Key-Indistinguishable Message Authentication Codes

Joël Alwen[1], Martin Hirt[1], Ueli Maurer[1], Arpita Patra[2], and Pavel Raykov[1]

[1] Department of Computer Science, ETH Zurich, Switzerland.
{alwenj,martin.hirt,ueli.maurer,pavel.raykov}@inf.ethz.ch
[2] Applied Statistics Unit, ISI Kolkata, India
arpitapatra10@gmail.com

Abstract. While standard message authentication codes (MACs) guarantee authenticity of messages, they do not, in general, guarantee the anonymity of the sender and recipient. For example it may be easy for an observer to determine whether or not two authenticated messages were sent by the same party even without any information about the secret key used. However preserving any uncertainty an attacker may have about the identities of honest parties engaged in authenticated communication is an important goal of many cryptographic applications. For example this is stated as an explicit goal of modern cellphone authentication protocols [rGPP12] and RFID based authentication systems [Vau10].

In this work we introduce and construct a new fundamental cryptographic primitive called *key indistinguishable* (KI) MACs. These can be used to realize many of the most important higher-level applications requiring some form of anonymity and authenticity [AHM+14a]. We show that much (though not all) of the modular MAC construction framework of [DKPW12] gives rise to several variants of KI MACs. On the one hand, we show that KI MACs can be built from hash proof systems and certain weak PRFs allowing us to base security on such assumption as DDH, CDH and LWE. Next we show that the two direct constructions from the LPN assumption of [DKPW12] are KI, resulting in particularly efficient constructions based on structured assumptions. On the other hand, we also give a very simple and efficient construction based on a PRF which allows us to base KI MACs on some ideal primitives such as an ideal compression function (using HMAC) or block-cipher (using say CBC-MAC). In particular, by using our PRF construction, many real-world implementations of MACs can be easily and cheaply modified to obtain a KI MAC. Finally we show that the transformations of [DKPW12] for increasing the domain size of a MAC as well as for strengthening the type of unforgeability it provides also preserve (or even strengthen) the type of KI enjoyed by the MAC. All together these results provide a wide range of assumptions and construction paths for building various flavors of this new primitive.

1 Introduction

1.1 Anonymous Authenticity

In many applications preserving anonymity can conflict with other desirable security properties such as secrecy and authenticity. In [BBDP01, KMO+13] the authors described and analyzed cryptographic primitives providing both anonymity and secrecy. In particular [BBDP01] define and realize the notion of *Key-Private* public key encryption (PKE) which, in addition to the usual secrecy provided by PKE, also guarantee that an adversary learns nothing about the target public key under which a given ciphertext was encrypted. Intuitively this can be used to provide reciever-anonymous private communication, a concept which was formalized in [KMO+13].

In this work we address the dual problem of providing anonymity in tandem with authenticity. That is we focus on the private key setting and define the notion of a *Key-Indistinguishable* Message Authentication Code (KI-MAC). These are MACs which have the added benefit that they reveal nothing about the keys used to generate the authentication tags. In [AHM+14b] it is shown how such schemes can be used to realize higher level applications such as anonymous authenticated or even secure message transmission and anonymous entity authentication each in strongly composable way.

1.2 Our Contributions

On the highest level we achieve our goal of constructing KI-MACs in three steps detailing a modular and flexible approach. First we formally define KI-MACs via a pair of games and describe some relevant variants thereof. Next we show the security of several constructions either based on Learning Parity with Noise assumption (LPN) or black-box primitives such as hash proof systems (HPS), certain weak pseudorandom functions (wPRF), and variable input-length PRFs. From a theoretical perspective the former constructions allow us to realize KI-MACs from a wide array of number-theoretic assumptions (beyond LPN) such as the Paillier assumption, DDH, CDH and LWE. From a practical perspective the PRF construction demonstrates how to base a KI-MAC on an ideal compression function (using HMAC), an ideal block-cipher (using CBC-MAC [BPR05] and several of its variable input-length extensions such as OMAC [IK03], ECBC [BPR05]) or a fixed-input length PRF (using SS-NMAC [DS09]). In the third step we show that various transformations on MACs for strengthening their security properties also preserve or strengthen the flavour of key-indistinguishability provided by the MAC.

We remark that all MAC schemes in this work are (neccecarily) probabilistic which may be a problem for extremely light-weight computing devices. However they can easily and generically be translated into stateful but deterministic parties by using a PRG.[3]

Exact Security. All security statements we give come with an exact security analysis (as opposed to asymptotic ones). We see at least two advantages in taking this approach. First, such results greatly facilitate comparing the quality/efficiency trade-off obtained via different constructions especially when based on the same underlying cryptographic assumptions. A somewhat less common but equally relevant advantage is that such statements make explicit the benefits obtained by enforcing

[3] In particular the security proofs for the probabilistic setting then automatically carry over (at least in a computational sense) by preceding the proof with a hybrid argument replacing the output of each call to the PRG with fresh random numbers.

constraints on the adversary through implementation choices. Take for example a protocol whose security degrades say in $q/|\mathcal{M}|$: the number of times an adversary can interact with a client divided by the size of the message space supported by a MAC. Normally such a protocol would require a MAC with at least 160-bit messages to be considered secure. However, if implemented on hardware which guarantees failure after a limited number of interactions, say $q \leq 2^{10}$ (a common assumption in the RFID setting) the MAC now need only support 100-bit messages potentially reducing the hardware costs of the resulting implementation significantly.

1.3 Related Work

MACs are one of the most fundamental, common and widely studied primitives in modern cryptography, especially in practice and a wide variety of constructions have been developed in the past. Most relevant to this work is [DKPW12] from which most of the MAC constructions and transformations analyzed in this work are taken (the notable exception being the PRF based construction of 3.1). That work focuses mainly on theoretical constructions of MACs with the aim of expanding the class of assumptions upon which we can base our security. However given practical efficiency constraints and the difficulty in designing secure symmetric key cryptographic primitives much attention has been focused on constructing secure (variable-input length) MACs from other existing (but idealized) symnetric key primitives such as block-ciphers [BPR05, IK03], compression functions [Bel06, BCK96a], and even fixed-input length PRFs [DS09]. Indeed, some of these have found wide acceptance in practice [Nat], however we stress that none of these constructions result in KI-MACs as they all result in deterministic MACs which trivially can not be key-indistinguishable.

On the other hand cryptographic applications ensuring anonymity have almost exclusively been studied in the context of interactive protocols and are therefore tailored to specific applications rather than providing a general tool with which anonymous applications can be constructed. The most relevant example to this work being [AHM+14b] which investigates how KI-MACs can be used in higher level protocols to construct various idealized multi-user anonymous functionalities. Some other notable examples are [Vau10, HPVP11, DLYZ11, BLdMT09, BM11] which primarily focus on entity authentication (often based on lite-weight RFID cards) and [AMRR11, TM12, AMR+12, LSWW13] which focus specifically on the requirements of mobile phone network communication protocols.

The most important exception to this trend is the work of [BBDP01] which investigates PKE schemes that additionaly hide all information about which public key was used to encrypt a given ciphertext. This is motivated by the higher level application of receiver-anonymous private message transmission as formalized in [KMO+13].

1.4 Outline

In Section 2 we briefly review various existing and some new security notions for MAC schemes. Next, in Section 3 we investigate a variety of constructions of varying strengths (and their consequences) based on both black-box and number-theoretic assumptions. Finally in Section 4 we describe how to strengthen the security properties of KI-MACs via some black-box transformations.

2 Definitions

We review some variants of secure message authentication codes and define the new property of key indistinguishability.

Syntax. A message authentication code $\mathsf{MAC} = \{\mathsf{KG}, \mathsf{TAG}, \mathsf{VRFY}\}$ is a triple of algorithms with associated key space $\mathcal{K}$, message space $\mathcal{M}$, and tag space $\mathcal{T}$.

- **Key Generation.** The probabilistic key generation algorithm $k \leftarrow \mathsf{KG}(1^\lambda)$ takes as input a security parameter $\lambda \in \mathbb{N}$ (in unary) and outputs a secret key $k \in \mathcal{K}$.
- **Tagging.** The probabilistic authentication algorithm $\tau \leftarrow \mathsf{TAG}_k(m)$ takes as input a secret key $k \in \mathcal{K}$ and a message $m \in \mathcal{M}$ and outputs an authentication tag $\tau \in \mathcal{T}$.
- **Verification.** The deterministic verification algorithm $\mathsf{VRFY}_k(m, \tau)$ takes as input a secret key $k \in \mathcal{K}$, a message $m \in \mathcal{M}$ and a tag $\tau \in \mathcal{T}$ and outputs an element of the set $\{\texttt{Accept}, \texttt{Reject}\}$.

Next we define some useful properties such a triple of algorithms can have such as completeness and unforgeability. We also discuss two further less common security notions for MACs, called message hiding and key indistinguishability which can only be achieved by *randomized* MACs. While the former notion was already introduced in [DKPW12] and recalled in Appendix A, the latter is defined for the first time in this work.

Completeness. We say that MAC has completeness error η if for all $m \in \mathcal{M}$ and $\lambda \in \mathbb{N}$,

$$\Pr[\mathsf{VRFY}_k(m, \tau) = \texttt{Reject} : \quad k \leftarrow \mathsf{KG}(1^\lambda), \tau \leftarrow \mathsf{TAG}_k(m)] \leq \eta.$$

Unforgeability. We recall the standard notion security for (randomized) MACs; namely unforgeability under a chosen message (and verification) attack (**uf-cmva**). We denote by $\mathbf{Adv}^{\mathbf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$, the *advantage* of the adversary A in forging the message for a random key $k \leftarrow \mathsf{KG}(1^\lambda)$. Formally it is the probability that the following experiment outputs 1.

Experiment. $\mathbf{Exp}^{\mathbf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$
- $k \leftarrow \mathsf{KG}(1^\lambda)$
- *Invoke* $\mathsf{A}^{\mathsf{TAG}_k(\cdot), \mathsf{VRFY}_k(\cdot,\cdot)}$.
- *Output* 1 *if* A *queried* (m^*, τ^*) *to* $\mathsf{VRFY}_k(\cdot,\cdot)$ *s.t.* $\mathsf{VRFY}_k(m^*, \tau^*) = \texttt{Accept}$ *and* A *did not receive* τ^* *by querying* m^* *to* $\mathsf{TAG}_k(\cdot)$.

The above experiment can be weakened in several ways to obtain useful variants. In the *selective* unforgeability (**suf-cmva**) notion defined in [DKPW12], A has to specify the target message m^* before making any queries to the oracles in $\mathbf{Exp}^{\mathbf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$. A yet weaker notion called *universal* unforgeability (**uuf-cmva**) requires the adversary to produce a fresh tag for a uniform random message $m^* \leftarrow \mathcal{M}$ given as input to the adversary. We call the modified experiments $\mathbf{Exp}^{\mathbf{suf\text{-}cmva}}_{\mathsf{MAC}}$ and $\mathbf{Exp}^{\mathbf{uuf\text{-}cmva}}_{\mathsf{MAC}}$, respectively. Another way in which the $\{\mathbf{uuf}, \mathbf{suf}, \mathbf{uf}\}$-**cmva** security notions can be weakened is to restrict the adversary A to making only a single query to the verification oracle.[4] To denote the resulting security notions we write $\{\mathbf{uuf}, \mathbf{suf}, \mathbf{uf}\}$-**cma** respectively.[5] Finally,

[4] Note that this is only a meaningful restriction for MACs with a randomized tagging algorithm since a deterministic tagging algorithm can trivially be used as a verification oracle.

[5] Equivalently we sometimes speak of the adversary simply having *no* access to the verification oracle and instead outputting an attempted forgery at the end of her execution in the **cma** type experiments.

if the winning condition of the experiment is to ask only those m^* that have not been previously queried to $\mathsf{TAG}_k(\cdot)$ then we refer to the resulting notion as *weakly* unforgeable while referring to the more stringent security notions as *strong*. In particular the $\{\mathbf{suf}, \mathbf{uf}\}$-$\{\mathbf{cma}, \mathbf{cmva}\}$ definitions in [DKPW12] are all weak variants. In general in this work unless stated otherwise we always mean the strong variants.

We refer to an efficient (i.e. PPT) adversary A playing a **cmva** type experiments as a (t, q_t, q_v)-adversary if it runs in time at most t, and for any pair of oracles with a fixed key A makes at most q_t tag and q_v verification queries.

Definition 1 (Unforgeability). *A* MAC *scheme is* (strongly) (t, q_t, q_v, ϵ)-**uf**-**cmva** secure *if for any* (t, q_t, q_v)*-adversary* A *we have:*

$$\mathbf{Adv}^{\mathbf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda) := \Pr[\mathbf{Exp}^{\mathbf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda) \to 1] \leq \epsilon.$$

It is (t, q_t, ϵ)-**uf**-**cmva** secure *if it is* $(t, q_t, 1, \epsilon)$-**uf**-**cmva** *secure.*

We omit the analogous definitions for the **suf**, and **uuf** variants with and without verification queries detailed above. From these definitions it is immediate that for any $t, q_t, q_v \in \mathbb{N}$ and $\epsilon \geq 0$ the following relation holds for both strong and weak variants:

$$(t, q_t, q_v, \epsilon)\text{-}\mathbf{uf\text{-}cmva} \implies (t, q_t, q_v, \epsilon)\text{-}\mathbf{suf\text{-}cmva} \implies (t, q_t, q_v, \epsilon)\text{-}\mathbf{uuf\text{-}cmva}.$$

Further, as observed in [DKPW12], every weakly (t, q_t, ϵ)-**suf**-**cma** MAC is also weakly $(t, q_t, \epsilon 2^{\mu})$-**uf**-**cma** secure where $|\mathcal{M}| = 2^{\mu}$, since the adversary can guess in advance for which message it can mount the forgery attack. The same observation holds for strong unforgeability.

