Unbounded HIBE with Tight Security

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

[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

Abstract. We propose the first tightly secure and *unbounded* hierarchical identity-based encryption (HIBE) scheme based on standard assumptions. Our main technical contribution is a novel proof strategy that allows us to tightly randomize user secret keys for identities with arbitrary hierarchy depths using low entropy hidden in a small and hierarchy-independent master public key.

The notion of unbounded HIBE is proposed by Lewko and Waters (Eurocrypt 2011). In contrast to most HIBE schemes, an unbounded scheme does not require any maximum depth to be specified in the setup phase, and user secret keys or ciphertexts can be generated for identities of arbitrary depths with hierarchy-independent system parameters.

While all the previous unbounded HIBE schemes have security loss that grows at least linearly in the number of user secret key queries, the security loss of our scheme is only dependent on the security parameter, even in the multi-challenge setting, where an adversary can ask for multiple challenge ciphertexts. We prove the adaptive security of our scheme based on the Matrix Decisional Diffie-Hellman assumption in prime-order pairing groups, which generalizes a family of standard Diffie-Hellman assumptions such as k-Linear.

Keywords. Unbounded hierarchical identity-based encryption, tight security, multi-challenge security

1 Introduction

1.1 Motivation

Hierarchical identity-based encryption (HIBE) [26,16] is a generalization of identity-based encryption (IBE) [36]. It offers more flexibility in sharing sensitive data than IBE or classical public-key encryption (PKE).

In an HIBE scheme, users' identities are arranged in an organizational hierarchy and, more precisely, a hierarchical identity is a vector of identities of some length $p > 0$. As in an IBE scheme, anyone can encrypt a message with respect to an identity $\mathsf{id} := (\mathsf{id}_1, ..., \mathsf{id}_p)$ by access to only the public parameters. To decrypt

* Part of the work done at Karlsruhe Institute of Technology, Karlsruhe, Germany. Supported in part by ERC CoG grant 724307.

this encrypted message, one of id's ascendants at level p' where $0 < p' < p$ can delegate a user secret key for id, in addition to asking the trusted authority for id's user secret key as in the IBE setting. Furthermore, a user at level p is not supposed to decrypt any ciphertext for a recipient who is not among its descendants.

The security we focus on in this paper is adaptive security, where an adversary is allowed to declare a fresh challenge identity $\mathsf{id}^\star$ adaptively and obtain a challenge ciphertext of $\mathsf{id}^\star$ after seeing user secret keys for arbitrary chosen identities and (master) public keys. It is a widely accepted security notion for both HIBE and IBE schemes. Most of the existing HIBE schemes in the standard model have a security loss of at least Q_e (such as [9,6]) or even Q_e^L [39], where Q_e is the maximum number of user secret key queries and L is the maximum hierarchy depth. Constructions from recent work of Langrehr and Pan (LP) [29,30] are the known exceptions. Their security loss depends only on the security parameter, but not Q_e. However, their master public key size[3] depends on L. As L grows, the master public key becomes larger.

In particular, the maximum hierarchy depth L needs to be fixed in the setup phase. Once it is fixed and master public keys are generated, there is no way to add new levels into the hierarchy. This can be an undesirable burden to deploy HIBE in practice since institutions grow rapidly nowadays. Hence, it is more desirable to construct a tightly secure HIBE scheme whose master public keys are independent of the maximum hierarchy depth.

We note that the limitation mentioned above exists not only in the LP schemes but also in almost all the HIBE schemes even with non-tight security in the standard model. The notion of unbounded HIBE from Lewko and Waters [33] is proposed to overcome this limitation. In an unbounded HIBE, the whole scheme is not bounded to the maximum depth L. In particular, its master public keys, user secret keys and ciphertexts are all independent of L. (Though the user secret keys and ciphertexts can still depend on the actual hierarchy depth of the identity.) They and the follow-up work [31,18] give constructions of unbounded HIBE in composite- and prime-order pairing groups, respectively, to implement this notion. Unfortunately, none of these constructions is tight.

Our goal: Tightly secure unbounded HIBE. In this paper, we aim at constructing unbounded HIBE with tight reductions based on standard assumptions. We start recalling tight security and then give some reasons about why it is technically challenging to achieve this goal.

A security reduction is usually used to prove the security of a cryptographic scheme S by reducing any attacker $\mathcal{A}$ against S to an attacker $\mathcal{R}$ against a corresponding computational hard problem P in an efficient way. After that, we can conclude that breaking the security of S is at least as hard as solving P. More precisely, we establish a relation that states $\varepsilon_{\mathcal{A}} \leq \ell \cdot \varepsilon_{\mathcal{R}}$. Here $\varepsilon_{\mathcal{A}}$ and $\varepsilon_{\mathcal{R}}$ are success probability of $\mathcal{A}$ and $\mathcal{R}$, respectively, and for simplicity we ignore the

[3] We measure the size of the master public key in terms of the number of group elements.

additive negligible terms and assume that the running time of $\mathcal{R}$ is approximately the same as that of $\mathcal{A}$.

Ideally, we want a reduction to be *tight*, namely, ℓ to be a small constant. Recent works are also interested in "almost tight security", where ℓ may be (for instance, linearly or logarithmically) dependent on the security parameter, but not the size of $\mathcal{A}$. We will not distinguish these two tightness notions, but state the precise security loss in security proofs and comparison of schemes. A tight security reduction means the security of S is tightly coupled with the hardness of P. A scheme with tight reductions is more desirable since it provides the same level of security regardless of the application size. Moreover, we can implement it with smaller parameters and do not need to compensate for the security loss. As a result, tightly secure schemes drew a lot of attention in the last few years, from basic primitives, such as PKE [13,14,21] and signature [1,15] schemes, to more advanced ones, such as (non-interactive) key exchange [17,22,10], zero-knowledge proof [3,2], IBE [9,6,20,23] and functional encryption [37] schemes. Currently, research is carried out to reduce the cost for tight security. For instance, for PKE, the public key size is shortened from being linear [13] (in the security parameter) to constant [14,21]. In particular, the scheme in [14] only has one element more in the ciphertext overhead than its non-tight counterpart [28] asymptotically. By taking the concrete security loss into account, we are optimistic that scheme in [14] will have shorter ciphertext length in terms of bits.

Difficulties in achieving our goal. Given the existing research, it is quite challenging to construct a tightly secure HIBE, even for a bounded one. Firstly, the potential difficulty of this task has been shown by Lewko and Waters [34], namely, it is hard to prove an HIBE scheme with security loss less than exponential in L, if its user secret keys are rerandomizable over all "functional" keys. Secondly, the work of Blazy, Kiltz, and Pan (BKP) [6] is the first that claimed to have solved this challenge by proposing a bounded tightly secure HIBE. Their scheme has indeed bypassed the impossibility result of [34] by having its user secret keys only rerandomizable in a subspace of all "functional" keys, which is similar to schemes based on the dual system technique [9,32]. Unfortunately, shortly after its publication, a technical flaw was found in their proof, which shows that their proof strategy is insufficient for HIBE with flexible identity depth.

Recently, Langrehr and Pan have proposed the first tightly secure HIBE in the standard model [29]. A very recent and concurrent work [30] improves this HIBE and proposes a tightly secure HIBE in the multi-challenge setting. Core techniques in both papers crucially require their master public key size depend on the maximum hierarchy, L. More precisely, they need to know L in advance so that they can choose independent master secret keys for different levels, which will be turned into master public keys. With these relatively large master secret keys, they can apply their independent randomization to isolate randomization for identities with different maximum levels. As a result, their scheme is bounded to the maximum level L of the whole HIBE scheme and its master public key size is dependent on L.

1.2 Our Contribution

We construct the *first* tightly secure unbounded HIBE based on standard assumptions. Our scheme is furthermore tightly multi-challenge secure. The multi-challenge security is a more realistic notion for (H)IBE, where an adversary is allowed to query multiple challenge identities adaptively and obtain the corresponding ciphertexts. It has comparable efficiency to its non-tight counterparts [31,18], and, in particular, it has shorter ciphertext and user secret key than the scheme of [31]. At the core of our construction is a novel technique that allows us to prove tight adaptive security of HIBE with "small", hierarchy-independent master public keys.

More precisely, the identity space for our scheme $\mathcal{ID} := \mathcal{S}^*$ has unbounded depth and the base set $\mathcal{S}$ can be arbitrary. In this section, we consider $\mathcal{S} := \{0, 1\}^n$ for simplicity, where n is the security parameter. The master public key of our scheme is independent of L and contains only $\mathbf{O}(n)$-many group elements, which is the same as the existing tightly secure IBE schemes [9,6,20,23].

All our security proofs are in the standard model and based on the Matrix Decisional Diffie-Hellman (MDDH) assumption [11] in prime-order asymmetric pairing groups. The MDDH assumption is a generalization of a class of Decisional Diffie-Hellman assumptions, such as the k-Lin [24] and aSymmetric eXternal Diffie-Hellman (SXDH) (for $k = 1$) assumptions. The security of our MAC requires an additional assumption on the existence of collision-resistant hash functions. There exist collision-resistant hash functions in the standard model that maps arbitrary-length bit-strings to fixed-length ones using fixed-length keys. For instance, one can use the Merkle-Damgård construction with hash functions from the SHA familiy or the less efficient but completely provably secure one from the discrete logarithm assumption.

Efficiency comparison. We compare the efficiency of bounded and unbounded HIBE schemes in the standard model with prime-order pairings in Table 1. We note that [35] achieves a weaker notion of unbounded HIBE in the sense that their master public key is independent of L, but the size of the user secret key is dependent on L. More precisely, their user secret key contains $\mathbf{\Omega}(L - p)$-many group elements for an identity $\mathsf{id} := (\mathsf{id}_1, \ldots, \mathsf{id}_p)$.

According to Table 1, our scheme has shorter ciphertexts and user secret keys than Lew12, which is comparable to GCTC16. We note that both Lew12 and GCTC16 are unbounded HIBE with non-tight reductions, while ours are tight. Thus, when accounting for a larger security loss in the reduction with larger groups, our scheme may have shorter ciphertexts and user secret keys than GCTC16 at the concrete level. We want to emphasize that our scheme is not fully practical yet, but it lays down a theoretical foundation for more efficient unbounded HIBE with tight security in the future.

Extensions. Our unbounded HIBE scheme directly implies a tightly secure unbounded identity-based signature by the Naor transformation. Furthermore, our HIBE is compatible with the Quasi-Adaptive NIZK (QANIZK) for linear subspaces and thus, similar to [23] it can be combined with a tightly simulation-

Scheme	U	$\lvert\mathsf{mpk}\rvert$	$\lvert\mathsf{usk}\rvert$	$\lvert\mathsf{C}\rvert$	Loss	MC	Assumption
Wat05 [39]	✗	$\mathbf{O}(nL)\lvert\mathbb{G}\rvert$	$\mathbf{O}(nL)\lvert\mathbb{G}\rvert$	$(1+p)\lvert\mathbb{G}\rvert$	$\mathbf{O}(nQ_{\mathsf{e}})^L$	✗	DBDH
Wat09 [38]	✗	$\mathbf{O}(L)\lvert\mathbb{G}\rvert$	$\mathbf{O}(p)(\lvert\mathbb{G}\rvert+\lvert\mathbb{Z}_q\rvert)$	$\mathbf{O}(p)(\lvert\mathbb{G}\rvert+\lvert\mathbb{Z}_q\rvert)$	$\mathbf{O}(Q_{\mathsf{e}})$	✗	2-LIN
Lew12[31]	✓	$60\lvert\mathbb{G}\rvert+2\lvert\mathbb{G}_T\rvert$	$(60+10p)\lvert\mathbb{G}\rvert$	$10p\lvert\mathbb{G}\rvert$	$\mathbf{O}(Q_{\mathsf{e}})$	✗	2-LIN
OT12 [35]	✗	$160\lvert\mathbb{G}\rvert$	$\mathbf{O}(p^2L)\lvert\mathbb{G}\rvert$	$3+6p\lvert\mathbb{G}\rvert$	$\mathbf{O}(Q_{\mathsf{e}}L^2)$	✗	2-LIN
CW13 [9]	✗	$\mathbf{O}(L)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$\mathbf{O}(L)\lvert\mathbb{G}_2\rvert$	$4\lvert\mathbb{G}_1\rvert$	$\mathbf{O}(Q_{\mathsf{e}})$	✗	SXDH
BKP14 [6]	✗	$\mathbf{O}(L)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$\mathbf{O}(L)\lvert\mathbb{G}_2\rvert$	$4\lvert\mathbb{G}_1\rvert$	$\mathbf{O}(Q_{\mathsf{e}})$	✗	SXDH
GCTC16 [18]	✓	$18(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)+3\lvert\mathbb{G}_T\rvert$	$(18\lceil p/3\rceil-3p+18)\lvert\mathbb{G}_2\rvert$	$9\lceil p/3\rceil\lvert\mathbb{G}_1\rvert$	$\mathbf{O}(QL)$	✗	SXDH
LP19$_1^{\mathcal{H}}$ [29]	✗	$\mathbf{O}(\gamma L)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$\mathbf{O}(\gamma L)\lvert\mathbb{G}_2\rvert$	$5\lvert\mathbb{G}_1\rvert$	$\mathbf{O}(\gamma L)$	✗	SXDH
LP19$_2^{\mathcal{H}}$ [29]	✗	$\mathbf{O}(\gamma L)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$(3p+2)\lvert\mathbb{G}_2\rvert$	$(3p+2)\lvert\mathbb{G}_1\rvert$	$\mathbf{O}(\gamma)$	✗	SXDH
LP20$_1^{\mathcal{H}}$ [30]	✗	$\mathbf{O}(\gamma L)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$\mathbf{O}(\gamma L)\lvert\mathbb{G}_2\rvert$	$5\lvert\mathbb{G}_1\rvert$	$\mathbf{O}(\gamma L)$	✓	SXDH
LP20$_2^{\mathcal{H}}$ [30]	✗	$\mathbf{O}(\gamma L)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$(3p+2)\lvert\mathbb{G}_2\rvert$	$(3p+2)\lvert\mathbb{G}_1\rvert$	$\mathbf{O}(\gamma L)$	✓	SXDH
Ours (Fig. 14)	✓	$\mathbf{O}(\gamma)(\lvert\mathbb{G}_1\rvert+\lvert\mathbb{G}_2\rvert)$	$(7p+2)\lvert\mathbb{G}_2\rvert$	$(7p+2)\lvert\mathbb{G}_1\rvert$	$\mathbf{O}(\gamma)$	✓	SXDH

Table 1. Comparison of bounded and unbounded HIBEs in prime-order pairing groups with adaptive security in the standard model based on static assumptions. The second column indicates whether an HIBE is bounded (✗) or unbounded (✓). The identity space for bounded HIBE is $(\{0,1\}^n)^{\leq L}$ and that for unbounded HIBE is $(\{0,1\}^n)^*$. γ is the bit length of the range of a collision-resistant hash function. '$\lvert\mathsf{mpk}\rvert$,' '$\lvert\mathsf{usk}\rvert$,' and '$\lvert\mathsf{C}\rvert$' stand for the size of the master public key, a user secret key and a ciphertext, respectively. 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 $\mathbb{G}(:= \mathbb{G}_1 = \mathbb{G}_2)$. In the '$\lvert\mathsf{usk}\rvert$' and '$\lvert\mathsf{C}\rvert$' columns p stands for the hierarchy depth of the identity vector. In bounded HIBEs, L denotes the maximum hierarchy depth. In the security loss, Q_{e} denotes the number of user secret key queries by the adversary. MC stands for multi-challenge and this column indicates whether the adversary is allowed to query multiple challenge ciphertexts (✓) or just one (✗). Lew12 is the prime-order variant of the unbounded scheme in [33].

sound QANIZK to construct a tightly CCA-secure unbounded HIBE in the multi-challenge setting. We give a detailed treatment in the full version for completeness.

1.3 Technical Overview

To achieve our goal, we develop a novel tight method that uses (limited) entropy hidden in hierarchy-independent master public key to generate enough entropy to randomize user secret keys of identities with unbounded hierarchy depths (in a computational manner). As a bonus, our technique naturally give us tight multi-challenge security.

A modular treatment: From MAC to HIBE. We follow the modular approach of Blazy, Kiltz, and Pan (BKP) [6] to construct our unbounded HIBE. The basis of our construction is a novel tightly secure message authentication code (MAC). Our MAC has *suitable algebraic structures* and thus can be turned into an unbounded HIBE tightly by adapting the BKP framework.

The BKP framework [6] tightly reduces constructing an (H)IBE to a suitable affine MAC. As a result, we only need to focus on constructing the suitable MAC. Affine MACs are algebraic MACs that have affine structures, and such structures allow transformation to (H)IBEs. This framework abstracts the first tightly secure IBE from Chen and Wee (CW) [9] and can be viewed as extending the "MAC→Signature" framework of Bellare and Goldwasser [5] to the IBE setting by using the affine structure and pairings. Most of the tightly secure IBE and HIBE schemes are related to this framework, such as [25,20,19,23,29,30].

Preparation: Shrinking the message space via hashing. We first apply a collision-resistant hash function to shrink the message space which the "bit-by-bit" argument applies on. More precisely, let $H : \{0,1\}^* \to \{0,1\}^n$ be a collision-resistant hash function. For an (unbounded) hierarchical message $\mathsf{m} := (\mathsf{m}_1, \dots, \mathsf{m}_p) \in (\{0,1\}^n)^p$, we hash every i-th prefix ($1 \leq i \leq p$) and have the hashed message $\mathsf{hm} := (\mathsf{hm}_1, \mathsf{hm}_2, \dots, \mathsf{hm}_p)$ where $\mathsf{hm}_i := H(\mathsf{m}_1, \dots, \mathsf{m}_i) \in \{0,1\}^n$. The collision-resistance guarantees that it is hard for an adversary to find two distinct m and $\mathsf{m}^\star$ messages with $H(\mathsf{m}) = H(\mathsf{m}^\star)$. In particular, after hashing every prefixes of a message, if a hierarchical message m is not a prefix of $\mathsf{m}^\star$, then the last hash value of m is different to every hash value of $\mathsf{m}^\star$. As a result, our argument is only applied on the last hash value.

