath}}{q^{2k}}$$

Proof. In game $\mathsf{G}_{2,\hat\imath,2,\hat\imath\alpha,0}$ the Chal-query evaluates $\mathsf{RF}_{\hat\imath,\hat\imath\alpha}$ only for the input value $\mathsf{m}_1^\star||\ldots||\mathsf{m}_{\hat\imath}^\star$ (if $p \geq \hat\imath$, otherwise it does not use $\mathsf{RF}_{\hat\imath,\hat\imath\alpha}$ at all). Assume $\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} = \emptyset$, otherwise the adversary has lost the game anyway. In each Eval query with $p = \hat\imath$ the value $\mathsf{RF}_{\hat\imath,\hat\imath\alpha}(\mathsf{m})\left(\mathbf{B}^\perp\right)^\top\mathbf{t}$ is part of u. This is the only place where $\mathsf{RF}_{\hat\imath,\hat\imath\alpha}(\mathsf{m})$ is used, since only the first Eval query for each message evaluates the random function. Thus each query outputs a uniformly random value for u when $\mathbf{t} \notin \mathsf{Span}(\mathbf{B})$, which happens with probability $\geq 1 - 1/\left(q^{2k}\right)$. In this case the games are distributed identically. □

Lemma 3.14 ($\mathsf{G}_{2,\hat\imath,3} \rightsquigarrow \mathsf{G}_{2,\hat\imath,4}$).

$$\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,3} \Rightarrow 1\right] = \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat\imath,4} \Rightarrow 1\right]$$

```
$\underline{\mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2):}$
$K \xleftarrow{\$} \mathsf{HGen}(1^\lambda)$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
$\textbf{for } i \in \{1, \dots, L\},\ j \in \{1, \dots, \gamma\},\ b \in \{0,1\} \textbf{ do } \mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$x'_0 \xleftarrow{\$} \mathbb{Z}_q$
$\textbf{return } \mathsf{sk}_{\mathsf{MAC}} := \left(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \gamma, b \in \{0,1\}}, x'_0, K\right)$

$\underline{\mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p):}$
$\textbf{Parse } \mathsf{sk}_{\mathsf{MAC}} =: \left(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \gamma, b \in \{0,1\}}, x'_0, K\right)$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{Bs}$
$u := \left(\sum_{i=1}^{p} \sum_{j=1}^{\gamma} \mathbf{x}^\top_{i,j,\mathsf{HEval}_K(\mathsf{m}_{|i})_j}\right)\mathbf{t} + x'_0$
$\textbf{return } \tau := \left([\mathbf{t}]_2, [u]_2\right)$

$\underline{\mathsf{Ver}_{\mathsf{MAC}}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p, \tau):}$
$\textbf{Parse } \mathsf{sk}_{\mathsf{MAC}} =: \left(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \gamma, b \in \{0,1\}}, x'_0, K\right)$
$\textbf{Parse } \mathsf{m} =: (\mathsf{m}_1, \dots, \mathsf{m}_p)$
$\textbf{Parse } \tau =: \left([\mathbf{t}]_2, [u]_2\right)$
$\textbf{return } u \stackrel{?}{=} \left(\sum_{i=1}^{p} \sum_{j=1}^{\gamma} \mathbf{x}^\top_{i,j,\mathsf{HEval}_K(\mathsf{m}_{|i})_j}\right)\mathbf{t} + x'_0$
```

Figure 7: The first affine MAC improved with a hash function

Proof. The games execute the same code if $p < \hat{\imath}$ and otherwise we can argue that $\mathbf{x}_{\hat{\imath},1,[\![\mathsf{m}^\star]\!]_1}$ and $\mathbf{x}_{\hat{\imath},1,[\![\mathsf{m}^\star]\!]_1} - \mathbf{B}^\perp(\mathsf{RF}_{\hat{\imath},\hat{\imath}\alpha}(\mathsf{m}^\star))^\top$ are identical distributed. All Eval queries and the "public key" returned by Init make only use of $\mathbf{x}_{\hat{\imath},1,[\![\mathsf{m}^\star]\!]_1}\mathbf{B}$, so the $\mathbf{B}^\perp(\mathsf{RF}_{\hat{\imath},\hat{\imath}\alpha}(\cdot))^\top$ part cancels out. □

Lemma 3.15 ($\mathsf{G}_{2,\hat{\imath},4} \rightsquigarrow \mathsf{G}_{2,\hat{\imath},5}$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat{\imath},4} \Rightarrow 1\right] - \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat{\imath},5} \Rightarrow 1\right]\right| \le 2k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{1}{q-1}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. The transition is the reverse of Lemma 3.6. □

Lemma 3.16 ($\mathsf{G}_{2,\hat{\imath},5} \rightsquigarrow \mathsf{G}_{2,\hat{\imath}+1,0}$). *For $\hat{\imath} < L$*

$$\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat{\imath},5} \Rightarrow 1\right] = \Pr\left[\mathsf{G}^{\mathcal{A}}_{2,\hat{\imath}+1,0} \Rightarrow 1\right].$$

Proof. These two games are equivalent. □

Lemma 3.17 ($\mathsf{G}_{2,L,5} \rightsquigarrow \mathsf{G}_3$).

$$\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,L,5} \Rightarrow 1\right] = \Pr\left[\mathsf{G}^{\mathcal{A}}_{3} \Rightarrow 1\right]$$

Proof. In game $\mathsf{G}_{2,L,5}$ the value x'_0 is only used to compute h_1, thus h_1 is a uniform random value to $\mathcal{A}$ and the games are distributed identical. □

Summary. To prove Theorem 3.3, we combine Lemmas 3.4–3.17 to change h_1 from real to random and then apply all Lemmas in reverse order to get to the $\mathsf{HPR}_0\text{-}\mathsf{CMA}_{\mathsf{rand}}$ game. □

Optimization. $\mathsf{MAC}_1[\mathcal{U}_{3k,k}]$ can be improved with a collision-resistant hash function. In $\mathsf{MAC}_1[\mathcal{U}_{3k,k}]$ in a tag for m we add to the value u for the i-th level $\sum_j f_j(\mathsf{m}_{|i})\mathbf{x}_j^\top$. The idea is to replace this by $\sum_j f_j(H(\mathsf{m}_{|i}))\mathbf{x}_j^\top$ for a collision resistant hash function H.

Formally, we need a family of hash functions $\mathcal{H} := (\mathsf{HGen}, \mathsf{HEval})$ with domain $\mathcal{S}^{\le L}$ and range $\{0,1\}^\gamma$ for all hash keys. The affine MAC $\mathsf{MAC}_1^{\mathcal{H}}[\mathcal{U}_{3k,k}]$ is shown in Figure 7.

```
$\underline{\mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2)\text{:}}$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
$\textbf{for } i \in \{1, \ldots, L\},\ j \in \{1, \ldots, i\alpha\},\ b \in \{0,1\} \textbf{ do } \mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$x'_0 \xleftarrow{\$} \mathbb{Z}_q$
$\textbf{return } \mathsf{sk}_{\mathsf{MAC}} := \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le i\alpha, b \in \{0,1\}}, x'_0\big)$

$\underline{\mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p)\text{:}}$
$\textbf{Parse } \mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le i\alpha, b \in \{0,1\}}, x'_0\big)$
$\textbf{for } i \in \{1, \ldots, p\} \textbf{ do } \mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t}_i := \mathbf{B}\mathbf{s}_i$
$u := \sum_{i=1}^{p} \Big( \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top \Big) \mathbf{t}_i + x'_0$
$\textbf{return } \Big( \big( [\mathbf{t}_i]_2 \big)_{1 \le i \le p}, [u]_2 \Big)$

$\underline{\mathsf{Ver}_{\mathsf{MAC}}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p, \tau)\text{:}}$
$\textbf{Parse } \mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le i\alpha, b \in \{0,1\}}, x'_0\big)$
$\textbf{Parse } \mathsf{m} =: (\mathsf{m}_1, \ldots, \mathsf{m}_p)$
$\textbf{Parse } \tau =: \Big( \big( [\mathbf{t}_i]_2 \big)_{1 \le i \le p}, [u]_2 \Big)$
$\textbf{return } u \overset{?}{=} \sum_{i=1}^{p} \Big( \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top \Big) \mathbf{t}_i + x'_0$
```

Figure 8: The second affine MAC with levels

Theorem 3.18 (Security of $\mathsf{MAC}_1^{\mathcal{H}}[\mathcal{U}_{3k,k}]$). $\mathsf{MAC}_1^{\mathcal{H}}[\mathcal{U}_{3k,k}]$ *is tightly* HPR_0-CMA *secure in* $\mathbb{G}_2$ *when* $\mathcal{H}$ *is collision resistant and the* $\mathcal{U}_k$-MDDH *assumption holds for* $\mathbb{G}_2$*. More precisely, for all adversaries* $\mathcal{A}$ *there exist adversaries* $\mathcal{B}$ *and* $\mathcal{C}$ *with*

$$\mathsf{Adv}^{\mathsf{hpr}_0\text{-}\mathsf{cma}}_{\mathsf{MAC}_1^{\mathcal{H}}[\mathcal{U}_{3k,k}],\mathbb{G}_2}(\mathcal{A}) \le (8kL(1+2\gamma))\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{4L(1+2\gamma)}{q-1} + 2L\mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H}}(\mathcal{C}) + \frac{2Q}{q^{2k}}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ *and* $T(\mathcal{C}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$*, where* Q *denotes the number of* Eval *queries of* $\mathcal{A}$ *and* poly *is a polynomial independent of* $\mathcal{A}$*.*

We omit the proof of this theorem, since it is very similar to the proof of Theorem 3.3. The main idea is rather standard: We use the collision resistance of $\mathcal{H}$ to reject the collision and then we apply the hybrid argument on the hash output.

3.2 The second construction

Let $(\mathbb{G}_2, q, P_2)$ be a group of prime order q. The second affine MAC with levels $\mathsf{MAC}_1[\mathcal{U}_{3k,k}] := (\mathsf{Gen}_{\mathsf{MAC}}, \mathsf{Tag}, \mathsf{Ver}_{\mathsf{MAC}})$ with message space $\mathcal{ID} := \mathcal{S}^{\le L} := (\{0,1\}^\alpha)^{\le L}$ is defined in Figure 8. The identity vectors bit-length α and the maximum length L of the identity vectors can be chosen freely. The difference to the first construction is that this MAC uses a different $\mathbf{t}_l$ on each level ($\ell(p) = p$) and thus needs no delegation keys. This leads to shorter user secret keys and allows a more efficient reduction. However, this comes at the price of larger ciphertexts. Formally, this MAC uses $\ell'(l,i) = 0$ for $i < p$ and $\ell'(l,i) = 2i\alpha$ for $i = p$.

Theorem 3.19 (Security of $\mathsf{MAC}_2[\mathcal{U}_{3k,k}]$). $\mathsf{MAC}_2[\mathcal{U}_{3k,k}]$ *is tightly* HPR_0-CMA *secure in* $\mathbb{G}_2$ *under the* $\mathcal{U}_k$-MDDH *assumption for* $\mathbb{G}_2$*. More precisely, for all adversaries* $\mathcal{A}$ *there exists an adversary* $\mathcal{B}$ *with*

$$\mathsf{Adv}^{\mathsf{hpr}_0\text{-}\mathsf{cma}}_{\mathsf{MAC}_2[\mathcal{U}_{3k,k}],\mathbb{G}_2}(\mathcal{A}) \le (4k + 16k\alpha L)\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{2 + 8k\alpha L}{q-1} + \frac{2Q}{q^{2k}}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$*, where* Q *denotes the number of* Eval *queries of* $\mathcal{A}$ *and* poly *is a polynomial independent of* $\mathcal{A}$*.*

Proof. The proof uses a hybrid argument with the hybrids G_0 (the HPR_0-$\mathsf{CMA}_{\mathsf{real}}$ game), G_1, G_2, $\mathsf{G}_{3,\hat{\jmath}}$ for $\hat{\jmath} \in \{0, \ldots, L\alpha\}$, $\mathsf{G}_{3,\hat{\jmath},1}$–$\mathsf{G}_{3,\hat{\jmath},3}$ for $\hat{\jmath} \in \{0, \ldots, L\alpha - 1\}$ and finally G_4. The hybrids are given in Figures 9 and 10. A summary can be found in Table 3. They make use of random functions $\mathsf{RF}_{i,\hat{\jmath}} : \{0,1\}^{\hat{\jmath}} \to \mathbb{Z}_q^{1 \times 2k}$, $\mathsf{RF}^{(0)}_{i,\hat{\jmath}} : \{0,1\}^{\hat{\jmath}} \to \mathbb{Z}_q^{1 \times k}$ and $\mathsf{RF}^{(1)}_{i,\hat{\jmath}} : \{0,1\}^{\hat{\jmath}} \to \mathbb{Z}_q^{1 \times k}$, defined on-the-fly.

Hybrid	$\mathbf{t}_i$ uniform in	$r_u(\mathsf{m}, i)$	$r_{\mathbf{h}_0}(\mathsf{m}, i)$	Transition
G_0	$\mathsf{Span}(\mathbf{B})$	0		Original game
G_1	$\mathsf{Span}(\mathbf{B})$	0		Identical
G_2	$\mathbb{Z}_q^{3k}$	0		$\mathcal{U}_k$-MDDH
$\mathsf{G}_{3,\hat{\jmath}}$	$\mathbb{Z}_q^{3k}$	$\mathsf{RF}_{i,\hat{\jmath}}\left(\llbracket\mathsf{m}\rrbracket_{\mid\max\{\hat{\jmath},i\alpha\}}\right)\left(\mathbf{B}^\perp\right)^\top$		Identical

$\mathsf{G}_{3,\hat{\jmath},1}$	**if** $\llbracket\mathsf{m}\rrbracket_{\hat{\jmath}+1} = 0$ **then**
	$\mid \mathsf{Span}(\mathbf{B}\mid\mathbf{B}_0)$
	else
	$\lfloor \mathsf{Span}(\mathbf{B}\mid\mathbf{B}_1)$
		$\mathsf{RF}_{i,\hat{\jmath}}\left(\llbracket\mathsf{m}\rrbracket_{\mid\max\{\hat{\jmath},i\alpha\}}\right)\left(\mathbf{B}^\perp\right)^\top$		$\mathcal{U}_k$-MDDH

$\mathsf{G}_{3,\hat{\jmath},2}$		$\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\left(\llbracket\mathsf{m}\rrbracket_{\mid\max\{\hat{\jmath}+1,i\alpha\}}\right)\left(\mathbf{B}^*_0\right)^\top$ $+ \mathsf{RF}^{(1)}_{i,\hat{\jmath}}\left(\llbracket\mathsf{m}\rrbracket_{\mid\max\{\hat{\jmath},i\alpha\}}\right)\left(\mathbf{B}^*_1\right)^\top$		Identical
$\mathsf{G}_{3,\hat{\jmath},3}$		$\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\left(\llbracket\mathsf{m}\rrbracket_{\mid\max\{\hat{\jmath}+1,i\alpha\}}\right)\left(\mathbf{B}^*_0\right)^\top$ $+ \mathsf{RF}^{(1)}_{i,\hat{\jmath}+1}\left(\llbracket\mathsf{m}\rrbracket_{\mid\max\{\hat{\jmath}+1,i\alpha\}}\right)\left(\mathbf{B}^*_1\right)^\top$		Identical
$\mathsf{G}_{3,\hat{\jmath}+1}$	$\mathbb{Z}_q^{3k}$	$\mathsf{RF}_{i,\hat{\jmath}+1}\left(\llbracket\mathsf{m}\rrbracket_{\mid\max\{\hat{\jmath}+1,i\alpha\}}\right)\left(\mathbf{B}^\perp\right)^\top$		$\mathcal{U}_k$-MDDH
G_4	$\mathbb{Z}_q^{3k}$	unif. random	$\mathsf{RF}_{i,i\alpha}\left(\mathsf{m}_{\mid i}\right)\left(\mathbf{B}^\perp\right)^\top$	Statistically close

Table 3: Summary of the hybrids in Figure 9 and 10. Non-duplicated Eval queries draw $\mathbf{t}_i$ from the set described by the second column and add the randomness $\sum_{i=1}^{p} r_u(\mathsf{m}, i)\mathbf{t}_i$ to u or choose u uniform random. The Chal query adds the term $r_{\mathbf{h}_0}(\mathsf{m}^\star, i)h$ to each $\mathbf{h}_{0,i}$. The background color indicates repeated transitions

Lemma 3.20 ($\mathsf{G}_0 \rightsquigarrow \mathsf{G}_1$).

$$\Pr\left[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\right]$$

Proof. In game G_1 each time the adversary queries a tag for a message m and he queried a tag for m before, the adversary will get a rerandomized version of the first tag he queried. The rerandomized tag is identically distributed to a fresh tag: $\mathbf{t}'_i := \mathbf{t}_i + \mathbf{B}\mathbf{s}'_i$ is uniformly random in $\mathsf{Span}(\mathbf{B})$, when $\mathbf{s}'_i$ is uniform random in $\mathbb{Z}_q^k$. Together with $u' := u + \sum_{i=1}^{p}\left(\sum_{j=1}^{i\alpha} \mathbf{x}^\top_{i,j,\llbracket\mathsf{m}\rrbracket_j}\mathbf{B}\mathbf{s}'_i\right)$ we get a valid message tag for m, when $\left(([\mathbf{t}_i]_2)_{1\le i\le p}, [u]_2\right)$ is a valid tag for m.