Key Indistinguishability. Intuitively, the notion of key indistinguishability (KI) ensures that tags leak no information about the secret key (or more generally the internal state of the tag algorithm). Indeed this permits the use of KI-MACs in implementing higher level anonymous authentication applications as detailed in [AHM+14b]. We note that such a property is not implied by even the strongest of unforgeability notions defined above.[6]

To capture the desired intuition we define a game where an adversary is given access to two sets of oracles. Its goal is to determine if the two sets use the same key or two independent random keys. To formalize this we introduce some notation. For keys $k_0, k_1 \in \mathcal{K}$ we write $[k_0, k_1]$ to denote the 4-tuple of oracles $(\mathsf{TAG}_{k_0}, \mathsf{VRFY}_{k_0}, \mathsf{TAG}_{k_1}, \mathsf{TAG}_{k_1})$. Moreover we write $[k_0, k_0]$ to denote a similar 4-tuple but where the TAG oracles share their entire internal state including secret key (and similarly for the VRFY oracles). In other words, calls to the first and third oracle of $[k_0, k_0]$ are answered by essentially the same oracle (and similarly for the second and fourth oracle).[7]

[6] Indeed this is not difficult to see. For example we can modify any (say strongly **ufcmva**) unforgeable scheme as follows such that it is clearly not KI yet maintains its original unforgeability property. We double the key size, use the first half of the key in conjunction with the original TAG algorithm to tag the message and then append the second half of the key to the resulting tag. Clearly the scheme remains unforgeable however yet it is trivial to tell apart tags issued under different keys.

[7] For stateful MACs it is important that the full state (and not just the secret key) be shared between matching oracles in $[k_0, k_0]$. Suppose we have a secure MAC which hides all information about the secret keys. We can modify the TAG algorithm to keep a counter which is appended to each tag τ. Clearly the scheme still hides all information about the secret key. However it is unclear how such a scheme might be used to achieve anonymity. Indeed it is trivial to tell say the 10^{th} tag issued for key k_0 from the 3^{rd} tag issued for different key k_1.

Experiment. $\mathbf{Exp}^{\mathbf{ki\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$
- $k_0, k_1 \leftarrow \mathsf{KG}(1^\lambda)$, $c \leftarrow \{0,1\}$
- *Sample output* $c' \leftarrow \mathsf{A}^{[k_0,k_c]}$.
- *If a tag obtained from the left oracle (namely* TAG_{k_0}*) was verified using the right verification oracle (namely* VRFY_{k_c}*) or vice versa, then output a uniform random bit.*
- *Otherwise if* $c = c'$ *output* 1 *and* 0 *otherwise.*

As usual, in the above experiment we have made a non-triviality constraint; namely that A is not allowed to make a verification query (m, τ) to the right oracle VRFY_{k_c} if τ was obtained from the left oracle TAG_{k_0} for message m (and vice versa).

As before in the following definition we say that an adversary A is a (t, q_t, q_v)-adversary if it runs in time at most t and for *each* pair of oracles with a given key makes at most q_t tag and q_v verification queries. So in total such an adversary can make up to $2q_t$ tag queries namely by making q_t queries to TAG_{k_0} and TAG_{k_c}.

Definition 2 (Key Indistinguishability). *A* MAC *scheme is* (t, q_t, q_v, ϵ)-**ki-cmva** secure *(informally:* key hiding*) if for any* (t, q_t, q_v)*-adversary* A *we have:*

$$\mathbf{Adv}^{\mathbf{ki\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda) := 2\left|\Pr[\mathbf{Exp}^{\mathbf{ki\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda) \rightarrow 1] - \frac{1}{2}\right| \leq \epsilon$$

Moreover if MAC *is* $(t, q_t, 0, \epsilon)$-**ki-cmva** *then we call it* (t, q_t, ϵ)-**ki-cma** secure*. In particular in the* **ki-cma** *experiment we simply omit all verification oracles.*

Multi-key KI Implies Plain KI. A possible extension of the KI notions involves giving the adversary access to n-tuples of pairs of oracles where either each of the pairs have their own states (and keys) or else all pairs share the same state. Indeed such a notion arises quite naturally in the context of a multi-user anonymous protocols as in the real world the adversary observes tags computed under many different states (one for each of the n users). Yet in the ideal, perfectly anonymous world the simulator uses the same state to answer all queries.

It turns out that (just as in the case for multi-message CPA encryption) the "one-key" KI notions defined above already implies such a multi-key variant with only a minimal loss of security. (Indeed this is implicitly proved in [AHM+14b].) As has been argued for CPA security, we view this as a further justification for the format of the KI notion defined above.

Message Hiding. Finally we require the somewhat non-standard security notion for MACs called *message hiding (under chosen message attacks)* [DKPW12] which we denote by **ind-cma**. In that work is is shown how message hiding MACs with (weak) unforgeability properties can be strengthened via a generic transformation. In this work we show that the same transformation preserves any KI properties the MAC may have. The formal definition of message hiding can be found in Appendix A.

3 Constructing Key Indistinguishable MACs

In this section we prove that various known constructions and transformations for MACs achieve KI. These results may be viewed as analogous to [BBDP01] with the difference that we consider the symmetric key MACs instead of public-key encryption. We now provide a more detailed overview of

our results and their relations in Figure 1. The letters "s" and "w" in the figure are used to denote the strong and weak unforgeability variants respectively. The figure consists of three columns. In the first column (AES, DDH, LWE, LPN) we put the underlying cryptographic assumptions upon which security is based. In the second column (HPS, PRF, weak PRF) we put common cryptographic primitives which the MAC constructions use in a black-box manner. In particular they may be implemented using the assumptions which are presented in the first column or any other computational problems. In the third column each box represents a generic MAC scheme characterized by the type of security it provides. Arrows from assumptions and primitives to such a box denote a particular construction. Additionally arrows between the generic MAC schemes represent transformations used to strengthen the security properties of MACs.

In the remainder of this section we detail two constructions (one from a PRF and one from the LPN assumption) and briefly describe three further constructions.

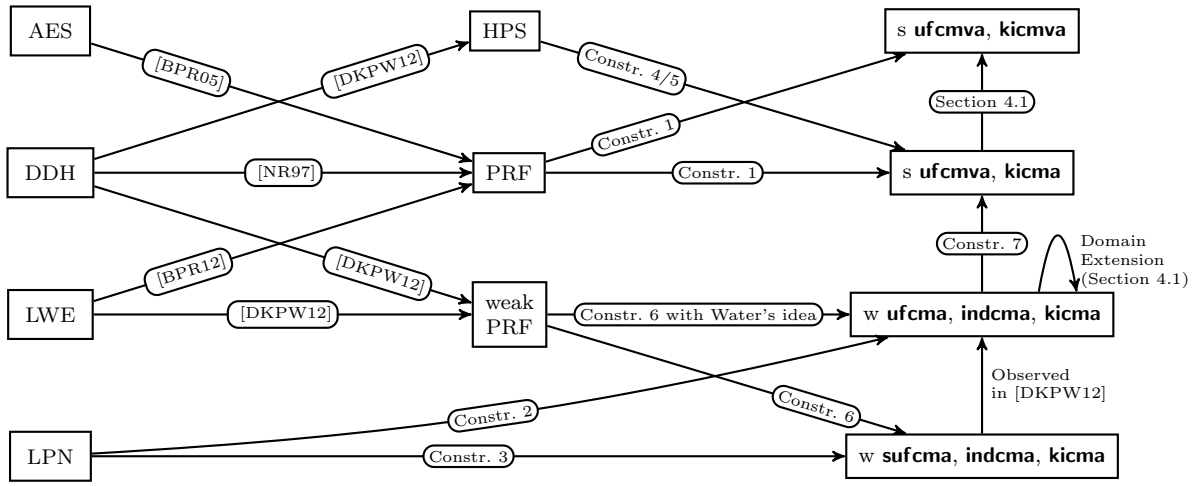

Fig. 1. Obtaining MACs and AA protocols from different assumptions

3.1 From PRFs

PRFs trivially give rise to deterministic MACs (simply by recasting them as the TAG algorithm). However deterministic MACs can not be key indistinguishable (even if they are only weakly universally unforgeable). Thus we now show an alternative construction called $\mathsf{MAC}_{\mathsf{PRF}}$ that is $\{\mathbf{uf}, \mathbf{ki}\}$-**cmva**. It is very efficient in practical terms (requiring a single call to the underlying PRF) while obtaining the strongest forms of unforgeability and key indistinguishability. Thus it represents potentially the most practically relevant of the construction methods of KI MACs detailed in this paper. In particular the PRF can be instantiated based on an block cipher using say CBC-MAC [BCK96b], OMAC [IK03] or ECBC [BPR05] modes of operation or using a compresion function via the HMAC [Bel06, BCK96a] construction. Alternatively, from a theoretical standpoint the PRF can also be based on a variety of well studied number theoretic assumptions such as the DDH family of assumptions [NR97, DY05] or LWE (using PRF from [BPR12]).

Pseudorandom Function Family (PRF) A PRF is a family of functions with the property that the input-output behavior of a random instance of the family is computationally indistinguishable from that of a random function.

Definition 3 (Pseudorandom Functions). *For arbitrary domain $\mathcal{X}$ and range $\mathcal{Y}$ let $\mathcal{R}$ denote the set of functions from $\mathcal{X}$ to $\mathcal{Y}$. Moreover let $\mathsf{PRF} := \{f_k : \mathcal{X} \to \mathcal{Y}\}_{k\in\mathcal{K}}$ be a set of efficiently computable functions indexed by key space $\mathcal{K}$. Then we call PRF a (t, q, ϵ)-*secure PRF *if for any (t, q)-adversary A (running in time at most t making at most q queries) we have:*

$$\mathbf{Adv}^{\mathbf{prf}}_{\mathsf{PRF}}(\mathsf{A}, \lambda) := \left|\Pr_{k\leftarrow\mathcal{K}}\left[\mathsf{A}^{f_k(\cdot)} \to 1\right] - \Pr_{R\leftarrow\mathcal{R}}\left[\mathsf{A}^{R(\cdot)} \to 1\right]\right| \leq \epsilon$$

For security parameter $\lambda \in \mathbb{N}$, let $\mathcal{M} = \mathcal{M}(\lambda)$ be a message space and $\mathcal{X} = \mathcal{X}(\lambda)$ be an arbitrary space such that $|\mathcal{X}| \geq 2^\lambda$. The construction makes use of pseudorandom function $\mathsf{PRF} = \{f_k : \mathcal{M} \times \mathcal{X} \to \mathcal{Y}\}_{k\in\mathcal{K}}$, that is, the domain of PRF is the set $\mathcal{M} \times \mathcal{X}$.

Construction 1 (MAC from PRF: $\mathsf{MAC}_{\mathsf{PRF}}$)

System Parameters: *The key space is $\mathcal{K}$, message space is $\mathcal{M}$ and tag space is $\mathcal{T} = \mathcal{Y} \times \mathcal{X}$.*

Key Generation: *The key generation algorithm $\mathsf{KG}(1^\lambda)$ samples $k \leftarrow \mathcal{K}$ and outputs k as the secret key.*

Tagging: *The tagging algorithm $\mathsf{TAG}_k(m)$ samples $r \leftarrow_R \mathcal{X}$, runs $z = f_k(m, r)$ and returns tag (r, z).*

Verification: *The verification algorithm $\mathsf{VRFY}_k(m, (r, z))$ outputs* `Accept` *if $f_k(m, r) = z$. Otherwise it outputs* `Reject`*.*

We now show that the MAC is $\{\mathbf{uf}, \mathbf{ki}\}$-**cmva** secure.

Theorem 1. *For any $t, q_t, q_v \in \mathbb{N}$, $\epsilon > 0$, if PRF is a $(t, q_t + q_v, \epsilon)$-secure, $(t, 2(q_t + q_v), \epsilon)$-secure and $(t, 2q_t, \epsilon)$-secure then for $t \approx t'$ we have that:*

- $\mathsf{MAC}_{\mathsf{PRF}}$ *has completeness error $\eta = 0$.*
- $\mathsf{MAC}_{\mathsf{PRF}}$ *is strongly $(t', q_t, q_v, \epsilon + \frac{q_v}{|\mathcal{Y}|})$-***uf-cmva** *secure.*
- $\mathsf{MAC}_{\mathsf{PRF}}$ *is $(t', q_t, q_v, 4\epsilon + \frac{4q_t^2}{|\mathcal{X}|} + \frac{2q_v}{|\mathcal{Y}|})$-***ki-cmva** *secure.*
- $\mathsf{MAC}_{\mathsf{PRF}}$ *is $(t', q_t, 4\epsilon + \frac{4q_t^2}{|\mathcal{X}|})$-***ki-cma** *secure respectively.*

Proof. The completeness follows by inspection of the scheme and the fact that all functions in PRF are deterministic.

Strong **uf-cmva** *Security.* To prove this we build a reduction to the security of underlying PRF. Let A be a (t, q_t, q_v)-adversary interacting with $\mathbf{Exp}^{\mathbf{uf\text{-}cmva}}_{\mathsf{MAC}}$. We give a reduction $\mathsf{R}(\mathsf{A})$ whoes advantage in the **prf** experiment implies an upper bound on the advantage of A. The reduction R expects oracle $\mathcal{O}$ and emulates the experiment $\mathbf{Exp}^{\mathbf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$ with the caveat that it uses $\mathcal{O}$ in place of PRF to implement the tag and verification oracles. Finally R outputs 1 if A ever makes a forgery query to the verification oracle; that is a query $(m^*, (r^*, z^*))$ such that $z^* = \mathcal{O}(m^*, r^*)$ and (r^*, z^*) was not obtained in response to a tag oracle query for message m^*. Otherwise R outputs 1. We note that R makes at most $q_t + q_v$ queries to $\mathcal{O}(\cdot)$ in order to simulate $\mathbf{Exp}^{\mathbf{uf\text{-}cmva}}_{\mathsf{MAC}}$ to A. We bound the probability $\Pr[\mathsf{R} \to 1]$ for the two possible types of oracle $\mathcal{O}$.

Case $\mathcal{O} = f$: When $\mathcal{O}$ is a PRF (with random key) R perfectly simulates $\mathbf{Exp}^{\mathsf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$. Therefore: $\Pr[\mathsf{R}^f \to 1] = \mathbf{Adv}^{\mathsf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$.

Case $\mathcal{O} = R$: Suppose $\mathcal{O}$ is a random function R and A makes q forgery attempts for message m^* of the form $(m^*, r^*, z_1), \ldots (m^*, r^*, z_q)$. Then the probability that for some i it holds that $z_i = R(m^*, r^*)$ is $\frac{q}{|\mathcal{Y}|}$. Moreover, if the forgery attempts involve more than one value of (m^*, r^*) then the probability of succesful forgery is even smaller. Thus after q_v verification attempts a forgery has occured with probability at most $\frac{q_v}{|Y|}$. That is: $\Pr[\mathsf{R}^R \to 1] \leq \frac{q_v}{|\mathcal{Y}|}$.