Our strategy: "Inject-and-Pack". Our strategy contains two steps: (1) injecting enough randomness into MAC tags locally and (2) packing the local randomness and lift it up to the global level. Both steps are compatible with each other, and they only rely on the limited entropy in the hierarchy-independent MAC keys and can provide tight security even in the multi-challenge setting.

Our MAC has the following structures that enable our "inject-and-pack" strategy. This is captured by our MAC scheme MAC_u in Section 3.2.

For a hierarchical message $\mathsf{m} := (\mathsf{m}_1, \dots, \mathsf{m}_p)$, our MAC tag $\tau_\mathsf{m} := (([\mathbf{t}_i]_2, [\tilde{\mathbf{t}}_i]_2, [\mathbf{u}_i]_2)_{1 \leq i \leq p}, [\tilde{\mathbf{u}}]_2)$ has the following form:

$$\begin{aligned} \mathbf{t}_i &:= \mathbf{B}\mathbf{s}_i \in \mathbb{Z}_q^{n_1} \text{ and } \tilde{\mathbf{t}}_i := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i \in \mathbb{Z}_q^{n_2} \quad \text{for} \quad \mathbf{s}_i, \tilde{\mathbf{s}}_i \overset{\$}{\leftarrow} \mathbb{Z}_q^{n_3} \\ \mathbf{u}_i &:= \boxed{\textstyle\sum_{j=1}^n \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]}\mathbf{t}_i} + \boxed{\tilde{\mathbf{X}}_1\tilde{\mathbf{t}}_i} \in \mathbb{Z}_q^{n_4} \\ \tilde{\mathbf{u}} &:= \boxed{\textstyle\sum_{j=1}^p \tilde{\mathbf{X}}_2\tilde{\mathbf{t}}_j} + \colorbox{lightgray}{$\mathbf{x}'$}, \end{aligned} \tag{1}$$

where $\mathbf{B} \overset{\$}{\leftarrow} \mathbb{Z}_q^{n_1 \times n_3}$, $\tilde{\mathbf{B}} \overset{\$}{\leftarrow} \mathbb{Z}_q^{n_2 \times n_3}$[4], $\mathbf{X}_{j,b} \overset{\$}{\leftarrow} \mathbb{Z}_q^{n_4 \times n_1}$ for $1 \leq j \leq n, b \in \{0,1\}$ and $\tilde{\mathbf{X}}_1, \tilde{\mathbf{X}}_2 \overset{\$}{\leftarrow} \mathbb{Z}_q^{n_4 \times n_2}$ and $\mathbf{x}' \overset{\$}{\leftarrow} \mathbb{Z}_q^{n_4}$ and they are all contained in the secret key of our MAC, namely, $\mathsf{sk}_{\mathsf{MAC}} := (\mathbf{B}, \tilde{\mathbf{B}}, (\mathbf{X}_{j,b})_{\text{for } 1 \leq j \leq n, b \in \{0,1\}}, \tilde{\mathbf{X}}_1, \tilde{\mathbf{X}}_2, \mathbf{x}')$. Here the (hierarchical) message space of a MAC is the identity space of the resulting HIBE.

We highlight different purposes of different parts in our MAC tags:

- randomizing $\colorbox{lightgray}{$\mathbf{x}'$}$ is our end goal. In the resulting HIBE, once $\mathbf{x}'$ is randomized, it will further randomize challenge ciphertexts;
- the linear part, $\boxed{\sum_{j=1}^n \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]}\mathbf{t}_i}$, is used to inject randomness;
- with the packing helpers, $\boxed{\tilde{\mathbf{X}}_1\tilde{\mathbf{t}}_i}$ and $\boxed{\sum_{j=1}^p \tilde{\mathbf{X}}_2\tilde{\mathbf{t}}_j}$, we can transfer the injected randomness in $\mathbf{u}_p$ to randomize $\colorbox{lightgray}{$\mathbf{x}'$}$.

[4] For simplicity, we choose $\mathbf{B}$ and $\tilde{\mathbf{B}}$ uniformly at random here, while in the actual scheme we choose them based on the underlying assumption.

We will discuss how to choose the dimensions of these random matrices and vectors to enable our strategy.

Before that, we stress that it is crucial to generate $([\mathbf{t}_i]_2, [\tilde{\mathbf{t}}_i]_2, [\mathbf{u}_i]_2)$ for all $1 \leq i \leq p$ and $\mathsf{hm}_i := H(\mathsf{m}_1, ..., \mathsf{m}_i)$ so that we can delegate and randomize MAC tags for further levels by publishing $([\mathbf{B}]_2, [\tilde{\mathbf{B}}]_2, ([\mathbf{X}_{j,b}\mathbf{B}]_2)_{j,b}, [\tilde{\mathbf{X}}_1\tilde{\mathbf{B}}]_2, [\tilde{\mathbf{X}}_2\tilde{\mathbf{B}}]_2)$. Details about public delegation can be found in Remark 1 and the full version.

Interlude: Security requirement. The MAC security we need for the "MAC-to-HIBE" transformation is pseudorandomness against adaptive chosen message attacks, which is a decisional version of the $\mathsf{EUF\text{-}CMA}$ security of MAC. To simplify our discussion, we use the $\mathsf{EUF\text{-}CMA}$ notion only in this chapter, but in the main body we prove the decisional one. In the $\mathsf{EUF\text{-}CMA}$ security game, an adversary can adaptively ask many MAC tag queries and at some point it will submit one forgery. For the multi-challenge security, we allow the adversary submit multiple forgeries. Here we only consider one forgery for simplicity. Note that our technique works tightly for multiple forgeries.

Local step: Injecting randomness. Here we only focus terms in the solid box of Equation (1) and find a right way to define the dimensions to implement the injection strategy. We note that one cannot use the idea of BKP MAC here, since it uses a square full-rank matrix $\mathbf{B} \in \mathbb{Z}_q^{k \times k}$ and there is no room to hide $\mathbf{X}_{j,b}$ from the published terms $[\mathbf{X}_{j,b}\mathbf{B}]_2$. These terms have to be public to delegate secret keys, while it is not a problem for IBE. Moreover, the same $(\mathbf{X}_{j,b})_{1 \leq j \leq n, b \in \{0,1\}}$ is re-used for all $\mathbf{u}_i$ and the injected randomness will be leaked along them, which is another issue we encounter with the BKP MAC.

To have control on where to inject randomness, we increase the number of row vectors in $\mathbf{B} \xleftarrow{\$} \mathbb{Z}_q^{3k \times k}$, namely, $n_1 := 3k$, as the LP method in [29], where $\mathbf{X}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{1 \times 3k}$ are row vectors. Now the column space of $\mathbf{B}$, $\mathsf{Span}(\mathbf{B}) := \{\mathbf{v} \mid \exists \mathbf{w} \in \mathbb{Z}_q^k \text{ s.t. } \mathbf{v} = \mathbf{B} \cdot \mathbf{w}\}$, is a subspace of $\mathbb{Z}_q^{3k}$ and there is a non-zero kernel matrix $\mathbf{B}^\perp \in \mathbb{Z}_q^{3k \times 2k}$ such that $(\mathbf{B}^\perp)^\top \mathbf{B} = \mathbf{0} \in \mathbb{Z}_q^{2k \times k}$. $\mathsf{Span}(\mathbf{B}^\perp)$ is orthogonal to $\mathsf{Span}(\mathbf{B})$.

We introduce a random function "inside" $\mathsf{Span}(\mathbf{B}^\perp)$ by tight reductions to the MDDH assumption and all $\mathbf{u}_i$ $(1 \leq i \leq p)$ in Equation (1) will distribute according to the following new form:

$$\mathbf{u}_i := \Big(\sum\nolimits_{j=1}^{n} \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]}^\top + \mathsf{RF}(\mathsf{hm}_i) \cdot (\mathbf{B}^\perp)^\top\Big)\mathbf{t}_i + \tilde{\mathbf{X}}_1 \tilde{\mathbf{t}}_i \in \mathbb{Z}_q \,. \tag{2}$$

Now $\mathsf{RF}(\mathsf{hm}_i)$ is multiplied by $\mathbf{B}^\perp$ and we can control where it gets introduced by choose $\mathbf{t}_i \notin \mathsf{Span}(\mathbf{B})$. More precisely, we only introduce the random function, RF, in $\mathbf{u}_p$ at level p for a hierarchical identity $\mathsf{m} := (\mathsf{m}_1, ..., \mathsf{m}_p)$.

The above idea is borrowed from [29], but it is still not enough to correctly inject randomness: It only helps us to hide RF in MAC tag queries, but we still have issue in answering the verification query for an adversary's forgery. The issue described below does not happen in the BKP and LP [29] schemes, since our MAC has more expressive structure. More precisely, on a forgery of message $\mathsf{m}^\star := (\mathsf{m}_1^\star, ..., \mathsf{m}_p^\star)$, we need to verify whether the forgery satisfies Equation (1), which form an explicit hierarchy. Since we have no control of how an adversary computes its random $\mathbf{t}_i^\star$, in answering one verification query, we compute RF on p many distinct messages, $\mathsf{hm}_1^\star, ..., \mathsf{hm}_p^\star$. This leaks too much information about RF.

Our solution is to increase the number of row vectors in $\mathbf{X}_{j,b}$ from 1 to k, namely, $n_4 := k$. As a result, there is room for us to use an assumption (namely, the MDDH assumption [11]) to tightly inject randomness into these row vectors. Thus, in the end, verification equations defined by Equation (1) get randomized and the information about RF is properly hidden. We refer Lemma 4 for technical details. The whole core step is formally captured by the Randomness Injection Lemma (cf. Lemma 4). Furthermore, this lemma abstracts the core ideas of [30].

Global step: Packing randomness. After the randomness is injected in $\mathbf{u}_i$ at the local level, we pack and move it into the global level to randomize $\mathbf{x}'$ which will be use to randomize the challenge ciphertexts. Implicitly, we pack the randomness firstly in $\tilde{\mathbf{t}}_p$ for an identity has p levels via the packing helper $\tilde{\mathbf{X}}_1\tilde{\mathbf{t}}_p$. Secondly, via another packing helper $\tilde{\mathbf{X}}_2\tilde{\mathbf{t}}_p$, we move the randomness into $\tilde{\mathbf{u}}$.

We choose $\tilde{\mathbf{B}} \xleftarrow{\$} \mathbb{Z}_q^{2k\times k}$, namely, $n_2 := 2k$, so that there is enough room to implement the above packing steps. Although the randomness is successfully injected, it may be leaked from MAC tag and verification queries during the packing process. In particular, we have small MAC secret keys. To accomplish the task, we carefully design several intermediate hybrid steps and apply the MDDH assumption several times. We refer Lemma 5 for details. The whole core step is formally captured by the Randomness Packing Lemma (cf. Lemma 5).

An alternative interpretation: Localizing HIBEs into IBEs, tightly. In contrast to the methods of Langrehr and Pan [29,30], our overall idea can be viewed as localizing a p-level HIBE into p IBE pieces which share the same master public and secret keys, and p is an arbitrary integer. In the security proof, we generate enough entropy locally and then extract it to the global level to argue the security of HIBE. Such an idea is borrowed from [33,31,18], where some variants of Boneh-Boyen's IBE [7] are used at the local level and all these IBE pieces are connected via a secret sharing method. However, implementing this idea with tight reductions is rather challenging, even with the existing tightly secure (H)IBEs (such as [9,6,20,29,30]). We observed that these techniques either fail to introduce local entropy or cannot collect the local randomness to argue the security of the (global) HIBE.

1.4 More Discussion on Related Work

The family of LP HIBE schemes. To implement the "level-by-level" argument, the LP HIBEs [29,30] require the size of master public keys dependent on the maximum hierarchy depth, L, so that they have enough entropy to randomize corresponding MAC tags.

Our approach provides an economic, tightly secure technique to do the randomization with more compact and hierarchy-independent master keys. Our technique uses and abstracts the core technique in a very recent and concurrent work [30] to inject randomness. As we showed above, injecting randomness is not enough for our goal and we require an additional suitable randomness packing technique. [30] achieves tight multi-challenge security for bounded HIBE, while ours is for unbounded HIBE.

Other techniques for tight multi-challenge security. Over the last few years, several techniques have been proposed for tightly secure IBE in the multi-challenge setting, such as [4,25,19,20,23], where [4,19] are based on strong and non-standard assumptions and [25] requires a composite-order group. Motivated by [25], the work of [20,23] construct the tightly multi-challenge secure IBE schemes in the prime-order group and they both follow the BKP method. They have the same limitation as discussed in the "Local Step: Injecting randomness" section and cannot be used for our goal, since their $\mathbf{B}$ is also full-rank square matrix. The same kind of information about $\mathbf{X}_{j,b}$ is leaked.

Furthermore, in the work of Hofheinz, Jia, and Pan [23] (also in [20] and BKP), they randomize their MAC by developing a random function, RF, in the $\mathbb{Z}_q$ full space gradually. This is problematic in the unbounded HIBE setting: When we "plug" their MAC into our framework, there is no room to hide RF and by a "mix-and-match" approach an adversary can learn $\mathsf{RF}(\mathsf{hm}^\star)$, where $\mathsf{hm}^\star := H(\mathsf{m}^\star)$. Imagine a challenge message $\mathsf{m}^\star \in \{0,1\}^n$. By asking a MAC tag of $(\mathsf{m}^\star, \mathsf{m})$, an adversary can easily learn $\mathsf{RF}(\mathsf{hm}^\star)$ from $\mathbf{u}_1$. Finally, [29] has discussed why these multi-challenge security techniques cannot be used for HIBEs.

Other unbounded technique. Chen et al. [8] proposes a variant of the bilinear entropy expansion lemma [27] in prime-order groups, which can be used to transform a (bounded) attribute-based encryption (ABE) scheme to an unbounded one in a tight manner. However, we note that their lemma requires a certain algebraic structure of the underlying scheme, which the LP schemes [29,30] do not have. Moreover, they only prove their scheme in the single-challenge setting, and it is not clear for us whether their single-challenge security tightly implies multi-challenge security.

Open problems. It is interesting to consider if we can extend our "inject-and-pack" strategy in a more general setting, such as predicate encryption schemes. Another open problem is to consider the Master-Key-KDM security [12] for HIBEs. Garg et al.[12] proposed a Master-Key-KDM secure IBE based on a tightly multi-challenge secure IBE. We are optimistic that our unbounded HIBE can be adapted to achieve the KDM security by following the approach of Garg et al., since our scheme has tight multi-challenge security as well. However, we leave a formal treatment of it as an open problem.

2 Preliminaries

Notations. We use $x \overset{\$}{\leftarrow} \mathcal{S}$ to denote the process of sampling an element x from $\mathcal{S}$ uniformly at random if $\mathcal{S}$ is a set and to denote the process of running $\mathcal{S}$ with its internal randomness and assign the output to x if $\mathcal{S}$ is an algorithm. The expression $a \overset{?}{=} b$ stands for comparing a and b on equality and returning the result in Boolean value. 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 $\left(\mathbf{A}^{-1}\right)^{\top}$. $\mathsf{GL}_k(\mathbb{Z}_q)$ denotes the set of invertible $k \times k$ matrices in $\mathbb{Z}_q$. $\mathbf{I}_k$ is the $k \times k$ identity matrix. 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 – unless state otherwise – $\mathbf{A}^{\perp}$ 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 $\mathcal{S}$ and $\mathcal{S}^* := \bigcup_{n=1}^{\infty} \mathcal{S}^n$. For an n-tuple or string $\mathsf{m} \in \mathcal{S}^n$, $\mathsf{m}_i \in \mathcal{S}$ and $\mathsf{m}[\![i]\!] \in \mathcal{S}$ both denote the i-th component of m $(1 \le i \le n)$ and $\mathsf{m}_{|i} \in \mathcal{S}^i$ denotes the prefix of length i of m.

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 [6], 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 pseudo-code. 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. The group elements P_1 and P_2 are generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. The function $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ 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 [11]. 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 $\mathbb{G}_s$. $\mathsf{Span}(\mathbf{A}) := \{\mathbf{A}\mathbf{r} | \mathbf{r} \in \mathbb{Z}_q^m\} \subset \mathbb{Z}_q^n$ denotes the linear span of $\mathbf{A}$, and similarly $\mathsf{Span}([\mathbf{A}]_s) := \{[\mathbf{A}\mathbf{r}]_s | \mathbf{r} \in \mathbb{Z}_q^m\} \subset \mathbb{G}_s^n$. Note that it is efficient to compute $[\mathbf{A}\mathbf{B}]_s$ given $([\mathbf{A}]_s, \mathbf{B})$ or $(\mathbf{A}, [\mathbf{B}]_s)$ with matching dimensions. We define $[\mathbf{A}]_1 \circ [\mathbf{B}]_2 := e([\mathbf{A}]_1, [\mathbf{B}]_2) = [\mathbf{A}\mathbf{B}]_T$, 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 [11].

Definition 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 ($\mathcal{D}_{\ell,k}$-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* PGGen *in group $\mathbb{G}_s$ if for all PPT adversaries $\mathcal{A}$, it holds that*

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

is negligible where the probability is taken over $\mathcal{PG} \xleftarrow{\$} \mathsf{PGGen}(1^\lambda)$, $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}$, $\mathbf{w} \xleftarrow{\$} \mathbb{Z}_q^k$ 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 [13] that $\mathcal{U}_k$-MDDH and $\mathcal{U}_{\ell,k}$-MDDH assumptions are equivalent.

Definition 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 1 ($\mathcal{U}_{\ell,k}$-MDDH $\Leftrightarrow$ $\mathcal{U}_k$-MDDH [13]). *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 ($\mathcal{D}_{\ell,k}$-MDDH $\Rightarrow$ $\mathcal{U}_k$-MDDH [11]). *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 $(\mathcal{PG}, [\mathbf{A}], [\mathbf{AW}])$ and $(\mathcal{PG}, [\mathbf{A}], [\mathbf{U}])$. 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 3 (Random self-reducibility [11]). *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 $Q \in \mathbb{N}_+$ and any adversary $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

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

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

To reduce the Q-fold $\mathcal{U}_{\ell,k}$-MDDH assumption to the $\mathcal{U}_k$-MDDH assumption we have to apply Lemma 3 to get from Q-fold $\mathcal{U}_{\ell,k}$-MDDH to standard $\mathcal{U}_{\ell,k}$-MDDH and then Lemma 1 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

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

Formal definitons of collision-resistant hash functions (CRHF) and message authentication codes (MACs) can be found in the full version.