Note that the rerandomization can be carried out by just using the "public key" returned by the Init oracle, so it could actually be carried out by the adversary herself. To put it in a nutshell, repeated Eval queries for one message m leak no information, that is not already leaked by the first Eval query for m or by the "public key". □

Lemma 3.21 ($\mathsf{G}_1 \rightsquigarrow \mathsf{G}_2$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1\right]\right| \le 2k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{1}{q-1}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. These two games are equivalent except that the values $\mathbf{t}_i$ are generated uniformly random from $\mathsf{Span}(\mathbf{B})$ in G_1 and uniformly random from $\mathbb{Z}_q^{3k}$ in game G_2. Since for all computed values it is enough to have $[\mathbf{B}]_2$ instead of $\mathbf{B}$, this leads to a straightforward reduction to the QL-fold $\mathcal{U}_{3k,k}$-MDDH assumption. Remember that by Lemma 2.4, the $\mathcal{U}_{3k,k}$-MDDH assumption is equivalent to the $\mathcal{U}_k$ assumption.

The running time of $\mathcal{B}$ is dominated by the running time of $\mathcal{A}$ plus some (polynomial) overhead that is independent of $T(\mathcal{A})$ for the group operations in each oracle query. □

Lemma 3.22 ($\mathsf{G}_2 \rightsquigarrow \mathsf{G}_{3,0}$).

$$\Pr\left[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathsf{G}_{3,0}^{\mathcal{A}} \Rightarrow 1\right]$$

Proof. The games are equivalent. When changing in G_2 the secret values $\mathbf{x}_{i,1,b}$ to $\mathbf{x}_{i,1,b} + \mathbf{B}^\perp(\mathsf{RF}_{i,0}(\varepsilon))^\top$ (for all $i \in \{1,\ldots,L\}$ and $b \in \{0,1\}$), we get game $\mathsf{G}_{3,0}$. The distribution of $\mathbf{x}_{i,1,b}$ and $\mathbf{x}_{i,1,b} + \mathbf{B}^\perp(\mathsf{RF}_{i,0}(\varepsilon))^\top$ is identical. Note that the term $\mathbf{B}^\perp(\mathsf{RF}_{i,0}(\varepsilon))^\top$ cancels out in the public key. □

G_0 G_1 G_2 $\mathsf{G}_{3,\hat{\jmath}}$ G_4

```
INIT:
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
for $i \in \{1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$ do
$\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$x'_0 \xleftarrow{\$} \mathbb{Z}_q$
$K : \mathcal{S}^{\leq L} \dashrightarrow (\mathbb{G}_2^{3k})^{\leq L} \times \mathbb{G}_2$
return $\left([\mathbf{B}]_2, \left(\left[\mathbf{x}_{i,j,b}^\top \mathbf{B}\right]_2\right)_{1\leq i\leq L, 1\leq j\leq i\alpha, b\in\{0,1\}}\right)$

CHAL($\mathsf{m}^\star \in \mathcal{S}^p$):                    // one query
$h \xleftarrow{\$} \mathbb{Z}_q$
for $i \in \{1,\dots,p\}$ do  $\mathbf{h}_{0,i} := \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}^\star]\!]_j} h$
$\hat{\imath} := \lfloor(\hat{\jmath}+1)/\alpha\rfloor$
$\hat{\imath} := L+1$
for $i \in \{1,\dots,\min\{p,\hat{\imath}-1\}\}$ do
    $\mathbf{h}_{0,i} := \mathbf{h}_{0,i} + \mathbf{B}^\perp \mathsf{RF}_{i,i\alpha}\left(\left[\!\left[\mathsf{m}^\star_{|i}\right]\!\right]\right)^\top h$
for $i \in \{\hat{\imath},\dots,p\}$ do
    $\mathbf{h}_{0,i} := \mathbf{h}_{0,i} + \mathbf{B}^\perp \mathsf{RF}_{i,\hat{\jmath}}\left([\![\mathsf{m}^\star]\!]_{|\hat{\jmath}}\right)^\top h$
$h_1 := x'_0 \cdot h$
$h_1 \xleftarrow{\$} \mathbb{Z}_q$
return $\left([h]_1, ([\mathbf{h}_{0,i}]_1)_{1\leq i\leq p}, [h_1]_T\right)$

FINALIZE($\beta \in \{0,1\}$):
return $\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_\mathcal{M} \overset{?}{=} \emptyset \wedge \beta$
```

```
EVAL($\mathsf{m} \in \mathcal{S}^p$):
if $\mathsf{m} \in \mathcal{Q}_\mathcal{M}$ then
    return $\mathsf{RerandTag}(\mathsf{m}, K(\mathsf{m}))$
$\mathcal{Q}_\mathcal{M} := \mathcal{Q}_\mathcal{M} \cup \{\mathsf{m}\}$
for $i \in \{1,\dots,p\}$ do
    $\mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t}_i := \mathbf{B}\mathbf{s}_i$
    $\mathbf{t}_i \xleftarrow{\$} \mathbb{Z}_q^{3k}$
for $i \in \{1,\dots,p\}$ do
    $u_i := \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top \mathbf{t}_i$
$\hat{\imath} := \lfloor(\hat{\jmath}+1)/\alpha\rfloor$
for $i \in \{1,\dots,\min\{p,\hat{\imath}-1\}\}$ do
    $u_i := u_i + \mathsf{RF}_{i,i\alpha}([\![\mathsf{m}_{|i}]\!])(\mathbf{B}^\perp)^\top \mathbf{t}_i$
for $i \in \{\hat{\imath},\dots,p\}$ do
    $u_i := u_i + \mathsf{RF}_{i,\hat{\jmath}}\left([\![\mathsf{m}]\!]_{|\hat{\jmath}}\right)(\mathbf{B}^\perp)^\top \mathbf{t}_i$
$u := \sum_{i=1}^p u_i + x'_0$
$u \xleftarrow{\$} \mathbb{Z}_q$
$K(\mathsf{m}) := \left(([\mathbf{t}_i]_2)_{1\leq i\leq p}, [u]_2\right)$
return $K(\mathsf{m})$

RerandTag($\mathsf{m} \in \mathcal{S}^p, \tau$):
Parse $\tau =: \left(([\mathbf{t}_i]_2)_{1\leq i\leq p}, [u]_2\right)$
for $i \in \{1,\dots,p\}$ do  $\mathbf{s}'_i \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t}'_i := \mathbf{t}_i + \mathbf{B}\mathbf{s}'_i$
$u' := u + \sum_{i=1}^p \left(\sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{m}]\!]_j}^\top \mathbf{B}\mathbf{s}'_i\right)$
return $\left(([\mathbf{t}'_i]_2)_{1\leq i\leq p}, [u']_2\right)$
```

Figure 9: Hybrids for the security proof of $\mathsf{MAC}_2[\mathcal{U}_{3k,k}]$. The algorithm $\mathsf{RerandTag}$ is only helper function and not an oracle for the adversary. The partial map K is initially totally undefined

Lemma 3.23 ($\mathsf{G}_{3,\hat{\jmath}} \rightsquigarrow \mathsf{G}_{3,\hat{\jmath}+1}$). *For all $\hat{\jmath} \in \{0,\dots,L\alpha-1\}$ and all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_{3,\hat{\jmath}}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_{3,\hat{\jmath}+1}^{\mathcal{A}} \Rightarrow 1\right]\right| \leq 8k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{4}{q-1}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. To prove this transition, we introduce new hybrids $\mathsf{G}_{3,\hat{\jmath},1}$, $\mathsf{G}_{3,\hat{\jmath},2}$ and $\mathsf{G}_{3,\hat{\jmath},3}$. The hybrids are given in Figure 10.

Lemma 3.23 follows directly from Lemma 3.24–3.27. □

Lemma 3.24 ($\mathsf{G}_{3,\hat{\jmath}} \rightsquigarrow \mathsf{G}_{3,\hat{\jmath},1}$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_{3,\hat{\jmath}}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_{3,\hat{\jmath},1}^{\mathcal{A}} \Rightarrow 1\right]\right| \leq 4k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{2}{q-1}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. These two games are equivalent except that the values $\mathbf{t}_i$ are generated uniformly random from $\mathbb{Z}_q^{3k}$ in game $\mathsf{G}_{3,\hat{\jmath}}$ and from $\mathsf{Span}(\mathbf{B}|\mathbf{B}_0)$ respectively $\mathsf{Span}(\mathbf{B}|\mathbf{B}_1)$ in game $\mathsf{G}_{3,\hat{\jmath},1}$ for $i \in \{\hat{\imath},\dots,p\}$. We can switch from $\mathsf{G}_{3,\hat{\jmath}}$ to $\mathsf{G}_{3,\hat{\jmath},1}$ with two QL-fold $\mathcal{U}_{3k,k}$-MDDH challenges. Remember that the $\mathcal{U}_{3k,k}$-MDDH assumption is equivalent to the $\mathcal{U}_k$ assumption by Lemma 2.4.

The first challenge is used to change the distribution of $\mathbf{t}_i$ from $\mathbb{Z}_q^{3k}$ to $\mathsf{Span}(\mathbf{B}|\mathbf{B}_0)$ for the EVAL-queries with $[\![\mathsf{m}]\!]_{\hat{\jmath}+1} = 0$. Therefore, on input of a QL-fold $\mathcal{U}_{3k,k}$-MDDH challenge $([\mathbf{B}_0]_2, [\mathbf{Z}]_2)$ where $\mathbf{B}_0 \in \mathbb{Z}_q^{3k\times k}$ and the column vectors of $\mathbf{Z} \in \mathbb{Z}_q^{3k\times QL}$ are either uniform random from $\mathbb{Z}_q^{3k}$ or uniform random from $\mathsf{Span}(\mathbf{B}_0)$.

```
$\mathsf{G}_{3,\hat{\jmath}}$  $\mathsf{G}_{3,\hat{\jmath},1}$  $\mathsf{G}_{3,\hat{\jmath},2}$  $\mathsf{G}_{3,\hat{\jmath},3}$

INIT:
$\mathbf{B}$ $\mathbf{B}_0, \mathbf{B}_1$ $\mathbf{B}_0^*, \mathbf{B}_1^* \xleftarrow{\$} \mathcal{U}_{3k,k}$
with the constrains:
    • $\mathbf{B}, \mathbf{B}_0, \mathbf{B}_1$ is a basis for $\mathbb{Z}_q^{3k}$
    • $\mathbf{B}_0^*, \mathbf{B}_1^*$ is a basis for $\mathsf{Span}(\mathbf{B}^\perp)$
    • $\mathbf{B}_1^\top \mathbf{B}_0^* = \mathbf{B}_0^\top \mathbf{B}_1^*$
for $i \in \{1, \ldots, L\},\ j \in \{1, \ldots, i\alpha\},\ b \in \{0,1\}$ do
$\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$x_0' \xleftarrow{\$} \mathbb{Z}_q$
$K : \mathcal{S}^{\leq L} \dashrightarrow (\mathbb{G}_2^{3k})^{\leq L} \times \mathbb{G}_2$
return $\left([\mathbf{B}]_2, \left(\left[\mathbf{x}_{i,j,b}^\top \mathbf{B}\right]_2\right)_{1 \leq i \leq L, 1 \leq j \leq i\alpha, b \in \{0,1\}}\right)$

CHAL($\mathsf{m}^\star \in \mathcal{S}^p$):                    // one query
$h \xleftarrow{\$} \mathbb{Z}_q$
for $i \in \{1, \ldots, p\}$ do  $\mathbf{h}_{0,i} := \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,\llbracket \mathsf{m}^\star \rrbracket_j} h$
$\hat{\imath} := \lfloor (\hat{\jmath}+1)/\alpha \rfloor$
for $i \in \{1, \ldots, \min\{p, \hat{\imath}-1\}\}$ do
    $\mathbf{h}_{0,i} := \mathbf{h}_{0,i} + \mathsf{RF}_{i,i\alpha}\left(\left\llbracket \mathsf{m}^\star_{|i} \right\rrbracket\right)(\mathbf{B}^\perp)^\top h$
for $i \in \{\hat{\imath}, \ldots, p\}$ do
    $\mathbf{h}_{0,i} := \mathbf{h}_{0,i}$
        $\boxed{+ \mathbf{B}^\perp \mathsf{RF}_{i,\hat{\jmath}}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}}\right)^\top h}$
        $+ \left(\mathbf{B}_0^* \mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}+1}\right)^\top + \mathbf{B}_1^* \mathsf{RF}^{(1)}_{i,\hat{\jmath}}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}}\right)^\top\right) h$
        $+ \left(\mathbf{B}_0^* \mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}+1}\right)^\top + \mathbf{B}_1^* \mathsf{RF}^{(1)}_{i,\hat{\jmath}+1}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}+1}\right)^\top\right) h$
$h_1 := x_0' \cdot h$
return $\left([h]_1, ([\mathbf{h}_{0,i}]_1)_{1 \leq i \leq p}, [h_1]_T\right)$
```

```
EVAL($\mathsf{m} \in \mathcal{S}^p$):
Parse $\mathsf{m} =: (\mathsf{m}_1, \ldots, \mathsf{m}_p)$
if $\mathsf{m} \in \mathcal{Q}_\mathcal{M}$ then
    return $\mathsf{RerandTag}(\mathsf{m}, K(\mathsf{m}))$
$\mathcal{Q}_\mathcal{M} := \mathcal{Q}_\mathcal{M} \cup \{\mathsf{m}\}$
for $i \in \{1, \ldots, p\}$ do  $\mathbf{t}_i \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$\hat{\imath} := \lfloor (\hat{\jmath}+1)/\alpha \rfloor$
    for $i \in \{\hat{\imath}, \ldots, p\}$ do
        $\mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q^{2k}$
        if $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0$ then
            $\mathbf{t}_i := (\mathbf{B}|\mathbf{B}_0)\mathbf{s}_i$
        else
            $\mathbf{t}_i := (\mathbf{B}|\mathbf{B}_1)\mathbf{s}_i$
for $i \in \{1, \ldots, p\}$ do $u_i := \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,\llbracket \mathsf{m} \rrbracket_j}^\top \mathbf{t}_i$
for $i \in \{1, \ldots, \min\{p, \hat{\imath}-1\}\}$ do
    $u_i := u_i + \mathsf{RF}_{i,i\alpha}(\llbracket \mathsf{m}_{|\hat{\imath}} \rrbracket)(\mathbf{B}^\perp)^\top \mathbf{t}_i$
for $i \in \{\hat{\imath}, \ldots, p\}$ do
    $u_i := u_i$
        $\boxed{+ \mathsf{RF}_{i,\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right)(\mathbf{B}^\perp)^\top \mathbf{t}_i}$
        $+ \left(\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\right)(\mathbf{B}_0^*)^\top + \mathsf{RF}^{(1)}_{i,\hat{\jmath}}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\right)(\mathbf{B}_1^*)^\top\right)\mathbf{t}_i$
        $+ \left(\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\right)(\mathbf{B}_0^*)^\top + \mathsf{RF}^{(1)}_{i,\hat{\jmath}+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\right)(\mathbf{B}_1^*)^\top\right)\mathbf{t}_i$
$u := \sum_{i=1}^{p} u_i + x_0'$
$K(\mathsf{m}) := \left(([\mathbf{t}_i]_2)_{1 \leq i \leq p}, [u]_2\right)$
return $K(\mathsf{m})$

FINALIZE($\beta \in \{0,1\}$):
return $\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_\mathcal{M} \overset{?}{=} \emptyset \wedge \beta$
```

Figure 10: Hybrids for the transition from $\mathsf{G}_{3,\hat{\jmath}}$ to $\mathsf{G}_{3,\hat{\jmath}+1}$

We can now switch from $\mathsf{G}_{3,\hat{\jmath}}$ to an intermediate hybrid, where the EVAL queries with $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0$ are distributed as in $\mathsf{G}_{3,\hat{\jmath},1}$ and everything else is distributed as in game $\mathsf{G}_{3,\hat{\jmath}}$. Therefore, first change in game $\mathsf{G}_{3,\hat{\jmath}}$ the generation of those $\mathbf{t}_i$ with $i \geq \hat{\imath}$ in EVAL queries with $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0$ to $\mathbf{s}_1 \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{s}_2 \xleftarrow{\$} \mathbb{Z}_q^{3k}$; $\mathbf{t} := \mathbf{B}\mathbf{s}_1 + \mathbf{s}_2$. Adversary $\mathcal{B}$ now derives $\mathbf{B}_0$ from the $\mathcal{U}_k$-MDDH challenge and draws $\mathbf{B}, \mathbf{B}_1 \xleftarrow{\$} \mathcal{U}_{3k,k}$ until $(\mathbf{B}|\mathbf{B}_1)$ has rank $2k$ and the kernels of $\mathbf{B}_0^\top$ and $(\mathbf{B}|\mathbf{B}_1)^\top$ are disjoint, i.e., all column vectors $\mathbf{b}$ of $(\mathbf{B}|\mathbf{B}_1)^\perp$ satisfy $\mathbf{B}_0^\top \mathbf{b} \neq \mathbf{0}$. The last check can be done with $[\mathbf{B}_0]_2$ over $\mathbb{G}_2$. Like this, the column vectors of $\mathbf{B}$, $\mathbf{B}_0$ and $\mathbf{B}_1$ form a random basis of $\mathbb{Z}_q^{3k}$.