Summing up, we have $\epsilon \geq \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{PRF}}(\mathsf{R}, \lambda) = \left|\Pr[\mathsf{R}^f \to 1] - \Pr[\mathsf{R}^R \to 1]\right| \geq \mathbf{Adv}^{\mathsf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda) - \frac{q_v}{|\mathcal{Y}|}$ or $\mathbf{Adv}^{\mathsf{uf\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda) \leq \epsilon + \frac{q_v}{|\mathcal{Y}|}$.

ki-cmva *and* **ki-cma** *Security.* Recall that the **ki-cmva** game involves two pairs of TAG and VRFY oracles associated with key k_0 and k_1 respectively. We define two experiments closely related to $\mathbf{Exp}^{\mathsf{ki\text{-}cmva}}_{\mathsf{MAC}}$ incrementally replacing the responses to tag and verification queries with responses that would be outputted when PRFs are replaced with random functions (i.e. independent of key for that oracle). As a result we obtain a **ki-cmva**-like experiment where both the pairs of TAG and VRFY oracles are implemented with a pair of random functions instead of a pair of PRFs. Subsequently we introduce another hybrid experiment where the responses to any non-trivial query to any of the verification oracle is immediately replied with `Reject`. This results the final experiment to be same as **ki-cma** experiment where both the TAG oracles are implemented with random functions. We prove the differences of the advantages of the hybrids are negligible and also prove that (unconditionally) the advantage in the **ki-cma** using truly random functions is $\frac{2q_t^2}{|\mathcal{X}|}$.

More precisely, for parameters $\lambda \in \mathbb{N}$ and any (t, q_t, q_v)-adversary A we define the experiment $\mathbf{Exp}_0 := \mathbf{Exp}^{\mathsf{ki\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$. Let experiment $\mathbf{Exp}_1$ be identical to $\mathbf{Exp}_0$ except for that any tag and verification query for the oracles associated key k_0 are responded after replacing the PRF with key k_0 with an random function R_0. Let $\mathbf{Exp}_2$ be identical to $\mathbf{Exp}_1$ except that also tag and verification queries for k_1 are responded after replacing the PRF with key k_1 with an random function R_1. Finally let $\mathbf{Exp}_3$ is identical to $\mathbf{Exp}_2$ except that all the non-trivial verification queries are immediately responded with `Reject` without performing any verification.

For $i \in [0, 3]$ we write $\epsilon_i := \mathbf{Adv}^{\mathbf{Exp}_i}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$ to denote the respective advantages of A at winning these experiments. Bellow we prove that $|\epsilon_0 - \epsilon_1| \leq 2\epsilon$. An almost identical argument will apply for $|\epsilon_1 - \epsilon_2|$. We show that $|\epsilon_2 - \epsilon_3| \leq \frac{2q_v}{|\mathcal{Y}|}$ holds unconditionally. Finally, the proof that $\epsilon_3 \leq \frac{4q_t^2}{|\mathcal{X}|}$ holds implies the result, as $|\epsilon_0 - \epsilon_3| \leq 2\epsilon + \frac{2q_v}{|\mathcal{Y}|}$ implies $\epsilon_0 = 4\epsilon + \frac{4q_t^2}{|\mathcal{X}|} + \frac{2q_v}{|\mathcal{Y}|}$.

Claim 1. $|\epsilon_0 - \epsilon_1| \leq 2\epsilon$.

Proof. Given A we define reduction R interacting with the **prf** experiment with access to oracles $\mathcal{O}$ as follows. Internally it runs $\mathbf{Exp}^{\mathsf{ki\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$ by sampling a random PRF f and then simulating $\mathsf{TAG}_0, \mathsf{VRFY}_0$ using $\mathcal{O}(\cdot)$ and $\mathsf{TAG}_1, \mathsf{VRFY}_1$ using f. Finally if A wins then R outputs 0, otherwise it outputs 1. We note that $\mathcal{O}(\cdot)$ might be queried $2(q_t + q_v)$ times in total when the bit c in **ki-cmva** experiment is chosen to be 0. This is the reason why we require the underlying PRF to be $(t, 2(q_t + q_v), \epsilon)$-secure.

Suppose now that $\mathcal{O} = f_k$ for a random $k \in \mathcal{K}$. Then the view of A is exactly that generated in $\mathbf{Exp}_0$. Therefore it must be that $\Pr[\mathsf{R}^{f_k} \to 0] = \frac{\epsilon_0}{2} + \frac{1}{2}$. On the other hand, if $\mathcal{O} = R$ is a random function then the view of A is identical to $\mathbf{Exp}_1$. This implies that $\Pr[\mathsf{R}^R \to 0] = \frac{\epsilon_1}{2} + \frac{1}{2}$.

Together, it implies $\left|\Pr[\mathsf{R}^{f_k} \to 0] - \Pr[\mathsf{R}^{R} \to 0]\right| = \frac{|\epsilon_0 - \epsilon_1|}{2}$. Due to the security of PRF, it now follows that $\frac{|\epsilon_0 - \epsilon_1|}{2} \leq \epsilon$ or $|\epsilon_0 - \epsilon_1| \leq 2\epsilon$.

□

Claim 2. $|\epsilon_2 - \epsilon_3| \leq \frac{2q_v}{|\mathcal{Y}|}$.

Proof. The only way A will behave differently in $\mathbf{Exp}_2$ and $\mathbf{Exp}_3$ is if she is able to produce a non-trivial query to any of the verification oracles that is accepted. I.e it makes a query $(m, (r, z))$ to a verification oracle using function f such that $f(m, r) = z$. Since both the pairs of oracles in the experiments are implemented with a pair of random functions and A has not seen the output of the random functions at point (m, r) the probability that she can produce the correct z is $\frac{1}{|\mathcal{Y}|}$. Thus via a hybrid argument over all verification queries we have that $|\epsilon_2 - \epsilon_3| \leq \frac{2q_v}{|\mathcal{Y}|}$. □

Claim 3. $\epsilon_3 \leq \frac{4q_t^2}{|\mathcal{X}|}$.

Proof. Our goal is to bound the advantage ϵ_3 of any adversary A for experiment $\mathbf{Exp}_3$; that is the experiment where two random functions R_0 and R_1 are used in place of PRFs for replying TAG queries and all the non-trivial verification queries are responded with immediate `Reject`. We define an event for which we can show that on the one hand if the event does not occur then the adversary has little chance of winning and moreover the event occurs with only a very small probability.

First we observe that in the experiment if the bit c is chosen to be 0 then R_0 is queried up to $2q_t$ times via tag oracles (and R_1 not at all) or, when $c = 1$, then both R_0 and R_1 are queried at most q_t via the respective tag oracles. Now consider calls to R_0 and R_1 (in $\mathbf{Exp}_3$) made through the tag oracles. Each such call has the form (m, r), where A chooses m but r is sampled uniformly at random from $\mathcal{X}$. Two such calls (m, r) and (m', r') are said to *collide* if $(m, r) = (m', r')$. We define the event C to occur when A produces output in $\mathbf{Exp}_3$ and at least one pair of colliding calls was made. Then we see that conditioned on C not occurring the view of A in $\mathbf{Exp}_3$ is independent of bit c which it must guess. Consequently we have $\Pr[\mathbf{Exp}_3 = 1 | \neg C] = \frac{1}{2}$.

It remains to bound $\Pr[C]$. During each of $2q_t$ queries r is chosen independently and uniformly at random. So $\Pr[C]$ is same as the probability that an $r \in \mathcal{X}$ is picked at least twice in these $2q_t$ queries, where there are $|\mathcal{X}|$ possibilities for r. We note that $\Pr[C] \leq \frac{\binom{2q_t}{2}}{|\mathcal{X}|} \leq \frac{2q_t^2}{|\mathcal{X}|}$. Now, we estimate the probability of A in winning $\mathbf{Exp}_3$.

$$\epsilon_3 = 2\left|\Pr[\mathbf{Exp}_3 = 1] - \frac{1}{2}\right| = 2\Big|\underbrace{\Pr[\mathbf{Exp}_3 = 1 \mid C] \cdot \Pr[C]}_{\leq \Pr[C]} + \underbrace{\Pr[\mathbf{Exp}_3 = 1 \mid \neg C] \cdot \Pr[\neg C]}_{= \frac{1}{2}(1 - \Pr[C])} - \frac{1}{2}\Big| \leq \Pr[C]$$

□

The proof for **ki-cma** follows from the above proof for **ki-cmva** where there is no verification oracles throughout and therefore experiments $\mathbf{Exp}_2$ and $\mathbf{Exp}_3$ become identical leading to the removal of the term $\frac{2q_v}{|\mathcal{Y}|}$ from the security parameter of **ki-cma**. □

3.2 From LPN

We now analyze the KI properties of the $\mathsf{MAC}_{\mathsf{LPN}}$ construction based directly on the LPN assumption taken from [DKPW12] where it was shown to be **ind-cma** and wealky **uf-cma** secure. We show that aditionally it is also **ki-cma**. The resulting scheme is the most efficient of the constructions based on number-theoretic assumptions analyzed in this work.

LPN and SLPN* Assumptions Following [Pie12], we briefly recall the LPN assumption defining it as a special case of the SLPN* assumption. Let $\mathtt{U}_n$ be the uniform distribution over $\mathbb{Z}_2^n$, $\mathtt{B}_\tau$ be the Bernoulli distribution with parameter τ and $\mathtt{B}_\tau^n$ be the n-dimensional Bernoulli distribution.[8] For a vector $\mathbf{x} \in \mathbb{Z}_2^n$ we denote by $\mathbf{x}^\mathsf{T}$ the transpose of $\mathbf{x}$. Moreover for a vector $\mathbf{a} \in \mathbb{Z}_2^n$ we denote by $hw(\mathbf{a})$ the hamming weight of $\mathbf{a}$. We write $\mathbf{a} \wedge \mathbf{b}$ for the component wise AND and $\mathbf{a}_{\downarrow \mathbf{b}}$ for the vector obtained from $\mathbf{a}$ by removing all components a_i of $\mathbf{a}$ where $b_i = 0$.

For $\ell \in \mathbb{N}$, $\tau \in (0, \frac{1}{2})$ and $\mathbf{s} \in \mathbb{Z}_2^\ell$ define SLPN* oracle $\Gamma_{\tau,\ell,d}(\mathbf{s}, \cdot)$ to take input vectors $\mathbf{v} \in \mathbb{Z}_2^\ell$ and return $\perp$ if $hw(\mathbf{v}) < d$. Otherwise the oracle samples fresh vector $\mathbf{r}$ according to $\mathtt{U}_\ell$ and bit e according to $\mathtt{B}_\tau$ and outputs $(\mathbf{r}^\mathsf{T}, \mathbf{r}^\mathsf{T}(\mathbf{s} \wedge \mathbf{v}) + e)$[9]. On the other hand the oracle $U_{\ell+1,d}(\cdot)$, on input $\mathbf{v} \in \mathbb{Z}_2^\ell$ outputs $\perp$ if $hw(\mathbf{v}) < d$. Otherwise it outputs a fresh sample from $\mathtt{U}_{\ell+1}$.

For $t, q \in \mathbb{N}$ we call a PPT oracle machine A a (t, q)-adversary if it runs in time at most t making at most q queries and produces binary output.

The $\mathbf{SLPN^*}_{\tau,\ell,d}$ assumption is said to be (t, q, ϵ)-*hard* if for secret $\mathbf{s}$ sampled according to $\mathtt{U}_{\ell+1}$ the distinguishing advantage between oracles $\Gamma_{\tau,\ell,d}$ and $U_{\ell+1,d}$ of any (t, q)-adversaries is at most ϵ. Similarly, the $\mathbf{LPN}_{\tau,\ell}$ assumption is (t, q, ϵ)-hard if no (t, q)-adversary can distinguish oracles $\Gamma_{\tau,\ell,\ell}$ and $U_{\ell+1,\ell}$ with greater then probability ϵ.

Roughly speaking, it was shown by Pietrzak in [Pie12] that the LPN implies the SLPN*.[10]

Lemma 1 ([Pie12]). *If the* $\mathbf{LPN}_{\tau,d}$ *is* (t, q, ϵ)*-hard then for any* $\delta \in \mathbb{N}$ *the* $\mathbf{SLPN^*}_{\tau,\ell,d+\delta}$ *is* (t', q, ϵ')*-hard where:*

$$t' = t - poly(q, \ell) \qquad\qquad \epsilon' = \epsilon + \frac{q}{2^\delta}.$$

We now turn to the second construction from [DKPW12] which was (implicitly) shown to be **ind-cma** and weakly **uf-cma** secure in [KPC+11]. Bellow we prove it to be **ki-cma** secure.

Construction 2 (MAC from LPN: $\mathsf{MAC}_{\mathbf{LPN}}$)

System Parameters: *Parameter* $\tau \in (0, \frac{1}{2})$ *and* $\ell \in \mathbb{N}$ *which control the security quality, and parameters* $\tau' = 1/4 + \tau/2$ *and* $n \in \mathbb{N}$ *which controls the correctness error. Finally parameter* $\alpha \in \mathbb{N}$ *controls the message length. The resulting key space is* $\mathcal{K} = \mathbb{Z}_2^{(\ell+1)\times\alpha}$*, the message space is* $\mathcal{M} = \mathbb{Z}_2^\alpha$ *and the tag space is* $\mathcal{T} = \mathbb{Z}_2^{(\ell+1)n}$*.*

Key Generation: *Algorithm* $\mathsf{KG}(\cdot)$ *samples* $\mathbf{X} \leftarrow \mathbb{Z}_2^{\ell\times\alpha}$ *and* $\bar{\mathbf{x}} \leftarrow \mathbb{Z}_2^\ell$ *both uniformly and outputs secret key* $(\mathbf{X}, \bar{\mathbf{x}})$*.*

Tagging: *For message* $\mathbf{m}$ *and secret key* $\mathbf{s} = (\mathbf{X}, \bar{\mathbf{x}})$ *the algorithm* $\mathsf{TAG}_\mathbf{s}(\mathbf{m})$ *first samples* $\mathbf{R} \leftarrow \mathbb{Z}_2^{\ell\times n}$ *uniformly and* $\mathbf{e}$ *according to* $\mathtt{B}_\tau^n$*. Then it outputs tag* $\sigma = (\mathbf{R}, \mathbf{R}^\mathsf{T} \cdot (\mathbf{X} \cdot \mathbf{m} + \bar{\mathbf{x}}) + \mathbf{e})$*.*

Verification: *To verify tag* $\sigma = (\mathbf{R}, z) \in \mathbb{Z}_2^{\ell\times n} \times \mathbb{Z}_2^n$ *with secret key* $\mathbf{s} = (\mathbf{X}, \bar{\mathbf{x}})$ *the algorithm* $\mathsf{VRFY}_\mathbf{s}(\mathbf{m}, \sigma)$ *outputs* `Accept` *if and only if* $hw(\mathbf{R}^\mathsf{T} \cdot (\mathbf{X} \cdot \mathbf{m} + \bar{\mathbf{x}}) - \mathbf{z}) \leq \tau' n$*.*

Theorem 2. *If* $\mathbf{LPN}_{\tau,\ell}$ *is* (t, q, ϵ)*-hard then* $\mathsf{MAC}_\mathbf{LPN}$ *is* $(t, \frac{q}{2}, 2\epsilon)$-**ki-cma** *secure.*

Proof. At its core the proof relies on a pair of reductions to the $\mathbf{LPN}_{\tau,\ell}$ problem. In a few words the LPN assumption tells us that for a given tagging key (component) $\bar{\mathbf{x}}$ we can replace all terms of the form $\mathbf{R}^T \cdot \bar{\mathbf{x}} + \mathbf{e}$ in all tag queries with fresh uniform random elements from $\mathbb{Z}_2^n$. By doing

[8] That is the distribution over $\mathbb{Z}_2^n$ where each bit is chosen independently according to $\mathtt{B}_\tau$.