3 Unbounded Affine MAC

3.1 Core Lemmata

The following two core Lemmata contain the main ingredient for the security proof of our new unbounded MAC. They form the main technical novelty of this work. Lemma 4 abstracts the technique used in [30] . It shows that the prototypic MAC $\mathsf{MAC}_{\mathsf{lin}}$ allows the injection of randomness in the tags.

We give a brief overview of how MAC_u is constructed from $\mathsf{MAC}_{\mathsf{lin}}$: For a p-level hierarchical message $\mathsf{m} := (\mathsf{m}_1, \ldots, \mathsf{m}_p) \in (\{0,1\}^\gamma)^p$, we divide it into p pieces $\mathsf{hm}_1, \ldots, \mathsf{hm}_p$ and each $\mathsf{hm}_i := H(\mathsf{m}_1, \ldots, \mathsf{m}_i)$ where H is a collision-resistant hash function (CRHF). For each hm_i we apply $\mathsf{MAC}_{\mathsf{lin}}$ on it and the purpose of $\mathsf{MAC}_{\mathsf{lin}}$ is to inject suitable randomness at the local level.

Lemma 5 is then used to move the entropy from $\mathbf{u}_p$ to the vector $\tilde{\mathbf{u}}$ and randomize it. This makes the user secret keys information-theoretically independent from the secret $\mathbf{x}'$ and allows us to randomize h_K in the $\textsc{Chal}$ queries.

```
$\mathsf{Gen}_{\mathsf{MAC}}(1^\lambda)$:
$\mathcal{PG} \xleftarrow{\$} \mathsf{PGGen}(1^\lambda)$
$\textbf{parse } \mathcal{PG} =: (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, P_1, P_2, e)$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
$\textbf{for } j \in \{1, \ldots, \gamma\},\ b \in \{0,1\} \textbf{ do}$
    $\mathbf{X}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{k \times 3k}$
$\textbf{return } \mathsf{sk}_{\mathsf{MAC}} := \big(\mathbf{B}, (\mathbf{X}_{j,b})_{1 \le j \le \gamma, b \in \{0,1\}}\big)$

$\mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{hm} \in \{0,1\}^\gamma)$:
$\textbf{parse } \mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{X}_{j,b})_{1 \le j \le \gamma, b \in \{0,1\}}\big)$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{B}\mathbf{s}$
$\mathbf{u} := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}[\![j]\!]} \mathbf{t} \in \mathbb{Z}_q^k$
$\textbf{return } \tau := \big([\mathbf{t}]_2, [\mathbf{u}]_2\big)$
```

```
$\mathsf{Ver}_{\mathsf{MAC}}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{hm} \in \{0,1\}^\gamma, \tau)$:
$\textbf{parse } \mathsf{sk}_{\mathsf{MAC}} =: \big(\mathbf{B}, (\mathbf{X}_{j,b})_{1 \le j \le \gamma, b \in \{0,1\}}\big)$
$\textbf{parse } \tau =: \big([\mathbf{t}]_2, [\mathbf{u}]_2\big)$
$\mathbf{h} \xleftarrow{\$} \mathbb{Z}_q^k$
$\mathbf{h}_0 := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}^\star[\![j]\!]}^\top \mathbf{h}$
$\textbf{return } e\big([\mathbf{h}^\top]_1, [\mathbf{u}]_2\big) \overset{?}{=} e\big([\mathbf{h}_0^\top]_1, [\mathbf{t}]_2\big)$
```

Fig. 1. Our linear MAC $\mathsf{MAC}_{\mathsf{lin}}$ for the message space $\{0,1\}^\gamma$

Randomness Injection Lemma. We start our exposition with a message authentication code (MAC) with linear structure[5] in Figure 1, $\mathsf{MAC}_{\mathsf{lin}}$. This MAC scheme is abstracted from [30]. The tags of this MAC can be verified by checking whether $\mathbf{u} = \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}[\![j]\!]} \mathbf{t}$, but we require the more sophisticated randomized verification procedure as in Figure 1 for the transformation to an unbounded HIBE later.

The MAC $\mathsf{MAC}_{\mathsf{lin}}$ is correct, since

$$e\big([\mathbf{h}^\top]_1, [\mathbf{u}]_2\big) = \Big[\mathbf{h}^\top \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}[\![j]\!]} \mathbf{t}\Big]_T = e\big([\mathbf{h}_0^\top]_1, [\mathbf{t}]_2\big)\,.$$

Our $\mathsf{MAC}_{\mathsf{lin}}$ is a stepping stone for our unbounded MAC for constructing HIBEs. For the transformation to unbounded HIBE our $\mathsf{MAC}_{\mathsf{lin}}$ satisfies a special security notion which is captured by Lemma 4. This security notion needs to combine with Lemma 5 to get a secure MAC for the unbounded HIBE (cf. Section 3.2).

In the security experiment (defined in Figure 2), the adversary gets values in dk_1 that allow her to rerandomize tags. These values also allows her to forge arbitrary tags. This is the reason why it is not a secure MAC, but the goal of the adversary here is not to forge a tag, but to distinguish two games $\mathsf{RI}_{\mathsf{real}}$ and $\mathsf{RI}_{\mathsf{rand}}$. More precisely, $\mathcal{A}$ gets access to two oracles, $\textsc{Eval}_{\mathsf{ri}}$ that gives her a tag for a message, and $\textsc{Chal}_{\mathsf{ri}}$ that gives her necessary values to check validity of a tag. She can query these two oracles arbitrary times in an adaptive manner, but for each message $\mathcal{A}$ can query it for either $\textsc{Eval}_{\mathsf{ri}}$ or $\textsc{Chal}_{\mathsf{ri}}$, but not both. $\mathcal{A}$ wins if she can distinguish game $\mathsf{RI}_{\mathsf{real}}$ from $\mathsf{RI}_{\mathsf{rand}}$. For technical reasons the

[5] We call it “linear” since it matches the affine MAC definition from [6] without using the affine part, i.e. the message dependent part $\mathbf{u}$ of the tags depends linear on the randomness $\mathbf{t}$ of the tags.

```
$\underline{\textsc{Init}_{\mathsf{ri}}:}$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
for $j \in \{1, \ldots, \gamma\},\ b \in \{0,1\}$ do
    $\mathbf{X}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{k \times 3k}$
$\mathsf{dk}_1 := \left([\mathbf{X}_{j,b}\mathbf{B}]_2\right)_{1 \leq j \leq \gamma, b \in \{0,1\}}$
return $\left([\mathbf{B}]_2, \mathsf{dk}_1\right)$

$\underline{\textsc{Chal}_{\mathsf{ri}}(\mathsf{hm}^\star \in \{0,1\}^\gamma):}$
$\mathcal{C}_{\mathsf{hm}} := \mathcal{C}_{\mathsf{hm}} \cup \{\mathsf{hm}^\star\}$
$\mathbf{h} \xleftarrow{\$} \mathbb{Z}_q^k$
$\mathbf{h}_0 := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}^\star[\![j]\!]}^\top \mathbf{h}$
$\boxed{\mathbf{h}_0 := \mathbf{h}_0 + \mathbf{B}^\perp \mathsf{RF}(\mathsf{hm}^\star)^\top \mathbf{h}}$
return $\left([\mathbf{h}]_1, [\mathbf{h}_0]_1\right)$
```

```
$\underline{\textsc{Eval}_{\mathsf{ri}}(\mathsf{hm} \in \{0,1\}^\gamma):}$
if $\mathsf{hm} \in \mathcal{Q}_{\mathsf{hm}}$ then return $\perp$
$\mathcal{Q}_{\mathsf{hm}} := \mathcal{Q}_{\mathsf{hm}} \cup \{\mathsf{hm}\}$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{Bs}$
$\boxed{\mathbf{t} \xleftarrow{\$} \mathbb{Z}_q^{3k}}$
$\mathbf{u} := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}[\![j]\!]}\mathbf{t} \in \mathbb{Z}_q^k$
$\boxed{\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^k}$
return $\tau := \left([\mathbf{t}]_2, [\mathbf{u}]_2\right)$

$\underline{\textsc{Finalize}_{\mathsf{ri}}(\beta \in \{0,1\}):}$
return $\left(\mathcal{C}_{\mathsf{hm}} \cap \mathcal{Q}_{\mathsf{hm}} \overset{?}{=} \emptyset\right) \wedge \beta$
```

Fig. 2. Games $\mathsf{RI}_{\mathsf{real}}$ and $\boxed{\mathsf{RI}_{\mathsf{rand}}}$ that define the security of $\mathsf{MAC}_{\mathsf{lin}}$. The function $\mathsf{RF} : \{0,1\}^\gamma \to \mathbb{Z}_q^{k \times 2k}$ is a random function, defined on-the-fly.

verification tokens are also randomized over $\mathsf{Span}\left(\mathbf{B}^\perp\right)$ when the tags are random. The formal security game can be found in Figure 2. Interestingly, Lemma 4 can be used to prove the security of LP HIBEs in [30] in a black-box manner. Essentially, Lemma 4 has a similar purpose as the core lemma in [15], namely, to inject randomness.

Lemma 4 (Randomness Injection Lemma). *For all adversaries $\mathcal{A}$ there exist adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ with*

$$\left|\Pr\left[\mathsf{RI}_{\mathsf{real}}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{RI}_{\mathsf{rand}}^{\mathcal{A}} \Rightarrow 1\right]\right| \leq (8k\gamma + 2k)\mathsf{Adv}_{\mathcal{U}_k,\mathsf{PGGen},2}^{\mathsf{mddh}}(\mathcal{B}_1) + k\gamma\mathsf{Adv}_{\mathcal{U}_k,\mathsf{PGGen},1}^{\mathsf{mddh}}(\mathcal{B}_2) + \frac{\gamma Q_c + 6\gamma + 1}{q-1} + \frac{Q_e}{q^{2k}}$$

and $T(\mathcal{B}_1) \approx T(\mathcal{B}_2) \approx T(\mathcal{A}) + (Q_e + Q_c) \cdot \mathsf{poly}(\lambda)$, where Q_e resp. Q_c denotes the number of $\textsc{Eval}_{\mathsf{ri}}$ *resp.* $\textsc{Chal}_{\mathsf{ri}}$ *queries of $\mathcal{A}$ and* poly *is a polynomial independent of $\mathcal{A}$.* $\mathsf{RI}_{\mathsf{real}}$ *and* $\mathsf{RI}_{\mathsf{rand}}$ *are defined as in Figure 2.*

We give the overall hybrids used to prove this Lemma in Figure 3. The proof can be found in the full version.

Randomness Packing Lemma. We will use a tight variant of the Lewko-Waters approach [33] to tie these local, linear tags together and move entropy from the local to the global part. Lemma 5 captures this approach.

```
$\mathsf{G}_0$ $\boxed{\mathsf{G}_1 \ulcorner\mathsf{G}_{2,\hat{j}}\lrcorner \fbox{$\mathsf{G}_3$}}$

$\underline{\text{Init}_{\mathsf{ri}}\text{:}}$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
for $j \in \{1,\ldots,\gamma\},\ b \in \{0,1\}$ do
    $\mathbf{X}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{k\times 3k}$
$\mathsf{dk}_1 := \big([\mathbf{X}_{j,b}\mathbf{B}]_2\big)_{1\le j\le\gamma, b\in\{0,1\}}$
return $\big([\mathbf{B}]_2, \mathsf{dk}_1\big)$

$\underline{\text{Chal}_{\mathsf{ri}}(\mathsf{hm}^\star \in \{0,1\}^\gamma)\text{:}}$
$\mathcal{C}_{\mathsf{hm}} := \mathcal{C}_{\mathsf{hm}} \cup \{\mathsf{hm}^\star\}$
$\mathbf{h} \xleftarrow{\$} \mathbb{Z}_q^k$
$\mathbf{h}_0 := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}^\star[\![j]\!]}^\top \mathbf{h}$
$\ulcorner \mathbf{h}_0 := \mathbf{h}_0 + \mathbf{B}^\perp \mathsf{RF}_{\hat{j}}\big(\mathsf{hm}^\star_{|\hat{j}}\big)^\top \mathbf{h} \lrcorner$
$\fbox{$\mathbf{h}_0 := \mathbf{h}_0 + \mathbf{B}^\perp \mathsf{RF}(\mathsf{hm}^\star)^\top \mathbf{h}$}$
return $\big([\mathbf{h}]_1, [\mathbf{h}_0]_1\big)$

$\underline{\text{Eval}_{\mathsf{ri}}(\mathsf{hm} \in \{0,1\}^\gamma)\text{:}}$
if $\mathsf{hm} \in \mathcal{Q}_{\mathsf{hm}}$ then return $\perp$
$\mathcal{Q}_{\mathsf{hm}} := \mathcal{Q}_{\mathsf{hm}} \cup \{\mathsf{hm}\}$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t} := \mathbf{B}\mathbf{s}$
$\boxed{\mathbf{t} \xleftarrow{\$} \mathbb{Z}_q^{3k}}$
$\mathbf{u} := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}[\![j]\!]} \mathbf{t}$
$\ulcorner \mathbf{u} := \mathbf{u} + \mathsf{RF}_{\hat{j}}\big(\mathsf{hm}_{|\hat{j}}\big)\big(\mathbf{B}^\perp\big)^\top \mathbf{t} \lrcorner$
$\fbox{$\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^k$}$
return $\tau := \big([\mathbf{t}]_2, [\mathbf{u}]_2\big)$

$\underline{\text{Finalize}_{\mathsf{ri}}(\beta \in \{0,1\})\text{:}}$
return $\Big(\mathcal{C}_{\mathsf{hm}} \cap \mathcal{Q}_{\mathsf{hm}} \overset{?}{=} \emptyset\Big) \wedge \beta$
```

Fig. 3. Hybrids for the security proof of Lemma 4.

Lemma 5 (Randomness Packing Lemma). *For all adversaries $\mathcal{A}$ there exist adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ with*

$$\Big|\Pr\Big[\mathsf{RP}_{\mathsf{real}}^{\mathcal{A}} \Rightarrow 1\Big] - \Pr\Big[\mathsf{RP}_{\mathsf{rand}}^{\mathcal{A}} \Rightarrow 1\Big]\Big| \le 2k\mathsf{Adv}_{\mathcal{U}_k,\mathsf{PGGen},2}^{\mathsf{mddh}}(\mathcal{B}_1) + k\mathsf{Adv}_{\mathcal{U}_k,\mathsf{PGGen},1}^{\mathsf{mddh}}(\mathcal{B}_2) + \frac{6}{q-1}$$

and $T(\mathcal{B}_1) \approx T(\mathcal{B}_2) \approx T(\mathcal{A}) + (Q_e + Q_c)\cdot \mathsf{poly}(\lambda)$, where Q_e resp. Q_c denotes the number of Eval$_{\mathsf{rp}}$ *resp.* Chal$_{\mathsf{rp}}$ *queries of $\mathcal{A}$ and* poly *is a polynomial independent of $\mathcal{A}$.* $\mathsf{RP}_{\mathsf{real}}$ *and* $\mathsf{RP}_{\mathsf{rand}}$ *are defined as in Figure 5.*

Proof. The proof uses a hybrid argument with hybrids G_0 (the $\mathsf{RP}_{\mathsf{real}}$ game), G_1, G_2, and G_3 (the $\mathsf{RP}_{\mathsf{rand}}$ game). The hybrids are given in Figure 6. A summary can be found in Table 2.