Now set in the ι-th EVAL query $\mathbf{t}_i := \mathbf{B}\mathbf{s}_1 + \mathbf{Z}[\iota L + i]$ where $\mathbf{s}_1 \xleftarrow{\$} \mathbb{Z}_q^k$ and $\mathbf{Z}[i]$ is the i-th column vector of $\mathbf{Z}$. If the column vectors of $\mathbf{Z}$ are uniform random from $\mathbb{Z}_q^{3k}$, $\mathcal{B}$ is simulating game $\mathsf{G}_{3,\hat{\jmath}}$. Otherwise, if the column vectors of $\mathbf{Z}$ are uniform random from $\mathsf{Span}(\mathbf{B}_0)$, $\mathbf{t}$ is uniformly random from $\mathsf{Span}(\mathbf{B}|\mathbf{B}_0)$ and $\mathcal{B}$ is simulating the intermediate hybrid.

We proceed analogously to switch from the intermediate hybrid to game $\mathsf{G}_{3,\hat{\jmath},1}$.

The running time of $\mathcal{B}$ is dominated by the running time of $\mathcal{A}$ plus some (polynomial) overhead that is independent of $T(\mathcal{A})$ for the group operations in each oracle query. □

Lemma 3.25 ($\mathsf{G}_{3,\hat{\jmath},1} \rightsquigarrow \mathsf{G}_{3,\hat{\jmath},2}$).

$$\Pr\left[\mathsf{G}^{\mathcal{A}}_{3,\hat{\jmath},1} \Rightarrow 1\right] = \Pr\left[\mathsf{G}^{\mathcal{A}}_{3,\hat{\jmath},2} \Rightarrow 1\right]$$

Proof. First of all, replace in game $\mathsf{G}_{3,\hat{\jmath},1}$ the term $\mathsf{RF}_{i,\hat{\jmath}}\left(\llbracket \mathsf{m}^\star \rrbracket_{|\hat{\jmath}}\right)(\mathbf{B}^\perp)^\top$ with $\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\right)(\mathbf{B}_0^*)^\top +$

$\mathsf{RF}^{(1)}_{i,\hat{\jmath}}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\Big)(\mathbf{B}_1^*)^\top$. This doesn't change the distribution, since $\mathbf{B}_0^*, \mathbf{B}_1^*$ is a basis for $\mathsf{Span}\big(\mathbf{B}^\perp\big)$.

We define

$$\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\Big) := \begin{cases} \mathsf{RF}^{(0)}_{i,\hat{\jmath}}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\Big) & \text{if } \llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0 \\ \mathsf{RF}^{(0)}_{i,\hat{\jmath}}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\Big) + \mathsf{RF}'^{(0)}_{i,\hat{\jmath}}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\Big) & \text{if } \llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 1 \end{cases},$$

where $\mathsf{RF}'^{(0)}_{i,\hat{\jmath}} : \{0,1\}^{\hat{\jmath}} \to \mathbb{Z}_q^{1\times k}$ is another independent random function. Since $\mathsf{RF}^{(0)}_{i,\hat{\jmath}}$ in not used in game $\mathsf{G}_{3,\hat{\jmath},2}$, $\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}$ is a random function.

Eval queries with $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0$ are distributed identically in both games, by definition of $\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}$.

Eval queries with $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 1$ are distributed identically in both games, since for those queries (for all $i \in \{\hat{\imath}, \dots, p\}$) $\mathbf{t}_i \in \mathsf{Span}(\mathbf{B}|\mathbf{B}_1)$ and both $\mathbf{B}$ and $\mathbf{B}_1$ are orthogonal to $\mathbf{B}_0^*$ and thus

$$\mathsf{RF}^{(0)}_{i,\hat{\jmath}+1}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\Big)(\mathbf{B}_0^*)^\top \mathbf{t}_i = 0.$$

The Chal query is distributed identically if $p < \hat{\imath}$ or $\llbracket \mathsf{m}^\star \rrbracket_{\hat{\jmath}+1} = 0$. For the case $p \geq \hat{\imath}$ and $\llbracket \mathsf{m}^\star \rrbracket_{\hat{\jmath}+1} = 1$ note that for all $i \in \{\hat{\imath}, \dots, p\}$, $\mathbf{x}_{i,\hat{\jmath}+1,1}$ is identically distributed as $\mathbf{x}_{i,\hat{\jmath}+1,1} + \mathbf{B}_0^*\mathbf{w}_i$ for $\mathbf{w}_i \overset{\$}{\leftarrow} \mathbb{Z}_q^k$ and $\mathbf{w}_i$ is hidden to the adversary since in all Eval queries (with $p \geq \hat{\imath}$) there is either $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0$ which means that $\mathbf{x}_{i,\hat{\jmath}+1,1}$ (for all $i \in \{\hat{\imath}, \dots, p\}$) is not used to compute the tag or there is $\llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 1$ which means that $\mathbf{t}_i \in \mathsf{Span}(\mathbf{B}|\mathbf{B}_1)$ and thus the $\mathbf{B}_0^*$ -part of $\mathbf{x}_{i,\hat{\jmath}+1,1}$ cancels out. All in all this means that the value $\mathbf{h}_{0,i}$ is the only one in the game that depends on $\mathbf{w}_i$ and thus the $\mathbf{B}_0^*$-part of $\mathbf{h}_{0,i}$ is uniformly random to the adversary. Especially it is distributed identically in both games. □

Lemma 3.26 ($\mathsf{G}_{3,\hat{\jmath},2} \rightsquigarrow \mathsf{G}_{3,\hat{\jmath},3}$).

$$\Pr\big[\mathsf{G}^{\mathcal{A}}_{3,\hat{\jmath},2} \Rightarrow 1\big] = \Pr\big[\mathsf{G}^{\mathcal{A}}_{3,\hat{\jmath},3} \Rightarrow 1\big]$$

Proof. We define

$$\mathsf{RF}^{(1)}_{i,\hat{\jmath}+1}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\Big) := \begin{cases} \mathsf{RF}^{(1)}_{i,\hat{\jmath}}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\Big) + \mathsf{RF}'^{(1)}_{i,\hat{\jmath}}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\Big) & \text{if } \llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 0 \\ \mathsf{RF}^{(1)}_{i,\hat{\jmath}}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\Big) & \text{if } \llbracket \mathsf{m} \rrbracket_{\hat{\jmath}+1} = 1 \end{cases},$$

where $\mathsf{RF}'^{(1)}_{i,\hat{\jmath}} : \{0,1\}^{\hat{\jmath}} \to \mathbb{Z}_q^{1\times k}$ is another independent random function. Since $\mathsf{RF}^{(1)}_{i,\hat{\jmath}}$ in not used in game $\mathsf{G}_{3,\hat{\jmath},3}$, $\mathsf{RF}^{(1)}_{i,\hat{\jmath}+1}$ is a random function.

The argument, that the games $\mathsf{G}_{3,\hat{\jmath},2}$ and $\mathsf{G}_{3,\hat{\jmath},3}$ are identically distributed, is the same as in Lemma 3.25, just with the roles of 0 and 1 swapped. □

Lemma 3.27 ($\mathsf{G}_{3,\hat{\jmath},3} \rightsquigarrow \mathsf{G}_{3,\hat{\jmath}+1}$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\big|\Pr\big[\mathsf{G}^{\mathcal{A}}_{3,\hat{\jmath},3} \Rightarrow 1\big] - \Pr\big[\mathsf{G}^{\mathcal{A}}_{3,\hat{\jmath}+1} \Rightarrow 1\big]\big| \leq 4k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{2}{q-1}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. In game $\mathsf{G}_{3,\hat{\jmath},3}$, replace all occurrences of the term $\mathsf{RF}^{(0)}_{\hat{\imath},\hat{\jmath}+1}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\Big)(\mathbf{B}_0^*)^\top + \mathsf{RF}^{(1)}_{\hat{\imath},\hat{\jmath}+1}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}+1}\Big)(\mathbf{B}_1^*)^\top$ with $\mathsf{RF}_{\hat{\imath},\hat{\jmath}}\Big(\llbracket \mathsf{m} \rrbracket_{|\hat{\jmath}}\Big)\big(\mathbf{B}^\perp\big)^\top$ to avoid computing $\mathbf{B}_0^*$ and $\mathbf{B}_1^*$. This does not change the distribution, since $\mathbf{B}_0^*, \mathbf{B}_1^*$ is a basis for $\mathsf{Span}\big(\mathbf{B}^\perp\big)$. The remaining transition is the reverse of Lemma 3.24. □

Lemma 3.28 ($\mathsf{G}_{3,L\alpha} \rightsquigarrow \mathsf{G}_4$).

$$\big|\Pr\big[\mathsf{G}^{\mathcal{A}}_{3,L\alpha} \Rightarrow 1\big] - \Pr\big[\mathsf{G}^{\mathcal{A}}_4 \Rightarrow 1\big]\big| \leq \frac{Q}{q^{2k}}$$

```
$\underline{\mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2):}$
$K \xleftarrow{\$} \mathsf{HGen}(1^\lambda)$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
$\mathbf{for}\ i \in \{1, \ldots, L\},\ j \in \{1, \ldots, \gamma\},\ b \in \{0,1\}\ \mathbf{do}\ \mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$x'_0 \xleftarrow{\$} \mathbb{Z}_q$
$\mathbf{return}\ \mathsf{sk}_{\mathsf{MAC}} := \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \gamma, b \in \{0,1\}}, x'_0, K\big)$

$\underline{\mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p):}$
$\mathbf{Parse}\ \mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \gamma, b \in \{0,1\}}, x'_0, K\big)$
$\mathbf{for}\ i \in \{1, \ldots, p\}\ \mathbf{do}\ \mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t}_i := \mathbf{B}\mathbf{s}_i$
$u := \sum_{i=1}^p \Big(\sum_{j=1}^\gamma \mathbf{x}_{i,j,\mathsf{HEval}_K(\mathsf{m}_{|i})_j}^\top\Big)\mathbf{t}_i + x'_0$
$\mathbf{return}\ \Big(\big([\mathbf{t}_i]_2\big)_{1 \le i \le p}, [u]_2\Big)$

$\underline{\mathsf{Ver}_{\mathsf{MAC}}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p, \tau):}$
$\mathbf{Parse}\ \mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \gamma, b \in \{0,1\}}, x'_0, K\big)$
$\mathbf{Parse}\ \mathsf{m} =: (\mathsf{m}_1, \ldots, \mathsf{m}_p)$
$\mathbf{Parse}\ \ \tau =: \Big(\big([\mathbf{t}_i]_2\big)_{1 \le i \le p}, [u]_2\Big)$
$\mathbf{return}\ u \stackrel{?}{=} \sum_{i=1}^p \Big(\sum_{j=1}^\gamma \mathbf{x}_{i,j,\mathsf{HEval}_K(\mathsf{m}_{|i})_j}^\top\Big)\mathbf{t}_i + x'_0$
```

Figure 11: The second affine MAC improved with a hash function

Proof. The challenge query evaluates $\mathsf{RF}_{i,i\alpha}$ only for the input value $\mathsf{m}^\star_{|i}$. Assume $\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} = \emptyset$, otherwise the adversary has lost the game anyway. In each user secret key the value $\mathsf{RF}_{p,p\alpha}(\mathsf{m})\big(\mathbf{B}^\perp\big)^\top \mathbf{t}_p$ is part of u. This is the only place where $\mathsf{RF}_{p,p\alpha}(\mathsf{m})$ is used, since only the first Eval query for each evaluates the random function. Each query outputs a uniformly random value for u when $\mathbf{t}_p \notin \mathsf{Span}(\mathbf{B})$, which happens with probability $\ge 1 - 1/q^{2k}$. In this case h_1 is the only value depending on x'_0 and thus uniform random as well. □

Summary. To prove Theorem 3.19, we combine Lemmas 3.20–3.28 to change h_1 from real to random and then apply all Lemmas in reverse order to get to the $\mathsf{HPR}_0\text{-}\mathsf{CMA}_{\mathsf{rand}}$ game. □

Optimization. $\mathsf{MAC}_2[\mathcal{U}_{3k,k}]$ can be improved, like $\mathsf{MAC}_1[\mathcal{U}_{3k,k}]$, with a collision-resistant hash function. Again we replace in the equation for u the term $\sum_j f_j\big(\mathsf{m}_{|i}\big)\mathbf{x}_j^\top$ with $\sum_j f_j\big(H\big(\mathsf{m}_{|i}\big)\big)\mathbf{x}_j^\top$, where H is a collision resistant hash function.

Formally, we need a family of hash functions $\mathcal{H} := (\mathsf{HGen}, \mathsf{HEval})$ with domain $\mathcal{S}^{\le L}$ and range $\{0,1\}^\gamma$ for all hash keys. The affine MAC $\mathsf{MAC}_2^{\mathcal{H}}[\mathcal{U}_{3k,k}]$ is shown in Figure 11.

Theorem 3.29 (Security of $\mathsf{MAC}_2^{\mathcal{H}}[\mathcal{U}_{3k,k}]$). $\mathsf{MAC}_2^{\mathcal{H}}[\mathcal{U}_{3k,k}]$ *is tightly* $\mathsf{HPR}_0\text{-}\mathsf{CMA}$ *secure in* $\mathbb{G}_2$ *when* $\mathcal{H}$ *is collision resistant and the* $\mathcal{U}_k$-MDDH *assumption holds for* $\mathbb{G}_2$*. More precisely, for all adversaries* $\mathcal{A}$ *there exist adversaries* $\mathcal{B}$ *and* $\mathcal{C}$ *with*

$$\mathsf{Adv}^{\mathsf{hpr_0\text{-}cma}}_{\mathsf{MAC}_2^{\mathcal{H}}[\mathcal{U}_{3k,k}],\mathbb{G}_2}(\mathcal{A}) \le (4k + 16k\gamma)\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{2 + 8k\gamma}{q-1} + \mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H}}(\mathcal{C}) + \frac{2Q}{q^{2k}}$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ *and* $T(\mathcal{C}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$*, where* Q *denotes the number of* Eval *queries of* $\mathcal{A}$ *and* poly *is a polynomial independent of* $\mathcal{A}$*.*

We omit the proof of this theorem because it is very similar to the proof of Theorem 3.3, just with the changes we already mentioned for theorem Theorem 3.18.