[9] The second component is same as $\mathbf{r}^\mathsf{T}_{\downarrow\mathbf{v}}\mathbf{s}_{\downarrow\mathbf{v}} + e$

[10] Actually a stronger result was shown, namely that the sub*space*-LPN assumption (which implies the SLPN*) is implied by the LPN.

this for both keys in the **ki-cma** game we obtain an experiment in which the bit c being guessed by adversary remains information theoretically hidden from her view.

More precisely, just as in the proof of Theorem 4, we define three experiments and show that on the one hand for any fixed adversary their outcomes are computationally indistinguishable and on the other hand all adversaries have no advantage at winning the third game. In fact we define experiments $\mathbf{Exp}_0, \mathbf{Exp}_1$ and $\mathbf{Exp}_2$ exactly as in that proof except that the construction $\mathsf{MAC}_{\mathbf{LPN}}$ is used. For example to answer tag queries for key k_0 in $\mathbf{Exp}_1$ simply returns a fresh uniform sample from $\mathbb{Z}_2^{(\ell+1)n}$ while tag queries for key k_1 are answered using the TAG algorithm of $\mathsf{MAC}_{\mathbf{LPN}}$.

Using the same notations of the previous proof, we observe that $\epsilon_2 = 0$ which holds unconditionally since in $\mathbf{Exp}_2$ the view of any adversary is information theoretically independent of the bit c which it is trying to guess. Thus it remains only to show that $|\epsilon_0 - \epsilon_1|$ and $|\epsilon_1 - \epsilon_2|$ (defined just as before) can be at most ϵ if the $\mathbf{LPN}_{\tau,\ell}$ is (t, q, ϵ)-hard.

Claim $|\epsilon_0 - \epsilon_1| \leq \epsilon$**:** We reduce the the $\mathbf{LPN}_{\tau,\ell}$ assumption with the following reduction R which has access to an oracle $\mathcal{O}$ that is either and LPN oracle or a uniform oracle. The reduction emulates an experiment internally to A and outputs 1 if and only if A wins. The experiment is identical to $\mathbf{Exp}_0$ except for the following:

1. Instead of generating k_0 according to $\mathsf{KG}(1^\lambda)$ it only samples and stores $\mathbf{X} \leftarrow \mathbb{Z}_2^{\ell \times \alpha}$.
2. When a tag query $m \in \mathbb{Z}_2^\alpha$ for key k_0 is made R first obtains n fresh samples $\{(\mathbf{r}_i, b_i) \in \mathbb{Z}_2^{\ell+1}\}_{i \in [n]}$ from $\mathcal{O}$. Let $\mathbf{R} \in \mathbb{Z}_2^{\ell \times n}$ be the matrix whose i^{th} column is $\mathbf{r}_i$ and $\mathbf{b} \in \mathbb{Z}_2^n$ be the vector whose i^{th} bit is b_i. Then R returns the tag $(\mathbf{R}, \mathbf{R}^T \cdot \mathbf{X} \cdot \mathbf{m} + \mathbf{b})$.

We claim that if $\mathcal{O}$ is an $\mathbf{LPN}_{\tau,\ell}$ oracle with secret $\mathbf{x}$ then R has perfectly emulated experiment $\mathbf{Exp}_0$ with key $k_0 = (\mathbf{X}, \mathbf{x})$. This follows from the calculation:

$$\mathbf{R}^T \cdot \mathbf{X} \cdot \mathbf{m} + \mathbf{b} = \mathbf{R}^T \cdot \mathbf{X} \cdot \mathbf{m} + \mathbf{R}^T \cdot \mathbf{x} + \mathbf{e} = \mathbf{R}^T \cdot (\mathbf{X} \cdot \mathbf{m} + \mathbf{x}) + \mathbf{e}$$

which implies that $\Pr[\mathsf{R} \to 1] = \epsilon_0$ for such an oracle.
On the other hand if $\mathcal{O}$ is a uniform oracle then in particular $\mathbf{b}$ is uniformly and independently distributed for each tag query. Thus all values $\mathbf{R}^T \cdot \mathbf{X} \cdot \mathbf{m} + \mathbf{b}$ are also uniformly and independently distributed exactly as in experiment $\mathbf{Exp}_1$. It follows that $\Pr[\mathsf{R} \to 1] = \epsilon_1$ when $\mathcal{O}$ is uniform.
Taken together we can conclude that $|\epsilon_0 - \epsilon_1| \leq \epsilon$. Moreover the reduction makes at most $2q$ queries to $\mathcal{O}$.[11]

Claim $|\epsilon_1 - \epsilon_2| \leq \epsilon$**:** Once again an almost identical argument applied to k_1 to the previous case also proves this claim.

3.3 Further Constructions

We briefly detail three further constructions which we prove KI in the appendix.

From the SLPN Assumption (Appendix C.1).* We also analyze the construction $\mathsf{MAC}_{\mathbf{SLPN^*}}$ based on the SLPN* assumption of [DKPW12] which was (implicitly) shown to be **ind-cma** and weakly **suf-cma** secure in [KPC+11]. In particular in Appendix C.1 we prove that it is also **ki-cma** secure based on the related Subset LPN (SLPN*) assumption which can be efficiently reduced to the more

[11] This occurs in the case when A makes q queries to both left and right oracle and $c = 0$ in the emulated $\mathbf{Exp}_1$ experiment

standard LPN assumption as shown by Pietrzak in [Pie12]. The two LPN based schemes $\mathsf{MAC}_{\mathbf{LPN}}$ and $\mathsf{MAC}_{\mathbf{SLPN}^*}$ are somewhat incomparable. On the one hand $\mathsf{MAC}_{\mathbf{LPN}}$ provides a stronger unforgeability property and comes with a tighter reduction to the LPN assumption. On the other hand $\mathsf{MAC}_{\mathbf{SLPN}^*}$ enjoys a smaller key size (though less efficient tagging operation) and can be instantiated with a comparatively smaller value of the security parameter to achieve the same level of security.

From Hash Proof Systems (Appendix C.2). In Appendix C.2 we show that the MAC construction $\mathsf{MAC}_{\mathsf{HPS}}$ given in [DKPW12] based on any labeled hash proof systems HPS is also **ki-cma** secure. The scheme has been shown to be weakly **uf-cmva** secure. Using a slightly stronger notion of a HPS, the proof for weak **uf-cmva** goes through unchanged resulting in the same parameters for strong **uf-cmva** security. This scheme provides the most efficient KI MAC in this work based on DDH assumption.

From Key-Homomorphic Weak PRFs (Appendix C.3). Next, in Appendix C.3 we show that the MAC construction $\mathsf{MAC}_{\mathsf{khwPRF}}$ of [DKPW12] based on any key-homomorphic weak PRF is **ki-cma** secure. The scheme has been proven to be weakly **suf-cma** secure. In [DKPW12] an extension of the wPRF construction is provided which makes use of Waters' argument [Wat05] to achieve weak **uf-cma** (and **ind-cma**) security at the cost of a somewhat less efficient scheme. We also observe that the modified scheme is **ki-cma** secure following essentially same argument that we use for the former wPRF based construction. As observed in [DKPW12] both DDH assumption and the LWE assumptions can (for example) be used to directly instantiate efficient key-homomorphic wPRF families.

4 Transformations for Strengthening MACs

We describe several transformations for strengthening the security properties of a MAC. All but one of the transformations were originally described in [DKPW12]. In this work we show that not only do they achieve their intended goal of producing MACs with better unforgeability properties but they also preserve the underlying KI property. Moreover we show that the most important such transformation even achieves a stronger unforgeability notion (namely *strong* **uf-cmva**) then originally claimed.

4.1 Adding Support for Verification Queries

We show how to add security in the face of verification queries while preserving KI by giving a full path from any weakly $\{\mathbf{uf}, \mathbf{ind}, \mathbf{ki}\}$-**cma** secure MAC to a strongly $\{\mathbf{uf}, \mathbf{ki}\}$-**cmva** secure one.

Verification Queries for Unforgeability. In [DKPW12] the authors present a very efficient transformation (detailed in Construction 7) which they show maps any weakly $\{\mathbf{uf}, \mathbf{ind}\}$-**cma** secure scheme MAC to a weakly **uf-cmva** secure scheme $\overline{\mathsf{MAC}}$. We show that the transformation achieves more namely the resulting MAC is even *strongly* **uf-cmva** secure. Indeed the original proof goes through unchanged also for the the stronger statement. We further show that the same transformation also preserves the **ki-cma** security of the underlying MAC.

Very briefly, the proof relies on a pair of reductions to **ki-cma** and **ind-cma** security of MAC. The keys for $\overline{\mathsf{MAC}}$ consist of a key k for MAC and h for a pairwise independent hash function. Recall that the experiment $\mathbf{Exp} = \mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\overline{\mathsf{MAC}}}$ involves distinguishing a setting where both oracles are

keyed with (k_0, h_0) from one where the oracles are keyed with (k_0, h_0) and (k_1, h_1). In the proof we define an experiment $\mathbf{Exp}_0$ where the oracles in the second setting are instead keyed with (k, h_0) and (k, h_1) and an experiment $\mathbf{Exp}_1$ where those oracles are keyed with (k_0, h) and (k_1, h). Then a standard hybrid argument indicates that an adversary A which can win in $\mathbf{Exp}$ is at least half as good at winning either $\mathbf{Exp}_1$ or $\mathbf{Exp}_2$. In the first case we show how to use A to break the **ind-cma** property of MAC while in the second case we instead break its **ki-cma** security. A detailed description can be found in Appendix D.1.

Verification Queries for KI. The above result shows that for certain MACs we can strengthen the type of unforgeability supported while preserving the KI property. Next we show that any (strong) **uf-cmva** and **ki-cma** secure MAC is also **ki-cmva** secure. Together these results provide a full path from any weakly $\{\mathbf{uf}, \mathbf{ind}, \mathbf{ki}\}$-**cma** secure MAC to a strongly $\{\mathbf{uf}, \mathbf{ki}\}$-**cmva** secure one.

Theorem 3 (uf-cmva + ki-cma $\implies$ ki-cmva). *For any $t, q_t, q_v \in \mathbb{N}$ and $\epsilon_1, \epsilon_2, \eta > 0$, if* MAC *is:*
- $(t, q_t, q_v, \epsilon_1)$-**uf-cmva** *(strongly existentially unforgeable with verification queries)*
- (t, q_t, ϵ_2)-**ki-cma** *(key indistinguishable)*
- *and has completeness error η*

then MAC *is $(t', q_t, q_v, 4\epsilon_1 + \epsilon_2 + 4\min(q_t, q_v)\eta)$-**ki-cmva** secure where $t' \approx t$.*

Proof. The proof formalizes the insight that an adversary playing the **ki-cmva** game for MAC can essentially simulate all verification queries by rejecting all but the trivial queries since otherwise it has created a forgery. Thus any **ki-cmva** adversary can be used either as a (strong) forger or as a **ki-cma** adversary.

More formally let A be an **ki-cmva** adversary and let B behave just as A except that each of its trivial verification queries[12] are automatically get response `Accept` rather then forwarding to the appropriate oracles. Then the views of A and B only differ if A makes a trivial query of the form (m, τ) which is rejected. For each new trivial query this happens with probability at most η (i.e. the completeness error of MAC). Thus using a hybrid argument over all such distinct queries made by A (of which there can be at most $2\min(q_t, q_v)$) a similar calculation to the one below implies that for security parameter $\lambda \in \mathbb{N}$ we have that:

$$\mathbf{Adv}^{\mathbf{ki\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{A}, \lambda) \leq \mathbf{Adv}^{\mathbf{ki\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{B}, \lambda) + 4\min(q_v, q_t)\eta.$$

Let experiment $\mathbf{Exp}_0 = \mathbf{Exp}^{\mathbf{ki\text{-}cmva}}_{\mathsf{MAC}}(\mathsf{B}, \lambda)$ and let $\mathbf{Exp}_1$ be identical to $\mathbf{Exp}_0$ except that all the non-trivial verification queries are answered with `Reject`. We define $\delta_i := \mathbf{Adv}^{\mathbf{Exp}_i}_{\mathsf{MAC}}(\mathsf{B}, \lambda)$ to be the advantages of B in the two experiments. Observe that the view of B in $\mathbf{Exp}_1$ is essentially that of $\mathbf{Exp}_2 = \mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\mathsf{MAC}}(\mathsf{B}, \lambda)$ since it can trivially simulate all verification queries (the trivial queries by `Accept` and `Reject` otherwise). Moreover winning $\mathbf{Exp}_1$ is no different then winning $\mathbf{Exp}_2$. Thus by assumption $\delta_1 \leq \epsilon_2$. So it remains only to show that $|\delta_0 - \delta_1| \leq 4\epsilon_1$.

Let E be the event that B makes a (non-trivial) verification query to one of the VRFY oracles in $\mathbf{Exp}_0$. Then $\Pr[E] \leq 2\epsilon_1$ since otherwise B can be used to break the **uf-cmva** security of MAC. The reduction needs to use its TAG_k and VRFY_k oracles from the **uf-cmva** game in place of either the key k_0 or k_1 oracles (with equal probability) in $\mathbf{Exp}_0(\mathsf{B}, \lambda)$. If E does occur then with probability $\frac{1}{2}$ it will happen for the verification oracle VRFY_k implying the reduction has produced a forgery.

[12] Recall that a query (m, τ) to verification oracle with key k is called non-trivial if and only if τ was not obtained as a response to tag query m for key k.