Lemma 6 ($\mathsf{G}_0 \rightsquigarrow \mathsf{G}_1$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\big|\Pr\big[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1\big] - \Pr\big[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1\big]\big| \le k\mathsf{Adv}_{\mathcal{U}_k,\mathsf{PGGen},2}^{\mathsf{mddh}}(\mathcal{B}) + \frac{1}{q-1}$$

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

Proof. The only difference between these two games is, that the Eval queries pick the vectors $\tilde{\mathbf{t}}$ uniformly random from $\mathbb{Z}_q^{2k}$ instead of only from $\mathsf{Span}(\tilde{\mathbf{B}})$.

```
$\underline{\mathsf{Gen}_{\mathsf{MAC}}(1^\lambda):}$
$\mathcal{PG} \overset{\$}{\leftarrow} \mathsf{PGGen}(1^\lambda)$
$\textbf{parse } \mathcal{PG} =: (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, P_1, P_2, e)$
$H \overset{\$}{\leftarrow} \mathcal{H}(1^\lambda)$; $\mathbf{B} \overset{\$}{\leftarrow} \mathcal{U}_{3k,k}$; $\tilde{\mathbf{B}} \overset{\$}{\leftarrow} \mathcal{U}_{2k,k}$
$\textbf{for } j \in \{1, \dots, \gamma\},\ b \in \{0,1\} \textbf{ do}$
    $\mathbf{X}_{j,b} \overset{\$}{\leftarrow} \mathbb{Z}_q^{k \times 3k}$
$\tilde{\mathbf{X}}_1 \overset{\$}{\leftarrow} \mathbb{Z}_q^{k \times 2k}$; $\tilde{\mathbf{X}}_2 \overset{\$}{\leftarrow} \mathbb{Z}_q^{k \times 2k}$; $\mathbf{x}' \overset{\$}{\leftarrow} \mathbb{Z}_q^{k}$
$\textbf{return } \mathsf{sk}_{\mathsf{MAC}}$

$\underline{\mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} = (\mathsf{m}_1, \dots, \mathsf{m}_p) \in \mathcal{S}^*):}$
$\textbf{for } i \in \{1, \dots, p\} \textbf{ do}$
    $\mathbf{s}_i \overset{\$}{\leftarrow} \mathbb{Z}_q^k$; $\mathbf{t}_i := \mathbf{B}\mathbf{s}_i$
    $\tilde{\mathbf{s}}_i \overset{\$}{\leftarrow} \mathbb{Z}_q^k$; $\tilde{\mathbf{t}}_i := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i$
    $\mathsf{hm}_i := H(\mathsf{m}_1, \dots, \mathsf{m}_i)$
    $\mathbf{u}_i := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]} \mathbf{t}_i + \tilde{\mathbf{X}}_1 \tilde{\mathbf{t}}_i$
$\tilde{\mathbf{u}} := \sum_{i=1}^{p} \tilde{\mathbf{X}}_2 \tilde{\mathbf{t}}_i + \mathbf{x}'$
$\textbf{return } \left( \left( [\mathbf{t}_i]_2, [\tilde{\mathbf{t}}_i]_2, [\mathbf{u}_i]_2 \right)_{1 \le i \le p}, [\tilde{\mathbf{u}}]_2 \right)$

$\underline{\mathsf{Ver}_{\mathsf{MAC}}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m} = (\mathsf{m}_1, \dots, \mathsf{m}_p) \in \mathcal{S}^*, \tau):}$
$\tau =: \left( \left( [\mathbf{t}_i]_2, [\tilde{\mathbf{t}}_i]_2, [\mathbf{u}_i]_2 \right)_{1 \le i \le p}, [\tilde{\mathbf{u}}]_2 \right)$
$\tilde{\mathbf{h}} \overset{\$}{\leftarrow} \mathbb{Z}_q^k$
$\textbf{for } i \in \{1, \dots, p\} \textbf{ do}$
    $\mathbf{h}_i \overset{\$}{\leftarrow} \mathbb{Z}_q^k$
    $\mathsf{hm}_i := H(\mathsf{m}_1, \dots, \mathsf{m}_i)$
    $\mathbf{h}_{0,i} := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]}^\top \mathbf{h}_i$
    $\tilde{\mathbf{h}}_{0,i} := \tilde{\mathbf{X}}_1^\top \mathbf{h}_i + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}}$
$h_K := (\mathbf{x}')^\top \tilde{\mathbf{h}}$
$\textbf{return } \sum_{i=1}^{p} \Big( e\big( [\mathbf{h}_i^\top]_1, [\mathbf{u}_i]_2 \big)$
    $- e\big( [\mathbf{h}_{0,i}^\top]_1, [\mathbf{t}_i]_2 \big) - e\big( [\tilde{\mathbf{h}}_{0,i}^\top]_1, [\tilde{\mathbf{t}}_i]_2 \big) \Big)$
        $+ e\big( [\tilde{\mathbf{h}}]_1, [\tilde{\mathbf{u}}]_2 \big) \overset{?}{=} [h_K]_T$
```

Fig. 4. Our unbounded affine MAC MAC_u. It uses a CRHF $\mathcal{H}$ with domain $\mathcal{S}^*$ and range $\{0,1\}^\gamma$. Throughout the scheme, $\mathsf{sk}_{\mathsf{MAC}} := \big(H, \mathbf{B}, \tilde{\mathbf{B}}, (\mathbf{X}_{j,b})_{1 \le j \le \gamma, b \in \{0,1\}}, \tilde{\mathbf{X}}_1, \tilde{\mathbf{X}}_2, \mathbf{x}'\big)$ with values generated in $\mathsf{Gen}_{\mathsf{MAC}}$. The linear MAC components are highlighted in gray.

This leads to a straightforward reduction to the Q_e-fold $\mathcal{U}_{2k,k}$-MDDH assumption on $\tilde{\mathbf{B}}$. □

Lemma 7 ($\mathsf{G}_1 \rightsquigarrow \mathsf{G}_2$). *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

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

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

Proof. In game G_2 the $\tilde{\mathbf{B}}^\perp$-part of $\tilde{\mathbf{h}}_0$ (for all $i \in \{1, \dots, p\}$) is uniformly random. To switch to this game, pick a Q_c-fold $\mathcal{U}_{2k,k}$-MDDH challenge and use the reduction in Figure 7.

Assume that $\overline{\mathbf{D}}$ is invertible. This happens with probability at least $(1 - 1/(q-1))$. The Init, Eval, and Finalize oracles are identical in both games. The reduction correctly simulates Init because the summand $\overline{\mathbf{D}}^{-\top} \underline{\mathbf{D}}^\top (\tilde{\mathbf{B}}^\perp)^\top$ cancels out in public key.

To analyze the Chal queries define $\mathbf{f}_c =: \begin{pmatrix} \overline{\mathbf{D}}\mathbf{w}_c \\ \underline{\mathbf{D}}\mathbf{w}_c + \mathbf{r}_c \end{pmatrix}$ where $\mathbf{w}_c$ is uniform random in $\mathbb{Z}_q^k$ and $\mathbf{r}_c$ is $\mathbf{0} \in \mathbb{Z}_q^k$ or uniform random in $\mathbb{Z}_q^k$. The reduction defines $\mathbf{h} := \overline{\mathbf{f}_c}$, which is a uniform random vector.

```
$\underline{\textsc{Init}_{\mathsf{rp}}:}$
$\tilde{\mathbf{B}} \xleftarrow{\$} \mathcal{U}_{2k,k}$
$\tilde{\mathbf{X}}_1 \xleftarrow{\$} \mathbb{Z}_q^{k\times 2k};\ \tilde{\mathbf{X}}_2 \xleftarrow{\$} \mathbb{Z}_q^{k\times 2k}$
$\mathsf{dk}_2 := \left(\left[\tilde{\mathbf{X}}_1\tilde{\mathbf{B}}\right]_2, \left[\tilde{\mathbf{X}}_2\tilde{\mathbf{B}}\right]_2\right)$
$\mathbf{return}\ \left(\left[\tilde{\mathbf{B}}\right]_2, \mathsf{dk}_2\right)$

$\underline{\textsc{Chal}_{\mathsf{rp}}\left(\tilde{\mathbf{h}} \in \mathbb{Z}_q^k\right):}$
$\mathbf{h} \xleftarrow{\$} \mathbb{Z}_q^k$
$\tilde{\mathbf{h}}_0 := \tilde{\mathbf{X}}_1^\top \mathbf{h} + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}}$
$\boxed{\mathbf{r} \xleftarrow{\$} \mathbb{Z}_q^k;\ \tilde{\mathbf{h}}_0 := \tilde{\mathbf{h}}_0 + \tilde{\mathbf{B}}^\perp \mathbf{r}}$
$\mathbf{return}\ \left([\mathbf{h}]_1, \left[\tilde{\mathbf{h}}_0\right]_1\right)$
```

```
$\underline{\textsc{Eval}_{\mathsf{rp}}:}$
$\tilde{\mathbf{s}} \xleftarrow{\$} \mathbb{Z}_q^k;\ \tilde{\mathbf{t}} := \tilde{\mathbf{B}}\tilde{\mathbf{s}}$
$\boxed{\tilde{\mathbf{t}} \xleftarrow{\$} \mathbb{Z}_q^{2k}}$
$\tilde{\mathbf{u}} := \tilde{\mathbf{X}}_2\tilde{\mathbf{t}}$
$\boxed{\tilde{\mathbf{u}} \xleftarrow{\$} \mathbb{Z}_q^k}$
$\mathbf{return}\ \left(\left[\tilde{\mathbf{t}}\right]_2, [\tilde{\mathbf{u}}]_2\right)$

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

Fig. 5. Games $\mathsf{RP}_{\mathsf{real}}$ and $\boxed{\mathsf{RP}_{\mathsf{rand}}}$ for Lemma 5.

Hybrid	$\tilde{\mathbf{t}}$ drawn from	$r_{\tilde{\mathbf{u}}}$	$r_{\tilde{\mathbf{h}}_0}$	Transition
G_0	$\mathsf{Span}(\tilde{\mathbf{B}})$	$\{0\}$	$\{0\}$	—
G_1	$\mathbb{Z}_q^{2k}$	$\{0\}$	$\{0\}$	$\mathcal{U}_k$-MDDH in $\mathbb{G}_2$
G_2	$\mathbb{Z}_q^{2k}$	$\{0\}$	$\mathsf{Span}(\tilde{\mathbf{B}}^\perp)$	$\mathcal{U}_k$-MDDH in $\mathbb{G}_1$
G_3	$\mathbb{Z}_q^{2k}$	$\mathbb{Z}_q^k$	$\mathsf{Span}(\tilde{\mathbf{B}}^\perp)$	$\mathcal{U}_k$-MDDH in $\mathbb{G}_2$

Table 2. Summary of the hybrids in Figure 6. $\textsc{Eval}_{\mathsf{rp}}$ queries draw $\tilde{\mathbf{t}}$ from the set described by the second column and add a uniform random element from the set $r_{\tilde{\mathbf{u}}}$ to $\tilde{\mathbf{u}}$. The $\textsc{Chal}_{\mathsf{rp}}$ queries add a uniform random element from $r_{\tilde{\mathbf{h}}_0}$ to each $\tilde{\mathbf{h}}_0$. The background color indicates repeated transitions.

The vector $\tilde{\mathbf{h}}_0$ is then computed as

$$\begin{aligned}\tilde{\mathbf{h}}_0 &:= \tilde{\mathbf{J}}_1^\top \mathbf{h} + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}} + \tilde{\mathbf{B}}^\perp \underline{\mathbf{f}_c} \\ &= \tilde{\mathbf{J}}_1^\top \mathbf{h} + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}} + \tilde{\mathbf{B}}^\perp \underline{\mathbf{D}}\overline{\mathbf{D}}^{-1}\mathbf{h} + \tilde{\mathbf{B}}^\perp \mathbf{r}_c \\ &= \tilde{\mathbf{X}}_1^\top \mathbf{h} + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}} + \tilde{\mathbf{B}}^\perp \mathbf{r}_c\end{aligned}$$

If $\mathbf{r}_c = \mathbf{0}$, the reduction is simulating game G_1 and if $\mathbf{r}_c$ is uniform, the reduction is simulating G_2. □

Lemma 8 ($\mathsf{G}_2 \rightsquigarrow \mathsf{G}_3$)**.** *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1\right]\right| \le k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}) + \frac{3}{q-1}$$

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

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

INIT_rp:
    $\tilde{\mathbf{B}} \xleftarrow{\$} \mathcal{U}_{2k,k}$
    $\tilde{\mathbf{X}}_1 \xleftarrow{\$} \mathbb{Z}_q^{k\times 2k};\ \tilde{\mathbf{X}}_2 \xleftarrow{\$} \mathbb{Z}_q^{k\times 2k}$
    $\mathsf{dk}_2 := \left(\left[\tilde{\mathbf{X}}_1\tilde{\mathbf{B}}\right]_2, \left[\tilde{\mathbf{X}}_2\tilde{\mathbf{B}}\right]_2\right)$
    return $\left(\left[\tilde{\mathbf{B}}\right]_2, \mathsf{dk}_2\right)$

CHAL_rp$\left(\tilde{\mathbf{h}} \in \mathbb{Z}_q^k\right)$:
    $\mathbf{h} \xleftarrow{\$} \mathbb{Z}_q^k$
    $\tilde{\mathbf{h}}_0 := \tilde{\mathbf{X}}_1^\top \mathbf{h} + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}}$
    $\boxed{\mathbf{r} \xleftarrow{\$} \mathbb{Z}_q^k;\ \tilde{\mathbf{h}}_0 := \tilde{\mathbf{h}}_0 + \tilde{\mathbf{B}}^\perp \mathbf{r}}$
    return $\left([\mathbf{h}]_1, \left[\tilde{\mathbf{h}}_0\right]_1\right)$

EVAL_rp:
    $\tilde{\mathbf{s}} \xleftarrow{\$} \mathbb{Z}_q^k;\ \tilde{\mathbf{t}} := \tilde{\mathbf{B}}\tilde{\mathbf{s}}$
    $\boxed{\tilde{\mathbf{t}} \xleftarrow{\$} \mathbb{Z}_q^{2k}}$
    $\tilde{\mathbf{u}} := \tilde{\mathbf{X}}_2\tilde{\mathbf{t}}$
    $\boxed{\tilde{\mathbf{u}} \xleftarrow{\$} \mathbb{Z}_q^k}$
    return $\left(\left[\tilde{\mathbf{t}}\right]_2, [\tilde{\mathbf{u}}]_2\right)$

FINALIZE_rp$(\beta \in \{0,1\})$:
    return $\beta$
```

Fig. 6. Hybrids for the security proof of Lemma 5.

```
INIT_rp:
    $\tilde{\mathbf{B}} \xleftarrow{\$} \mathcal{U}_{2k,k}$
    $\tilde{\mathbf{J}}_1 \xleftarrow{\$} \mathbb{Z}_q^{k\times 2k};\ \tilde{\mathbf{X}}_2 \xleftarrow{\$} \mathbb{Z}_q^{k\times 2k}$
    // Implicit: $\tilde{\mathbf{X}}_1 := \tilde{\mathbf{J}}_1 + \overline{\mathbf{D}}^{-\top}\underline{\mathbf{D}}^\top\left(\tilde{\mathbf{B}}^\perp\right)^\top$
    $\mathsf{dk}_2 := \left(\left[\tilde{\mathbf{J}}_1\tilde{\mathbf{B}}\right]_2, \left[\tilde{\mathbf{X}}_2\tilde{\mathbf{B}}\right]_2\right)$
    return $\left(\left[\tilde{\mathbf{B}}\right]_2, \mathsf{dk}_2\right)$

FINALIZE_rp$(\beta \in \{0,1\})$:
    return $\beta$

EVAL_rp:
    $\tilde{\mathbf{t}} \xleftarrow{\$} \mathbb{Z}_q^{2k}$
    $\tilde{\mathbf{u}} := \tilde{\mathbf{X}}_2\tilde{\mathbf{t}}$
    return $\left(\left[\tilde{\mathbf{t}}\right]_2, [\tilde{\mathbf{u}}]_2\right)$

CHAL_rp$\left(\tilde{\mathbf{h}} \in \mathbb{Z}_q^k\right)$:
    Let this be the $c$-th CHAL query.
    $\mathbf{h} := \overline{\mathbf{f}_c}$
    $\tilde{\mathbf{h}}_0 := \tilde{\mathbf{J}}_1^\top \mathbf{h} + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}} + \tilde{\mathbf{B}}^\perp \underline{\mathbf{f}_c}$
    return $\left([\mathbf{h}]_1, \left[\tilde{\mathbf{h}}_0\right]_1\right)$
```