4 Transformation to HIBE

Any affine MAC with levels can be transformed tightly to a hierarchical identity-based key encapsulation mechanism (HIBKEM) under the $\mathcal{D}_k$-MDDH assumption in $\mathbb{G}_1$. The transformation is shown in Figure 12. It is a generalization of the transformation from delegatable, affine MACs to HIBKEMs in

```
$\mathsf{Gen}(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, P_1, P_2, e)$:
$\mathsf{sk}_{\mathsf{MAC}} \overset{\$}{\leftarrow} \mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2)$
$\mathsf{sk}_{\mathsf{MAC}} =: \left(\mathbf{B}, (\mathbf{x}_{l,i,j})_{1 \le l \le \ell(L), 1 \le i \le L, 1 \le j \le \ell'(l,i)}, x'_0\right)$
$\mathbf{A} \overset{\$}{\leftarrow} \mathcal{D}_k$
for $l \in \{1, \dots, \ell(L)\},\ i \in \{1, \dots, L\},\ j \in \{1, \dots$
$\dots, \ell'(l,i)\}$ do
    $\mathbf{Y}_{l,i,j} \overset{\$}{\leftarrow} \mathbb{Z}_q^{k \times n};\ \mathbf{Z}_{l,i,j} := \left(\mathbf{Y}_{l,i,j}^\top \mid \mathbf{x}_{l,i,j}\right) \cdot \mathbf{A}$
    $\mathbf{d}_{l,i,j} := \mathbf{x}_{l,i,j}^\top \cdot \mathbf{B};\ \mathbf{E}_{l,i,j} := \mathbf{Y}_{l,i,j} \cdot \mathbf{B}$
$\mathbf{y}'_0 \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{z}'_0 := \left(\mathbf{y}_0'^\top \mid x'_0\right) \cdot \mathbf{A}$
$\mathsf{pk} := \left(\mathcal{G}, [\mathbf{A}]_1, ([\mathbf{Z}_{l,i,j}]_1)_{\substack{1 \le l \le \ell(L), 1 \le i \le L,, \\ 1 \le j \le \ell'(l,i)}}, [\mathbf{z}'_0]_1\right)$
$\mathsf{dk} := \left([\mathbf{B}]_2, ([\mathbf{d}_{l,i,j}]_2, [\mathbf{E}_{l,i,j}]_2)_{\substack{1 \le l \le \ell(L), 1 \le i \le L, \\ 1 \le j \le \ell'(l,i)}}\right)$
$\mathsf{sk} := \left(\mathsf{sk}_{\mathsf{MAC}}, (\mathbf{Y}_{l,i,j})_{1 \le l \le \ell(p), 1 \le i \le L, 1 \le j \le \ell'(l,i)}, \mathbf{y}'_0\right)$
return $(\mathsf{pk}, \mathsf{dk}, \mathsf{sk})$
```

```
$\mathsf{Enc}(\mathsf{pk}, \mathsf{id} \in \mathcal{S}^p)$:
$\mathbf{r} \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{c}_0 := \mathbf{A}\mathbf{r}$
for $l \in \{1, \dots, \ell(p)\}$ do
    $\mathbf{c}_{1,l} := \sum_{i=1}^{p} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}\left(\mathsf{id}_{|i}\right)\mathbf{Z}_{l,i,j}\mathbf{r}$
$\mathsf{C} := \left([\mathbf{c}_0]_1, ([\mathbf{c}_{1,l}]_1)_{1 \le l \le \ell(p)}\right)$
$\mathsf{K} := \mathbf{z}'_0 \cdot \mathbf{r}$
return $([K]_T, \mathsf{C})$
```

```
$\mathsf{Dec}(\mathsf{usk}[\mathsf{id}], \mathsf{id} \in \mathcal{S}^p, \mathsf{C})$:
$\mathsf{usk}[\mathsf{id}] =: \left(([\mathbf{t}_l]_2)_{1 \le l \le \ell(p)}, [u]_2, [\mathbf{v}]_2\right)$
Parse $\mathsf{C} =: \left([\mathbf{c}_0]_1, ([\mathbf{c}_{1,l}]_1)_{1 \le l \le \ell(p)}\right)$
$[\mathsf{K}]_T := e\left([\mathbf{c}_0^\top]_1, \begin{bmatrix} \mathbf{v} \\ u \end{bmatrix}_2\right) - \sum_{l=1}^{\ell(p)} e\left(\left[\mathbf{c}_{1,l}^\top\right]_1, [\mathbf{t}_l]_2\right)$
return $[\mathsf{K}]_T$
```

```
$\mathsf{Ext}(\mathsf{sk}, \mathsf{id} \in \mathcal{S}^p)$:
$\left(([\mathbf{t}_l]_2)_{1 \le l \le \ell(p)}, [u]_2\right) \overset{\$}{\leftarrow} \mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{id})$
$\mathbf{v} := \sum_{l=1}^{\ell(p)} \left(\sum_{i=1}^{p} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}\left(\mathsf{m}_{|i}\right)\mathbf{Y}_{l,i,j}\right)\mathbf{t}_l + \mathbf{y}'_0$
for $l \in \{1, \dots, \ell(p)\},\ i \in \{p+1, \dots, L\},\ j \in$
$\{1, \dots, \ell'(l,i)\}$ do
    $d_{l,i,j} := \mathbf{x}_{l,i,j}^\top \mathbf{t}_l;\ \mathbf{e}_{l,i,j} := \mathbf{Y}_{l,i,j}\mathbf{t}_l$
$\mathsf{usk}[\mathsf{id}] := \left(([\mathbf{t}_l]_2)_{1 \le l \le \ell(p)}, [u]_2, [\mathbf{v}]_2\right)$
$\mathsf{udk}[\mathsf{id}] := \left([d_{l,i,j}]_2, [\mathbf{e}_{l,i,j}]_2\right)_{\substack{1 \le l \le \ell(L), p+1 \le i \le L, \\ 1 \le j \le \ell'(l,i)}}$
return $(\mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}])$
```

```
$\mathsf{Del}(\mathsf{dk}, \mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}], \mathsf{id} \in \mathcal{S}^p, \mathsf{id}_{p+1} \in \mathcal{S})$:
Parse $\mathsf{usk}[\mathsf{id}] =: \left(([\mathbf{t}_l]_2)_{1 \le l \le \ell(p)}, [u]_2, [\mathbf{v}]_2\right)$
$\mathsf{udk}[\mathsf{id}] =: \left([d_{l,i,j}]_2, [\mathbf{e}_{l,i,j}]_2\right)_{\substack{1 \le l \le \ell(L), p+1 \le i \le L, \\ 1 \le j \le \ell'(l,i)}}$
for $l \in \{\ell(p)+1, \dots, \ell(p+1)\}$ do $\mathbf{t}_l := \mathbf{0}$
for $l \in \{1, \dots, \ell(p+1)\}$ do
    $\mathbf{s}'_l \overset{\$}{\leftarrow} \mathbb{Z}_q^{n'};\ \mathbf{t}'_l := \mathbf{t}_l + \mathbf{B}\mathbf{s}'_l$
$\mathsf{id}' := (\mathsf{id}_1, \dots, \mathsf{id}_p, \mathsf{id}_{p+1})$
$u' := u + \sum_{l=1}^{\ell(p)} \sum_{j=1}^{\ell'(l,p+1)} f_{l,p+1,j}\left(\mathsf{id}'\right)d_{l,p+1,j}$
$\qquad + \sum_{l=1}^{\ell(p+1)} \left(\sum_{i=1}^{p+1} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}\left(\mathsf{id}'_{|i}\right)\mathbf{d}_{l,i,j}\right)\mathbf{s}'_l$
$\mathbf{v}' := \mathbf{v} + \sum_{l=1}^{\ell(p)} \sum_{j=1}^{\ell'(l,p+1)} f_{l,p+1,j}\left(\mathsf{id}'\right)\mathbf{e}_{l,p+1,j}$
$\qquad + \sum_{l=1}^{\ell(p+1)} \left(\sum_{i=1}^{p+1} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}\left(\mathsf{id}'_{|i}\right)\mathbf{E}_{l,i,j}\right)\mathbf{s}'_l$
for $l \in \{1, \dots, \ell(p)\},\ i \in \{p+2, \dots, L\},\ j \in$
$\{1, \dots, \ell'(l,i)\}$ do
    $d'_{l,i,j} := d_{l,i,j} + \mathbf{d}_{l,i,j}\mathbf{s}'_l$
    $\mathbf{e}'_{l,i,j} := \mathbf{e}_{l,i,j} + \mathbf{E}_{l,i,j}\mathbf{s}'_l$
for $l \in \{\ell(p)+1, \dots, \ell(p+1)\},\ i \in \{p+2, \dots, L\},$
$j \in \{1, \dots, \ell'(l,i)\}$ do
    $d'_{l,i,j} := \mathbf{d}_{l,i,j}\mathbf{s}'_l$
    $\mathbf{e}'_{l,i,j} := \mathbf{E}_{l,i,j}\mathbf{s}'_l$
$\mathsf{usk}[\mathsf{id}'] := \left(([\mathbf{t}'_l]_2)_{1 \le l \le \ell(p+1)}, [u']_2, [\mathbf{v}']_2\right)$
$\mathsf{udk}[\mathsf{id}'] := \left(\left[d'_{l,i,j}\right]_2, \left[\mathbf{e}'_{l,i,j}\right]_2\right)_{\substack{1 \le l \le \ell(p+1), p+2 \le i \le L, \\ 1 \le j \le \ell'(l,i)}}$
return $\left(\mathsf{usk}[\mathsf{id}'], \mathsf{udk}[\mathsf{id}']\right)$
```

Figure 12: The Transformation HIBKEM of an affine MAC with levels to an HIBKEM

[BKP14]. We only consider HIBKEM here, and one can easily prove that every HIBKEM can be transformed (tightly) into an HIBE scheme with a (one-time secure) symmetric cipher by adapting a similar transformation for public-key encryption in [HK07].

Theorem 4.1 (Security of the HIBKEM transformation). *The HIBKEM* HIBKEM[MAC, $\mathcal{D}_k$] *is* IND-HID-CPA *secure in* $\mathcal{G}$ *under the* $\mathcal{D}_k$-MDDH *assumption for* $\mathbb{G}_1$ *if* MAC *is* HPR$_0$-CMA *secure in* $\mathbb{G}_2$*. More precisely, for all adversaries* $\mathcal{A}$ *there exist adversaries* $\mathcal{B}_1$ *and* $\mathcal{B}_2$ *with*

$$\mathsf{Adv}^{\mathsf{ind\text{-}hid\text{-}cpa}}_{\mathsf{HIBKEM}[\mathsf{MAC},\mathcal{D}_k],\mathcal{G}}(\mathcal{A}) \le \mathsf{Adv}^{\mathsf{hpr_0\text{-}cma}}_{\mathsf{MAC},\mathbb{G}_2}(\mathcal{B}_1) + 2\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{D}_k,\mathsf{PGGen},1}(\mathcal{B}_2)$$

and $T(\mathcal{B}_1) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ *and* $T(\mathcal{B}_2) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$*, where* Q *denotes the number of* Ext *queries of* $\mathcal{A}$ *and* poly *is a polynomial independent of* $\mathcal{A}$*.*

The detailed proof of Theorem 4.1 can be found in Appendix B.

```
$\mathsf{Gen}(1^\lambda)$:
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$; $\mathbf{A} \xleftarrow{\$} \mathcal{D}_k$
for $i \in \{1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$ do
    $\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{3k}$; $\mathbf{Y}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{k\times 3k}$
    $\mathbf{Z}_{i,j,b} := \left(\mathbf{Y}_{i,j,b}^\top \mid \mathbf{x}_{i,j,b}\right) \cdot \mathbf{A}$
    $\mathbf{d}_{i,j,b} := \mathbf{x}_{i,j,b}^\top \mathbf{B}$; $\mathbf{E}_{i,j,b} := \mathbf{Y}_{i,j,b}\mathbf{B}$
$x_0' \xleftarrow{\$} \mathbb{Z}_q$; $\mathbf{y}_0' \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{z}_0' := \left(\mathbf{y}_0'^\top \mid x_0'\right) \cdot \mathbf{A}$
$\widetilde{\mathbf{Z}} := \left([\mathbf{Z}_{i,j,b}]_1\right)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}$
$\mathsf{pk} := \left(\mathcal{G}, [\mathbf{A}]_1, \widetilde{\mathbf{Z}}, [\mathbf{z}_0']_1\right)$
$\widetilde{\mathsf{dk}} := \left([\mathbf{d}_{i,j,b}]_2, [\mathbf{E}_{i,j,b}]_2\right)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}$
$\mathsf{dk} := \left([\mathbf{B}]_2, \widetilde{\mathsf{dk}}\right)$
$\mathsf{sk} := \left(\mathsf{sk}_{\mathsf{MAC}}, (\mathbf{Y}_{i,j,b})_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}, \mathbf{y}_0'\right)$
return $(\mathsf{pk}, \mathsf{dk}, \mathsf{sk})$

$\mathsf{Enc}(\mathsf{pk}, \mathsf{id} \in \mathcal{S}^p)$:
$\mathbf{r} \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{c}_0 := \mathbf{A}\mathbf{r}$
$\mathbf{c}_1 := \sum_{i=1}^{p} \sum_{j=1}^{i\alpha} \mathbf{Z}_{i,j,[\![\mathsf{id}]\!]_j} \mathbf{r}$
$\mathsf{C} := ([\mathbf{c}_0]_1, [\mathbf{c}_1]_1)$
$\mathsf{K} := \mathbf{z}_0' \cdot \mathbf{r}$
return $([\mathsf{K}]_T, \mathsf{C})$

$\mathsf{Dec}(\mathsf{usk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{C})$:
Parse $\mathsf{usk} =: ([\mathbf{t}]_2, [u]_2, [\mathbf{v}]_2)$
Parse $\mathsf{C} =: ([\mathbf{c}_0]_1, [\mathbf{c}_1]_1)$
$[\mathsf{K}]_T := e\left([\mathbf{c}_0^\top]_1, \begin{bmatrix} \mathbf{v} \\ u \end{bmatrix}_2\right) - e\left([\mathbf{c}_1^\top]_1, [\mathbf{t}]_2\right)$
return $[\mathsf{K}]_T$
```

```
$\mathsf{Ext}(\mathsf{sk}, \mathsf{id} \in \mathcal{S}^p)$:
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t} := \mathbf{B}\mathbf{s}$
$u := \sum_{i=1}^{p} \sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{id}]\!]_j}^\top \mathbf{t} + x_0'$
$\mathbf{v} := \sum_{i=1}^{p} \sum_{j=1}^{i\alpha} \mathbf{Y}_{i,j,[\![\mathsf{id}]\!]_j} \mathbf{t} + \mathbf{y}_0'$
for $i \in \{p+1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$
do $d_{i,j,b} := \mathbf{x}_{i,j,b}^\top \mathbf{t}$; $\mathbf{e}_{i,j,b} := \mathbf{Y}_{i,j,b}\mathbf{t}$
$\mathsf{usk}[\mathsf{id}] := ([\mathbf{t}]_2, [u]_2, [\mathbf{v}]_2)$
$\mathsf{udk}[\mathsf{id}] := \left([d_{i,j,b}]_2, [\mathbf{e}_{i,j,b}]_2\right)_{\substack{p+1\le i\le L, b\in\{0,1\},\\ 1\le j\le i\alpha}}$
return $(\mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}])$

$\mathsf{Del}(\mathsf{dk}, \mathsf{usk}, \mathsf{udk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{id}_{p+1} \in \mathcal{S})$:
Parse $\mathsf{usk} =: ([\mathbf{t}]_2, [u]_2, [\mathbf{v}]_2)$
$\mathbf{s}' \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t}' := \mathbf{t} + \mathbf{B}\mathbf{s}'$
$\mathsf{id}' := (\mathsf{id}_1, \dots, \mathsf{id}_p, \mathsf{id}_{p+1})$
$u' := u + \sum_{j=1}^{(p+1)\alpha} d_{p+1,j,[\![\mathsf{id}']\!]_j} + \left(\sum_{i=1}^{p+1} \sum_{j=1}^{i\alpha} \mathbf{d}_{i,j,[\![\mathsf{id}']\!]_j}\right)\mathbf{s}'$
$\mathbf{v}' := \mathbf{v} + \sum_{j=1}^{(p+1)\alpha} \mathbf{e}_{p+1,j,[\![\mathsf{id}']\!]_j} + \left(\sum_{i=1}^{p+1} \sum_{j=1}^{i\alpha} \mathbf{E}_{i,j,[\![\mathsf{id}']\!]_j}\right)\mathbf{s}'$
for $i \in \{p+2,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$
do
    $d_{i,j,b}' := d_{i,j,b} + \mathbf{d}_{i,j,b}\mathbf{s}'$
    $\mathbf{e}_{i,j,b}' := \mathbf{e}_{i,j,b} + \mathbf{E}_{i,j,b}\mathbf{s}'$
$\mathsf{usk}' := ([\mathbf{t}']_2, [u']_2, [\mathbf{v}']_2)$
$\mathsf{udk}' := \left(\left[d_{i,j,b}'\right]_2, \left[\mathbf{e}_{i,j,b}'\right]_2\right)_{\substack{p+2\le i\le L, b\in\{0,1\},\\ 1\le j\le i\alpha}}$
return $(\mathsf{usk}', \mathsf{udk}')$
```

Figure 13: The resulting scheme $\mathsf{HIBKEM}_1 := \mathsf{HIBKEM}[\mathsf{MAC}_1[\mathcal{U}_{3k,k}], \mathcal{D}_k]$

4.1 Instantiations

4.1.1 MDDH.

The result of applying the HIBKEM transformation to $\mathsf{MAC}_1[\mathcal{U}_{3k,k}]$ is shown in Figure 13. The scheme has $\alpha\left(L^2+L\right)\left(4k^2+k\right)+3k^2+2k$ group elements in the public key and $4k+1$ group elements in the ciphertext. The user secret keys have at most $\alpha\left(L^2/2+L/2-1\right)(k+1)+4k+1$ group elements. Identities that are deeper in the hierarchy have smaller secret keys since the user secret key size is dominated by the size of the delegation keys. On the last level, the user secret keys consist of only $4k+1$ group elements.

The result of applying the HIBKEM transformation to $\mathsf{MAC}_2[\mathcal{U}_{3k,k}]$ is shown in Figure 14. The scheme has $\alpha\left(L^2+L\right)\left(4k^2+k\right)+3k^2+2k$ group elements in the public key and $3Lk+k+1$ group elements in the ciphertext. The user secret keys have at most $3Lk+k+1$ group elements. Identities that are deeper in the hierarchy have larger secret keys.

The schemes have both the same public key. The first scheme has smaller ciphertexts, while the second has a more efficient reduction and smaller user secret keys in the worst case.

4.1.2 SXDH.

With a type III pairing, both of our schemes can be instantiated with the SXDH assumption.

The result (HIBKEM_1) of instantiating scheme $\mathsf{HIBKEM}[\mathsf{MAC}_1[\mathcal{U}_{3k,k}], \mathcal{D}_k]$ with the SXDH assumption is shown in Figure 15. The scheme has $5\alpha\left(L^2+L\right)+5$ group elements in the public key and 5 group elements in the ciphertext. The user secret keys have at most $2\alpha\left(L^2+L-2\right)+5$ group elements.