Now conditioned on E *not* occurring the view of B in both experiments is identical. More precisely $\Pr[\mathbf{Exp}_0 = 1|\neg E] = \Pr[\mathbf{Exp}_1 = 1]$. Thus we can write:

$$\begin{aligned}
|\delta_0 - \delta_1| &= \left|2\left|\Pr[\mathbf{Exp}_0 = 1] - \frac{1}{2}\right| - 2\left|\Pr[\mathbf{Exp}_1 = 1] - \frac{1}{2}\right|\right| = |2\left(\Pr[\mathbf{Exp}_0 = 1] - \Pr[\mathbf{Exp}_1 = 1]\right)| \\
&= |2\left(\Pr[\mathbf{Exp}_0 = 1|E]\Pr[E] + \Pr[\mathbf{Exp}_0 = 1|\neg E](1 - \Pr[E]) - \Pr[\mathbf{Exp}_1 = 1]\right)| \\
&= 2\Pr[E]\,|\Pr[\mathbf{Exp}_0 = 1|E] - \Pr[\mathbf{Exp}_0 = 1|\neg E]| \\
&= 2\Pr[E] \leq 4\epsilon_1
\end{aligned}$$

□

KI Preserving Domain Extension Recall that (for both weak and strong variants) an **suf-cma** secure MAC is also **uf-cma** secure albiet at a cost of degrading security by a multiplicative factor of $2^\mu = |\mathcal{M}|$; the size of the message space. In order to keep this 2^μ factor small, we start with (t, q_t, ϵ)-**suf-cma** MAC with very small message space and then after recasting it as **uf-cma** secure scheme we can apply the domain extension transformation of [DKPW12] to grow the message space. In Appendix D.2 we show that the transformation also preserves the KI of the original scheme as long as the original MAC is also **ind-cma** secure (though not necessarily unforgeable in any sense).

References

[AHM+14a] J. Alwen, M. Hirt, U. Maurer, A. Patra, and P. Raykov. Anonymous authentication with shared secrets. Cryptology ePrint Archive, Report 2014/073, 2014. `http://eprint.iacr.org/`.

[AHM+14b] J. Alwen, M. Hirt, U. Maurer, A. Patra, and P. Raykov. Anonymous authentication with shared secrets. Cryptology ePrint Archive, to Appear, 2014.

[AMR+12] M. Arapinis, L. I. Mancini, E. Ritter, M. Ryan, N. Golde, K. Redon, and R. Borgaonkar. New privacy issues in mobile telephony: fix and verification. In T. Yu, G. Danezis, and V. D. Gligor, editors, *ACM Conference on Computer and Communications Security*, pages 205–216. ACM, 2012.

[AMRR11] M. Arapinis, L. I. Mancini, E. Ritter, and M. Ryan. Formal analysis of umts privacy. *CoRR*, abs/1109.2066, 2011.

[BBDP01] M. Bellare, A. Boldyreva, A. Desai, and D. Pointcheval. Key-privacy in public-key encryption. In *ASIACRYPT*, pages 566–582, 2001.

[BCK96a] M. Bellare, R. Canetti, and H. Krawczyk. Keying hash functions for message authentication. In *CRYPTO*, pages 1–15, 1996.

[BCK96b] M. Bellare, R. Canetti, and H. Krawczyk. Pseudorandom functions revisited: The cascade construction and its concrete security. In *FOCS*, pages 514–523, 1996.

[Bel06] M. Bellare. New proofs for NMAC and MAC. In *CRYPTO*, pages 602–619, 2006.

[BLdMT09] M. Burmester, T. V. Le, B. de Medeiros, and G. Tsudik. Universally composable RFID identification and authentication protocols. *ACM Trans. Inf. Syst. Secur.*, 12(4), 2009.

[BM11] M. Burmester and J. Munilla. Lightweight RFID authentication with forward and backward security. *ACM Trans. Inf. Syst. Secur.*, 14(1):11, 2011.

[BPR05] M. Bellare, K. Pietrzak, and P. Rogaway. Improved security analyses for CBC MACs. In *CRYPTO*, pages 527–545, 2005.

[BPR12] A. Banerjee, C. Peikert, and A. Rosen. Pseudorandom functions and lattices. In *EUROCRYPT*, pages 719–737, 2012.

[CS02] R. Cramer and V. Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In *EUROCRYPT*, pages 45–64, 2002.

[DKPW12] Y. Dodis, E. Kiltz, K. Pietrzak, and D. Wichs. Message authentication, revisited. In *EUROCRYPT*, pages 355–374, 2012.

[DLYZ11] R. H. Deng, Y. Li, M. Yung, and Y. Zhao. A zero-knowledge based framework for RFID privacy. *Journal of Computer Security*, 19(6):1109–1146, 2011.

[DS09] Y. Dodis and J. P. Steinberger. Message authentication codes from unpredictable block ciphers. In S. Halevi, editor, *CRYPTO*, volume 5677 of *Lecture Notes in Computer Science*, pages 267–285. Springer, 2009.

[DY05] Y. Dodis and A. Yampolskiy. A verifiable random function with short proofs and keys. In *Public Key Cryptography*, pages 416–431, 2005.

[HPVP11] J. Hermans, A. Pashalidis, F. Vercauteren, and B. Preneel. A new RFID privacy model. In *ESORICS*, pages 568–587, 2011.

[IK03] T. Iwata and K. Kurosawa. Omac: One-key cbc mac. In T. Johansson, editor, *FSE*, volume 2887 of *Lecture Notes in Computer Science*, pages 129–153. Springer, 2003.

[KMO+13] M. Kohlweiss, U. Maurer, C. Onete, B. Tackmann, and D. Venturi. Anonymity-preserving public-key encryption: A constructive approach. In E. D. Cristofaro and M. Wright, editors, *Privacy Enhancing Technologies*, volume 7981 of *Lecture Notes in Computer Science*, pages 19–39. Springer, 2013.

[KPC+11] E. Kiltz, K. Pietrzak, D. Cash, A. Jain, and D. Venturi. Efficient authentication from hard learning problems. In K. G. Paterson, editor, *EUROCRYPT*, volume 6632 of *Lecture Notes in Computer Science*, pages 7–26. Springer, 2011.

[LSWW13] M.-F. Lee, N. P. Smart, B. Warinschi, and G. Watson. Anonymity guarantees of the umts/lte authentication and connection protocol. Cryptology ePrint Archive, Report 2013/027, 2013. `http://eprint.iacr.org/`.

[Nat] National Institute of Standards and Technology, U.S. Department of Commerce, M Dworkin. Recommendation for block cipher modes of operation: the CMAC mode for authentication. NIST Special Publication 800-38B.

[NR97] M. Naor and O. Reingold. Number-theoretic constructions of efficient pseudo-random functions. In *FOCS*, pages 458–467, 1997.

[Pie12] K. Pietrzak. Subspace LWE. In R. Cramer, editor, *TCC*, volume 7194 of *Lecture Notes in Computer Science*, pages 548–563. Springer, 2012.

[rGPP12] 3rd Generation Partnership Project. Ts 33.102 - 3g security; Security architecture v11.5.0, 2012.

[TM12] J.-K. Tsay and S. F. Mjølsnes. A vulnerability in the umts and lte authentication and key agreement protocols. In I. V. Kotenko and V. A. Skormin, editors, *MMM-ACNS*, volume 7531 of *Lecture Notes in Computer Science*, pages 65–76. Springer, 2012.

[Vau10] S. Vaudenay. Privacy models for RFID schemes. In *RFIDSec*, page 65, 2010.

[Wat05] B. Waters. Efficient identity-based encryption without random oracles. In R. Cramer, editor, *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127. Springer, 2005.

A More Security Notions for MAC

Message Hiding. A message hiding MAC (defined in [DKPW12]) has the property that the tags leak no information about the message, making the tags for a message indistinguishable from the tags for a fixed message, say 0. Such a notion can only be achieved by *randomized* MACs as reasoned below [DKPW12].

Definition 4 (Message Hiding). *A message authentication scheme* MAC *is called* (t, q_t, ϵ)-**ind-cma** secure *(informally:* message hiding*) if no* (t, q_t)*-adversary* A *(i.e. running in time at most* t *making at most* q_t *queries) can distinguish tags for chosen messages from tags for a fixed message, say* 0 *i.e.*

$$\mathbf{Adv}^{\mathsf{ind\text{-}cma}}_{\mathsf{MAC}}(\mathsf{A}, \lambda) := \left| \Pr_{k \leftarrow \mathsf{KG}(1^\lambda)}[\mathsf{A}^{\mathsf{TAG}_k(\cdot)}(1^\lambda) \to 1] - \Pr_{k \leftarrow \mathsf{KG}(1^\lambda)}[\mathsf{A}^{\mathsf{TAG}_k(0)}(1^\lambda) \to 1] \right| \leq \epsilon.$$

The probability is taken over the coins of algorithms KG *and* TAG *as well as the adversary* A*.*

Here $\mathsf{TAG}_k(0)$ is an oracle that ignores its input and outputs a tag for some fixed message 0 using key k. A deterministic MAC can not be **ind-cma** secure since the adversary A can trivially distinguish the oracles $\mathsf{TAG}_k(\cdot)$ and $\mathsf{TAG}_k(0)$ by making queries on two different message $m \neq m'$ and checking if the returned tags are identical, which will be the case if the oracle implements $\mathsf{TAG}_k(0)$.

B Universal and Pairwise Independent Hash Functions

The transformations for MACs employ universal and pairwise independent hash functions. We define them below.

Definition 5 (Almost Universal Hash Function). *A keyed hash function $h : \{0,1\}^\ell \times \{0,1\}^m \to \{0,1\}^n$ is δ-almost universal if any two distinct inputs collide with probability at most δ over the choices of the random key, i.e., for all $x \neq x' \in \{0,1\}^m$*

$$\Pr_{k \leftarrow_R \{0,1\}^\ell}[h_k(x) = h_k(x')] \leq \delta.$$

Definition 6 (Pairwise Independent Hash Function). *A keyed hash function $h : \{0,1\}^\ell \times \{0,1\}^m \to \{0,1\}^n$ is pairwise independent if it behaves like a uniformly random function on any two inputs, i.e., for all $x \neq x' \in \{0,1\}^m$ and $y, y' \in \{0,1\}^n$*

$$\Pr_{k \leftarrow_R \{0,1\}^\ell}[h_k(x) = y \wedge h_k(x') = y'] = 2^{-2n}.$$

A pairwise independent hash function as above is 2^{-n}-universal.

Proposition 1. *There exists a 2^{-n+1}-universal hash function as above with key length $\ell = 4(n + \log m)$. There exists a pairwise independent hash function with key length $\ell = 2\max\{m, n\}$.*

C Constructing key indistinguishable MACs

We discuss further KI MAC constructions in detail.

C.1 Weakly suf-cma/uf-cma and ki-cma MACs from SLPN*

In this section we show that the MAC constructions based on the Subspace Learning Parity with Noise (SLPN*) assumption of [DKPW12] are also **ki-cma**.

We briefly recall the first construction of [DKPW12] which was (implicitly) shown to be **ind-cma** and weakly **suf-cma** secure in [KPC+11].

Construction 3 (MAC from SLPN*: $\mathsf{MAC}_{\mathsf{SLPN}^*}$)

System Parameters: *Parameter $\tau \in (0, \frac{1}{2})$ and $\ell \in \mathbb{N}$ control the security quality, and parameters $\tau' = 1/4 + \tau/2$ and $n \in \mathbb{N}$ controls the correctness error. The resulting key space is $\mathcal{K} = \mathbb{Z}_2^{2\ell}$, the message space is $\mathcal{M} = \{\mathbf{m} \in \mathbb{Z}_2^{2\ell} : hw(\mathbf{m}) = \ell\}$ and the tag space is $\mathcal{T} = \mathbb{Z}_2^{(\ell+1)n}$.*

Key Generation: *Algorithm $\mathsf{KG}(\cdot)$ samples secret key $\mathbf{s} \in \mathbb{Z}_2^{2\ell}$ according to $\mathtt{U}_{2\ell}$.*

Tagging: *For message $\mathbf{m}$ the algorithm $\mathsf{TAG}_\mathbf{s}(\mathbf{m})$ first samples $\mathbf{R} \leftarrow \mathbb{Z}_2^{\ell \times n}$ uniformly and $\mathbf{e}$ according to $\mathtt{B}_\tau^n$. Then it outputs tag $\sigma = (\mathbf{R}, \mathbf{R}^\mathsf{T} \cdot \mathbf{s}_{\downarrow \mathbf{m}} + \mathbf{e})$.*

Verification: *To verify tag $\sigma = (\mathbf{R}, z) \in \mathbb{Z}_2^{\ell \times n} \times \mathbb{Z}_2^n$ the algorithm $\mathsf{VRFY}_\mathbf{s}(\mathbf{m}, \sigma)$ outputs* `Accept` *if and only if $hw(\mathbf{R}^\mathsf{T} \cdot \mathbf{s}_{\downarrow \mathbf{m}} - \mathbf{z}) \leq \tau' n$.*

Theorem 4. *If $\mathsf{SLPN}^*_{\tau, 2\ell, \ell}$ is (t, q, ϵ)-hard then $\mathsf{MAC}_{\mathsf{SLPN}^*}$ is $(t, \frac{q}{2}, 2\epsilon)$-**ki-cma** secure.*

Proof. On the highest level the proof has the following structure. We define a pair of experiments which are slight alterations of the **ki-cma** experiment for the above construction. Then we give a pair of reductions showing how an adversary behaving significantly different in any of these experiments can be used to break the **SLPN*** assumption.

Fix any (t, q)-adversary A and security parameter $\lambda \in \mathbb{N}$. For the sake of legibility we define the experiment $\mathbf{Exp}_0 := \mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\mathsf{MAC}_1}(\mathsf{A}, \lambda)$. Moreover we define experiment $\mathbf{Exp}_1$ to be identical to $\mathbf{Exp}_0$ except any query to the TAG_{k_0} oracle is answered with a fresh uniform sample from $\mathbb{Z}_2^{(\ell+1)\times n}$.[13] Finally experiment $\mathbf{Exp}_2$ is defined just as $\mathbf{Exp}_1$ except that any query to TAG_{k_1} is also answered with a fresh uniform sample from $\mathbb{Z}_2^{(\ell+1)\times n}$.

For $i \in [0, 2]$ we write $\epsilon_i := \mathbf{Adv}^{\mathbf{Exp}_i}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$ to denote the respective advantages of A at winning these experiments. We observe that the value of bit c is information theoretically hidden from A in $\mathbf{Exp}_2$ and so unconditionally $\epsilon_2 = 0$. It remains only to show that $|\epsilon_0 - \epsilon_1|$ and $|\epsilon_1 - \epsilon_2|$ can be at most ϵ if the **SLPN***$_{\tau,\ell,d}$ is (t, q, ϵ)-hard.