Fig. 7. Reduction for the transition from G_1 to G_2 to the Q_c-fold $\mathcal{U}_{2k,k}$-MDDH challenge $\left([\mathbf{D}]_1, [\mathbf{f}_1]_1, \ldots, [\mathbf{f}_{Q_c}]_1\right)$.

```
$\underline{\textsc{Init}_{\mathsf{rp}}:}$
$\tilde{\mathbf{B}} \overset{\$}{\leftarrow} \mathcal{U}_{2k,k}$
if $\mathrm{rank}\big(\underline{\tilde{\mathbf{B}}}\big) \neq k \vee \mathrm{rank}\big(\overline{\tilde{\mathbf{B}}}\big) \neq k$ then
    abort
$\tilde{\mathbf{B}}^{\perp} := \begin{pmatrix} \overline{\tilde{\mathbf{B}}}^{-\top} \\ -\underline{\tilde{\mathbf{B}}}^{-\top} \end{pmatrix};\ \tilde{\mathbf{B}}' := \frac{1}{2}\begin{pmatrix} \overline{\tilde{\mathbf{B}}} \\ -\underline{\tilde{\mathbf{B}}} \end{pmatrix}$
$\tilde{\mathbf{X}}_1 \overset{\$}{\leftarrow} \mathbb{Z}_q^{k \times 2k};\ \tilde{\mathbf{J}}_2 \overset{\$}{\leftarrow} \mathbb{Z}_q^{k \times 2k}$
// Implicit: $\tilde{\mathbf{X}}_2 := \tilde{\mathbf{J}}_2 + \underline{\mathbf{D}}\overline{\mathbf{D}}^{-1}\big(\tilde{\mathbf{B}}^{\perp}\big)^{\top}$
$\mathsf{dk}_2 := \big(\big[\tilde{\mathbf{X}}_1\tilde{\mathbf{B}}\big]_2, \big[\tilde{\mathbf{J}}_2\tilde{\mathbf{B}}\big]_2\big)$
return $\big(\big[\tilde{\mathbf{B}}\big]_2, \mathsf{dk}_2\big)$
```

```
$\underline{\textsc{Eval}_{\mathsf{rp}}:}$
Let this be the $c$-th Eval query.
$\tilde{\mathbf{s}} \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \tilde{\mathbf{t}} := \tilde{\mathbf{B}}\tilde{\mathbf{s}} + \tilde{\mathbf{B}}'\overline{\mathbf{f}_c}$
$\tilde{\mathbf{u}} := \sum_{i=1}^{p} \tilde{\mathbf{J}}_2\tilde{\mathbf{t}} + \underline{\mathbf{f}_c}$
return $\big(\big[\tilde{\mathbf{t}}\big]_2, [\tilde{\mathbf{u}}]_2\big)$

$\underline{\textsc{Chal}_{\mathsf{rp}}\big(\tilde{\mathbf{h}} \in \mathbb{Z}_q^k\big):}$
$\mathbf{h} \overset{\$}{\leftarrow} \mathbb{Z}_q^k$
$\mathbf{r} \overset{\$}{\leftarrow} \mathbb{Z}_q^k;\ \tilde{\mathbf{h}}_0 := \tilde{\mathbf{X}}_1^{\top}\mathbf{h} + \tilde{\mathbf{J}}_2^{\top}\tilde{\mathbf{h}} + \tilde{\mathbf{B}}^{\perp}\mathbf{r}$
return $\big([\mathbf{h}]_1, \big[\tilde{\mathbf{h}}_0\big]_1\big)$

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

Fig. 8. Reduction for the transition from G_2 to G_3 to the Q_e-fold $\mathcal{U}_{2k,k}$-MDDH challenge $\big([\mathbf{D}]_2, [\mathbf{f}_1]_2, \ldots, [\mathbf{f}_{Q_e}]_2\big)$.

Proof. In game G_3 the vector $\tilde{\mathbf{u}}$ is chosen uniformly random. For the transition to this game, we need a Q_e-fold $\mathcal{U}_{2k,k}$-MDDH challenge. The reduction is given in Figure 8.

The reduction aborts if the upper or lower $k \times k$-submatrix of $\tilde{\mathbf{B}}$ does not have full rank. This happens only with probability at most $2/(q-1)$. Assume in the following, that the reduction does not abort. Furthermore assume $q > 2$.

The way we defined $\tilde{\mathbf{B}}^{\perp}$ and $\tilde{\mathbf{B}}'$ we get the following three properties:

$$\big(\tilde{\mathbf{B}}^{\perp}\big)^{\top}\tilde{\mathbf{B}} = \overline{\tilde{\mathbf{B}}}^{-1}\overline{\tilde{\mathbf{B}}} - \underline{\tilde{\mathbf{B}}}^{-1}\underline{\tilde{\mathbf{B}}} = \mathbf{I}_k - \mathbf{I}_k = \mathbf{0} \tag{3}$$

$$\big(\tilde{\mathbf{B}}^{\perp}\big)^{\top}\tilde{\mathbf{B}}' = \frac{1}{2}\left(\overline{\tilde{\mathbf{B}}}^{-1}\overline{\tilde{\mathbf{B}}} + \underline{\tilde{\mathbf{B}}}^{-1}\underline{\tilde{\mathbf{B}}}\right) = \frac{1}{2}(\mathbf{I}_k + \mathbf{I}_k) = \mathbf{I}_k \tag{4}$$

$$\tilde{\mathbf{B}}, \tilde{\mathbf{B}}' \text{ is a basis of } \mathbb{Z}_q^{2k} \tag{5}$$

To see Equation (5), note that this is equivalent to the column vectors $\mathbf{b}_1, \ldots, \mathbf{b}_{2k}$ of

$$\big(\tilde{\mathbf{B}}|2\tilde{\mathbf{B}}'\big) = \begin{pmatrix} \overline{\tilde{\mathbf{B}}} & \overline{\tilde{\mathbf{B}}} \\ \underline{\tilde{\mathbf{B}}} & -\underline{\tilde{\mathbf{B}}} \end{pmatrix}$$

being linear independent. Assume there exist $\mu_1, \ldots, \mu_{2k} \in \mathbb{Z}_q$ with

$$\mu_1\mathbf{b}_1 + \cdots + \mu_{2k}\mathbf{b}_{2k} = \mathbf{0}\,.$$

Looking at the first k entries in each vector and using that $\overline{\tilde{\mathbf{B}}}$ has full rank we get

$$\mu_1 = -\mu_{k+1}, \ldots, \mu_k = -\mu_{2k}\,.$$

Now looking at the remaining lower k entries and using that the column vectors of $\underline{\tilde{\mathbf{B}}}$ can not be $\mathbf{0}$ (because we already assumed that $\underline{\tilde{\mathbf{B}}}$ has full rank) we get that

$$\mu_1 = 0, \ldots, \mu_{2k} = 0\,.$$

The Init oracle is identically distributed in both games and correctly simulated by the reduction, because the $\underline{\mathbf{D}}\overline{\mathbf{D}}^{-1}\left(\tilde{\mathbf{B}}^{\perp}\right)^{\top}$ cancels out in the public key.

The Chal oracle is also distributed identically in both games and simulated correctly since the $\tilde{\mathbf{B}}^{\perp}$-part of $\tilde{\mathbf{h}}_0$ is uniform random. More precisely, $\mathbf{r}$ is identically distributed to $\mathbf{r} + \overline{\mathbf{D}}^{-\top}\underline{\mathbf{D}}^{\top}\tilde{\mathbf{h}}$. Thus $\tilde{\mathbf{h}}_0$ as computed by the reduction:

$$\tilde{\mathbf{h}}_0 := \tilde{\mathbf{X}}_1^{\top}\mathbf{h} + \tilde{\mathbf{J}}_2^{\top}\tilde{\mathbf{h}} + \tilde{\mathbf{B}}^{\perp}\mathbf{r}$$

is identically distributed to

$$\begin{aligned} &\tilde{\mathbf{X}}_1^{\top}\mathbf{h} + \tilde{\mathbf{J}}_2^{\top}\tilde{\mathbf{h}} + \tilde{\mathbf{B}}^{\perp}\left(\mathbf{r} + \overline{\mathbf{D}}^{-\top}\underline{\mathbf{D}}^{\top}\tilde{\mathbf{h}}\right) \\ = &\tilde{\mathbf{X}}_1^{\top}\mathbf{h} + \tilde{\mathbf{X}}_2^{\top}\tilde{\mathbf{h}} + \tilde{\mathbf{B}}^{\perp}\mathbf{r}\,, \end{aligned}$$

which is the real $\tilde{\mathbf{h}}_0$.

To analyze the Eval queries, define $\mathbf{f}_c =: \left(\begin{smallmatrix}\overline{\mathbf{D}}\mathbf{w}_c \\ \underline{\mathbf{D}}\mathbf{w}_c + \mathbf{r}_c\end{smallmatrix}\right)$ where $\mathbf{w}_c$ is uniform random in $\mathbb{Z}_q^k$ and $\mathbf{r}_c$ is $\mathbf{0} \in \mathbb{Z}_q^k$ or uniform random in $\mathbb{Z}_q^k$. In the Eval queries the reduction computes $\tilde{\mathbf{t}}$ as $\tilde{\mathbf{t}} := \tilde{\mathbf{B}}\tilde{\mathbf{s}} + \tilde{\mathbf{B}}'\overline{\mathbf{f}_c}$, but this is distributed identically to a uniform random vector, because $\tilde{\mathbf{s}}$ and $\overline{\mathbf{f}_c}$ are uniform random and $\tilde{\mathbf{B}}, \tilde{\mathbf{B}}'$ are a basis of $\mathbb{Z}_q^{2k}$ (see Equation (5)).

The vector $\tilde{\mathbf{u}}$ is computed as

$$\begin{aligned} \tilde{\mathbf{u}} &:= \sum\nolimits_{i=1}^{p} \tilde{\mathbf{J}}_2\tilde{\mathbf{t}} + \underline{\mathbf{f}_c} \\ &= \sum\nolimits_{i=1}^{p} \tilde{\mathbf{J}}_2\tilde{\mathbf{t}} + \underline{\mathbf{D}}\overline{\mathbf{D}}^{-1}\overline{\mathbf{f}_c} + \mathbf{r}_c \\ &= \sum\nolimits_{i=1}^{p} \tilde{\mathbf{J}}_2\tilde{\mathbf{t}} + \underline{\mathbf{D}}\overline{\mathbf{D}}^{-1} \underbrace{\left(\tilde{\mathbf{B}}^{\perp}\right)^{\top}\tilde{\mathbf{B}}'}_{=\mathbf{I}_k \text{ (Eq. (4))}} \overline{\mathbf{f}_c} + \mathbf{r}_c \\ &\overset{\text{Eq. (3)}}{=} \sum\nolimits_{i=1}^{p} \tilde{\mathbf{J}}_2\tilde{\mathbf{t}} + \underline{\mathbf{D}}\overline{\mathbf{D}}^{-1}\left(\tilde{\mathbf{B}}^{\perp}\right)^{\top} \underbrace{\left(\tilde{\mathbf{B}}\tilde{\mathbf{s}} + \tilde{\mathbf{B}}'\overline{\mathbf{f}_c}\right)}_{=\tilde{\mathbf{t}}} + \mathbf{r}_c \\ &= \sum\nolimits_{i=1}^{p} \tilde{\mathbf{X}}_2\tilde{\mathbf{t}} + \mathbf{r}_c\,. \end{aligned}$$

If $\mathbf{r}_c = \mathbf{0}$, the reduction is simulating game G_2 and if $\mathbf{r}_c$ is uniform, the reduction is simulating G_3. □

Summary. To prove Lemma 5, we combine Lemmata 6–8. □

3.2 An Unbounded Affine MAC

Our next step is to construct an unbounded affine MAC as in Figure 4. Again, our idea is to divide a hierarchical message $(\mathsf{m}_1, \ldots, \mathsf{m}_p)$ into p pieces $\mathsf{hm}_i := H(\mathsf{m}_1 || \ldots || \mathsf{m}_i)$ $(1 \leq i \leq p)$ by using a CRHF H. In stark contrast to methods in [29,30], we generate a MAC tag for each hm_i with the same secret key. More precisely, we apply $\mathsf{MAC}_{\mathsf{lin}}$ on each hm_i, and additionally we have a wrapper, namely, $\tilde{\mathbf{X}}_1 \cdot \tilde{\mathbf{t}}_i$ to connect all these p pieces together.

```
$\underline{\textsc{Init}_{\mathsf{MAC}}:}$
$\mathsf{sk}_{\mathsf{MAC}} \xleftarrow{\$} \mathsf{Gen}_{\mathsf{MAC}}(1^\lambda)$
$\textbf{parse } \mathsf{sk}_{\mathsf{MAC}} =: \big(H, \mathbf{B}, \tilde{\mathbf{B}}, (\mathbf{X}_{j,b})_{1 \le j \le \gamma, b \in \{0,1\}},$
$\tilde{\mathbf{X}}_1, \tilde{\mathbf{X}}_2, \mathbf{x}'\big)$
$\mathsf{dk}_1 := \big([\mathbf{X}_{j,b}\mathbf{B}]_2\big)_{1 \le j \le \gamma, b \in \{0,1\}}$
$\mathsf{dk}_2 := \big([\tilde{\mathbf{X}}_1\tilde{\mathbf{B}}]_2, [\tilde{\mathbf{X}}_2\tilde{\mathbf{B}}]_2\big)$
$\textbf{return } \big(\mathcal{PG}, H, [\mathbf{B}]_2, [\tilde{\mathbf{B}}]_2, \mathsf{dk}_1, \mathsf{dk}_2\big)$

$\underline{\textsc{Eval}(\mathsf{m} = (\mathsf{m}_1, \dots, \mathsf{m}_p) \in \mathcal{S}^*):}$
$\mathcal{Q}_{\mathcal{M}} := \mathcal{Q}_{\mathcal{M}} \cup \{\mathsf{m}\}$
$\textbf{return } \mathsf{Tag}(\mathsf{sk}_{\mathsf{MAC}}, \mathsf{m})$

$\underline{\textsc{Finalize}_{\mathsf{MAC}}(\beta \in \{0,1\}):}$
$\textbf{return } \Big(\bigcup_{\mathsf{m}^\star \in \mathcal{C}_{\mathcal{M}}} \mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} = \emptyset\Big) \wedge \beta$
```

```
$\underline{\textsc{Chal}\big(\mathsf{m}^\star = (\mathsf{m}_1^\star, \dots, \mathsf{m}_p^\star) \in \mathcal{S}^*\big):}$
$\mathcal{C}_{\mathcal{M}} := \mathcal{C}_{\mathcal{M}} \cup \{\mathsf{m}^\star\}$
$\tilde{\mathbf{h}} \xleftarrow{\$} \mathbb{Z}_q^\eta$
$\textbf{for } i \in \{1, \dots, p\} \textbf{ do}$
    $\mathbf{h}_i \xleftarrow{\$} \mathbb{Z}_q^\eta$
    $\mathsf{hm}_i^\star := H(\mathsf{m}_1^\star, \dots, \mathsf{m}_i^\star)$
    $\mathbf{h}_{0,i} := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i^\star[\![j]\!]}^\top \mathbf{h}_i$
    $\tilde{\mathbf{h}}_{0,i} := \tilde{\mathbf{X}}_1^\top \mathbf{h}_i + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}}$
$h_K := (\mathbf{x}')^\top \tilde{\mathbf{h}}$
$\boxed{h_K \xleftarrow{\$} \mathbb{Z}_q}$
$\mathbf{H} := \big([\mathbf{h}_i]_1, [\mathbf{h}_{0,i}]_1, [\tilde{\mathbf{h}}_{0,i}]_1\big)_{1 \le i \le p}$
$\textbf{return } \big([\tilde{\mathbf{h}}]_1, \mathbf{H}, [h_K]_T\big)$
```

Fig. 9. Games $\mathsf{uMAC}_{\mathsf{real}}$ and $\boxed{\mathsf{uMAC}_{\mathsf{rand}}}$ for defining security for MAC_u.

One can show MAC_u is a secure MAC according to the (standard) UF-CMA security (cf. the full version). Our MAC_u has stronger security which is formally stated in Theorem 1.[6] It is not a standard security for a MAC scheme, but it is exactly what we need for the transformation to unbounded HIBE. As in the security game for linear MACs, values in dk_1 and dk_2 can be used to rerandomize tags (cf. Remark 1). Oracle Eval is available to an adversary $\mathcal{A}$ for a tag on any message of her choice. Moreover, oracle Chal provides $\mathcal{A}$ necessary values to check validity of a tag. She can query these two oracles arbitrary many times in an adaptive manner. In the end, $\mathcal{A}$ needs to distinguish during the experiment Chal always gives her the real values or the random ones. Of course, we exclude the case where $\mathcal{A}$ trivially wins by asking Eval for any prefix of a challenge message $\mathsf{m}^\star$. The formal security game can be found in Figure 9.

Remark 1 (Delegation). The tags of MAC_u are delegatable in the following sense: Given a tag $\tau = \Big(\big([\mathbf{t}_i]_2, [\tilde{\mathbf{t}}_i]_2, [\mathbf{u}_i]_2\big)_{1 \le i \le p}, [\tilde{\mathbf{u}}]_2\Big)$ for a message $\mathsf{m} = (\mathsf{m}_1, \dots, \mathsf{m}_p)$, one can compute a fresh tag τ'' for a message $\mathsf{m}' := (\mathsf{m}_1, \dots, \mathsf{m}_p, \mathsf{m}_{p+1})$ for arbitrary $\mathsf{m}_{p+1} \in \mathcal{S}$ using only the "public key" returned from the $\textsc{Init}_{\mathsf{MAC}}$ oracle in the $\mathsf{uMAC}_{\mathsf{real}}$ game. We call the tag τ'' *fresh*, because its distribution is independent of τ.

First, we define the tag τ' for m' as $\tau' := \Big(\big([\mathbf{t}'_i]_2, [\tilde{\mathbf{t}}'_i]_2, [\mathbf{u}'_i]_2\big)_{1 \le i \le p+1}, [\tilde{\mathbf{u}}']_2\Big)$. This tag is identical to τ on the first p levels, i.e., for all $i \in \{1, \dots, p\}$ we define $\mathbf{t}'_i := \mathbf{t}_i$, $\tilde{\mathbf{t}}'_i := \tilde{\mathbf{t}}_i$ and $\mathbf{u}'_i := \mathbf{u}_i$. Furthermore we define $\mathbf{t}'_{p+1} := \mathbf{0}$, $\tilde{\mathbf{t}}'_{p+1} := \mathbf{0}$,

[6] Our security notion is stronger than UF-CMA since a forged tag could be used to distinguish the real from the random Chal queries.

$\mathbf{u}'_{p+1} = \mathbf{0}$ and $\tilde{\mathbf{u}}' := \tilde{\mathbf{u}}$. The resulting tag τ is indeed a valid tag for m', but it is not fresh.

To get a fresh tag $\tau'' := \Big(\big([\mathbf{t}''_i]_2, [\tilde{\mathbf{t}}''_i]_2, [\mathbf{u}''_i]_2\big)_{1 \le i \le p+1}, [\tilde{\mathbf{u}}'']_2\Big)$, we rerandomize the tag τ'. That is, for all $i \in \{1, \dots, p+1\}$ we define $\mathbf{t}''_i := \mathbf{t}'_i + \mathbf{B}\mathbf{s}'_i$ and $\tilde{\mathbf{t}}''_i := \tilde{\mathbf{t}}'_i + \mathbf{B}\tilde{\mathbf{s}}'_i$ for uniform random $\mathbf{s}'_i \xleftarrow{\$} \mathbb{Z}_q^{n'}$ and $\tilde{\mathbf{s}}_i \xleftarrow{\$} \mathbb{Z}_q^{\tilde{n}'}$. Moreover, we adapt $\mathbf{u}_i$ and $\tilde{\mathbf{u}}$ to the new $\mathbf{t}''_i$ and $\tilde{\mathbf{t}}''_i$ in the following way:

$$\mathbf{u}''_i := \mathbf{u}'_i + \sum\nolimits_{j=1}^{\gamma} \mathbf{X}_{j,b}\mathbf{B}\mathbf{s}'_i + \tilde{\mathbf{X}}_1\tilde{\mathbf{B}}\tilde{\mathbf{s}}'_i$$

$$\tilde{\mathbf{u}}'' := \tilde{\mathbf{u}}' + \sum\nolimits_{i=1}^{p} \tilde{\mathbf{X}}_2\tilde{\mathbf{B}}\tilde{\mathbf{s}}'_i$$

Theorem 1 (Security of MAC_u). MAC_u *is tightly secure under the* $\mathcal{U}_k$-MDDH *assumption for* $\mathbb{G}_1$*, the* $\mathcal{U}_k$-MDDH *assumption for* $\mathbb{G}_2$ *and the collision resistance of* $\mathcal{H}$*. More precisely, for all adversaries* $\mathcal{A}$ *there exist adversaries* $\mathcal{B}_1$*,* $\mathcal{B}_2$ *and* $\mathcal{B}_3$ *with*

$$\Big|\Pr\Big[\mathsf{uMAC}^{\mathcal{A}}_{\mathsf{real}} \Rightarrow 1\Big] - \Pr\Big[\mathsf{uMAC}^{\mathcal{A}}_{\mathsf{rand}} \Rightarrow 1\Big]\Big| \le (8k + 16k\gamma)\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}_1)$$

$$+ (1 + 2k(\gamma+1))\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},1}(\mathcal{B}_2) + 2\mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H}}(\mathcal{B}_3) + \frac{16 + (12 + 2Q_cL)\gamma}{q-1} + \frac{2Q_e}{q^{2k}}$$

and $T(\mathcal{B}_1) \approx T(\mathcal{B}_2) \approx T(\mathcal{B}_3) \approx T(\mathcal{A}) + (Q_e + Q_c)L \cdot \mathsf{poly}(\lambda)$*, where* Q_e *resp.* Q_c *denotes the number of* Eval *resp.* Chal *queries of* $\mathcal{A}$*,* L *denotes the maximum length of the messages for which the adversary queried a tag or a challenge, and* poly *is a polynomial independent of* $\mathcal{A}$*.*

Proof. The proof uses a hybrid argument with hybrids G_0–G_5, where G_0 is the $\mathsf{uMAC}_{\mathsf{real}}$ game. The hybrids are given in Figure 10. They make use of the random function $\mathsf{RF} : \{0,1\}^{\gamma} \to \mathbb{Z}_q^{k \times 2k}$, defined on-the-fly.

Lemma 9 ($\mathsf{G}_0 \rightsquigarrow \mathsf{G}_1$).

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

Proof. In game G_1 each time the adversary queries a tag for a message m, where she queried a tag for m before, the adversary will get a rerandomized version of the first tag she queried. The $\mathsf{RerandTag}$ algorithm chooses $\mathbf{t}'_i := \mathbf{t}_i + \mathbf{B}\mathbf{s}'_i$ and $\tilde{\mathbf{t}}'_i := \tilde{\mathbf{t}}_i + \tilde{\mathbf{B}}\tilde{\mathbf{s}}'_i$, which is uniformly random in $\mathsf{Span}(\mathbf{B})$ resp. $\mathsf{Span}\big(\tilde{\mathbf{B}}\big)$, independent of $\mathbf{t}_i$ and $\tilde{\mathbf{t}}_i$, because $\mathbf{s}'_i$ and $\tilde{\mathbf{s}}'_i$ are uniform random in $\mathbb{Z}_q^k$. The $\mathsf{RerandTag}$ algorithm then computes $\mathbf{u}'_i$ and $\tilde{\mathbf{u}}'$ such to get another valid tag for m, that is distributed like a fresh tag, independent of the input tag. Thus the games are equivalent.

Note that the rerandomization uses only the "public key" returned by the Init oracle so that it could be carried out by the adversary herself. In the following, we will ignore these duplicated Eval queries. □

Lemma 10 ($\mathsf{G}_1 \rightsquigarrow \mathsf{G}_2$). *For all adversaries* $\mathcal{A}$ *there exist adversaries* $\mathcal{B}_1$ *and* $\mathcal{B}_2$ *with*

$$\big|\Pr\big[\mathsf{G}^{\mathcal{A}}_1 \Rightarrow 1\big] - \Pr\big[\mathsf{G}^{\mathcal{A}}_2 \Rightarrow 1\big]\big| \le \mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H}}(\mathcal{B})$$

and $T(\mathcal{B}) \approx T(\mathcal{A}) + (Q_e + Q_c)L \cdot \mathsf{poly}(\lambda)$*.*

G_0 G_1 G_2 G_3 G_4 $\boxed{\mathsf{G}_5}$