The result (HIBKEM_2) of instantiating scheme $\mathsf{HIBKEM}[\mathsf{MAC}_2[\mathcal{U}_{3,1}], \mathcal{U}_{2,1}]$ with the SXDH assumption is shown in Figure 16. The scheme has $5\alpha\left(L^2+L\right)+5$ group elements in the public key and $3L+2$ group elements in the ciphertext. The user secret keys have at most $3L+2$ group elements.

```
$\underline{\mathsf{Gen}(1^\lambda)}$:
$\mathbf{B} \overset{\$}{\leftarrow} \mathcal{U}_{3k,k};\ \mathbf{A} \overset{\$}{\leftarrow} \mathcal{D}_k$
for $i \in \{1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$ do
    $\mathbf{x}_{i,j,b} \overset{\$}{\leftarrow} \mathbb{Z}_q^{3k};\ \mathbf{Y}_{i,j,b} \overset{\$}{\leftarrow} \mathbb{Z}_q^{k\times 3k}$
    $\mathbf{Z}_{i,j,b} := \left(\mathbf{Y}_{i,j,b}^\top \mid \mathbf{x}_{i,j,b}\right) \cdot \mathbf{A}$
    $\mathbf{d}_{i,j,b} := \mathbf{x}_{i,j,b}^\top \mathbf{B};\ \mathbf{E}_{i,j,b} := \mathbf{Y}_{i,j,b}\mathbf{B}$
$x'_0 \overset{\$}{\leftarrow} \mathbb{Z}_q;\ \mathbf{y}'_0 \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{z}'_0 := \left(\mathbf{y}_0'^\top \mid x'_0\right) \cdot \mathbf{A}$
$\mathsf{pk} := \left(\mathcal{G}, [\mathbf{A}]_1, ([\mathbf{Z}_{i,j,b}]_1)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}, [\mathbf{z}'_0]_1\right)$
$\mathsf{dk} := \left([\mathbf{B}]_2, ([\mathbf{d}_{i,j,b}]_2, [\mathbf{E}_{i,j,b}]_2)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}\right)$
$\mathsf{sk} := \left(\mathsf{sk}_{\mathsf{MAC}}, (\mathbf{Y}_{i,j,b})_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}, \mathbf{y}'_0\right)$
return $(\mathsf{pk}, \mathsf{dk}, \mathsf{sk})$

$\underline{\mathsf{Ext}(\mathsf{sk}, \mathsf{id} \in \mathcal{S}^p)}$:
for $i \in \{1,\dots,p\}$ do $\mathbf{s}_i \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{t}_i := \mathbf{B}\mathbf{s}_i$
$u := \sum_{i=1}^{p}\left(\sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{id}]\!]_j}^\top\right)\mathbf{t}_i + x'_0$
$\mathbf{v} := \sum_{i=1}^{p}\left(\sum_{j=1}^{i\alpha} \mathbf{Y}_{i,j,[\![\mathsf{id}]\!]_j}\right)\mathbf{t}_i + \mathbf{y}'_0$
return $\left(([\mathbf{t}_i]_2)_{1\le i\le p}, [u]_2, [\mathbf{v}]_2\right)$

$\underline{\mathsf{Del}(\mathsf{dk}, \mathsf{usk}, \mathsf{udk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{id}_{p+1} \in \mathcal{S})}$:
Parse $\mathsf{usk} =: \left(([\mathbf{t}_i]_2)_{1\le i\le p}, [u]_2, [\mathbf{v}]_2\right)$
for $i \in \{1,\dots,p\}$ do $\mathbf{s}'_i \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{t}'_i := \mathbf{t}_i + \mathbf{B}\mathbf{s}'_i$
$\mathbf{s}'_{p+1} \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{t}'_{p+1} := \mathbf{B}\mathbf{s}'_{p+1}$
$\mathsf{id}' := (\mathsf{id}_1, \dots, \mathsf{id}_p, \mathsf{id}_{p+1})$
$u' := u + \sum_{i=1}^{p+1}\left(\sum_{j=1}^{i\alpha} \mathbf{d}_{i,j,[\![\mathsf{id}']\!]_j}\right)\mathbf{s}'_i$
$\mathbf{v}' := \mathbf{v} + \sum_{i=1}^{p+1}\left(\sum_{j=1}^{i\alpha} \mathbf{E}_{i,j,[\![\mathsf{id}']\!]_j}\right)\mathbf{s}'_i$
return $\left(([\mathbf{t}']_2)_{1\le i\le p+1}, [u']_2, [\mathbf{v}']_2\right)$

$\underline{\mathsf{Enc}(\mathsf{pk}, \mathsf{id} \in \mathcal{S}^p)}$:
$\mathbf{r} \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{c}_0 := \mathbf{A}\mathbf{r}$
for $i \in \{1,\dots,p\}$ do $\mathbf{c}_{1,i} := \sum_{j=1}^{i\alpha} \mathbf{Z}_{i,j,[\![\mathsf{id}]\!]_j}\mathbf{r}$
$\mathsf{C} := \left([\mathbf{c}_0]_1, ([\mathbf{c}_{1,i}]_1)_{1\le i\le p}\right)$
$\mathsf{K} := \mathbf{z}'_0 \cdot \mathbf{r}$
return $([\mathsf{K}]_T, \mathsf{C})$

$\underline{\mathsf{Dec}(\mathsf{usk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{C})}$:
Parse $\mathsf{usk} =: \left(([\mathbf{t}_i]_2)_{1\le i\le p}, [u]_2, [\mathbf{v}]_2\right)$
Parse $\mathsf{C} =: \left([\mathbf{c}_0]_1, ([\mathbf{c}_{1,i}]_1)_{1\le i\le p}\right)$
$[\mathsf{K}]_T := e\left(\left[\mathbf{c}_0^\top\right]_1, \begin{bmatrix} \mathbf{v} \\ u \end{bmatrix}_2\right)$
            $- \sum_{i=1}^{p}\left(e\left(\left[\mathbf{c}_{1,i}^\top\right]_1, [\mathbf{t}_i]_2\right)\right)$
return $[\mathsf{K}]_T$
```

Figure 14: The resulting scheme $\mathsf{HIBKEM}_2 := \mathsf{HIBKEM}[\mathsf{MAC}_2[\mathcal{U}_{3k,k}], \mathcal{D}_k]$

```
$\underline{\mathsf{Gen}(1^\lambda)}$:
$\mathbf{B} \overset{\$}{\leftarrow} \mathcal{U}_{3,1};\ \mathbf{A} \overset{\$}{\leftarrow} \mathcal{U}_{2,1}$
for $i \in \{1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$ do
    $\mathbf{x}_{i,j,b} \overset{\$}{\leftarrow} \mathbb{Z}_q^{3}\ \mathbf{Y}_{i,j,b} \overset{\$}{\leftarrow} \mathbb{Z}_q^{1\times 3}$
    $\mathbf{Z}_{i,j,b} := \left(\mathbf{Y}_{i,j,b}^\top \mid \mathbf{x}_{i,j,b}\right) \cdot \mathbf{A}$
    $\mathbf{d}_{i,j,b} := \mathbf{x}_{i,j,b}^\top \mathbf{B};\ \mathbf{E}_{i,j,b} := \mathbf{Y}_{i,j,b}\mathbf{B}$
$x'_0 \overset{\$}{\leftarrow} \mathbb{Z}_q;\ \mathbf{y}'_0 \overset{\$}{\leftarrow} \mathbb{Z}_q;\ \mathbf{z}'_0 := \left(\mathbf{y}_0'^\top \mid x'_0\right) \cdot \mathbf{A}$
$\mathsf{pk} := \left(\mathcal{G}, [\mathbf{A}]_1, ([\mathbf{Z}_{i,j,b}]_1)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}, [\mathbf{z}'_0]_1\right)$
$\mathsf{dk} := \left([\mathbf{B}]_2, ([\mathbf{d}_{i,j,b}]_2, [\mathbf{E}_{i,j,b}]_2)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}\right)$
$\mathsf{sk} := \left(\mathsf{sk}_{\mathsf{MAC}}, (\mathbf{Y}_{i,j,b})_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}, \mathbf{y}'_0\right)$
return $(\mathsf{pk}, \mathsf{dk}, \mathsf{sk})$

$\underline{\mathsf{Ext}(\mathsf{sk}, \mathsf{id} \in \mathcal{S}^p)}$:
$\mathbf{s} \overset{\$}{\leftarrow} \mathbb{Z}_q;\ \mathbf{t} := \mathbf{B}\mathbf{s}$
$u := \left(\sum_{i=1}^{p}\sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{id}]\!]_j}^\top\right)\mathbf{t} + x'_0$
$\mathbf{v} := \left(\sum_{i=1}^{p}\sum_{j=1}^{i\alpha} \mathbf{Y}_{i,j,[\![\mathsf{id}]\!]_j}\right)\mathbf{t} + \mathbf{y}'_0$
for $i \in \{p+1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,1\}$
do $d_{i,j,b} := \mathbf{x}_{i,j,b}^\top\mathbf{t};\ \mathbf{e}_{i,j,b} := \mathbf{Y}_{i,j,b}\mathbf{t}$
$\mathsf{usk} := ([\mathbf{t}]_2, [u]_2, [\mathbf{v}]_2)$
$\mathsf{udk} := \left([d_{i,j,b}]_2, [\mathbf{e}_{i,j,b}]_2\right)_{p+1\le i\le L, b\in\{0,1\}, 1\le j\le i\alpha}$
return $(\mathsf{usk}, \mathsf{udk})$

$\underline{\mathsf{Del}(\mathsf{dk}, \mathsf{usk}, \mathsf{udk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{id}_{p+1} \in \mathcal{S})}$:
Parse $\mathsf{usk} =: ([\mathbf{t}]_2, [u]_2, [\mathbf{v}]_2)$
$\mathbf{s}' \overset{\$}{\leftarrow} \mathbb{Z}_q;\ \mathbf{t}' := \mathbf{t} + \mathbf{B}\mathbf{s}'$
$u' := u + \left(\sum_{i=1}^{p+1}\sum_{j=1}^{i\alpha} \mathbf{d}_{i,j,[\![\mathsf{id}]\!]_j}\right)\mathbf{s}'$
$\mathbf{v}' := \mathbf{v} + \left(\sum_{i=1}^{p+1}\sum_{j=1}^{i\alpha} \mathbf{E}_{i,j,[\![\mathsf{id}]\!]_j}\right)\mathbf{s}'$
for $i \in \{p+1,\dots,L\},\ j \in \{1,\dots,i\alpha\},\ b \in \{0,$
$1\}$ do
    $d'_{i,j,b} := d_{i,j,b} + \mathbf{d}_{i,j,b}\mathbf{s}$
    $\mathbf{e}'_{i,j,b} := \mathbf{e}_{i,j,b} + \mathbf{E}_{i,j,b}\mathbf{s}$
$\mathsf{usk}' := ([\mathbf{t}']_2, [u']_2, [\mathbf{v}']_2)$
$\mathsf{udk}' := \left(\left[d'_{i,j,b}\right]_2, \left[\mathbf{e}'_{i,j,b}\right]_2\right)_{\substack{p+2\le i\le L, b\in\{0,1\},\\ 1\le j\le i\alpha}}$
return $(\mathsf{usk}', \mathsf{udk}')$

$\underline{\mathsf{Enc}(\mathsf{pk}, \mathsf{id} \in \mathcal{S}^p)}$:
$\mathbf{r} \overset{\$}{\leftarrow} \mathbb{Z}_q;\ \mathbf{c}_0 := \mathbf{A}\mathbf{r}$
$\mathbf{c}_1 := \sum_{i=1}^{p}\sum_{j=1}^{\ell'(l,i)} \mathbf{Z}_{i,j,[\![\mathsf{id}]\!]_j}\mathbf{r}$
$\mathsf{C} := ([\mathbf{c}_0]_1, [\mathbf{c}_1]_1)$
$\mathsf{K} := \mathbf{z}'_0 \cdot \mathbf{r}$
return $([\mathsf{K}]_T, \mathsf{C})$

$\underline{\mathsf{Dec}(\mathsf{usk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{C})}$:
Parse $\mathsf{usk} =: ([\mathbf{t}]_2, [u]_2, [\mathbf{v}]_2)$
Parse $\mathsf{C} =: ([\mathbf{c}_0]_1, [\mathbf{c}_1]_1)$
$[\mathsf{K}]_T := \left[\mathbf{c}_0^\top\right]_1 \circ \begin{bmatrix} \mathbf{v} \\ u \end{bmatrix}_2 - \left[\mathbf{c}_1^\top\right]_1 \circ [\mathbf{t}]_2$
return $[\mathsf{K}]_T$
```

Figure 15: The resulting scheme $\mathsf{HIBKEM}[\mathsf{MAC}_1[\mathcal{U}_{3k,k}], \mathcal{D}_k]$

```
$\underline{\mathsf{Gen}(1^\lambda)}$:
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3,1};\ \mathbf{A} \xleftarrow{\$} \mathcal{U}_{2,1}$
for $i \in \{1, \dots, L\},\ j \in \{1, \dots, i\alpha\},\ b \in \{0,1\}$ do
    $\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^3\ \mathbf{Y}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^{1\times 3}$
    $\mathbf{Z}_{i,j,b} := \left(\mathbf{Y}_{i,j,b}^\top \mid \mathbf{x}_{i,j,b}\right) \cdot \mathbf{A}$
    $\mathbf{d}_{i,j,b} := \mathbf{x}_{i,j,b}^\top \mathbf{B};\ \mathbf{E}_{i,j,b} := \mathbf{Y}_{i,j,b}\mathbf{B}$
$x_0' \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{y}_0' \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{z}_0' := \left(\mathbf{y}_0'^\top \mid x_0'\right) \cdot \mathbf{A}$
$\mathsf{pk} := \left(\mathcal{G}, [\mathbf{A}]_1, ([\mathbf{Z}_{i,j,b}]_1)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}, [\mathbf{z}_0']_1\right)$
$\mathsf{dk} := \left([\mathbf{B}]_2, ([\mathbf{d}_{i,j,b}]_2, [\mathbf{E}_{i,j,b}]_2)_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}\right)$
$\mathsf{sk} := \left(\mathsf{sk}_{\mathsf{MAC}}, (\mathbf{Y}_{i,j,b})_{1\le i\le L, 1\le j\le i\alpha, b\in\{0,1\}}, \mathbf{y}_0'\right)$
return $(\mathsf{pk}, \mathsf{dk}, \mathsf{sk})$

$\underline{\mathsf{Ext}(\mathsf{sk}, \mathsf{id} \in \mathcal{S}^p)}$:
for $i \in \{1, \dots, p\}$ do $\mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{t}_i := \mathbf{B}\mathbf{s}_i$
$u := \sum_{i=1}^p \left(\sum_{j=1}^{i\alpha} \mathbf{x}_{i,j,[\![\mathsf{id}]\!]_j}^\top\right)\mathbf{t}_i + x_0'$
$\mathbf{v} := \sum_{i=1}^p \left(\sum_{j=1}^{i\alpha} \mathbf{Y}_{i,j,[\![\mathsf{id}]\!]_j}\right)\mathbf{t}_i + \mathbf{y}_0'$
return $\left(([\mathbf{t}_i]_2)_{1\le i\le p}, [u]_2, [\mathbf{v}]_2\right)$
```

```
$\underline{\mathsf{Del}(\mathsf{dk}, \mathsf{usk}, \mathsf{udk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{id}_{p+1} \in \mathcal{S})}$:
Parse $\mathsf{usk} =: \left(([\mathbf{t}_i]_2)_{1\le i\le p}, [u]_2, [\mathbf{v}]_2\right)$
for $i \in \{1, \dots, p\}$ do $\mathbf{s}_i' \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{t}_i' := \mathbf{t}_i + \mathbf{B}\mathbf{s}_i'$
$u' := u + \sum_{i=1}^{p+1} \left(\sum_{j=1}^{i\alpha} \mathbf{d}_{i,j,[\![\mathsf{id}]\!]_j}\right)\mathbf{s}_i'$
$\mathbf{v}' := \mathbf{v} + \sum_{i=1}^{p+1} \left(\sum_{j=1}^{\ell'(l,i)} \mathbf{E}_{i,j,[\![\mathsf{id}]\!]_j}\right)\mathbf{s}_i'$
return $\left(([\mathbf{t}']_2)_{1\le i\le p}, [u']_2, [\mathbf{v}']_2\right)$

$\underline{\mathsf{Enc}(\mathsf{pk}, \mathsf{id} \in \mathcal{S}^p)}$:
$\mathbf{r} \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{c}_0 := \mathbf{A}\mathbf{r}$
for $i \in \{1, \dots, p\}$ do $\mathbf{c}_{1,i} := \sum_{j=1}^{\ell'(l,i)} \mathbf{Z}_{i,j,[\![\mathsf{id}]\!]_j}\mathbf{r}$
$\mathsf{C} := \left([\mathbf{c}_0]_1, ([\mathbf{c}_1]_1)_{1\le i\le p}\right)$
$\mathsf{K} := \mathbf{z}_0' \cdot \mathbf{r}$
return $([\mathsf{K}]_T, \mathsf{C})$

$\underline{\mathsf{Dec}(\mathsf{usk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{C})}$:
Parse $\mathsf{usk} =: \left(([\mathbf{t}_i]_2)_{1\le i\le p}, [u]_2, [\mathbf{v}]_2\right)$
Parse $\mathsf{C} =: \left([\mathbf{c}_0]_1, ([\mathbf{c}_1]_1)_{1\le i\le p}\right)$
$[\mathsf{K}]_T := \left[\mathbf{c}_0^\top\right]_1 \circ \begin{bmatrix} \mathbf{v} \\ u \end{bmatrix}_2 - \sum_{i=1}^{p}\left(\left[\mathbf{c}_{1,i}^\top\right]_1 \circ [\mathbf{t}]_2\right)$
return $[\mathsf{K}]_T$
```

Figure 16: The resulting scheme $\mathsf{HIBKEM}[\mathsf{MAC}_2[\mathcal{U}_{3,1}],\mathcal{U}_{2,1}]$

Acknowledgment Open Access funding provided by NTNU Norwegian University of Science and Technology (incl St. Olavs Hospital - Trondheim University Hospital).