Claim $|\epsilon_0 - \epsilon_1| \leq \epsilon$**:** We construct a reduction R which acts as a (t, q)-adversary breaking the **SLPN***$_{\tau,\ell,d}$ with probability ϵ as follows. Recall that R is given access to an oracle $\mathcal{O}$ for which it must decide if it is a uniform oracle or an **SLPN*** oracle. To do this the reduction emulates an execution of $\mathbf{Exp}_0$ to A except that whenever A makes a query $\mathbf{m} \in \mathbb{Z}_2^{2\ell}$ to TAG_{k_0} then R responds with a fresh sample obtained from $\mathcal{O}(\mathbf{m})$. Finally if A guesses the value of bit c in the emulated **Exp** correctly then R outputs 1. Otherwise it outputs 0.

We observe that if $\mathcal{O} = \Gamma_{\tau,\ell,d}(\mathbf{x}, \cdot)$ for some secret $\mathbf{x} \in \mathbb{Z}_2^{\ell}$ then R emulates $\mathbf{Exp}_0$ perfectly (with $k_0 := \mathbf{x}$) implying that $\Pr[\mathsf{R} \to 1] = \epsilon_0$. On the other hand, if $\mathcal{O} = U_{\ell+1,d}(\cdot)$ then R emulates $\mathbf{Exp}_1$ perfectly which implies $\Pr[\mathsf{R} \to 1] = \epsilon_1$.

Thus the distinguishing advantage of R is $|\epsilon_0 - \epsilon_1|$ which is at most ϵ. Moreover, as R runs A a single time and has essentially no overhead. R runs in time at most t making at most $2q$ queries to $\mathcal{O}$ (in the case when A makes q queries to both left and right oracle and $c = 0$ and both of those oracles are keyed with k_0).

Claim $|\epsilon_1 - \epsilon_2| \leq \epsilon$**:** An essentially identical argument as in the previous case (but for key k_1 and experiments $\mathbf{Exp}_1$ and $\mathbf{Exp}_2$) implies the result.

C.2 uf-cmva and ki-cma MACs from Hash Proof Systems

As stated we show that the construction given in [DKPW12] based on labeled hash proof systems (HPS) is key indistinguishable. In what follows, we recall the necessary details for labeled HPS and the construction for MAC from [DKPW12].

Labeled HPS We present the framework of HPS, introduced by Cramer and Shoup [CS02]. For simplicity we frame the description by viewing HPS as key-encapsulation mechanisms (KEM). A KEM is a public-key encryption scheme that is used for encrypting random messages that are used as encryption keys for a symmetric-key encryption scheme, which in turn encrypts the actual plaintext. A HPS can be viewed as a labeled KEM in which ciphertexts can be generated in two modes. The ciphertexts that are generated using the first mode are referred to as *valid* ciphertexts. For such ciphertexts the encapsulated key is well defined, and can be decapsulated using the secret key and also using the public key along with the "witness" of the ciphertext validity. The ciphertexts that

[13] As apposed to using the tagging algorithm of MAC_1 as in $\mathbf{Exp}_0$.

are generated using the second mode are referred to as *invalid* ciphertexts and essentially contain no information on the encapsulated key. That is, given a public key and an invalid ciphertext, the distribution of the encapsulated key (as it will be produced by the decapsulation process using the secret key) is almost uniform. This is achieved by introducing redundancy into the secret key: each public key has many corresponding secret keys. It might not be even possible to decapsulate the key using the public key. The only computational requirement is that the two modes are computational indistinguishable: any efficient adversary cannot distinguish with a noticeable advantage between valid ciphertexts and invalid ciphertexts.

Let $\mathcal{C}$ be the set of all ciphertexts, $\mathcal{V} \subset \mathcal{C}$ be the set of all *valid* ciphertexts, $\mathcal{K}$ be the set of all symmetric keys, $\mathcal{L}$ be the set of labels, $\mathcal{PK}$ be the set of all public keys and $\mathcal{SK}$ be the set of all secret keys. We assume that there are efficient algorithms for sampling $sk \in \mathcal{SK}$, $\ell \in \mathcal{L}$ $c \in \mathcal{V}$ together with a witness w, and $c \in \mathcal{C} \setminus \mathcal{V}$. Let $\Lambda_{sk}^{\ell} : \mathcal{C} \times \mathcal{L} \to \mathcal{K}$ be a labeled hash function indexed with $sk \in \mathcal{SK}$ and $\ell \in \mathcal{L}$ that maps ciphertexts in $\mathcal{C}$ to symmetric keys in $\mathcal{K}$. A hash function Λ_{sk}^{ℓ} is *projective* if there exists a projection $\mu : \mathcal{SK} \to \mathcal{PK}$ such that $\mu(sk) \in \mathcal{PK}$ defines the action of Λ_{sk}^{ℓ} over the subset $\mathcal{V}$. That is, for every $c \in \mathcal{V}$, the value $k = \Lambda_{sk}^{\ell}(c)$ is uniquely determined by $pk = \mu(sk)$ and c. In other words, even though there are many different secret keys sk corresponding to the same public key pk, the action of Λ_{sk}^{ℓ} over the subset of valid ciphertexts in completely determined by the public key pk. In contrast, the action of Λ_{sk}^{ℓ} over the subset of invalid ciphertexts should be completely undetermined and it might not be possible to compute Λ_{sk}^{ℓ} from pk and $c \in \mathcal{C} \setminus \mathcal{V}$. A projective hash function is *2-universal* if for all $c, c^* \in \mathcal{C} \setminus \mathcal{V}$, $\ell, \ell^* \in \mathcal{L}$ with $(c, \ell) \neq (c^*, \ell^*)$

$$SD\left((pk, \Lambda_{sk}^{\ell^*}(c^*), \Lambda_{sk}^{\ell}(c)), (pk, k, \Lambda_{sk}^{\ell}(c))\right) = 0$$

where $sk \leftarrow_R \mathcal{SK}$ and $k \leftarrow_R \mathcal{K}$ and $pk = \mu(sk)$.[14] A projective hash function is *extracting* if any ciphertext can be mapped to a symmetric key under any secret key, i.e., for all $c \in \mathcal{C}$, $\ell \in \mathcal{L}$, $sk \in \mathcal{SK}$ and some $k \in \mathcal{K}$

$$\Lambda_{sk}^{\ell}(c) = k.$$

A labeled hash proof system $\mathsf{HPS} = (\mathsf{Params}, \mathsf{Pub}, \mathsf{Priv})$ consists of three algorithms. The randomized algorithm $\mathsf{Params}(1^{\lambda})$ generates parameterized instances of the form $\left(group, \mathcal{K}, \mathcal{C}, \mathcal{V}, \mathcal{L}, \mathcal{PK}, \mathcal{SK}, \Lambda_{(\cdot)}^{(\cdot)}, \mu\right)$, where *group* may contain some additional structural parameters and $\Lambda_{(\cdot)}^{(\cdot)}, \mu$ are efficiently computable functions. The deterministic public evaluation algorithm Pub is used to decapsulate valid ciphertexts $c \in \mathcal{V}$ given a witness w of the fact that c is indeed valid (one can think of w as the random coins used to sample c from $\mathcal{V}$). That is, Pub receives a public key $pk = \mu(sk)$, ciphertext $c \in \mathcal{V}$, the witness w and a label $\ell \in \mathcal{L}$ as input and returns the value $\Lambda_{sk}^{\ell}(c)$. The deterministic private evaluation algorithm Priv is used to decapsulate valid ciphertexts without knowing a witness w, but by using the secret key sk. That is, Priv receives a secret key sk, ciphertext $c \in \mathcal{C}$ and a label $\ell \in \mathcal{L}$ as input and returns the value $\Lambda_{sk}^{\ell}(c)$. The labeled hash proof system is said to be *2-universal* and *extracting* if the underlying function Λ is so over the outcomes of Params. We require that the *subset membership problem* is hard in HPS, which means that for random valid ciphertext $c_0 \in \mathcal{V}$ and random invalid ciphertext $c_1 \in \mathcal{C} \setminus \mathcal{V}$, the two ciphertexts c_0 and c_1 are computationally indistinguishable. For formally, the subset membership problem is said to be (ϵ, t)-hard in HPS if

[14] We note that this definition is slightly stronger then that of [DKPW12] since we require the equality to hold also for the case when $\ell = \ell^*$ but $c \neq c^*$. Fortunately the DDH based labeled HPS in that paper also satisfies this stronger definition.

for all adversaries A that runs in time t,

$$\mathbf{Adv}^{\mathbf{sm}}_{\mathsf{HPS}}(\mathsf{A},\lambda) = \left|\Pr_{c_0 \leftarrow_R \mathcal{V}}[\mathsf{A}(\mathcal{C},\mathcal{V},c_0) \to 1] - \Pr_{c_1 \leftarrow_R \mathcal{C}\setminus\mathcal{V}}[\mathsf{A}(\mathcal{C},\mathcal{V},c_1) \to 1]\right| \leq \epsilon.$$

Construction 4 (MAC from HPS: $\mathsf{MAC}_{\mathsf{HPS}}$)

System Parameters: *The key space is $\bar{\mathcal{K}} = \mathcal{SK}$, message space is $\mathcal{M} = \mathcal{L}$ and tag space is $\mathcal{T} = \mathcal{V} \times \mathcal{K}$.*

Key Generation: *The key generation algorithm $\mathsf{KG}(1^\lambda)$ samples $sk \leftarrow \mathcal{SK}$ and outputs secret key sk.*

Tagging: *On input message $m \in \mathcal{L}$ the tagging algorithm TAG_{sk} samples a uniform $c \leftarrow \mathcal{V}$ and outputs tag $(c, \Lambda^m_{sk}(c))$.*

Verification: *On input message $m \in \mathcal{L}$ and tag (c,d) the verification algorithm VRFY_{sk} outputs* `Accept` *if and only if $\Lambda^m_{sk}(c) = d$.*

Note that the construction does not use algorithm Pub of HPS. The scheme has been shown to be weakly $(t, q_t, q_v, q_t\epsilon + \frac{2q_tq_v+q_t+q_v)}{|\mathcal{K}|})$-**uf-cmva** secure given that HPS is extracting, 2-universal and that the subset membership is (t,ϵ)-hard [DKPW12]. In fact, using the slightly strong notion of 2-universality in this paper, the proof goes through unchanged resulting in the same parameters for strong **uf-cmva** security.

Next we show that the scheme is also **ki-cma** secure.

Theorem 5. *If HPS is extracting and 2-universal and the subset membership problem for HPS is (t,ϵ)-hard, then $\mathsf{MAC}_{\mathsf{HPS}}$ has completeness error $\eta = 0$ and is $(t', q_t, 2q_t\epsilon)$-**ki-cma** secure where $t \approx t'$.*

Proof. The completeness follows directly by inspection since Λ is a deterministic function. Now to prove the theorem, we recall that the experiment for the **ki-cma** involves a pair of TAG oracles and two secret keys sk_0 and sk_1 (where sk_1 may or may not be used depending on if the bit c chosen in the **ki-cma** experiment). We define a number of hybrid experiments starting with $\mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\mathsf{MAC}}$, incrementally replacing responses to tag queries with uniform samples for the tag space. The result is an experiment $\mathbf{Exp}_2$ where the view of the adversary is completely independent of the underlying keys. Thus in $\mathbf{Exp}_2$ the advantage of the adversary is 0 and so by showing that the difference in an adversary's advantage between the experiments is small we conclude the proof.

In more detail, for any (t,q_t)-adversary A and $\lambda \in \mathbb{N}$, we define the experiments as follows: $\mathbf{Exp}_0 = \mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\mathsf{MAC}}(\mathsf{A},\lambda)$. Next the experiment $\mathbf{Exp}_1$ is same as $\mathbf{Exp}_0$ except that all the queries to the first TAG oracle are answered with fresh random tags from the space $\mathcal{C}\setminus\mathcal{V}\times\mathcal{K}$ and finally $\mathbf{Exp}_2$ is same as $\mathbf{Exp}_1$ except that all the queries to the second TAG oracle are answered with fresh random tags from the space $\mathcal{C}\setminus\mathcal{V}\times\mathcal{K}$. For $i \in [0,2]$ we define $\epsilon_i := \mathbf{Adv}^{\mathbf{Exp}_i}_{\mathsf{MAC}}(\mathsf{A},\lambda)$ to denote the respective advantages of A in winning the experiments. Clearly $\epsilon_2 = 0$ since all the responses from both the tag oracles (and so the entire view of A) is independent of the secret keys under-use. Bellow we prove that $|\epsilon_0 - \epsilon_1| \leq q_t\epsilon$ and $|\epsilon_1 - \epsilon_2| \leq q_t\epsilon$. Combining these results, we get that $|\epsilon_0 - \epsilon_2| \leq 2q_t\epsilon$ and thus $\epsilon_0 \leq 2q_t\epsilon$ as desired.

Claim 4. $|\epsilon_0 - \epsilon_1| \leq q_t\epsilon$.

Proof. We define a series of $q_t + 1$ hybrid experiments starting from $\mathbf{Exp}_0$ and ending with $\mathbf{Exp}_1$. More precisely for $i \in \{0,\ldots,q_t\}$ in experiment $\mathbf{Exp}_{0,i}$ the first i queries to (either) tag oracles with

key sk_0 receive a fresh random tag in response sampled uniformly from the set $(c,k) \leftarrow_R \mathcal{C} \setminus \mathcal{V} \times \mathcal{K}$. In particular $\mathbf{Exp}_0 = \mathbf{Exp}_{0,0}$ and $\mathbf{Exp}_1 = \mathbf{Exp}_{0,q_t}$. We write $\delta_i := \mathbf{Adv}^{\mathbf{Exp}_{0,i}}_{\mathsf{MAC}}(\mathsf{A},\lambda)$ to denote the respective advantages of A at winning these experiments.

Fix any $i \in \{0,\ldots,q_t\}$. We show that if HPS is 2-universal and the subset membership problem is (t,ϵ)-hard then $|\delta_{i-1} - \delta_i| \leq \epsilon$. In particular this implies $|\epsilon_0 - \epsilon_1| = |\delta_0 - \delta_{q_t}| \leq q_t\epsilon$.