```
INIT_MAC:
$\mathcal{PG} \xleftarrow{\$} \mathsf{PGGen}(1^\lambda)$
parse $\mathcal{PG} =: (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, P_1, P_2, e)$
$H \xleftarrow{\$} \mathcal{H}(1^\lambda)$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$; $\tilde{\mathbf{B}} \xleftarrow{\$} \mathcal{U}_{2k,k}$
for $j \in \{1, \ldots, \gamma\}$, $b \in \{0,1\}$ do
    $\mathbf{X}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{k \times 3k}$
$\tilde{\mathbf{X}}_1 \xleftarrow{\$} \mathbb{Z}_q^{k \times 2k}$; $\tilde{\mathbf{X}}_2 \xleftarrow{\$} \mathbb{Z}_q^{k \times 2k}$; $\mathbf{x}' \xleftarrow{\$} \mathbb{Z}_q^k$
$\Phi : \mathcal{S}^* \dashrightarrow (\mathbb{G}_2^{6k})^* \times \mathbb{G}_2^k$
$\mathsf{dk}_1 := \left([\mathbf{X}_{j,b}\mathbf{B}]_2\right)_{1 \le j \le \gamma, b \in \{0,1\}}$
$\mathsf{dk}_2 := \left([\tilde{\mathbf{X}}_1\tilde{\mathbf{B}}]_2, [\tilde{\mathbf{X}}_2\tilde{\mathbf{B}}]_2\right)$
return $\left(\mathcal{PG}, H, [\mathbf{B}]_2, [\tilde{\mathbf{B}}]_2, \mathsf{dk}_1, \mathsf{dk}_2\right)$

CHAL$(\mathsf{m}^\star = (\mathsf{m}_1^\star, \ldots, \mathsf{m}_p^\star) \in \mathcal{S}^*)$:
$\mathcal{C}_\mathcal{M} := \mathcal{C}_\mathcal{M} \cup \{\mathsf{m}^\star\}$
$\tilde{\mathbf{h}} \xleftarrow{\$} \mathbb{Z}_q^k$
for $i \in \{1, \ldots, p\}$ do
    $\mathbf{h}_i \xleftarrow{\$} \mathbb{Z}_q^k$
    $\mathsf{hm}_i^\star := H(\mathsf{m}_1^\star, \ldots, \mathsf{m}_i^\star)$
    $\mathcal{C}_\mathsf{hm} := \mathcal{C}_\mathsf{hm} \cup \{\mathsf{hm}_i^\star\}$
    $\mathbf{h}_{0,i} := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i^\star[\![j]\!]}^\top \mathbf{h}_i$
    $\mathbf{h}_{0,i} := \mathbf{h}_{0,i} + \mathbf{B}^\perp \mathsf{RF}(\mathsf{m}_{|i}^\star)^\top \mathbf{h}_i$
    $\tilde{\mathbf{h}}_{0,i} := \tilde{\mathbf{X}}_1^\top \mathbf{h}_i + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}}$
    $\mathbf{r}_i \xleftarrow{\$} \mathbb{Z}_q^k$; $\tilde{\mathbf{h}}_{0,i} := \tilde{\mathbf{h}}_{0,i} + \tilde{\mathbf{B}}^\perp \mathbf{r}_i$
$h_K := (\mathbf{x}')^\top \tilde{\mathbf{h}}$; $\boxed{h_K \xleftarrow{\$} \mathbb{Z}_q}$
$\mathbf{H} := \left([\mathbf{h}_i]_1, [\mathbf{h}_{0,i}]_1, [\tilde{\mathbf{h}}_{0,i}]_1\right)_{1 \le i \le p}$
return $\left([\tilde{\mathbf{h}}]_1, \mathbf{H}, [h_K]_T\right)$

FINALIZE_MAC$(\beta \in \{0,1\})$:
return $(\mathcal{C}_\mathsf{hm} \cap \mathcal{Q}_\mathsf{hm} = \emptyset) \wedge \beta$
return $\left(\bigcup_{\mathsf{m}^\star \in \mathcal{C}_\mathcal{M}} \mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_\mathcal{M} = \emptyset\right) \wedge \beta$
```

```
EVAL$(\mathsf{m} = (\mathsf{m}_1, \ldots, \mathsf{m}_p) \in \mathcal{S}^*)$:
if $\mathsf{m} \in \mathcal{Q}_\mathcal{M}$ then
    return $\mathsf{RerandTag}(\mathsf{m}, \Phi(\mathsf{m}))$
$\mathcal{Q}_\mathcal{M} := \mathcal{Q}_\mathcal{M} \cup \{\mathsf{m}\}$
for $i \in \{1, \ldots, p-1\}$ do
    $\mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t}_i := \mathbf{B}\mathbf{s}_i$; $\tilde{\mathbf{s}}_i \xleftarrow{\$} \mathbb{Z}_q^k$; $\tilde{\mathbf{t}}_i := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i$
    $\mathsf{hm}_i := H(\mathsf{m}_1, \ldots, \mathsf{m}_i)$
    $\mathbf{u}_i := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]} \mathbf{t}_i + \tilde{\mathbf{X}}_1 \tilde{\mathbf{t}}_i$
$\mathbf{s}_p \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t}_p := \mathbf{B}\mathbf{s}_p$; $\tilde{\mathbf{s}}_p \xleftarrow{\$} \mathbb{Z}_q^k$; $\tilde{\mathbf{t}}_p := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_p$
$\mathbf{t}_p \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$\tilde{\mathbf{t}}_p \xleftarrow{\$} \mathbb{Z}_q^{2k}$
$\mathsf{hm}_p := H(\mathsf{m})$
if $\mathsf{hm}_p \in \mathcal{Q}_\mathsf{hm}$ then abort
$\mathcal{Q}_\mathsf{hm} := \mathcal{Q}_\mathsf{hm} \cup \{\mathsf{hm}_p\}$
$\mathbf{u}_p := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_p[\![j]\!]} \mathbf{t}_p + \tilde{\mathbf{X}}_1 \tilde{\mathbf{t}}_p$
$\mathbf{u}_p \xleftarrow{\$} \mathbb{Z}_q^k$
$\tilde{\mathbf{u}} := \sum_{i=1}^{p} \tilde{\mathbf{X}}_2 \tilde{\mathbf{t}}_i + \mathbf{x}'$
$\tilde{\mathbf{u}} \xleftarrow{\$} \mathbb{Z}_q^k$
$\Phi(\mathsf{m}) := \left(\left([\mathbf{t}_i]_2, [\tilde{\mathbf{t}}_i]_2, [\mathbf{u}_i]_2\right)_{1 \le i \le p}, [\tilde{\mathbf{u}}]_2\right)$
return $\Phi(\mathsf{m})$

RerandTag$(\mathsf{m} = (\mathsf{m}_1, \ldots, \mathsf{m}_p) \in \mathcal{S}^*, \tau)$:
$\tau =: \left(\left([\mathbf{t}_i]_2, [\tilde{\mathbf{t}}_i]_2, [\mathbf{u}_i]_2\right)_{1 \le i \le p}, [\tilde{\mathbf{u}}]_2\right)$
for $i \in \{1, \ldots, p\}$ do
    $\mathbf{s}_i' \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t}_i' := \mathbf{t}_i + \mathbf{B}\mathbf{s}_i'$
    $\tilde{\mathbf{s}}_i' \xleftarrow{\$} \mathbb{Z}_q^k$; $\tilde{\mathbf{t}}_i' := \tilde{\mathbf{t}}_i + \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i'$
    $\mathbf{u}_i' := \mathbf{u}_i + \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]} \mathbf{B}\mathbf{s}_i' + \tilde{\mathbf{X}}_1 \tilde{\mathbf{B}} \tilde{\mathbf{s}}_i'$
$\tilde{\mathbf{u}}' := \tilde{\mathbf{u}} + \sum_{i=1}^{p} \tilde{\mathbf{X}}_2 \tilde{\mathbf{B}} \tilde{\mathbf{s}}_i'$
return $\left(\left([\mathbf{t}_i']_2, [\tilde{\mathbf{t}}_i']_2, [\mathbf{u}_i']_2\right)_{1 \le i \le p}, [\tilde{\mathbf{u}}']_2\right)$
```

Fig. 10. Hybrids G_0–G_5 for the security proof of MAC_u. The algorithm $\mathsf{RerandTag}$ is only helper function and not an oracle for the adversary. The partial map Φ is initially totally undefined.

Proof. Compared to G_1, the hybrid G_2 aborts if two different messages, for which the adversary queried a tag, have the same hash value. Furthermore, in G_2 the adversary looses (i.e., the output of $\textsc{Finalize}_{\mathsf{MAC}}$ is always 0), if the hash of a prefix of a message sent to the $\textsc{Chal}$ oracle is identical to the hash of a message send to the $\textsc{Eval}$ oracle. So the two games are identical, except when a hash function collision occurs. □

Lemma 11 **($\mathsf{G}_2 \rightsquigarrow \mathsf{G}_3$).** *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1\right]\right| \leq (8k\gamma + 2k)\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}_1) + k\gamma\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},1}(\mathcal{B}_2) + \frac{\gamma Q_c L + 6\gamma + 1}{q-1} + \frac{Q_e}{q^{2k}}$$

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

Proof. In game G_3 the value $\mathbf{u}_p$ is chosen uniformly random (and some side-effect changes are made). For the transition to this game, we use the security of the underlying linear MAC. The reduction is given in Figure 11.

```
$\underline{\textsc{Init}_{\mathsf{MAC}}:}$
$H \xleftarrow{\$} \mathcal{H}(1^\lambda);\ \tilde{\mathbf{B}} \xleftarrow{\$} \mathcal{U}_{2k,k}$
$(\mathcal{PG}, [\mathbf{B}]_2, \mathsf{dk}_1) \xleftarrow{\$} \textsc{Init}_{\mathsf{ri}}$
parse $\mathsf{dk}_1 =: \left([\mathbf{D}_{j,b}]_2\right)_{1\leq j\leq\gamma, b\in\{0,1\}}$
$\tilde{\mathbf{X}}_1 \xleftarrow{\$} \mathbb{Z}_q^{k\times 2k};\ \tilde{\mathbf{X}}_2 \xleftarrow{\$} \mathbb{Z}_q^{k\times 2k};\ \mathbf{x}' \xleftarrow{\$} \mathbb{Z}_q^k$
$\varPhi : \mathcal{S}^* \dashrightarrow \left(\mathbb{G}_2^{6k}\right)^* \times \mathbb{G}_2^k$
$\mathsf{dk}_2 := \left(\left[\tilde{\mathbf{X}}_1\tilde{\mathbf{B}}\right]_2, \left[\tilde{\mathbf{X}}_2\tilde{\mathbf{B}}\right]_2\right)$
return $\left(\mathcal{PG}, H, [\mathbf{B}]_2, \left[\tilde{\mathbf{B}}\right]_2, \mathsf{dk}_1, \mathsf{dk}_2\right)$

$\underline{\textsc{Chal}\left(\mathsf{m}^\star = \left(\mathsf{m}_1^\star, \ldots, \mathsf{m}_p^\star\right) \in \mathcal{S}^*\right):}$
$\mathcal{C}_{\mathcal{M}} := \mathcal{C}_{\mathcal{M}} \cup \{\mathsf{m}^\star\}$
$\tilde{\mathbf{h}} \xleftarrow{\$} \mathbb{Z}_q^k;\ h_K := (\mathbf{x}')^\top \tilde{\mathbf{h}}$
for $i \in \{1, \ldots, p\}$ do
    $\mathsf{hm}_i^\star := H(\mathsf{m}_1^\star, \ldots, \mathsf{m}_i^\star)$
    $\mathcal{C}_{\mathsf{hm}} := \mathcal{C}_{\mathsf{hm}} \cup \{\mathsf{hm}_i^\star\}$
    $(\mathbf{h}_i, \mathbf{h}_{0,i}) \xleftarrow{\$} \textsc{Chal}_{\mathsf{ri}}(\mathsf{hm}_i^\star)$
    $\tilde{\mathbf{h}}_{0,i} := \tilde{\mathbf{X}}_1^\top \mathbf{h}_i + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}}$
$\mathbf{H} := \left([\mathbf{h}_i]_1, [\mathbf{h}_{0,i}]_1, \left[\tilde{\mathbf{h}}_{0,i}\right]_1\right)_{1\leq i\leq p}$
return $\left(\left[\tilde{\mathbf{h}}\right]_1, \mathbf{H}, [h_K]_T\right)$
```

```
$\underline{\textsc{Eval}(\mathsf{m} = (\mathsf{m}_1, \ldots, \mathsf{m}_p) \in \mathcal{S}^*):}$
if $\mathsf{m} \in \mathcal{Q}_{\mathcal{M}}$ then
    return $\mathsf{RerandTag}(\mathsf{m}, \varPhi(\mathsf{m}))$
$\mathcal{Q}_{\mathcal{M}} := \mathcal{Q}_{\mathcal{M}} \cup \{\mathsf{m}\}$
for $i \in \{1, \ldots, p-1\}$ do
    $\mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t}_i := \mathbf{B}\mathbf{s}_i;\ \tilde{\mathbf{s}}_i \xleftarrow{\$} \mathbb{Z}_q^k;\ \tilde{\mathbf{t}}_i := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i$
    $\mathsf{hm}_i := H(\mathsf{m}_1, \ldots, \mathsf{m}_i)$
    $\mathbf{u}_i := \sum_{j=1}^{\gamma} \mathbf{D}_{j,\mathsf{hm}_i[\![j]\!]}\mathbf{s}_i + \tilde{\mathbf{X}}_1\tilde{\mathbf{t}}_i$
$\mathsf{hm}_p := H(\mathsf{m})$
if $\mathsf{hm}_p \in \mathcal{Q}_{\mathsf{hm}}$ then abort
$\mathcal{Q}_{\mathsf{hm}} := \mathcal{Q}_{\mathsf{hm}} \cup \{\mathsf{hm}_p\}$
$\left([\mathbf{t}_p]_2, \left[\mathbf{u}_p'\right]_2\right) \xleftarrow{\$} \textsc{Eval}_{\mathsf{ri}}(\mathsf{hm}_p)$
$\tilde{\mathbf{s}}_p \xleftarrow{\$} \mathbb{Z}_q^k;\ \tilde{\mathbf{t}}_p := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_p$
$\mathbf{u}_p := \mathbf{u}_p' + \tilde{\mathbf{X}}_1\tilde{\mathbf{t}}_p$
$\tilde{\mathbf{u}} := \sum_{i=1}^{p} \tilde{\mathbf{X}}_2\tilde{\mathbf{t}}_i + \mathbf{x}'$
$\varPhi(\mathsf{m}) := \left(\left([\mathbf{t}_i]_2, \left[\tilde{\mathbf{t}}_i\right]_2, [\mathbf{u}_i]_2\right)_{1\leq i\leq p}, [\tilde{\mathbf{u}}]_2\right)$
return $\varPhi(\mathsf{m})$

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

Fig. 11. Reduction for the transition from G_2 to G_3 to the Randomness Injection Lemma.

We use the Randomness Injection Lemma to compute the components $\mathbf{h}_i$ and $\mathbf{h}_{0,i}$ for all levels i in the $\textsc{Chal}$ oracle and to compute $\mathbf{t}_p$ and $\mathbf{u}_p'$, i.e. the last-level components of the tags. For the other components, we use the public key returned from $\textsc{Init}_{\mathsf{ri}}$. This is important to avoid asking both the $\textsc{Eval}_{\mathsf{ri}}$ and $\textsc{Chal}_{\mathsf{ri}}$ oracles on common prefixes of $\textsc{Eval}_{\mathsf{ri}}$-messages and $\textsc{Chal}_{\mathsf{ri}}$-messages.