A Semi-adaptive Security of the BKP MAC

Blazy, Kiltz, and Pan proposed in [BKP14] an affine MAC with a tight security reduction in the non-hierarchical setting. This MAC can be generalized to the hierarchical setting in the semi-adaptive security model, as shown in Figure 17. In the semi-adaptive security model, the adversary has to send the challenge identity $\mathsf{id}^\star$ before asking for user secret keys and after seeing the public key, in contrast to the adaptive security model described above. Formally, the semi-adaptive security model is defined via the games in Figure 18.

```
$\underline{\mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2):}$
$\mathbf{A} \xleftarrow{\$} \mathcal{D}_k;\ \mathbf{B} := \overline{\mathbf{A}}$
for $i \in \{1, \dots, L\},\ j \in \{1, \dots, \alpha\},\ b \in \{0,1\}$ do $\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^k$
$x'_0 \xleftarrow{\$} \mathbb{Z}_q$
return $\mathsf{sk}_{\mathsf{MAC}} := \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \alpha, b \in \{0,1\}}, x'_0\big)$

$\underline{\mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p):}$
Parse $\mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \alpha, b \in \{0,1\}}, x'_0\big)$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{B}\mathbf{s}$
$u := \Big(\sum_{i=1}^{p} \sum_{j=1}^{\alpha} \mathbf{x}_{i,j,(\mathsf{m}_i)_j}^{\top}\Big)\mathbf{t} + x'_0$
return $\tau := \big([\mathbf{t}]_2, [u]_2\big)$

$\underline{\mathsf{Ver}_{\mathsf{MAC}}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} \in \mathcal{S}^p, \tau):}$
Parse $\mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{x}_{i,j,b})_{1 \le i \le L, 1 \le j \le \alpha, b \in \{0,1\}}, x'_0\big)$
Parse $\mathsf{m} =: (\mathsf{m}_1, \dots, \mathsf{m}_p)$
Parse $\tau =: \big([\mathbf{t}]_2, [u]_2\big)$
return $u \overset{?}{=} \Big(\sum_{i=1}^{p} \sum_{j=1}^{\alpha} \mathbf{x}_{i,j,(\mathsf{m}_i)_j}^{\top}\Big)\mathbf{t} + x'_0$
```

Figure 17: Description of $\mathsf{MAC}_{\mathsf{NR}}^{h}[\mathcal{D}_k]$, a hierarchical version of $\mathsf{MAC}_{\mathsf{NR}}[\mathcal{D}_k]$ from [BKP14].

```
$\underline{\textsc{Init}:}$
$(\mathsf{pk}, \mathsf{sk}, \mathsf{dk}) \xleftarrow{\$} \mathsf{Gen}(\lambda)$
$c :=$ false
return $(\mathsf{pk}, \mathsf{dk})$

$\underline{\textsc{Ext}(\mathsf{id}):}$
if $c =$ false then return $\perp$
$\mathcal{Q}_{\mathcal{ID}} \leftarrow \mathcal{Q}_{\mathcal{ID}} \cup \{\mathsf{id}\}$
return $(\mathsf{usk}[\mathsf{id}], \mathsf{udk}[\mathsf{id}]) \xleftarrow{\$} \mathsf{Ext}(\mathsf{sk}, \mathsf{id})$

$\underline{\textsc{Enc}(\mathsf{id}^\star):}$                //one query
$(\mathsf{K}^\star, \mathsf{C}^\star) \xleftarrow{\$} \mathsf{Enc}(\mathsf{pk}, \mathsf{id}^\star)$
$\boxed{\mathsf{K}^\star \xleftarrow{\$} \mathcal{K}}$
$c :=$ true
return $(\mathsf{K}^\star, \mathsf{C}^\star)$

$\underline{\textsc{Finalize}(\beta \in \{0,1\}):}$
return $(\mathsf{Prefix}(\mathsf{id}^\star) \cap \mathcal{Q}_{\mathcal{ID}} \overset{?}{=} \emptyset \wedge \beta$
```

Figure 18: Games $\mathsf{IND\text{-}saHID\text{-}CPA}_{\mathsf{real}}$ and $\boxed{\mathsf{IND\text{-}saHID\text{-}CPA}_{\mathsf{rand}}}$ for defining $\mathsf{IND\text{-}saHID\text{-}CPA}$-security. For any identity $\mathsf{id} \in \mathcal{S}^p$, $\mathsf{Prefix}(\mathsf{id})$ denotes the set of all prefixes of id.

Definition A.1 ($\mathsf{IND\text{-}saHID\text{-}CPA}$ security). A hierarchical identity-based key encapsulation scheme HIBKEM is $\mathsf{IND\text{-}saHID\text{-}CPA}$-*secure* if for all PPT $\mathcal{A}$,

$$\mathsf{Adv}_{\mathsf{HIBKEM}}^{\mathsf{ind\text{-}sahid\text{-}cpa}}(\mathcal{A}) := |\Pr[\mathsf{IND\text{-}saHID\text{-}CPA}_{\mathsf{real}}^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{IND\text{-}saHID\text{-}CPA}_{\mathsf{rand}}^{\mathcal{A}}]|$$

is negligible.

For MACs we define $\mathsf{saHPR}_0\text{-}\mathsf{CMA}$ security similar to $\mathsf{HPR}_0\text{-}\mathsf{CMA}$ security, just with the difference that the adversary has to send the challenge message before any Tag queries. Such a MAC can be transformed into an $\mathsf{IND\text{-}saHID\text{-}CPA}$ secure HIBE in the same way as in the full security setting.

The main difference to the original, non-hierarchical MAC is that the randomness in the Tag algorithm $\mathbf{s}$ is random and not pseudo-random. Thus the tags are not deterministic, and we need to take care of duplicated tag queries for a single message. We do this in our proof by storing for each message m the first tag we computed and return a randomized version of this tag for each follow-up Tag query for m. This randomization only requires knowledge of the public key.

We call messages $(\mathsf{m}_1, \dots, \mathsf{m}_p)$ critical, if $(\mathsf{m}_1, \dots, \mathsf{m}_{p-1})$ is a prefix of the challenge message $\mathsf{m}^\star$. For each message m the critical prefix is the prefix of m, that is a critical message. The critical prefix exists and is unique for all messages, that are not a prefix of the challenge message $\mathsf{m}^\star$. Our overall proof strategy is to randomize u in all Tag oracle queries for critical messages and for Tag oracle queries of non-critical messages we simulate a Tag oracle query the critical prefix and use the key delegation mechanism to obtain a tag for the actual message.

Theorem A.2 (Security of $\mathsf{MAC}_{\mathsf{NR}}^{h}[\mathcal{D}_k]$). $\mathsf{MAC}_{\mathsf{NR}}^{h}[\mathcal{D}_k]$ *is tightly* $\mathsf{saHPR}_0\text{-}\mathsf{CMA}$ *secure in* $\mathbb{G}_2$ *under the* $\mathcal{D}_k$-MDDH *assumption for* $\mathbb{G}_2$*. More precisely, for all adversaries* $\mathcal{A}$ *there exists an adversary* $\mathcal{B}$ *with*

$$\mathsf{Adv}_{\mathsf{MAC}_{\mathsf{NR}}^{h}[\mathcal{D}_k],\mathbb{G}_2}^{\mathsf{sa\text{-}hpr_0\text{-}cma}}(\mathcal{A}) \le 2\alpha L \mathsf{Adv}_{\mathcal{D}_k,\mathsf{PGGen},2}^{\mathsf{mddh}}(\mathcal{B})$$

```
$\boxed{\mathsf{G}_1}$ $\boxed{\mathsf{G}_{2,\hat{\imath}}}$ $\boxed{\mathsf{G}_3}$

INIT:
$\mathbf{A} \xleftarrow{\$} \mathcal{D}_k;\ \mathbf{B} := \overline{\mathbf{A}}$
for $i \in \{1,\dots,L\},\ j \in \{1,\dots,\alpha\},\ b \in \{0,1\}$ do
    $\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^k$
$\boxed{x'_0 \xleftarrow{\$} \mathbb{Z}_q}$
$K : \mathcal{S}^{\leq L} \dashrightarrow (\mathbb{G}_2^k \times \mathbb{G}_2)$
return $\left([\mathbf{B}]_2, \left(\left[\mathbf{x}_{i,j,b}^\top \mathbf{B}\right]_2\right)_{1\leq i\leq L, 1\leq j\leq\alpha, b\in\{0,1\}}\right)$

CHAL($\mathsf{m}^\star \in \mathcal{S}^p$):                    // one query
$h \xleftarrow{\$} \mathbb{Z}_q$
$\mathbf{h}_0 := \left(\sum_{i=1}^{p}\sum_{j=1}^{\alpha} \mathbf{x}_{i,j,(\mathsf{m}^\star_i)_j}^\top\right) h$
$\boxed{h_1 = x'_0 \cdot h}$
$\boxed{h_1 = \mathsf{RF}_{\hat{\imath}}\left([\![\mathsf{m}^\star]\!]_{|\hat{\imath}}\right) \cdot h}$
$\boxed{h_1 \xleftarrow{\$} \mathbb{Z}_q}$
return $([h]_1, [\mathbf{h}_0]_1, [h_1]_T)$

EVAL($\mathsf{m}$):
Let $p'$ s.t. $\forall i < p' : \mathsf{m}_i = \mathsf{m}^\star_i \wedge \mathsf{m}_{p'} \neq \mathsf{m}^\star_{p'}$
$\mathsf{m}' := (\mathsf{m}_1,\dots,\mathsf{m}_{p'})$
if $K(\mathsf{m}') = \perp$ then
    $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{B}\mathbf{s}$
    $u := \left(\sum_{i=1}^{p'}\sum_{j=1}^{\alpha} \mathbf{x}_{i,j,(\mathsf{m}'_i)_j}^\top\right)\mathbf{t}\ \boxed{+\, x'_0}$
                                        $\boxed{+\, \mathsf{RF}_{\hat{\imath}}\left([\![\mathsf{m}']\!]_{|\hat{\imath}}\right)}$
    $\boxed{u \xleftarrow{\$} \mathbb{Z}_q}$
    for $i \in \{p'+1,\dots,L\},\ j \in \{1,\dots,\alpha\},\ b \in \{0,1\}$ do  $d_{i,j,b} := \mathbf{x}_{i,j,b}^\top \mathbf{B}\mathbf{s}$
    $\mathsf{tdk} := \left([d_{i,j,b}]_2\right)_{p'+1\leq i\leq L, 1\leq j\leq\alpha, b\in\{0,1\}}$
    $K(\mathsf{m}') := ([\mathbf{t}]_2, [u]_2, \mathsf{tdk})$
return $\mathsf{Delegate}(\mathsf{m}', K(\mathsf{m}'), (\mathsf{m}_{p'+1},\dots,\mathsf{m}_p))$

FINALIZE($\beta \in \{0,1\}$):
return $\beta \wedge (\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} \overset{?}{=} \emptyset$

$\mathsf{Delegate}\left(\mathsf{m}' \in \mathcal{S}^{p'}, \tau, (\mathsf{m}_{p'+1},\dots,\mathsf{m}_p) \in \mathcal{S}^{p-p'}\right)$:
Parse $\tau =: \left([\mathbf{t}]_2, [u]_2, \left([d_{i,j,b}]_2\right)_{p'+1\leq i\leq L, 1\leq j\leq\alpha, b\in\{0,1\}}\right)$
Parse $\mathsf{m}' =: (\mathsf{m}_1,\dots,\mathsf{m}_{p'})$
$\mathbf{s}' \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t}' := \mathbf{t} + \mathbf{B}\mathbf{s}'$
$u' := u + \left(\sum_{i=p'}^{p}\sum_{j=1}^{\alpha} d_{i,j,(\mathsf{m}'_i)_j}\right) + \left(\sum_{i=1}^{p}\sum_{j=1}^{\alpha} \mathbf{x}_{i,j,(\mathsf{m}_i)_j}^\top\right)\mathbf{B}\mathbf{s}'$
for $i \in \{p+1,\dots,L\},\ j \in \{1,\dots,\alpha\},\ b \in \{0,1\}$ do  $d'_{i,j,b} := d_{i,j,b} + \mathbf{x}_{i,j,b}^\top \mathbf{B}\mathbf{s}'$
return $\left([\mathbf{t}']_2, [u']_2, \left(\left[d'_{i,j,b}\right]_2\right)_{p+1\leq i\leq L, 1\leq j\leq\alpha, b\in\{0,1\}}\right)$
```

Figure 19: Hybrids for the security proof of $\mathsf{MAC}_{\mathsf{NR}}^{h}[\mathcal{D}_k]$. The algorithm $\mathsf{Delegate}$ is only helper function and not an oracle for the adversary.

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. The proof uses a hybrid argument with hybrids G_0, G_1, $\mathsf{G}_{2,\hat{\imath}}$ for $\hat{\imath} \in \{0,\dots,\alpha L\}$ and G_3. G_0 is the $\mathsf{saPR\text{-}CMA}_{\mathsf{real}}$ game, the other games are defined in Figure 19. They make use of a random functions $\mathsf{RF}_{\hat{\imath}} : \{0,1\}^{\hat{\imath}} \to \mathbb{Z}_q^{1\times k}$, for $i \in \{0,\dots,\alpha L\}$, defined on-the-fly.

Lemma A.3 ($\mathsf{G}_0 \rightsquigarrow \mathsf{G}_1$).

$$\Pr\left[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\right]$$

Proof. In game G_1 the queried message tags are not computed directly; instead we compute a tag for the prefix m' of m, that is one component longer than the longest common prefix of m and $\mathsf{m}^\star$. This prefix exists when m is not a prefix of $\mathsf{m}^\star$. The tag for m' is then delegated to a tag for m and re-randomized. This requires only the public key. The distribution of a delegated and re-randomized tag is the same as the distribution of a fresh tag; thus, the games are identical. □

Lemma A.4 ($\mathsf{G}_1 \rightsquigarrow \mathsf{G}_{2,0}$).

$$\Pr\left[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathsf{G}_{2,0}^{\mathcal{A}} \Rightarrow 1\right]$$

Proof. In game $\mathsf{G}_{2,0}$ we replace x'_0 with $\mathsf{RF}_0(\varepsilon)$, which is equivalent. □

Lemma A.5 ($\mathsf{G}_{2,\hat{\imath}} \rightsquigarrow \mathsf{G}_{2,\hat{\imath}+1}$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_{2,\hat{\imath}}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_{2,\hat{\imath}+1}^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \mathsf{Adv}_{\mathcal{D}_k,\mathsf{PGGen},2}^{\mathsf{mddh}}(\mathcal{B})$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. We define

$$\mathsf{RF}_{\hat\imath+1}\left(\llbracket \mathsf{m} \rrbracket_{|\hat\imath+1}\right) := \begin{cases} \mathsf{RF}_{\hat\imath}\left(\llbracket \mathsf{m} \rrbracket_{|\hat\imath}\right) & \text{if } \llbracket \mathsf{m} \rrbracket_{\hat\imath+1} = \llbracket \mathsf{m}^\star \rrbracket_{\hat\imath+1} \\ \mathsf{RF}_{\hat\imath}\left(\llbracket \mathsf{m} \rrbracket_{|\hat\imath}\right) + \mathsf{RF}'_{\hat\imath}\left(\llbracket \mathsf{m} \rrbracket_{|\hat\imath}\right) & \text{if } \llbracket \mathsf{m} \rrbracket_{\hat\imath+1} = 1 - \llbracket \mathsf{m}^\star \rrbracket_{\hat\imath+1} \end{cases},$$

where $\mathsf{RF}'_{\hat\imath} : \{0,1\}^{\hat\imath} \to \mathbb{Z}_q$ is another independent random function. Note that whenever $\mathsf{RF}_{\hat\imath+1}$ is evaluated, $\mathsf{RF}_{\hat\imath}$ is not evaluated. This is because the adversary is not allowed to query the tags for any prefix of m' since each such message would be a prefix of the challenge message $\mathsf{m}^\star$.

The adversary $\mathcal{B}$ uses the random self reducibility to get a Q-fold $\mathcal{D}_k$-MDDH challenge $([\mathbf{A}]_2, [\mathbf{H}]_2)$, where Q denotes the number of tag queries. The reduction uses a function $\phi : \{0,1\}^{\hat\imath} \to \{1, \ldots, Q\}$ that has to be injective on all prefixes of queried messages. It can be computed on-the-fly. The reduction is given in Figure 20.

The reduction computes the public key honestly except for $\mathbf{x}_{i^\star, j^\star, 1-\left(\mathsf{m}^\star_{i^\star}\right)_{j^\star}}$. The public key is distributed correctly since $\mathbf{r}^\top \mathbf{A}$ is a uniform random k dimensional row vector, just like $\mathbf{x}^\top_{i^\star, j^\star, 1-\left(\mathsf{m}^\star_{i^\star}\right)_{j^\star}} \mathbf{B}$ would be for a uniform random $\mathbf{x}^\top_{i^\star, j^\star, 1-\left(\mathsf{m}^\star_{i^\star}\right)_{j^\star}}$.

Chal, Finalize and Eval queries with $K(\mathsf{m}') \neq \perp$ are the same as in $\mathsf{G}_{2,\hat\imath}$ and $\mathsf{G}_{2,\hat\imath+1}$. The Eval queries with $p' < i^\star \vee \llbracket \mathsf{m}' \rrbracket_{\hat\imath} = \llbracket \mathsf{m}^\star \rrbracket_{\hat\imath}$ remain also unchanged. Note that for these queries, the tag delegation keys $[d_{i,j,b}]_2$ can be computed from $\left[\mathbf{x}^\top_{i,j,b} \mathbf{B}\right]_2$ and $\mathbf{s}$.