Let $\mathbf{Exp}'_{0,i}$ be identical to $\mathbf{Exp}_{0,i}$ except that response to the $i+1^{st}$ query m to a tag oracle is computed by first selecting random $c \leftarrow \mathcal{C} \setminus \mathcal{V}$ (but k is still computed as $k = \Lambda^m_{sk_0}(c)$). Let γ_i denote the advantage of A in $\mathbf{Exp}'_{0,i}$. Then $|\delta_i - \gamma_i| \leq \epsilon$ via a simple reduction to the subset membership problem. The reduction simply runs $\mathbf{Exp}_{0,i}$ and plants the subset membership challenge $c \in \mathcal{C}$ as the ciphertext for the $i+1^{st}$ query to a tag oracle with key sk_0. If and only if the adversary wins the experiment, the reduction outputs 1. Thus the value $|\delta_i - \gamma_i|$ is nothing other then the reduction's distinguishing advantage ϵ in the subset membership game. Finally $|\gamma_i - \delta_{i+1}| = 0$ follows from the 2-universality of HPS. In particular the view of A is identically distributed in the two experiments. Summing over all $i \in \{0,\ldots,q_t\}$, we prove the claim. □

The fact that $|\epsilon_1 - \epsilon_2| \leq q_t\epsilon$ follows from a similar argument and so we obtain the theorem. □

HPS based MAC from DDH Assumption [DKPW12]. Let $\mathbb{G}$ be a group of prime order p and let g be a random generator of $\mathbb{G}$. Let $H : \mathbb{G}^2 \times \mathcal{M} \to \mathbb{Z}_p$ be a collision resistant hash function.

Construction 5 ($\mathsf{MAC}^{\mathsf{DDH}}_{\mathsf{HPS}}$)

System Parameters: *The key space is $\mathcal{K} = \mathbb{Z}_p^3$, message space is $\mathcal{M}$ and tag space is $\mathcal{T} = \mathbb{G}^3$.*

Key Generation: *The key generation algorithm $\mathsf{KG}(1^\lambda)$ outputs a secret key $k = (w,x,x') \leftarrow_R \mathbb{Z}_p^3$.*

Tagging: *On input message $m \in \mathcal{M}$ the tagging algorithm TAG_k samples a uniform $c \leftarrow \mathbb{G}$ and outputs tag $(c, c^w, c^{x\ell+x'}) \in \mathbb{G}$, where $\ell = H(c,c^w,m)$.*

Verification: *On input message $m \in \mathcal{M}$ and tag $t = (c,d,e)$ the verification algorithm VRFY_k outputs* `Accept` *if and only if $c^w = d$ and $e = c^{x\ell+x'}$ such that $\ell = H(c,d,m)$.*

C.3 Weakly suf-cma/uf-cma and ki-cma MACs from Weak PRFs

In this section we show that the construction for MACs from key-homomorphic weak pseudorandom functions (kh-wPRF) described in [DKPW12] is also **ki-cma**. We prove this using a similar trick as was used in that paper to show that the construction is **ind-cma** and weakly **suf-cma** secure. We recall the definition of key homomorphic weak PRF and subsequently the construction.

Key-homomorphic weak PRF First, weak PRFs (wPRF) are the PRFs that are indistinguishable from random functions only for uniform random inputs. More precisely for a function $f : \mathcal{X} \to \mathcal{Y}$ let oracle $\mathcal{O}_f$ be such that upon each invocation it samples independent uniform $x \leftarrow_R \mathcal{X}$ and return samples $(x, f(x))$. Then we define a wPRF as follows.

Definition 7 (Weak PRFs). *Let m, n and $\mathcal{R}$ be as above. Moreover let $\mathsf{wPRF} := \{f_k : \mathcal{X} \to \mathcal{Y}\}_{k \in \mathcal{K}}$ be a set of efficiently computable functions indexed by keys pace $\mathcal{K}$. Then we call wPRF a (t,q,ϵ)-*secure wPRF *if for any (t,q)-adversary A (running in time at most t making at most q oracle calls) we have:*

$$\mathbf{Adv}^{\mathbf{wprf}}_{\mathsf{wPRF}}(\mathsf{A},\lambda) := \left| \Pr_{k \leftarrow \mathcal{K}}\left[\mathsf{A}^{\mathcal{O}_{f_k}} \to 1\right] - \Pr_{R \leftarrow \mathcal{R}}\left[\mathsf{A}^{\mathcal{O}_R} \to 1\right] \right| \leq \epsilon$$

Definition 8 (Key-Homomorphic wPRF). *A PRF* $\mathsf{khwPRF} = \{f_k : \mathcal{X} \to \mathcal{Y}\}_{k\in\mathcal{K}}$ *is called* key-homomorphic *wPRF if it is wPRF and* $\mathcal{K}$ *and* $\mathcal{Y}$ *are (additive) groups of prime order* p *equipped with an efficient (additive) group operation such that for any fixed* $x \in \mathcal{X}$ *function* $f_k(x)$ *is a group isomorphism from* $\mathcal{K} \to \mathcal{Y}$*. In particular for all keys* $k_1, k_2 \in \mathcal{K}$*, integers* $a, b \in \mathbb{Z}_p$ *and* $x \in \mathbb{Z}_2^m$ *we have:*

$$a \cdot f_{k_1}(x) + b \cdot f_{k_2}(x) = f_{a\cdot k_1 + b\cdot k_2}(x).$$

For security parameter $\lambda \in \mathbb{N}$ let $\mathcal{X} = \mathcal{X}(\lambda)$, $\mathcal{Y} = \mathcal{Y}(\lambda)$ and $\mathcal{K} = \mathcal{K}(\lambda)$ be sets such that $\mathsf{khwPRF} = \{f_k : \mathcal{X} \to \mathcal{Y}\}_{k\in\mathcal{K}}$ is a kh-wPRF where $\mathcal{K}$ and $\mathcal{Y}$ are additive groups of prime order $p = p(\lambda)$. The construction of [DKPW12] goes as follows.

Construction 6 (MAC from khwPRF: $\mathsf{MAC}_{\mathsf{khwPRF}}$)

System Parameters: *The key space is* $\bar{\mathcal{K}} = \mathcal{K} \times \mathcal{K}$*, message space is* $\mathcal{M} = \mathbb{Z}_p$ *and tag space is* $\mathcal{T} = \mathcal{X} \times \mathcal{Y}$*.*

Key Generation: *The key generation algorithm* $\mathsf{KG}(1^\lambda)$ *samples* $k_1, k_2 \leftarrow \mathcal{K}$ *and outputs secret key* (k_1, k_2)*.*

Tagging: *On input message* $m \in \mathbb{Z}_p$ *the tagging algorithm* $\mathsf{TAG}_{(k_1,k_2)}$ *samples a uniform* $x \leftarrow \mathcal{X}$ *and outputs tag* $(x, f_{k_1\cdot m+k_2}(x))$*.*

Verification: *On input message* $m \in \mathbb{Z}_p$ *and tag* (x, y) *the verification algorithm* $\mathsf{VRFY}_{(k_1,k_2)}$ *outputs* `Accept` *if and only if* $f_{k_1\cdot m+k_2}(x) = y$*.*

We prove that this construction is indeed KI.

Theorem 6. *If* khwPRF *is* (t, q, ϵ)*-secure kh-wPRF then* $\mathsf{MAC}_{\mathsf{khwPRF}}$ *is a* $(t, q, 2\epsilon)$-**ki-cma** *secure MAC.* $\mathsf{MAC}_{\mathsf{khwPRF}}$ *has completeness error of* 0*.*

Proof. The completeness follows by inspection of the scheme. Now recall that the KI game involves two instances of MAC. We define two experiments closely related to $\mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\mathsf{MAC}}$ incrementally replacing the responses to tag queries with uniform random elements in the tag space (i.e. independent of key for that oracle). Thus we obtain a KI-like experiment where the bit to be guessed is perfectly hidden from the adversary. To argue that we can make these switches we use the trick employed in [DKPW12] to prove the message-hiding and unforgeability properties which crucially relies on the key-homomorphicity of khwPRF.

More precisely, for parameters $\lambda, t, q \in \mathbb{N}$ and any (t, q)-adversary A we define the experiment $\mathbf{Exp}_0 := \mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$. Moreover let experiment $\mathbf{Exp}_1$ be identical to $\mathbf{Exp}_0$ except for that any tag query for key k_0 receives a fresh uniform sample from $\mathcal{T}$. Finally let $\mathbf{Exp}_2$ be identical to $\mathbf{Exp}_1$ except that also tag queries for k_1 result in uniform samples from $\mathcal{T}$.

For $i \in [0, 2]$ we write $\epsilon_i := \mathbf{Adv}^{\mathbf{Exp}_i}_{\mathsf{MAC}}(\mathsf{A}, \lambda)$ to denote the respective advantages of A at winning these experiments. Then clearly $\epsilon_2 = 0$ since all responses from the tag oracles (and so the entire view of A) are independent of bit c. Bellow we prove that $|\epsilon_0 - \epsilon_1| \leq \epsilon$. An almost identical argument for $|\epsilon_1 - \epsilon_2|$ implies the result.

Claim 5. $|\epsilon_0 - \epsilon_1| \leq \epsilon$

Proof. We build a reduction R to the **wprf** game which runs in time t making at most q queries and wins with probability at least ϵ. Given access to an oracle $\mathcal{O}$ it first obtains q samples of the form (x_i, y_i) from $\mathcal{O}$. Then it simulates $\mathbf{Exp}_0$ except for the following changes outputting 1 if and only if A wins.

1. Instead of using $\mathsf{KG}(1^\lambda)$ to generate key $k_0 \in \bar{\mathcal{K}}$ it only samples a uniform $k_2 \leftarrow \mathcal{K}$.
2. For each new query $m \in \mathbb{Z}_p$ made to TAG_{k_0} the reduction uses a fresh sample (x_i, y_i) and responds with tag $\sigma_i = (x_i, m \cdot y_i + f_{k_2}(x_i))$.

We claim that if $\mathcal{O} = R$ is a random function then the view of A is identical to $\mathbf{Exp}_1$. Indeed in this case (x_i, y_i) if uniformly distributed over $\mathcal{T} = \mathcal{X} \times \mathcal{Y}$ by the definition of R. Thus so is σ_i (regardless of the value of m and $f_{k_2}(x_i)$). But this implies that if $\mathcal{O} = R$ then $\Pr[\mathsf{R} \to 1] = \epsilon_1$.

Suppose now that $\mathcal{O} = f_k$ for a random $k \in \mathcal{K}$. Then we claim that the view of A is exactly that generated in $\mathbf{Exp}_0$ when $k_0 := (k, k_2)$. This follows from the following calculation:

$$\sigma_i = (x_i, m \cdot y_i + f_{k_2}(x_i)) = (x_i \cdot m \cdot f_k(x_i) + f_{k_2}(x_i)) = (x_i, f_{m \cdot k + k_2}(x_i)) = \mathsf{TAG}_{k_0}(m).$$

Therefore if $\mathcal{O} = f_k$ it must be that $\Pr[\mathsf{R} \to 1] = \epsilon_1$.

So by assumption on the security of khwPRF we obtain the claim, and thus the result. □

□

As observed in [DKPW12] both DDH assumption and the LWE assumptions can (for example) be used to directly instantiate efficient key-homomorphic wPRF families.

Let $\mathbb{G}$ be a group of prime order p. The functions family $\{f_k(x) = x^k\}_{k \in \mathbb{Z}_p}$ is wPRF assuming DDH holds for $\mathbb{G}$. Furthermore, it is key homomorphic with $f_{a \cdot k_1 + b \cdot k_2}(x) = (f_{k_1}(x))^a (f_{k_2}(x))^b$. Therefore this construction for weak PRFs gives us a MAC that is **suf-cma**, **ki-cma** and **ind-cma** with key space $\mathbb{Z}_p^2$, message space $\mathcal{M} = \mathbb{Z}_p$, tag space $\mathcal{T} = \mathbb{G}^2$ and tagging function $\mathsf{TAG}_{k_1,k_2} = (g, h) = (g, g^{k_1 m + k_2})$.

To use LWE we first make the following definition. For integers $p < q$ and $x \in \mathbb{Z}_q$. Let $\lceil x \rfloor_p := \lceil (p/q) \cdot x \rfloor \mod q$ and for $\mathbf{x} \in \mathbb{Z}_q^n$ we let $\lceil \mathbf{x} \rfloor_p$ be defined as the natural component-wise extension. Then for $\mathcal{K} = \mathbb{Z}_q^{m \times n}$, $\mathcal{X} = \mathbb{Z}_q^n$ and $\mathcal{Y} = \mathbb{Z}_q^m$ the set $\{f_{\mathbf{K}}(\mathbf{x}) = \lceil \mathbf{K} \cdot \mathbf{x} \rfloor_p\}$ is a wPRF [BPR12] based on the hardness of LWE (for appropriate choice of q, p and the error parameter of the underlying LWE problem). Moreover for almost all inputs $x \in \mathcal{X}$ the functions are key-homomorphic which suffices for the construction as it only evaluates the wPRF on uniform random points. In particular with overwhelming probability the reduction needs only to simulate the wPRF at points for which the key-homomorphicity holds.

In [DKPW12] an extension of the wPRF construction is provided which makes use of Waters' argument [Wat05] to achieve weak **uf-cma** (and **ind-cma**) security at the cost of a somewhat less efficient scheme. We also observe that the modified scheme is **ki-cma** secure following essentially same argument that we use in the proof of Theorem 6.

D Transformations

D.1 Adding Support for Verification Queries

We briefly recall the transformation of [DKPW12] transformation mapping any weakly {**uf**, **ind**}-**cma** secure MAC to a weakly **uf-cmva** secure MAC. In fact the transformation achieves more namely the resulting MAC is even *strongly* **uf-cmva** secure. Indeed the original proof goes through unchanged also for the the stronger statement. In this section, we show that the same transformation also preserves the **ki-cma** security of the underlying MAC. We start with the transformation of [DKPW12].

Let $\mu = \mu(\lambda)$ denote a statistical security parameter and let $\mathcal{H}$ be the family of pairwise independent hash function $h : \mathcal{T} \to \{0,1\}^\mu$. Given an MAC $\mathsf{MAC} = \{\mathsf{KG}, \mathsf{TAG}, \mathsf{VRFY}\}$ with key space

$\mathcal{K}$, message space $\mathcal{M} \times \{0,1\}^\mu$, and tag space $\mathcal{T}$, a new MAC $\overline{\mathsf{MAC}} = \{\overline{\mathsf{KG}}, \overline{\mathsf{TAG}}, \overline{\mathsf{VRFY}}\}$ with key space $\mathcal{K} \times \mathcal{H}$, message space $\mathcal{M}$ and tag space $\mathcal{T} \times \{0,1\}^\mu$ is constructed as follows.