If the reduction is accessing the $\mathsf{RI}_{\mathsf{real}}$ game, it simulates G_2. Otherwise, it simulates G_3. □

Lemma 12 **($\mathsf{G}_3 \rightsquigarrow \mathsf{G}_4$).** *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_4^{\mathcal{A}} \Rightarrow 1\right]\right| \leq 2k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},2}(\mathcal{B}_1) + k\mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},1}(\mathcal{B}_2) + \frac{6}{q-1}$$

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

Proof. In game G_4 the value $\tilde{\mathbf{u}}$ is chosen uniformly random (and some side-effect changes are made). For the transition to this game, we use the Randomness Packing Lemma (Lemma 5). The reduction is given in Figure 12.

```
$\textsc{Init}_{\mathsf{MAC}}$:
$H \xleftarrow{\$} \mathcal{H}(1^\lambda)$; $\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k}$
for $j \in \{1,\dots,\gamma\}$, $b \in \{0,1\}$ do
    $\mathbf{X}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{k \times 3k}$
$\mathsf{dk}_1 := \left([\mathbf{X}_{j,b}\mathbf{B}]_2\right)_{1 \leq j \leq \gamma, b \in \{0,1\}}$
$\left(\left[\tilde{\mathbf{B}}\right]_2, \mathsf{dk}_2\right) \xleftarrow{\$} \textsc{Init}_{\mathsf{rp}}$
parse $\mathsf{dk}_2 =: \left(\left[\tilde{\mathbf{D}}_1\right]_2, \left[\tilde{\mathbf{D}}_2\right]_2\right)$
$\varPhi : \mathcal{S}^* \dashrightarrow \left(\mathbb{G}_2^{6k}\right)^* \times \mathbb{G}_2^k$
return $\left(\mathcal{PG}, H, [\mathbf{B}]_2, \left[\tilde{\mathbf{B}}\right]_2, \mathsf{dk}_1, \mathsf{dk}_2\right)$

$\textsc{Chal}\left(\mathsf{m}^\star = \left(\mathsf{m}_1^\star, \dots, \mathsf{m}_p^\star\right) \in \mathcal{S}^*\right)$:
$\mathcal{C}_{\mathcal{M}} := \mathcal{C}_{\mathcal{M}} \cup \{\mathsf{m}^\star\}$
$\tilde{\mathbf{h}} \xleftarrow{\$} \mathbb{Z}_q^k$; $h_K := (\mathbf{x}')^\top \tilde{\mathbf{h}}$
for $i \in \{1,\dots,p\}$ do
    $\mathsf{hm}_i^\star := H(\mathsf{m}_1^\star, \dots, \mathsf{m}_i^\star)$
    $\mathcal{C}_{\mathsf{hm}} := \mathcal{C}_{\mathsf{hm}} \cup \{\mathsf{hm}_i^\star\}$
    $\left(\mathbf{h}_i, \tilde{\mathbf{h}}_{0,i}\right) \xleftarrow{\$} \textsc{Chal}_{\mathsf{rp}}\left(\tilde{\mathbf{h}}\right)$
    $\mathbf{h}_{0,i} := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i^\star[\![j]\!]}^\top \mathbf{h}_i$
$\mathbf{H} := \left([\mathbf{h}_i]_1, [\mathbf{h}_{0,i}]_1, \left[\tilde{\mathbf{h}}_{0,i}\right]_1\right)_{1 \leq i \leq p}$
return $\left(\left[\tilde{\mathbf{h}}\right]_1, \mathbf{H}, [h_K]_T\right)$
```

```
$\textsc{Eval}(\mathsf{m} = (\mathsf{m}_1, \dots, \mathsf{m}_p) \in \mathcal{S}^*)$:
if $\mathsf{m} \in \mathcal{Q}_{\mathcal{M}}$ then
    return $\mathsf{RerandTag}(\mathsf{m}, \varPhi(\mathsf{m}))$
$\mathcal{Q}_{\mathcal{M}} := \mathcal{Q}_{\mathcal{M}} \cup \{\mathsf{m}\}$
for $i \in \{1,\dots,p-1\}$ do
    $\mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q^k$; $\mathbf{t}_i := \mathbf{B}\mathbf{s}_i$; $\tilde{\mathbf{s}}_i \xleftarrow{\$} \mathbb{Z}_q^k$; $\tilde{\mathbf{t}}_i := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i$
    $\mathsf{hm}_i := H(\mathsf{m}_1, \dots, \mathsf{m}_i)$
    $\mathbf{u}_i := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]}\mathbf{t}_i + \tilde{\mathbf{D}}_1\tilde{\mathbf{s}}_i$
$\mathsf{hm}_p := H(\mathsf{m})$
if $\mathsf{hm}_p \in \mathcal{Q}_{\mathsf{hm}}$ then abort
$\mathcal{Q}_{\mathsf{hm}} := \mathcal{Q}_{\mathsf{hm}} \cup \{\mathsf{hm}_p\}$
$\mathbf{t}_p \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$\mathbf{u}_p \xleftarrow{\$} \mathbb{Z}_q^k$
$\left(\left[\tilde{\mathbf{t}}_p\right]_2, [\tilde{\mathbf{u}}']_2\right) \xleftarrow{\$} \textsc{Eval}_{\mathsf{ri}}(\mathsf{hm}_p)$
$\tilde{\mathbf{u}} := \sum_{i=1}^{p-1} \tilde{\mathbf{D}}_2\tilde{\mathbf{s}}_i + \tilde{\mathbf{u}}' + \mathbf{x}'$
$\varPhi(\mathsf{m}) := \left(\left([\mathbf{t}_i]_2, \left[\tilde{\mathbf{t}}_i\right]_2, [\mathbf{u}_i]_2\right)_{1 \leq i \leq p}, [\tilde{\mathbf{u}}]_2\right)$
return $\varPhi(\mathsf{m})$

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

Fig. 12. Reduction for the transition from G_3 to G_4 to the Randomness Packing Lemma.

We use the Randomness Packing Lemma to compute the components $\mathbf{h}_i$ and $\tilde{\mathbf{h}}_{0,i}$ for all levels i in the $\textsc{Chal}$ oracle and to compute $\tilde{\mathbf{t}}_p$ and $\tilde{\mathbf{u}}'$. Everything else can be computed with the delegation key returned from $\textsc{Init}_{\mathsf{rp}}$.

If the reduction is accessing the $\mathsf{RP}_{\mathsf{real}}$ game, it simulates G_3. Otherwise, it simulates G_4. □

Lemma 13 **($\mathsf{G}_4 \rightsquigarrow \mathsf{G}_5$).** *For all adversaries $\mathcal{A}$ there exists an adversary $\mathcal{B}$ with*

$$\left|\Pr\left[\mathsf{G}_4^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{G}_5^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{U}_k,\mathsf{PGGen},1}(\mathcal{B}) + \frac{2}{q-1}$$

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

Proof. In game G_5 the value h_K is chosen uniformly random. For the transition to this game, we need a Q_c-fold $\mathcal{U}_k$-MDDH challenge $\left([\mathbf{D}]_1, [\mathbf{f}_1]_1, \ldots, [\mathbf{f}_{Q_c}]_1\right)$. The reduction is given in Figure 13.

```
$\underline{\textsc{Init}_{\mathsf{MAC}}:}$
$H \xleftarrow{\$} \mathcal{H}(1^\lambda);\ \mathbf{B} \xleftarrow{\$} \mathcal{U}_{3k,k};\ \tilde{\mathbf{B}} \xleftarrow{\$} \mathcal{U}_{2k,k}$
for $j \in \{1, \ldots, \gamma\},\ b \in \{0,1\}$ do
    $\mathbf{X}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{k \times 3k}$
$\tilde{\mathbf{X}}_1 \xleftarrow{\$} \mathbb{Z}_q^{k \times 2k};\ \tilde{\mathbf{X}}_2 \xleftarrow{\$} \mathbb{Z}_q^{k \times 2k};\ \mathbf{j}' \xleftarrow{\$} \mathbb{Z}_q^k$
// Implicit: $\mathbf{x}' := \mathbf{j}' + \left(\underline{\mathbf{D}}\overline{\mathbf{D}}^{-1}\right)^\top$
$\varPhi : \mathcal{S}^* \dashrightarrow \left(\mathbb{G}_2^{6k}\right)^* \times \mathbb{G}_2^k$
$\mathsf{dk}_1 := \left([\mathbf{X}_{j,b}\mathbf{B}]_2\right)_{1 \leq j \leq \gamma, b \in \{0,1\}}$
$\mathsf{dk}_2 := \left(\left[\tilde{\mathbf{X}}_1\tilde{\mathbf{B}}\right]_2, \left[\tilde{\mathbf{J}}_2\tilde{\mathbf{B}}\right]_2\right)$
return $\left(\mathcal{PG}, H, [\mathbf{B}]_2, \left[\tilde{\mathbf{B}}\right]_2, \mathsf{dk}_1, \mathsf{dk}_2\right)$

$\underline{\textsc{Chal}\left(\mathsf{m}^\star = \left(\mathsf{m}_1^\star, \ldots, \mathsf{m}_p^\star\right) \in \mathcal{S}^*\right):}$
$\mathcal{C}_{\mathcal{M}} := \mathcal{C}_{\mathcal{M}} \cup \{\mathsf{m}^\star\}$
Let this be the $c$-th $\textsc{Chal}$ query.
$\tilde{\mathbf{h}} := \mathbf{f}_c;\ h_K := (\mathbf{j}')^\top \tilde{\mathbf{h}} + \underline{\mathbf{f}_c}$
for $i \in \{1, \ldots, p\}$ do
    $\mathbf{h}_i \xleftarrow{\$} \mathbb{Z}_q^k$
    $\mathsf{hm}_i^\star := H(\mathsf{m}_1^\star, \ldots, \mathsf{m}_i^\star)$
    $\mathbf{h}_{0,i} := \left(\sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i^\star[\![j]\!]}^\top + \mathbf{B}^\perp \mathsf{RF}\left(\mathsf{m}_{|i}^\star\right)^\top\right)\mathbf{h}_i$
    $\mathbf{r}_i \xleftarrow{\$} \mathbb{Z}_q^k;\ \tilde{\mathbf{h}}_{0,i} := \tilde{\mathbf{X}}_1^\top \mathbf{h}_i + \tilde{\mathbf{X}}_2^\top \tilde{\mathbf{h}} + \tilde{\mathbf{B}}^\perp \mathbf{r}_i$
$\mathbf{H} := \left(\left[\tilde{\mathbf{h}}\right]_1, \left([\mathbf{h}_i]_1, [\mathbf{h}_{0,i}]_1, \left[\tilde{\mathbf{h}}_{0,i}\right]_1\right)_{1 \leq i \leq p}\right)$
return $\left(\mathbf{H}, [h_K]_T\right)$
```

```
$\underline{\textsc{Eval}(\mathsf{m} = (\mathsf{m}_1, \ldots, \mathsf{m}_p) \in \mathcal{S}^*):}$
if $\mathsf{m} \in \mathcal{Q}_{\mathcal{M}}$ then
    return $\mathsf{RerandTag}(\mathsf{m}, \varPhi(\mathsf{m}))$
$\mathcal{Q}_{\mathcal{M}} := \mathcal{Q}_{\mathcal{M}} \cup \{\mathsf{m}\}$
for $i \in \{1, \ldots, p-1\}$ do
    $\mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q^k;\ \mathbf{t}_i := \mathbf{B}\mathbf{s}_i$
    $\tilde{\mathbf{s}}_i \xleftarrow{\$} \mathbb{Z}_q^k;\ \tilde{\mathbf{t}}_i := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i$
    $\mathsf{hm}_i := H(\mathsf{m}_1, \ldots, \mathsf{m}_i)$
    $\mathbf{u}_i := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hm}_i[\![j]\!]}\mathbf{t}_i + \tilde{\mathbf{X}}_1\tilde{\mathbf{t}}_i$
$\mathbf{t}_p \xleftarrow{\$} \mathbb{Z}_q^{3k}$
$\tilde{\mathbf{t}}_p \xleftarrow{\$} \mathbb{Z}_q^{2k}$
$\mathbf{u}_p \xleftarrow{\$} \mathbb{Z}_q^k$
$\tilde{\mathbf{u}} \xleftarrow{\$} \mathbb{Z}_q^k$
$\varPhi(\mathsf{m}) := \left(\left([\mathbf{t}_i]_2, \left[\tilde{\mathbf{t}}_i\right]_2, [\mathbf{u}_i]_2\right)_{1 \leq i \leq p}, [\tilde{\mathbf{u}}]_2\right)$
return $\varPhi(\mathsf{m})$

$\underline{\textsc{Finalize}_{\mathsf{MAC}}(\beta \in \{0,1\}):}$
return $\left(\bigcup_{\mathsf{m}^\star \in \mathcal{C}_{\mathcal{M}}} \mathsf{Prefix}(\mathsf{m}^\star) \cap \mathcal{Q}_{\mathcal{M}} = \emptyset\right) \wedge \beta$
```

Fig. 13. Reduction for the transition from G_4 to G_5 to the Q_c-fold $\mathcal{U}_k$-MDDH challenge $\left([\mathbf{D}]_1, [\mathbf{f}_1]_1, \ldots, [\mathbf{f}_{Q_c}]_1\right)$.

Assume that $\overline{\mathbf{D}}$ is invertible. This happens with probability at least $(1 - 1/(q-1))$. The $\textsc{Init}$ and $\textsc{Eval}$ oracles are identical in both games and simulated correctly by the reduction, because they do not return anything depending on $\mathbf{x}'$. Write $\mathbf{f}_c =: \left(\begin{smallmatrix}\overline{\mathbf{D}}\mathbf{w}_c \\ \underline{\mathbf{D}}\mathbf{w}_c + r_c\end{smallmatrix}\right)$ where $\mathbf{w}_c$ is uniform random in $\mathbb{Z}_q^k$ and r_c is 0 or uniform random in $\mathbb{Z}_q$. In the CHAL queries the reduction picks $\tilde{\mathbf{h}} := \overline{\mathbf{f}_c}$. Since $\overline{\mathbf{f}_c}$ is a uniform random vector, $\tilde{\mathbf{h}}$ is distributed correctly. Furthermore, h_K is computed as

$$h_K := (\mathbf{j}')^\top \tilde{\mathbf{h}} + \underline{\mathbf{f}_c} = (\mathbf{j}')^\top \tilde{\mathbf{h}} + \underline{\mathbf{D}}\overline{\mathbf{D}}^{-1}\overline{\mathbf{f}_c} + r_c = (\mathbf{x}')^\top \tilde{\mathbf{h}} + r_c\,.$$

If $r_c = 0$, we are simulating game G_4. If r_c is uniform random we are simulating game G_5. □

SUMMARY. To prove Theorem 1, we combine Lemmata 9–13 to change h_K from real to random and then apply all Lemmata (except Lemma 13) in reverse order to get to the $\mathsf{uMAC_{rand}}$ game. □

4 Transformation to unbounded HIBE

Our unbounded affine MAC can be tightly transformed to an unbounded HIBE under the $\mathcal{U}_k$-MDDH assumption in $\mathbb{G}_1$. The transformation follows the same idea as [6]. It can be found in the full version.

The unbounded HIBE obtained from our unbounded affine MAC can be instantiated with any MDDH assumption. The result for the SXDH assumption can be found in Figure 14.

References

1. Abe, M., Hofheinz, D., Nishimaki, R., Ohkubo, M., Pan, J.: Compact structure-preserving signatures with almost tight security. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part II. LNCS, vol. 10402, pp. 548–580. Springer, Heidelberg (Aug 2017)
2. Abe, M., Jutla, C.S., Ohkubo, M., Pan, J., Roy, A., Wang, Y.: Shorter QA-NIZK and SPS with tighter security. In: Galbraith, S.D., Moriai, S. (eds.) ASIACRYPT 2019, Part III. LNCS, vol. 11923, pp. 669–699. Springer, Heidelberg (Dec 2019)
3. Abe, M., Jutla, C.S., Ohkubo, M., Roy, A.: Improved (almost) tightly-secure simulation-sound QA-NIZK with applications. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part I. LNCS, vol. 11272, pp. 627–656. Springer, Heidelberg (Dec 2018)
4. Attrapadung, N., Hanaoka, G., Yamada, S.: A framework for identity-based encryption with almost tight security. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 521–549. Springer, Heidelberg (Nov / Dec 2015)
5. Bellare, M., Goldwasser, S.: New paradigms for digital signatures and message authentication based on non-interative zero knowledge proofs. In: Brassard, G. (ed.) CRYPTO'89. LNCS, vol. 435, pp. 194–211. Springer, Heidelberg (Aug 1990)
6. Blazy, O., Kiltz, E., Pan, J.: (Hierarchical) identity-based encryption from affine message authentication. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 408–425. Springer, Heidelberg (Aug 2014)
7. Boneh, D., Boyen, X.: Efficient selective identity-based encryption without random oracles. Journal of Cryptology 24(4), 659–693 (Oct 2011)