For the Eval queries with $p' \geq i^\star \wedge \llbracket \mathsf{m}' \rrbracket_{\hat\imath} \neq \llbracket \mathsf{m}^\star \rrbracket_{\hat\imath}$ write $\mathbf{H}_c =: \mathbf{A}\mathbf{W}_c + \mathbf{R}_c$ for some $\mathbf{W}_c \in \mathbb{Z}_q$ and $\mathbf{R}_c = \mathbf{0}$ if $\mathbf{H}_c$ was drawn from $\mathcal{D}_k$ or uniform random $\mathbf{R} \in \mathbb{Z}_q^{k+1}$ if $\mathbf{H}_c$ was chosen uniformly random. Then

$$\begin{aligned} u &= \left(\sum_{\substack{(i,j)=(1,1) \\ (i,j)\neq(i^\star,j^\star)}}^{(p',\alpha)} \mathbf{x}^\top_{i,j,\left(\mathsf{m}'_i\right)_j} \right) \mathbf{t} + \mathbf{r}^\top \mathbf{A}(\mathbf{s} + \mathbf{W}_c) + \mathbf{r}^\top \mathbf{R}_c + \mathsf{RF}_{\hat\imath}\left(\llbracket \mathsf{m}' \rrbracket_{|\hat\imath}\right) \\ &= \left(\sum_{\substack{(i,j)=(1,1) \\ (i,j)\neq(i^\star,j^\star)}}^{(p',\alpha)} \mathbf{x}^\top_{i,j,\left(\mathsf{m}'_i\right)_j} \right) \mathbf{t} + \mathbf{x}^\top_{i^\star, j^\star, 1-\left(\mathsf{m}^\star_{i^\star}\right)_{j^\star}} \underbrace{\mathbf{B}(\mathbf{s} + \mathbf{W}_c)}_{\mathbf{t}} + \mathbf{r}^\top \mathbf{R}_c + \mathsf{RF}_{\hat\imath}\left(\llbracket \mathsf{m}' \rrbracket_{|\hat\imath}\right) \\ &= \left(\sum_{(i,j)=(1,1)}^{(p',\alpha)} \mathbf{x}^\top_{i,j,\left(\mathsf{m}'_i\right)_j} \right) \mathbf{t} + \mathbf{r}^\top \mathbf{R}_c + \mathsf{RF}_{\hat\imath}\left(\llbracket \mathsf{m}' \rrbracket_{|\hat\imath}\right). \end{aligned}$$

If $\mathbf{R}_c = 0$ the reduction is simulating $\mathsf{G}_{2,\hat\imath}$, if $\mathbf{R}_c$ is uniform random, define $\mathsf{RF}'_{\hat\imath}\left(\llbracket \mathsf{m}' \rrbracket_{|\hat\imath}\right) := \mathbf{r}^\top \mathbf{R}_c$ and the reduction is simulating $\mathsf{G}_{2,\hat\imath}$. Note that for these queries, the tag delegation keys $[d_{i,j,b}]_2$ can be computed from $\mathbf{x}_{i,j,b}$ and $[\mathsf{t}]_2$ since $\mathbf{x}_{i,j,b}$ is known to the adversary for all $i > p' \geq i^\star$.

□

Lemma A.6 ($\mathsf{G}_{2,L\alpha} \rightsquigarrow \mathsf{G}_3$).

$$\Pr\left[\mathsf{G}^{\mathcal{A}}_{2,L\alpha} \Rightarrow 1\right] = \Pr\left[\mathsf{G}^{\mathcal{A}}_3 \Rightarrow 1\right]$$

Proof. In game $\mathsf{G}_{2,L\alpha}$ all the us in tags for a message prefix m' are masked by $\mathsf{RF}_{L\alpha}$ evaluated at the unique value $\llbracket \mathsf{m}' \rrbracket$. So they are uniformly random as in G_3. □

Summary. To prove Theorem A.2, we combine Lemmas A.3–A.6 to change h_1 from real to random and then apply all Lemmas in reverse order to get to the $\mathsf{saHPR}_0\text{-}\mathsf{CMA}_{\mathsf{rand}}$ game. □

B Security of the HIBKEM transformation

Proof (of Theorem 4.1). The proof makes use of the hybrids G_0–G_4 defined in Figure 21. G_0 is the $\mathsf{IND\text{-}HID\text{-}CPA}_{\mathsf{real}}$ game.

```
INIT:
$\mathbf{B} := \overline{\mathbf{A}}$
for $i \in \{1,\dots,L\},\ j \in \{1,\dots,\alpha\},\ b \in \{0,1\}$ do
    if $i\alpha + j \neq \hat{\imath} \vee b = (\mathsf{m}^\star_i)_j$ then $\mathbf{x}_{i,j,b} \xleftarrow{\$} \mathbb{Z}_q^k$
$i^\star := (\hat{\imath} \operatorname{div} \alpha) + 1;\ j^\star := (\hat{\imath} \mod \alpha) + 1$
$\mathbf{r} \xleftarrow{\$} \mathbb{Z}_q^{k+1};\ b := 1 - (\mathsf{m}^\star_{i^\star})_{j^\star};\ \mathbf{x}^\top_{i^\star,j^\star,b}\mathbf{B} := \mathbf{r}^\top\mathbf{A}$
$K : \mathcal{S}^{\le L} \dashrightarrow (\mathbb{G}_2^k \times \mathbb{G}_2)$
return $\left([\mathbf{B}]_2, \left(\left[\mathbf{x}^\top_{i,j,b}\mathbf{B}\right]_2\right)_{1\le i\le L, 1\le j\le\alpha, b\in\{0,1\}}\right)$

CHAL($\mathsf{m}^\star \in \mathcal{S}^p$):                          // one query
$h \xleftarrow{\$} \mathbb{Z}_q$
$\mathbf{h}_0 := \left(\sum_{i=1}^{p}\sum_{j=1}^{\alpha} \mathbf{x}^\top_{i,j,(\mathsf{m}^\star_i)_j}\right)h$
$h_1 = \mathsf{RF}_{\hat{\imath}}\left([\![\mathsf{m}^\star]\!]_{|\hat{\imath}}\right) \cdot h$
return $([h]_1, [\mathbf{h}_0]_1, [h_1]_T)$

FINALIZE($\beta \in \{0,1\}$):
return $\beta \wedge (\mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} \overset{?}{=} \emptyset$

EVAL(m):
Let $p'$ s.t. $\forall i < p' : \mathsf{m}_i = \mathsf{m}^\star_i \wedge \mathsf{m}_{p'} \neq \mathsf{m}^\star_{p'}$
$\mathsf{m}' := (\mathsf{m}_1, \dots, \mathsf{m}_{p'})$
if $K(\mathsf{m}') = \perp$ then
    if $p' < i^\star \vee [\![\mathsf{m}']\!]_{\hat{\imath}} = [\![\mathsf{m}^\star]\!]_{\hat{\imath}}$ then
        $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{B}\mathbf{s}$
        $i' := \min\{\hat{\imath}, p'\alpha\}$
        $u := \left(\sum_{i=1}^{p'}\sum_{j=1}^{\alpha} \mathbf{x}^\top_{i,j,(\mathsf{m}'_i)_j}\right)\mathbf{t} + \mathsf{RF}_{i'}\left([\![\mathsf{m}']\!]_{|i'}\right)$
        for $i \in \{p'+1,\dots,L\},\ j \in \{1,\dots,\alpha\},\ b \in \{0,1\}$ do $d_{i,j,b} := \mathbf{x}^\top_{i,j,b}\mathbf{B}\mathbf{s}$
    else
        $c := \phi\left([\![\mathsf{m}']\!]_{|\hat{\imath}}\right)$
        $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{B}\mathbf{s} + \overline{\mathbf{H}}_c$
        $u := \left(\sum_{(i,j)=(1,1), (i,j)\neq(i^\star,j^\star)}^{(p',\alpha)} \mathbf{x}^\top_{i,j,(\mathsf{m}'_i)_j}\right)\mathbf{t} + \mathbf{r}^\top(\mathbf{A}\mathbf{s} + \mathbf{H}_c) + \mathsf{RF}_{\hat{\imath}}\left([\![\mathsf{m}']\!]_{|\hat{\imath}}\right)$
        for $i \in \{p'+1,\dots,L\},\ j \in \{1,\dots,\alpha\},\ b \in \{0,1\}$ do $d_{i,j,b} := \mathbf{x}^\top_{i,j,b}\mathbf{t}$
    $\mathsf{tdk} := \left([d_{i,j,b}]_2\right)_{p'+1\le i\le L, 1\le j\le\alpha, b\in\{0,1\}}$
    $K(\mathsf{m}') := ([\mathbf{t}]_2, [u]_2, \mathsf{tdk})$
return $\mathsf{Delegate}(\mathsf{m}', K(\mathsf{m}'), (\mathsf{m}_{p'+1}, \dots, \mathsf{m}_p))$
```

Figure 20: Adversary $\mathcal{B}$ for the proof of Lemma A.5. The helper algorithm Delegate is given in Figure 19.

Lemma B.1 ($\mathsf{G}_0 \rightsquigarrow \mathsf{G}_1$).

$$\Pr\left[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\right]$$

Proof. The only difference between these games is that $\mathbf{c}^*_{1,l}$ and K^* are computed with the public value $\mathbf{Z}_{l,i,j}$ in game G_0 and with the secret key $\mathbf{x}_{l,i,j}$ and $\mathbf{Y}_{l,i,j}$ in G_1. □

Lemma B.2 ($\mathsf{G}_1 \rightsquigarrow \mathsf{G}_2$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}_2$ with*

$$\left|\Pr\left[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1\right]\right| \le \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_{2,1},\mathsf{PGGen},1}(\mathcal{B}_2)$$

and $T(\mathcal{B}_2) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

Proof. The only difference between these is that $\mathbf{c}^*_0$ is chosen from $\mathsf{Span}(\mathbf{A})$ in G_1 and from $\mathbb{Z}_q^2$ in G_2.

The running time of $\mathcal{B}_2$ is dominated by the running time of $\mathcal{A}$ plus some (polynomial) overhead that is independent of $T(\mathcal{A})$ for the group operations in each oracle query. □

Lemma B.3 ($\mathsf{G}_2 \rightsquigarrow \mathsf{G}_3$).

$$\Pr\left[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1\right]$$

Proof. These two games are equivalent. First notice that the values $\mathbf{Z}_{l,i,j}$ and $\mathbf{z}'_0$ are uniform random when $\mathbf{Y}_{l,i,j}$ and $\mathbf{y}'_0$ are hidden, so $\mathbf{Z}_{l,i,j}$ and $\mathbf{z}'_0$ are distributed identical in both games. Second notice

$$\mathbf{Z}_{l,i,j} := \left(\mathbf{Y}^\top_{l,i,j} \mid \mathbf{x}_{l,i,j}\right) \cdot \mathbf{A} \Longleftrightarrow \mathbf{Y}^\top_{l,i,j} = (\mathbf{Z}_{l,i,j} - \mathbf{x}_{l,i,j}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1}$$

and similarly

$$\mathbf{z}'_0 := \left(\mathbf{y}'^\top_0 \mid x'_0\right) \cdot \mathbf{A} \Longleftrightarrow \mathbf{y}'^\top_0 = (\mathbf{z}'_0 - x'_0\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1}.$$

Game G_3 is obtained from G_2 by choosing $\mathbf{Z}_{l,i,j}$ and $\mathbf{z}'_0$ uniform random and replacing all occurrences of the values $\mathbf{Y}_{l,i,j}$ and $\mathbf{y}'_0$ with the above equation. Thus the games are equally distributed. □

Lemma B.4 ($\mathsf{G}_3 \rightsquigarrow \mathsf{G}_4$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}_1$ with*

$$\left|\Pr\left[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_4^{\mathcal{A}} \Rightarrow 1\right]\right| \le \mathsf{Adv}^{\mathsf{hpr}_0\text{-}\mathsf{cma}}_{\mathsf{MAC},\mathbb{G}_2}(\mathcal{B}_1)$$

and $T(\mathcal{B}_1) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$.

```
$\mathsf{G}_0$ $\boxed{\mathsf{G}_1\ \mathsf{G}_2}$ $\mathsf{G}_3$ $\mathsf{G}_4$

INIT:
$\mathsf{sk}_{\mathsf{MAC}} \overset{\$}{\leftarrow} \mathsf{Gen}_{\mathsf{MAC}}(\mathbb{G}_2, q, P_2)$
$\mathsf{sk}_{\mathsf{MAC}} =: \left(\mathbf{B}, (\mathbf{x}_{l,i,j})_{1 \le l \le \ell(L), 1 \le i \le L, 1 \le j \le \ell'(l,i)}, x'_0\right)$
$\mathbf{A} \overset{\$}{\leftarrow} \mathcal{D}_k$
for $l \in \{1, \ldots, \ell(L)\},\ i \in \{1, \ldots, L\},\ j \in \{1, \ldots \ldots, \ell'(l,i)\}$ do
    $\mathbf{Y}_{l,i,j} \overset{\$}{\leftarrow} \mathbb{Z}_q^{k \times n};\ \mathbf{Z}_{l,i,j} := \left(\mathbf{Y}_{l,i,j}^\top \mid \mathbf{x}_{l,i,j}\right) \cdot \mathbf{A}$
    $\mathbf{d}_{l,i,j} := \mathbf{x}_{l,i,j}^\top \cdot \mathbf{B};\ \mathbf{E}_{l,i,j} := \mathbf{Y}_{l,i,j} \cdot \mathbf{B}$
    $\mathbf{E}_{l,i,j} := \overline{\mathbf{A}}^{-\top}\left(\mathbf{Z}_{l,i,j}^\top \mathbf{B} - \underline{\mathbf{A}}^\top \mathbf{d}_{l,i,j}\right)$
$\mathbf{y}'_0 \overset{\$}{\leftarrow} \mathbb{Z}_q^k$
$\mathbf{z}'_0 := \left(\mathbf{y}'^\top_0 \mid x'_0\right) \cdot \mathbf{A}$
$\mathsf{pk} := \left(\mathcal{G}, [\mathbf{A}]_1, \left([\mathbf{Z}_{l,i,j}]_1\right)_{1 \le l \le \ell(L), 1 \le i \le L, 1 \le j \le \ell'(l,i)}, [\mathbf{z}'_0]_1\right)$
$\mathsf{dk} := \left([\mathbf{B}]_2, \left([\mathbf{d}_{l,i,j}]_2, [\mathbf{E}_{l,i,j}]_2\right)_{1 \le l \le \ell(L), 1 \le i \le L, 1 \le j \le \ell'(l,i)}\right)$
return $(\mathsf{pk}, \mathsf{dk})$

EXT($\mathsf{id} \in \mathcal{S}^p$):
$\mathcal{Q}_{\mathcal{ID}} \leftarrow \mathcal{Q}_{\mathcal{ID}} \cup \{\mathsf{id}\}$
$\left(([\mathbf{t}_l]_2)_{1 \le l \le \ell(p)}, [u]_2\right) \overset{\$}{\leftarrow} \mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{id})$
$\mathbf{v} := \sum_{l=1}^{\ell(p)} \left(\sum_{i=1}^{p} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}(\mathsf{id}_{|i}) \mathbf{Y}_{l,i,j}\right) \mathbf{t}_l + \mathbf{y}'_0$
$\mathbf{v}^\top := \left(\sum_{l=1}^{\ell(p)} \left(\mathbf{t}_l^\top \sum_{i=1}^{p} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}(\mathsf{id}_{|i}) \mathbf{Z}_{l,i,j}\right) + \mathbf{z}'_0 - u \cdot \underline{\mathbf{A}}\right) \overline{\mathbf{A}}^{-1}$
for $l \in \{1, \ldots, \ell(p)\},\ i \in \{p+1, \ldots, L\},\ j \in \{1, \ldots \ldots, \ell'(l,i)\}$ do
    $d_{l,i,j} := \mathbf{x}_{l,i,j}^\top \mathbf{t}_l;\ \mathbf{e}_{l,i,j} := \mathbf{Y}_{l,i,j} \mathbf{t}_l$
    $\mathbf{e}_{l,i,j}^\top := \left(\mathbf{t}_l^\top \mathbf{Z}_{l,i,j} - d_{l,i,j} \underline{\mathbf{A}}\right) \overline{\mathbf{A}}^{-1}$
$\mathsf{udk}[\mathsf{id}] := \left([d_{l,i,j}]_2, [\mathbf{e}_{l,i,j}]_2\right)_{1 \le l \le \ell(L), p+1 \le i \le L, 1 \le j \le \ell'(l,i)}$
return $\left(([\mathbf{t}_l]_2)_{1 \le l \le \ell(p)}, [u]_2, [\mathbf{v}]_2, \mathsf{udk}[\mathsf{id}]\right)$

ENC($\mathsf{id}^\star \in \mathcal{S}^p$):    // one query
$\mathbf{r} \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \mathbf{c}_0^\star := \mathbf{A}\mathbf{r}$
$\mathbf{c}_0^\star \overset{\$}{\leftarrow} \mathbb{Z}_q^{k+1}$
$h \overset{\$}{\leftarrow} \mathbb{Z}_q$
$\overline{\mathbf{c}_0^\star} \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \underline{\mathbf{c}_0^\star} := h + \underline{\mathbf{A}} \cdot \overline{\mathbf{A}}^{-1} \overline{\mathbf{c}_0^\star}$
for $l \in \{1, \ldots, \ell(p)\}$ do
    $\mathbf{c}_{1,l}^\star := \sum_{i=1}^{p} \sum_{j=0}^{\ell'(l,i)} f_{l,i,j}\left(\mathsf{id}^\star_{|i}\right) \mathbf{Z}_{l,i,j} \mathbf{r}$
    $\boxed{\mathbf{c}_{1,l}^\star := \sum_{i=1}^{p} \sum_{j=0}^{\ell'(l,i)} \left(f_{l,i,j}\left(\mathsf{id}^\star_{|i}\right)\left(\mathbf{Y}_{l,i,j}^\top \mid \mathbf{x}_{l,i,j}\right)\right) \mathbf{c}_0^\star}$
    $\mathbf{c}_{1,l}^\star := \sum_{i=1}^{p} \sum_{j=0}^{\ell'(l,i)} \left(f_{l,i,j}\left(\mathsf{id}^\star_{|i}\right) \left(\mathbf{Z}_{l,i,j} \overline{\mathbf{A}}^{-1} \overline{\mathbf{c}_0^\star} + \mathbf{x}_{l,i,j} h\right)\right)$
$\mathsf{K}^\star := \mathbf{z}'_0 \cdot \mathbf{r}$
$\boxed{\mathsf{K}^\star := \left(\mathbf{y}'^\top_0 \mid x'_0\right) \mathbf{c}_0^\star}$
$\mathsf{K}^\star := \mathbf{z}'_0 \overline{\mathbf{A}}^{-1} \overline{\mathbf{c}_0^\star} + x'_0 h$
$\mathsf{K}^\star \overset{\$}{\leftarrow} \mathbb{Z}_q$
return $\left([\mathsf{K}^\star]_T, \left([\mathbf{c}_0^\star]_1, \left(\left[\mathbf{c}_{1,l}^\star\right]_1\right)_{1 \le l \le \ell(p)}\right)\right)$

FINALIZE($\beta \in \{0,1\}$):
return $\beta \wedge \mathsf{Prefix}(\mathsf{id}^\star) \cap \mathcal{Q}_{\mathcal{ID}} \overset{?}{=} \emptyset$
```