Construction 7 ($\overline{\mathsf{MAC}}$ from MAC)

Key Generation: *The key generation algorithm* $\overline{\mathsf{KG}}(1^\lambda)$ *runs* $k \leftarrow \mathsf{KG}(1^\lambda)$*, samples a pairwise independent hash function* $h \leftarrow \mathcal{H}$ *and outputs* (k, h) *as the secret key.*

Tagging: *The tagging algorithm* $\overline{\mathsf{TAG}}_{(k,h)}(m)$ *samples* $b \leftarrow_R \{0,1\}^\mu$*, runs* $z = \mathsf{TAG}_k(m||b)$ *and returns* $(z, h(z) \oplus b)$ *as the tag.*

Verification: *The verification algorithm* $\overline{\mathsf{VRFY}}_{(k,h)}(m, (z, y))$ *computes* $b = y \oplus h(z)$ *and outputs* $\mathsf{VRFY}_k(m||b, z)$*.*

The following theorem was proved in [DKPW12].

Theorem 7 (Weak $\{\mathsf{uf}, \mathsf{ind}\}$-cma $\implies$ Strong uf-cmva). *For any* $t, q_t, q_v \in \mathbb{N}$ *and* $\epsilon \geq 0$*, if* MAC *is:*

- *weakly* (t, q_t, ϵ_1)-**uf-cma** *(existentially unforgeable)*
- (t, q_t, ϵ_2)-**ind-cma** *(message hiding)*

then $\overline{\mathsf{MAC}}$ *is* $(t', q_t, q_v, 2q_v \max\{\epsilon_1, \epsilon_2\} + 2q_v q_t/2^\mu)$-**uf-cmva** *secure where* $t' \approx t$*.*

We now prove the following theorem which shows that the above transformation essentially preserves the **ki-cma** property of the underlying MAC.

Theorem 8 (Preserving ki-cma Security). *For any* $t, q_t \in \mathbb{N}$ *and* $\epsilon_1, \epsilon_2 \geq 0$*, if* MAC *is:*

- (t, q_t, ϵ_1)-**ki-cma** *(key indistinguishable)*
- (t, q_t, ϵ_2)-**ind-cma** *(message hiding)*

then $\overline{\mathsf{MAC}}$ *is* $(t', q_t, \epsilon_1 + 2\epsilon_2)$-**ki-cma** *secure where* $t' \approx t$*.*

Proof. We prove the theorem by building two reductions, namely to **ki-cma** and **ind-cma** security of MAC.

For security parameter $\lambda \in \mathbb{N}$, and $\delta = \delta(\lambda)$ let B be a (t, q_t)-adversary such that $\mathbf{Adv}^{\mathsf{ki\text{-}cma}}_{\overline{\mathsf{MAC}}}(\mathsf{B}, \lambda) = \delta$. We upper bound $\delta \leq \epsilon_1 + 2\epsilon_2$ using a hybrid argument. That is we define two experiments such that the advantage in the **ki-cma** game for $\overline{\mathsf{MAC}}$ is at most the sum of the advantages at winning the two experiments.

Experiment $\mathbf{Exp}_0$ is identical to $\mathbf{Exp}^{\mathsf{ki\text{-}cma}}_{\overline{\mathsf{MAC}}}$ except that the same key k_0 (and state) is used for MAC in both instances of $\overline{\mathsf{MAC}}$ i.e. the keys for $\overline{\mathsf{MAC}}$ are (k_0, h_0) and (k_0, h_1). Similarly experiment $\mathbf{Exp}_1$ is defined to use the same hash function for both the keys of $\overline{\mathsf{MAC}}$ namely (k_0, h_1) and (k_1, h_1). For $i \in [0, 1]$ we define $\gamma_i := \mathbf{Adv}^{\mathbf{Exp}_i}_{\overline{\mathsf{MAC}}}(\mathsf{B}, \lambda)$ to denote the respective advantages of B in winning the experiments.

We now show that $\gamma_0 \leq 2\epsilon_2$ and $\gamma_1 \leq \epsilon_1$. The triangle inequality, then bounds $\delta \leq \gamma_0 + \gamma_1 \leq 2\epsilon_2 + \epsilon_1$, as required.

Claim $\gamma_0 \leq 2\epsilon_2$: We construct a reduction R that interacts in **ind-cma** game and is provided with the oracle $\mathcal{O}$. It must tell if the oracle returns tag of the message that it supplies or returns the tag of zero message after making q_t queries to the oracle in time t. In order to use B to make its decision, R runs $\mathbf{Exp}_0(\mathsf{B}, \lambda)$ except that all evaluations of the TAG algorithm are instead computed via a call to $\mathcal{O}$. Finally R outputs 1 iff B wins.

Case $\mathcal{O} = \mathsf{TAG}_k(\cdot)$: R perfectly simulates $\mathbf{Exp}_0$ to B and so $2\left|\Pr[\mathsf{R}^{\mathsf{TAG}_k(\cdot)} \to 1] - \frac{1}{2}\right| = \gamma_0$.

Case $\mathcal{O} = \mathsf{TAG}_k(0)$: In this case for $i \in \{0, c\}$ any tag $\tau = (z, h_i(z) \oplus b)$ for a message m is such that $z = \mathsf{TAG}_k(0)$. Thus b is independent of z (since $\mathsf{TAG}_k(0)$ ignores the message). This means that h_i is independent of τ and so h_i and in particular c remains information theoretically hidden from B. Since $c \in \{0, 1\}$ is a uniform random and R outputs 1 iff B guesses c we get that $\Pr[\mathsf{R}^{\mathsf{TAG}_k(0)} \to 1] = \frac{1}{2}$.

Summing up, we can use the **ind-cma** security of MAC to write

$$\gamma_0 = 2\left|\Pr[\mathsf{R}^{\mathsf{TAG}_k(\cdot)} \to 1] - \frac{1}{2}\right| = 2\left|\Pr[\mathsf{R}^{\mathsf{TAG}_k(\cdot)} \to 1] - \Pr[\mathsf{R}^{\mathsf{TAG}_k(0)} \to 1]\right| \leq 2\epsilon_2.$$

Claim $\gamma_1 \leq \epsilon_1$: We construct a reduction R that interacts in $\mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\mathsf{MAC}}$ experiment and is provided with two tag oracles $\mathcal{O}_0, \mathcal{O}_c$. The reduction now simulates $\mathbf{Exp}_1(\mathsf{B}, \lambda)$ forwarding all TAG_{k_0} queries to $\mathcal{O}_0$ and all TAG_{k_c} queries to $\mathcal{O}_c$. Finally, R produces the same output as B.

Observe that R perfectly simulates $\mathbf{Exp}_1$ to B and moreover R wins the **ki-cma** game if and only if B wins $\mathbf{Exp}_1$. Thus the **ki-cma** security of MAC implies

$$\gamma_1 = 2\left|\Pr[\mathbf{Exp}_1(\mathsf{B}, \lambda) = 1] - \frac{1}{2}\right| = 2\left|\Pr[\mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\mathsf{MAC}}(\mathsf{R}, \lambda) = 1] - \frac{1}{2}\right| \leq \epsilon_1.$$

□

D.2 KI Preserving Domain Extension

In [DKPW12], the authors have shown that any **ind-cma** and weakly **uf-cma** secure MAC supports domain extension using universal hash function following the 'hash and then MAC' paradigm. A ν-bit message is hashed down to a μ-bit input and then the shorter μ-bit input is fed to the MAC. The domain extension for the MAC in this way is otherwise not secure since typical MACs are not message hiding and so might reveal the hash function which would allow an adversary to find colliding messages making forgery trivial. More formally, the following proposition is shown in [DKPW12].

Proposition 2 (Domain Extension). *Consider* $\mathsf{MAC} = \{\mathsf{KG}, \mathsf{TAG}, \mathsf{VRFY}\}$ *with small message space* $\mathcal{M} = \{0,1\}^\mu$ *and let* $\overline{\mathsf{MAC}} = \{\overline{\mathsf{KG}}, \overline{\mathsf{TAG}}, \overline{\mathsf{VRFY}}\}$ *for large message space* $\{0,1\}^\nu$ *be derived from* MAC *by first hashing the message using a* β*-universal hash function* $g : \{0,1\}^\ell \times \{0,1\}^\nu \to \{0,1\}^\mu$.

If MAC *is* (t, q_t, ϵ_1)-**ind-cma** *secure and weakly* (t, q_t, ϵ_2)-**uf-cma** *secure then* $\overline{\mathsf{MAC}}$ *is* $(t', q_t, 2\epsilon_1)$-**ind-cma** *secure and weakly* $(t', q_t, \epsilon_1 + \epsilon_2 + q_t\beta)$-**uf-cma** *secure, where* $t' \approx t$.

We now prove the following theorem which shows that KI is preserved by the domain extension transformation, using similar ideas to the proof of Proposition 2 in [DKPW12].

Theorem 9 (Domain Extension Preserves KI). *Let* MAC *and* $\overline{\mathsf{MAC}}$ *be as in Proposition 2. If* MAC *is* (t, q_t, ϵ_1)-**ki-cma** *secure and* (t, q_t, ϵ_2)-**ind-cma** *secure, then* $\overline{\mathsf{MAC}}$ *is* $(t', q_t, \epsilon_1 + 2\epsilon_2)$-**ki-cma** *secure for* $t \approx t'$.

Proof. We prove this theorem using almost the same line of reasoning as in the proof of Theorem 8. For completeness we provide a full proof below.

Let B be a (t, q_t) adversary interacting in $\mathbf{Exp}^{\mathbf{ki\text{-}cma}}_{\overline{\mathsf{MAC}}}$. For any security parameter $\lambda \in \mathbb{N}$ we upperbound $\bar{\epsilon} = \mathbf{Adv}^{\mathbf{ki\text{-}cma}}_{\overline{\mathsf{MAC}}}(\mathsf{B}, \lambda)$ by first designing a pair of reductions for the **ind-cma** and **ki-cma**

security (respectively) of MAC and then showing that $\bar{\epsilon}$ is at most the sum of the advantages of the reductions.

Recall that the keys for $\overline{\mathsf{MAC}}$ have the form (k, h) where k is a key for MAC and h is a key for a hash function. We define experiment $\mathbf{Exp}_0$ to be identical to $\mathbf{Exp}^{\textsf{ki-cma}}_{\overline{\mathsf{MAC}}}$ except that the same key is used for MAC in both instances of $\overline{\mathsf{MAC}}$. In other words $\overline{\mathsf{MAC}}$ is instantiated with keys (k_0, h_0) and (k_0, h_1). Experiment $\mathbf{Exp}_2$ on the other hand is identical to $\mathbf{Exp}_0$ but instead uses the same hash function for both instances of $\overline{\mathsf{MAC}}$.

For $i \in [0, 1]$ we define $\gamma_i := \mathbf{Adv}^{\mathbf{Exp}_i}_{\overline{\mathsf{MAC}}}(\mathsf{B}, \lambda)$ to denote the respective advantages of B in winning the experiments. A standard hybrid argument shows that an adversary which can tell $\overline{\mathsf{MAC}}$ instances keyed with (k_0, h_0) and (k_1, h_1) apart is at least half as likely to tell either the $\overline{\mathsf{MAC}}$ instances from $\mathbf{Exp}_0$ or $\mathbf{Exp}_2$ apart. In other words $\bar{\epsilon} \leq \gamma_0 + \gamma_1$. Thus the result follows by showing that $\gamma_0 \leq 2\epsilon_2$ and $\gamma_1 \leq \epsilon_1$.

Claim $\gamma_0 \leq 2\epsilon_2$**:** We provide a reduction R to the **ind-cma** security of MAC using B. Given oracle $\mathcal{O}$ in **ind-cma** experiment, R simulates $\mathbf{Exp}_0(\mathsf{B}, \lambda)$ as faithfully except that it computes all TAG_{k_0} queries using $\mathcal{O}$ instead. Finally R outputs 1 iff B wins.

Case $\mathcal{O} = \mathsf{TAG}_k(\cdot)$**:** In this case R perfectly simulates $\mathbf{Exp}_0$ to B and thus

$$2\left|\Pr[\mathsf{R}^{\mathsf{TAG}_k(\cdot)} \to 1] - \frac{1}{2}\right| = \gamma_0.$$

Case $\mathcal{O} = \mathsf{TAG}_k(0)$**:** In this case, in $\mathbf{Exp}_0$ all tags z from either TAG oracle are computed as $z = \mathsf{TAG}_k(0)$ and so are independent of the particular hash function used by R. Thus the view of B is independent of the value c which it must guess. In particular we have

$$2\left|\Pr[\mathsf{R}^{\mathsf{TAG}_k(\cdot)} \to 1] - \frac{1}{2}\right| = 0.$$

Summing up, we have $\left|\Pr[\mathsf{R}^{\mathsf{TAG}_k(\cdot)} \to 1] - \Pr[\mathsf{R}^{\mathsf{TAG}_k(0)} \to 1]\right| = \frac{\gamma_0}{2}$. By the **ind-cma** security of MAC, it now follows that $\frac{\gamma_0}{2} \leq \epsilon_2$.

Claim $\gamma_1 \leq \epsilon_1$**:** We construct a reduction R to the **ki-cma** security of MAC using B. Given oracles $\mathcal{O}_0, \mathcal{O}_c$, R simulates $\mathbf{Exp}_2$ to B faithfully except that it computes all TAG_{k_0} queries with calls to $\mathcal{O}_0$ and all TAG_{k_c} queries with calls to $\mathcal{O}_c$. Finally R produces the same output as B.

Reduction R perfectly simulates $\mathbf{Exp}_2$ to B and moreover R wins the **ki-cma** game if and only if B wins in $\mathbf{Exp}_2$. Thus we can write:

$$\gamma_1 = 2\left|\Pr[\mathbf{Exp}^{\textsf{ki-cma}}_{\mathsf{MAC}}(\mathsf{R}, \lambda) = 1] - \frac{1}{2}\right| \leq \epsilon_1.$$

□

Applying the domain extension trick with the observation that any **suf-cma** scheme is also **uf-cma** up to a security loss, we can turn any (t, q_t, ϵ_1)-**ind-cma** secure, weakly (t, q_t, ϵ_2)-**suf-cma** secure and (t, q_t, ϵ_3)-**ki-cma** secure MAC with $\mathcal{M} = \{0, 1\}^\mu$ into a MAC which is:

- $(t', q_t, 2\epsilon_1)$-**ind-cma** secure
- weakly $(t', q_t, \epsilon_1 + \epsilon_2 2^\mu + q_t\beta)$-**uf-cma** secure
- $(t', q_t, 2\epsilon_1 + \epsilon_3)$-**ki-cma** secure

where $t' \approx t$.