```
$\underline{\mathsf{Gen}(1^\lambda):}$
$\mathcal{PG} \xleftarrow{\$} \mathsf{PGGen}(1^\lambda);\ H \xleftarrow{\$} \mathcal{H}(1^\lambda)$
$\mathbf{parse}\ \mathcal{PG} =: (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, P_1, P_2, e)$
$\mathbf{B} \xleftarrow{\$} \mathcal{U}_{3,1};\ \tilde{\mathbf{B}} \xleftarrow{\$} \mathcal{U}_{2,1};\ \mathbf{A} \xleftarrow{\$} \mathcal{U}_1$
$\mathbf{for}\ j \in \{1, \ldots, \gamma\},\ b \in \{0,1\}\ \mathbf{do}$
    $\mathbf{X}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{1\times 3};\ \mathbf{Y}_{j,b} \xleftarrow{\$} \mathbb{Z}_q^{1\times 3}$
    $\mathbf{Z}_{j,b} := \left(\mathbf{Y}_{j,b}^\top \mid \mathbf{X}_{j,b}^\top\right)\mathbf{A}$
    $\mathbf{D}_{j,b} := \mathbf{X}_{j,b}\mathbf{B};\ \mathbf{E}_{j,b} := \mathbf{Y}_{j,b}\mathbf{B}$
$\mathbf{for}\ \delta \in \{1,2\}\ \mathbf{do}$
    $\tilde{\mathbf{X}}_\delta \xleftarrow{\$} \mathbb{Z}_q^{1\times 2};\ \tilde{\mathbf{Y}}_\delta \xleftarrow{\$} \mathbb{Z}_q^{1\times 2}$
    $\tilde{\mathbf{Z}}_\delta := \left(\tilde{\mathbf{Y}}_\delta^\top \mid \tilde{\mathbf{X}}_\delta^\top\right)\mathbf{A}$
    $\tilde{\mathbf{D}}_\delta := \tilde{\mathbf{X}}_\delta\tilde{\mathbf{B}};\ \tilde{\mathbf{E}}_\delta := \tilde{\mathbf{Y}}_\delta\tilde{\mathbf{B}}$
$\mathbf{x}' \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{y}' \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{z}' := (\mathbf{y}' \mid \mathbf{x}') \cdot \mathbf{A}$
$\mathsf{pk} := \Big(\mathcal{PG}, H, [\mathbf{A}]_1, \left(\left[\mathbf{Z}_{j,b}\right]_1\right)_{1\le j\le\gamma, b\in\{0,1\}},$
        $\left[\tilde{\mathbf{Z}}_1\right]_1, \left[\tilde{\mathbf{Z}}_2\right]_1, \left[\mathbf{z}'\right]_1\Big)$
$\mathsf{dk} := \Big([\mathbf{B}]_2, \left[\tilde{\mathbf{B}}\right]_2, \left(\left[\mathbf{D}_{j,b}\right]_2, \left[\mathbf{E}_{j,b}\right]_2\right)_{\substack{1\le j\le\gamma,,\\ b\in\{0,1\}}}$
        $\left[\tilde{\mathbf{D}}_1\right]_2, \left[\tilde{\mathbf{D}}_2\right]_2, \left[\tilde{\mathbf{E}}_1\right]_2, \left[\tilde{\mathbf{E}}_2\right]_2\Big)$
$\mathsf{sk} := \Big(\mathsf{sk}_{\mathsf{MAC}}, \left(\mathbf{Y}_{j,b}\right)_{\substack{1\le j\le\gamma,,\\ b\in\{0,1\}}}, \tilde{\mathbf{Y}}_1, \tilde{\mathbf{Y}}_2, \mathbf{y}'\Big)$
$\mathbf{return}\ (\mathsf{pk}, \mathsf{dk}, \mathsf{sk})$

$\underline{\mathsf{Ext}(\mathsf{sk}, \mathsf{id} = (\mathsf{id}_1, \ldots, , \mathsf{id}_p) \in \mathcal{S}^*):}$
$\mathbf{for}\ i \in \{1, \ldots, p\}\ \mathbf{do}$
    $\mathbf{s}_i \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{t}_i := \mathbf{B}\mathbf{s}_i;\ \tilde{\mathbf{s}}_i \xleftarrow{\$} \mathbb{Z}_q;\ \tilde{\mathbf{t}}_i := \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i$
    $\mathsf{hid}_i := H(\mathsf{id}_1, \ldots, \mathsf{id}_i)$
    $\mathbf{u}_i := \sum_{j=1}^{\gamma} \mathbf{X}_{j,\mathsf{hid}_i[\![j]\!]}\mathbf{t}_i + \tilde{\mathbf{X}}_1\tilde{\mathbf{t}}_i$
    $\mathbf{v}_i := \sum_{j=1}^{\gamma} \mathbf{Y}_{j,\mathsf{hid}_i[\![j]\!]}\mathbf{t}_i + \tilde{\mathbf{Y}}_1\tilde{\mathbf{t}}_i$
$\tilde{\mathbf{u}} := \sum_{i=1}^{p} \tilde{\mathbf{X}}_2\tilde{\mathbf{t}}_i + \mathbf{x}';\ \tilde{\mathbf{v}} := \sum_{i=1}^{p} \tilde{\mathbf{Y}}_2\tilde{\mathbf{t}}_i + \mathbf{y}'$
$\mathbf{return}\ \Big(\left([\mathbf{t}_i]_2, \left[\tilde{\mathbf{t}}_i\right]_2, [\mathbf{u}_i]_2, [\mathbf{v}_i]_2\right)_{1\le i\le p},$
        $[\tilde{\mathbf{u}}]_2, [\tilde{\mathbf{v}}]_2\Big)$

$\underline{\mathsf{Enc}(\mathsf{pk}, \mathsf{id} = (\mathsf{id}_1, \ldots, \mathsf{id}_p) \in \mathcal{S}^*):}$
$\mathbf{r} \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{c}_4 := \mathbf{A}\mathbf{r};\ \mathsf{K} := \mathbf{z}' \cdot \mathbf{r}$
$\mathbf{for}\ i \in \{1, \ldots, p\}\ \mathbf{do}$
    $\mathbf{r}_i \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{c}_{2,i} := \mathbf{A}\mathbf{r}_i$
    $\mathsf{hid}_i := H(\mathsf{id}_1, \ldots, \mathsf{id}_i)$
    $\mathbf{c}_{1,i} := \sum_{j=1}^{\gamma} \mathbf{Z}_{j,\mathsf{hid}_i[\![j]\!]}\mathbf{r}_i$
    $\mathbf{c}_{3,i} := \tilde{\mathbf{Z}}_1\mathbf{r}_i + \tilde{\mathbf{Z}}_2\mathbf{r}$
$\mathsf{C} := \Big(\left([\mathbf{c}_{1,i}]_1, [\mathbf{c}_{2,i}]_1, [\mathbf{c}_{3,i}]_1\right)_{1\le i\le p}, [\mathbf{c}_4]_1\Big)$
$\mathbf{return}\ \left([\mathsf{K}]_T, \mathsf{C}\right)$

$\underline{\mathsf{Del}(\mathsf{dk}, \mathsf{usk}[\mathsf{id}], \mathsf{id} \in \mathcal{S}^p, \mathsf{id}_{p+1} \in \mathcal{S}):}$
$\mathbf{parse}\ \mathsf{usk}[\mathsf{id}] =: \Big(\Big([\mathbf{t}_i]_2, \left[\tilde{\mathbf{t}}_i\right]_2, [\mathbf{u}_i]_2,$
        $[\mathbf{v}_i]_2\Big)_{1\le i\le p}, [\tilde{\mathbf{u}}]_2, [\tilde{\mathbf{v}}]_2\Big)$
$\mathbf{t}_{p+1} := \mathbf{0};\ \tilde{\mathbf{t}}_{p+1} := \mathbf{0}$
$\mathbf{u}_{p+1} := \mathbf{0};\ \mathbf{v}_{p+1} := \mathbf{0}$
$\mathsf{id}' := (\mathsf{id}_1, \ldots, \mathsf{id}_p, \mathsf{id}_{p+1})$
$\mathsf{usk}[\mathsf{id}'] := \Big(\left([\mathbf{t}_i]_2, [\mathbf{u}_i]_2, [\mathbf{v}_i]_2\right)_{1\le i\le p+1},$
        $[\tilde{\mathbf{u}}]_2, [\tilde{\mathbf{v}}]_2\Big)$
$\mathbf{return}\ \mathsf{RerandUSK}(\mathsf{dk}, \mathsf{id}', \mathsf{usk}[\mathsf{id}'])$

$\underline{\mathsf{RerandUSK}(\mathsf{dk}, \mathsf{id} \in \mathcal{S}^p, \mathsf{usk}[\mathsf{id}]):}$
$\mathbf{parse}\ \mathsf{usk}[\mathsf{id}] =: \Big(\Big([\mathbf{t}_i]_2, \left[\tilde{\mathbf{t}}_i\right]_2, [\mathbf{u}_i]_2,$
        $[\mathbf{v}_i]_2\Big)_{1\le i\le p}, [\tilde{\mathbf{u}}]_2, [\tilde{\mathbf{v}}]_2\Big)$
$\mathbf{for}\ i \in \{1, \ldots, p\}\ \mathbf{do}$
    $\mathbf{s}_i' \xleftarrow{\$} \mathbb{Z}_q;\ \mathbf{t}_i' := \mathbf{t}_i + \mathbf{B}\mathbf{s}_i'$
    $\tilde{\mathbf{s}}_i' \xleftarrow{\$} \mathbb{Z}_q;\ \tilde{\mathbf{t}}_i' := \tilde{\mathbf{t}}_i \xleftarrow{\$} \tilde{\mathbf{B}}\tilde{\mathbf{s}}_i'$
    $\mathsf{hid}_i := H(\mathsf{id}_1, \ldots, \mathsf{id}_i)$
    $\mathbf{u}_i' := \mathbf{u}_i + \sum_{j=1}^{\gamma} \mathbf{D}_{j,\mathsf{hid}_i[\![j]\!]}\mathbf{s}_i' + \tilde{\mathbf{D}}_1\tilde{\mathbf{s}}_i'$
    $\mathbf{v}_i' := \mathbf{v}_i + \sum_{j=1}^{\gamma} \mathbf{E}_{j,\mathsf{hid}_i[\![j]\!]}\mathbf{s}_i' + \tilde{\mathbf{E}}_1\tilde{\mathbf{s}}_i'$
$\tilde{\mathbf{u}}' := \tilde{\mathbf{u}} + \sum_{i=1}^{p} \tilde{\mathbf{D}}_2\tilde{\mathbf{s}}_i'$
$\tilde{\mathbf{v}}' := \tilde{\mathbf{v}} + \sum_{i=1}^{p} \tilde{\mathbf{E}}_2\tilde{\mathbf{s}}_i'$
$\mathbf{return}\ \Big(\left([\mathbf{t}_i']_2, \left[\tilde{\mathbf{t}}_i'\right]_2, [\mathbf{u}_i']_2, [\mathbf{v}_i']_2\right)_{1\le i\le p},$
        $[\tilde{\mathbf{u}}']_2, [\tilde{\mathbf{v}}']_2\Big)$

$\underline{\mathsf{Dec}(\mathsf{usk}[\mathsf{id}], \mathsf{id} = (\mathsf{id}_1, \ldots, \mathsf{id}_p) \in \mathcal{S}^*, \mathsf{C}):}$
$\mathbf{parse}\ \mathsf{usk}[\mathsf{id}] =: \Big(\Big([\mathbf{t}_i]_2, \left[\tilde{\mathbf{t}}_i\right]_2, [\mathbf{u}_i]_2,$
        $[\mathbf{v}_i]_2\Big)_{1\le i\le p}, [\tilde{\mathbf{u}}]_2, [\tilde{\mathbf{v}}]_2\Big)$
$\mathbf{parse}\ \mathsf{C} =: \Big(\Big([\mathbf{c}_{1,i}]_1, [\mathbf{c}_{2,i}]_1,$
        $[\mathbf{c}_{3,i}]_1\Big)_{1\le i\le p}, [\mathbf{c}_4]_1\Big)$
$[\mathsf{K}]_T := \sum_{i=1}^{p}\Big(e\Big(\left[\mathbf{c}_{2,i}^\top\right]_1, \begin{bmatrix}\mathbf{v}_i\\ \mathbf{u}_i\end{bmatrix}_2\Big)$
        $- e\left(\left[\mathbf{c}_{1,i}^\top\right]_1, [\mathbf{t}_i]_2\right) - e\left(\left[\mathbf{c}_{3,i}^\top\right]_1, \left[\tilde{\mathbf{t}}_i\right]_2\right)\Big)$
        $+ e\Big(\left[\mathbf{c}_4^\top\right]_1, \begin{bmatrix}\tilde{\mathbf{v}}\\ \tilde{\mathbf{u}}\end{bmatrix}_2\Big)$
$\mathbf{return}\ [\mathsf{K}]_T$
```

Fig. 14. The scheme obtained from MAC_u instantiated with the SXDH assumption.

8. Chen, J., Gong, J., Kowalczyk, L., Wee, H.: Unbounded ABE via bilinear entropy expansion, revisited. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part I. LNCS, vol. 10820, pp. 503–534. Springer, Heidelberg (Apr / May 2018)
9. Chen, J., Wee, H.: Fully, (almost) tightly secure IBE and dual system groups. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 435–460. Springer, Heidelberg (Aug 2013)
10. Cohn-Gordon, K., Cremers, C., Gjøsteen, K., Jacobsen, H., Jager, T.: Highly efficient key exchange protocols with optimal tightness. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part III. LNCS, vol. 11694, pp. 767–797. Springer, Heidelberg (Aug 2019)
11. Escala, A., Herold, G., Kiltz, E., Ràfols, C., Villar, J.: An algebraic framework for Diffie-Hellman assumptions. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 129–147. Springer, Heidelberg (Aug 2013)
12. Garg, S., Gay, R., Hajiabadi, M.: Master-key KDM-secure IBE from pairings. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020, Part I. LNCS, vol. 12110, pp. 123–152. Springer, Heidelberg (May 2020)
13. Gay, R., Hofheinz, D., Kiltz, E., Wee, H.: Tightly CCA-secure encryption without pairings. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part I. LNCS, vol. 9665, pp. 1–27. Springer, Heidelberg (May 2016)
14. Gay, R., Hofheinz, D., Kohl, L.: Kurosawa-desmedt meets tight security. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 133–160. Springer, Heidelberg (Aug 2017)
15. Gay, R., Hofheinz, D., Kohl, L., Pan, J.: More efficient (almost) tightly secure structure-preserving signatures. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part II. LNCS, vol. 10821, pp. 230–258. Springer, Heidelberg (Apr / May 2018)
16. Gentry, C., Silverberg, A.: Hierarchical ID-based cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 548–566. Springer, Heidelberg (Dec 2002)
17. Gjøsteen, K., Jager, T.: Practical and tightly-secure digital signatures and authenticated key exchange. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part II. LNCS, vol. 10992, pp. 95–125. Springer, Heidelberg (Aug 2018)
18. Gong, J., Cao, Z., Tang, S., Chen, J.: Extended dual system group and shorter unbounded hierarchical identity based encryption. Designs, Codes and Cryptography 80(3), 525–559 (Sep 2016), https://doi.org/10.1007/s10623-015-0117-z
19. Gong, J., Chen, J., Dong, X., Cao, Z., Tang, S.: Extended nested dual system groups, revisited. In: Cheng, C.M., Chung, K.M., Persiano, G., Yang, B.Y. (eds.) PKC 2016, Part I. LNCS, vol. 9614, pp. 133–163. Springer, Heidelberg (Mar 2016)
20. Gong, J., Dong, X., Chen, J., Cao, Z.: Efficient IBE with tight reduction to standard assumption in the multi-challenge setting. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 624–654. Springer, Heidelberg (Dec 2016)
21. Han, S., Liu, S., Lyu, L., Gu, D.: Tight leakage-resilient CCA-security from quasi-adaptive hash proof system. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part II. LNCS, vol. 11693, pp. 417–447. Springer, Heidelberg (Aug 2019)
22. Hesse, J., Hofheinz, D., Kohl, L.: On tightly secure non-interactive key exchange. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part II. LNCS, vol. 10992, pp. 65–94. Springer, Heidelberg (Aug 2018)
23. Hofheinz, D., Jia, D., Pan, J.: Identity-based encryption tightly secure under chosen-ciphertext attacks. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 190–220. Springer, Heidelberg (Dec 2018)

24. Hofheinz, D., Kiltz, E.: Secure hybrid encryption from weakened key encapsulation. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 553–571. Springer, Heidelberg (Aug 2007)
25. Hofheinz, D., Koch, J., Striecks, C.: Identity-based encryption with (almost) tight security in the multi-instance, multi-ciphertext setting. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 799–822. Springer, Heidelberg (Mar / Apr 2015)
26. Horwitz, J., Lynn, B.: Toward hierarchical identity-based encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 466–481. Springer, Heidelberg (Apr / May 2002)
27. Kowalczyk, L., Lewko, A.B.: Bilinear entropy expansion from the decisional linear assumption. In: Gennaro, R., Robshaw, M.J.B. (eds.) CRYPTO 2015, Part II. LNCS, vol. 9216, pp. 524–541. Springer, Heidelberg (Aug 2015)
28. Kurosawa, K., Desmedt, Y.: A new paradigm of hybrid encryption scheme. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 426–442. Springer, Heidelberg (Aug 2004)
29. Langrehr, R., Pan, J.: Tightly secure hierarchical identity-based encryption. In: Lin, D., Sako, K. (eds.) PKC 2019, Part I. LNCS, vol. 11442, pp. 436–465. Springer, Heidelberg (Apr 2019)
30. Langrehr, R., Pan, J.: Hierarchical identity-based encryption with tight multi-challenge security. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020, Part I. LNCS, vol. 12110, pp. 153–183. Springer, Heidelberg (May 2020)
31. Lewko, A.B.: Tools for simulating features of composite order bilinear groups in the prime order setting. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 318–335. Springer, Heidelberg (Apr 2012)
32. Lewko, A.B., Waters, B.: New techniques for dual system encryption and fully secure HIBE with short ciphertexts. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 455–479. Springer, Heidelberg (Feb 2010)
33. Lewko, A.B., Waters, B.: Unbounded HIBE and attribute-based encryption. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 547–567. Springer, Heidelberg (May 2011)
34. Lewko, A.B., Waters, B.: Why proving HIBE systems secure is difficult. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 58–76. Springer, Heidelberg (May 2014)
35. Okamoto, T., Takashima, K.: Fully secure unbounded inner-product and attribute-based encryption. In: Wang, X., Sako, K. (eds.) ASIACRYPT 2012. LNCS, vol. 7658, pp. 349–366. Springer, Heidelberg (Dec 2012)
36. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakley, G.R., Chaum, D. (eds.) CRYPTO'84. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (Aug 1984)
37. Tomida, J.: Tightly secure inner product functional encryption: Multi-input and function-hiding constructions. In: Galbraith, S.D., Moriai, S. (eds.) ASIACRYPT 2019, Part III. LNCS, vol. 11923, pp. 459–488. Springer, Heidelberg (Dec 2019)
38. Waters, B.: Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 619–636. Springer, Heidelberg (Aug 2009)
39. Waters, B.R.: Efficient identity-based encryption without random oracles. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127. Springer, Heidelberg (May 2005)