Figure 21: Hybrids for the security proof of the HIBKEM transformation.

```
$\underline{\textsc{Init}:}$
$\mathsf{pk_{MAC}} \overset{\$}{\leftarrow} \textsc{Init}_{\mathsf{MAC}}$
Parse $\mathsf{pk_{MAC}} =: \left([\mathbf{B}]_2, \left(\left[\mathbf{x}_{l,i,j}^\top \mathbf{B}\right]_2\right)_{1 \le l \le \ell(L), 1 \le i \le L, 1 \le j \le \ell'(l,i)}\right)$
$\mathbf{A} \overset{\$}{\leftarrow} \mathcal{D}_k$
for $l \in \{1, \ldots, \ell(L)\}$, $i \in \{1, \ldots, L\}$, $j \in \{1, \ldots, \ell'(l,i)\}$ do
    $\mathbf{d}_{l,i,j} := \mathbf{x}_{l,i,j}^\top \mathbf{B}$; $\mathbf{Z}_{l,i,j} \overset{\$}{\leftarrow} \mathbb{Z}_q^{n \times k}$
    $\mathbf{E}_{l,i,j} := \overline{\mathbf{A}}^{-\top}\left(\mathbf{Z}_{l,i,j}^\top \mathbf{B} - \underline{\mathbf{A}}^\top \mathbf{d}_{l,i,j}\right)$
$\mathbf{z}_0' \overset{\$}{\leftarrow} \mathbb{Z}_q^{1 \times k}$
$\mathsf{pk} := \left(\mathcal{G}, [\mathbf{A}]_1, \left([\mathbf{Z}_{l,i,j}]_1\right)_{1 \le l \le \ell(L), 1 \le i \le L,, 1 \le j \le \ell'(l,i)}, [\mathbf{z}_0']_1\right)$
$\mathsf{dk} := \left([\mathbf{B}]_2, \left([\mathbf{d}_{l,i,j}]_2, [\mathbf{E}_{l,i,j}]_2\right)_{1 \le l \le \ell(L), 1 \le i \le L, 1 \le j \le \ell'(l,i)}\right)$
return $(\mathsf{pk}, \mathsf{dk})$

$\underline{\textsc{Enc}(\mathsf{id}^\star \in \mathcal{S}^p):}$                                        // one query
$\mathbf{H} \overset{\$}{\leftarrow} \textsc{Chal}(\mathsf{id}^\star)$
Parse $\mathbf{H} =: \left([h]_1, \left([\mathbf{h}_{0,l}]_1\right)_{1 \le l \le \ell(p)}, [h_1]_T\right)$
$\overline{\mathbf{c}_0^\star} \overset{\$}{\leftarrow} \mathbb{Z}_q^k$; $\underline{\mathbf{c}_0^\star} := h + \underline{\mathbf{A}} \cdot \overline{\mathbf{A}}^{-1} \overline{\mathbf{c}_0^\star}$
for $l \in \{1, \ldots, \ell(p)\}$ do
    $\mathbf{c}_{1,l}^\star := \left(\sum_{i=1}^p \sum_{j=0}^{\ell'(l,i)} f_{l,i,j}\left(\mathsf{id}_{|i}^\star\right) \mathbf{Z}_{l,i,j} \overline{\mathbf{A}}^{-1} \overline{\mathbf{c}_0^\star}\right) + \mathbf{h}_{0,l}$
$\mathsf{K}^\star := \mathbf{z}_0' \overline{\mathbf{A}}^{-1} \overline{\mathbf{c}_0^\star} + h_T$
return $\left([\mathsf{K}^\star]_T, \left([\mathbf{c}_0^\star]_1, \left(\left[\mathbf{c}_{1,l}^\star\right]_1\right)_{1 \le l \le \ell(p)}\right)\right)$

$\underline{\textsc{Ext}(\mathsf{id} \in \mathcal{S}^p):}$
$\mathcal{Q}_{\mathcal{ID}} \leftarrow \mathcal{Q}_{\mathcal{ID}} \cup \{\mathsf{id}\}$
$\tau \overset{\$}{\leftarrow} \textsc{Eval}(\mathsf{id})$
Parse $\tau =: \left(([\mathbf{t}_l]_2)_{1 \le l \le \ell(p)}, [u]_2, \mathsf{tdk}\right)$
$\mathsf{tdk} =: \left([d_{l,i,j}]_2\right)_{1 \le l \le \ell(p), p+1 \le i \le L, 1 \le j \le \ell'(l,i)}$
$\mathbf{v}^\top := \left(\sum_{l=1}^{\ell(p)} \left(\mathbf{t}_l^\top \sum_{i=1}^{p} \sum_{j=1}^{\ell'(l,i)} f_{l,i,j}\left(\mathsf{id}_{|i}\right) \mathbf{Z}_{l,i,j}\right) + \mathbf{z}_0' - u \cdot \underline{\mathbf{A}}\right) \overline{\mathbf{A}}^{-1}$
for $l \in \{1, \ldots, \ell(p)\}$, $i \in \{p+1, \ldots\ldots, L\}$, $j \in \{1, \ldots, \ell'(l,i)\}$ do
    $\mathbf{e}_{l,i,j}^\top := \left(\mathbf{t}_l^\top \mathbf{Z}_{l,i,j} - d_{l,i,j} \underline{\mathbf{A}}\right) \overline{\mathbf{A}}^{-1}$
$\mathsf{udk}[\mathsf{id}] := \left([d_{l,i,j}]_2, [\mathbf{e}_{l,i,j}]_2\right)_{1 \le l \le \ell(L), p+1 \le i \le L, 1 \le j \le \ell'(l,i)}$
return $\left(([\mathbf{t}_l]_2)_{1 \le l \le \ell(p)}, [u]_2, [\mathbf{v}]_2, \mathsf{udk}[\mathsf{id}]\right)$

$\underline{\textsc{Finalize}(\beta \in \{0,1\}):}$
return $\textsc{Finalize}_{\mathsf{MAC}}(\beta)$
```

Figure 22: Adversary $\mathcal{B}$ for Lemma B.4.

Proof. The adversary $\mathcal{B}$ is given in Figure 22. When $\mathcal{B}$ plays the $\mathsf{HPR_0\text{-}CMA_{real}}$ game with the affine MAC with levels challenger, he simulates the game G_3 for $\mathcal{A}$. On the other hand, when $\mathcal{B}$ plays the $\mathsf{HPR\text{-}CMA_{rand}}$ game with the MAC challenger, he simulates the game G_4 for $\mathcal{A}$.

The running time of $\mathcal{B}_1$ is dominated by the running time of $\mathcal{A}$ plus some (polynomial) overhead that is independent of $T(\mathcal{A})$ for the group operations in each oracle query. □

Summary. To prove Theorem 4.1, we combine Lemmas B.1–B.4 to change the key K from real to random and then apply all Lemmata (except Lemma B.4) in reverse order to get to the $\mathsf{IND\text{-}HID\text{-}CPA_{rand}}$ game. □

References

[AHN+17] Masayuki Abe, Dennis Hofheinz, Ryo Nishimaki, Miyako Ohkubo, and Jiaxin Pan. Compact structure-preserving signatures with almost tight security. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part II*, volume 10402 of *LNCS*, pages 548–580. Springer, Heidelberg, August 2017. (Cited on page 1.)

[BF01] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil pairing. In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 213–229. Springer, Heidelberg, August 2001. (Cited on page 2.)

[BHJ+15] Christoph Bader, Dennis Hofheinz, Tibor Jager, Eike Kiltz, and Yong Li. Tightly-secure authenticated key exchange. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part I*, volume 9014 of *LNCS*, pages 629–658. Springer, Heidelberg, March 2015. (Cited on page 6.)

[BKP14] Olivier Blazy, Eike Kiltz, and Jiaxin Pan. (Hierarchical) identity-based encryption from affine message authentication. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 408–425. Springer, Heidelberg, August 2014. (Cited on page 1, 2, 3, 6, 9, 10, 11, 23, 26, 27.)

[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, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 595–624. Springer, Heidelberg, April 2015. (Cited on page 3.)

[CHK04] Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 207–222. Springer, Heidelberg, May 2004. (Cited on page 2, 3.)

[CLL$^+$13] Jie Chen, Hoon Wei Lim, San Ling, Huaxiong Wang, and Hoeteck Wee. Shorter IBE and signatures via asymmetric pairings. In Michel Abdalla and Tanja Lange, editors, *PAIRING 2012*, volume 7708 of *LNCS*, pages 122–140. Springer, Heidelberg, May 2013. (Cited on page 2.)

[Coc01] Clifford Cocks. An identity based encryption scheme based on quadratic residues. In Bahram Honary, editor, *8th IMA International Conference on Cryptography and Coding*, volume 2260 of *LNCS*, pages 360–363. Springer, Heidelberg, December 2001. (Cited on page 2.)

[CW13] Jie Chen and Hoeteck Wee. Fully, (almost) tightly secure IBE and dual system groups. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 435–460. Springer, Heidelberg, August 2013. (Cited on page 1, 2, 3, 4.)

[EHK$^+$13] Alex Escala, Gottfried Herold, Eike Kiltz, Carla Ràfols, and Jorge Villar. An algebraic framework for Diffie-Hellman assumptions. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 129–147. Springer, Heidelberg, August 2013. (Cited on page 3, 7, 8.)

[GCTC16] Junqing Gong, Zhenfu Cao, Shaohua Tang, and Jie Chen. Extended dual system group and shorter unbounded hierarchical identity based encryption. *Designs, Codes and Cryptography*, 80(3):525–559, Sep 2016. (Cited on page 2.)

[GDCC16] Junqing Gong, Xiaolei Dong, Jie Chen, and Zhenfu Cao. Efficient IBE with tight reduction to standard assumption in the multi-challenge setting. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 624–654. Springer, Heidelberg, December 2016. (Cited on page 1, 2, 6.)

[Gen06] Craig Gentry. Practical identity-based encryption without random oracles. In Serge Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 445–464. Springer, Heidelberg, May / June 2006. (Cited on page 2.)

[GHKP18] Romain Gay, Dennis Hofheinz, Lisa Kohl, and Jiaxin Pan. More efficient (almost) tightly secure structure-preserving signatures. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part II*, volume 10821 of *LNCS*, pages 230–258. Springer, Heidelberg, April / May 2018. (Cited on page 1, 3.)

[GHKW16] Romain Gay, Dennis Hofheinz, Eike Kiltz, and Hoeteck Wee. Tightly CCA-secure encryption without pairings. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part I*, volume 9665 of *LNCS*, pages 1–27. Springer, Heidelberg, May 2016. (Cited on page 1, 3, 4, 5, 6, 7, 8.)

[GS02] Craig Gentry and Alice Silverberg. Hierarchical ID-based cryptography. In Yuliang Zheng, editor, *ASIACRYPT 2002*, volume 2501 of *LNCS*, pages 548–566. Springer, Heidelberg, December 2002. (Cited on page 2.)

[HHK18] Julia Hesse, Dennis Hofheinz, and Lisa Kohl. On tightly secure non-interactive key exchange. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part II*, volume 10992 of *LNCS*, pages 65–94. Springer, Heidelberg, August 2018. (Cited on page 1.)

[HJ12] Dennis Hofheinz and Tibor Jager. Tightly secure signatures and public-key encryption. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 590–607. Springer, Heidelberg, August 2012. (Cited on page 1.)

[HJP18] Dennis Hofheinz, Dingding Jia, and Jiaxin Pan. Identity-based encryption tightly secure under chosen-ciphertext attacks. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part II*, volume 11273 of *LNCS*, pages 190–220. Springer, Heidelberg, December 2018. (Cited on page 2, 3, 6.)

[HK07] Dennis Hofheinz and Eike Kiltz. Secure hybrid encryption from weakened key encapsulation. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 553–571. Springer, Heidelberg, August 2007. (Cited on page 8, 23.)

[HKS15] Dennis Hofheinz, Jessica Koch, and Christoph Striecks. Identity-based encryption with (almost) tight security in the multi-instance, multi-ciphertext setting. In Jonathan Katz, editor, *PKC 2015*, volume 9020 of *LNCS*, pages 799–822. Springer, Heidelberg, March / April 2015. (Cited on page 2.)

[HL02] Jeremy Horwitz and Ben Lynn. Toward hierarchical identity-based encryption. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 466–481. Springer, Heidelberg, April / May 2002. (Cited on page 2.)

[JR13] Charanjit S. Jutla and Arnab Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part I*, volume 8269 of *LNCS*, pages 1–20. Springer, Heidelberg, December 2013. (Cited on page 2.)

[KN09] Eike Kiltz and Gregory Neven. Identity-based signatures. In Marc Joye and Gregory Neven, editors, *Identity-Based Cryptography*. IOS Press, 2009. (Cited on page 2.)

[KPW15] Eike Kiltz, Jiaxin Pan, and Hoeteck Wee. Structure-preserving signatures from standard assumptions, revisited. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015, Part II*, volume 9216 of *LNCS*, pages 275–295. Springer, Heidelberg, August 2015. (Cited on page 3.)

[KW15] Eike Kiltz and Hoeteck Wee. Quasi-adaptive NIZK for linear subspaces revisited. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 101–128. Springer, Heidelberg, April 2015. (Cited on page 3.)

[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, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 318–335. Springer, Heidelberg, April 2012. (Cited on page 2.)

[LP19] Roman Langrehr and Jiaxin Pan. Tightly secure hierarchical identity-based encryption. In Dongdai Lin and Kazue Sako, editors, *PKC 2019, Part I*, volume 11442 of *LNCS*, pages 436–465. Springer, Heidelberg, April 2019. (Cited on page 6, 8.)

[LW14] Allison B. Lewko and Brent Waters. Why proving HIBE systems secure is difficult. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 58–76. Springer, Heidelberg, May 2014. (Cited on page 2.)

[NR97] Moni Naor and Omer Reingold. On the construction of pseudo-random permutations: Luby-Rackoff revisited (extended abstract). In *29th ACM STOC*, pages 189–199. ACM Press, May 1997. (Cited on page 3.)

[Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and David Chaum, editors, *CRYPTO'84*, volume 196 of *LNCS*, pages 47–53. Springer, Heidelberg, August 1984. (Cited on page 2.)

[SOK00] Ryuichi Sakai, Kiyoshi Ohgishi, and Masao Kasahara. Cryptosystems based on pairing. In *SCIS 2000*, Okinawa, Japan, January 2000. (Cited on page 2.)

[SW08] Elaine Shi and Brent Waters. Delegating capabilities in predicate encryption systems. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *ICALP 2008, Part II*, volume 5126 of *LNCS*, pages 560–578. Springer, Heidelberg, July 2008. (Cited on page 9.)

[Wat05] Brent R. Waters. Efficient identity-based encryption without random oracles. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 114–127. Springer, Heidelberg, May 2005. (Cited on page 2.)

[Wat09] Brent Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 619–636. Springer, Heidelberg, August 2009. (Cited on page 2.)

[Wee14] Hoeteck Wee. Dual system encryption via predicate encodings. In Yehuda Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 616–637. Springer, Heidelberg, February 2014. (Cited on page 3.)