IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 2017, No. 2, pp. 27–58. DOI:10.13154/tosc.v2017.i2.27-58

New Constructions of MACs from (Tweakable) Block Ciphers

Benoît Cogliati[1], Jooyoung Lee[2] and Yannick Seurin[3]

[1] University of Luxembourg, Luxembourg, Luxembourg
benoitcogliati@hotmail.fr
[2] Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea
hicalf@kaist.ac.kr
[3] Agence nationale de la sécurité des systèmes d'information (ANSSI), Paris, France
yannick.seurin@m4x.org

Abstract. We propose new constructions of Message Authentication Codes (MACs) from tweakable or conventional block ciphers. Our new schemes are either stateless and deterministic, nonce-based, or randomized, and provably secure either in the standard model for tweakable block cipher-based ones, or in the ideal cipher model for block cipher-based ones. All our constructions are very efficient, requiring only one call to the underlying (tweakable) block cipher in addition to universally hashing the message. Moreover, the security bounds we obtain are quite strong: they are beyond the birthday bound, and nonce-based/randomized variants provide graceful security degradation in case of misuse, i.e., the security bound degrades linearly with the maximal number of repetitions of nonces/random values.

Keywords: MAC · tweakable block cipher · nonce-misuse resistance · graceful security degradation

1 Introduction

MACs. A *Message Authentication Code* (MAC) is a fundamental symmetric primitive allowing two entities sharing a secret key to verify that a received message originates from one of the two parties and was not modified by an attacker. Most existing MACs are built from a block cipher, e.g., CBC-MAC [BKR00] or OMAC [IK03], or from a cryptographic hash function, e.g., HMAC [BCK96]. At a high level, many of these constructions follow the well-established *UHF-then-PRF* design paradigm: the message M is first mapped onto a short string through a universal hash function (UHF), and then "encrypted" through a fixed-input-length PRF to obtain a short tag.[1] This method is simple (in particular, it is deterministic and stateless), yet its security caps at the so-called birthday-bound since any collision at the output of the UHF, which translate into a tag collision, is usually enough to break the security of the scheme. Better security bounds can be obtained by incorporating in the tag computation a nonce (a value that never repeats), e.g., in Wegman-Carter type MACs [WC81, Sho96, Ber05, CS16] or a random value [BGK99, JJV02, JL04, Min10].

Our Contribution. We propose new MAC constructions, which are either nonce-based/randomized or stateless and deterministic, and which are based on a universal hash function and either on a (conventional) block cipher or a tweakable block cipher. Hence, in total, we propose four new constructions, two of which can be analyzed in two slightly different (but related) security models (namely nonce-based or randomized). Tweakable

[1] Actually, this kind of construction yields a variable-input-length PRF rather than a mere MAC.

Received: 2017-03-01, Accepted: 2017-05-01, Published: 2017-06-19

block ciphers (TBCs) are a generalization of conventional block ciphers which, in addition to a message and a cryptographic key, take another (public, or even controlled by the adversary) input called a *tweak*. This tweak should provide inherent variability to the block cipher and plays a similar role to an IV or a nonce in an encryption scheme. The security notion for this primitive was first formalized in [LRW02], where it was pointed out that tweakable block ciphers are very useful for building various higher level cryptographic schemes.

Our two TBC-based constructions follow the traditional UHF-then-PRF approach, the PRF being "instantiated" from the TBC $\widetilde{E}$. The starting point of our nonce-based construction, called NaT (*Nonce-as-Tweak*) and depicted on Figure 1 (top left), is the simple remark that, as long as tweaks do not repeat, a tweakable block cipher behaves as a random function. Hence, if the hash of each message is encrypted with a fresh nonce as tweak, collisions among hash values don't matter since the hashes are encrypted by "independent" random functions $\widetilde{E}_K^N$. Even if tweaks (i.e., nonces) repeat, the security loss is negligible as long as the number of repetitions is small. The provable security bound for the NaT construction is dominated by terms of the form $\mu q \varepsilon$, where μ denotes the maximal number of repetitions of any nonce, q denotes the number of adversarial (MAC or verification) queries, and ε is the parameter characterizing the collision probability of the UHF. A typical value (e.g., for polynomial-based hashing [Sho96, Ber07]) for ε is $\ell/2^n$, where n is the output length of the UHF (which is also the block length of the TBC) and ℓ is the maximal length of messages in n-bit blocks. Hence, in the nonce-respecting case (i.e., $\mu = 1$) the adversary's advantage is of the form $q\ell/2^n$, whereas in the nonce-misusing case (where μ might be as large as q), it becomes $q^2\ell/2^n$, i.e., a birthday-type bound. The security bound degrades linearly with μ, the maximal number of repetitions of nonces. We note that the NaT construction is used in version 1.41 of the authenticated encryption scheme Deoxys [JNPS16], a third round candidate of the CAESAR competition.

To obtain a stateless deterministic TBC-based construction, one simply replaces the nonce by an independent hash of the message. The resulting construction is called HaT (*Hash-as-Tweak*), see Figure 1 (top right). This construction is secure beyond the birthday bound. Both NaT and HaT are provably secure in the standard model, assuming only that the TBC is a secure pseudorandom tweakable permutation.

Our two block cipher-based constructions, on the other hand, depart from the standard UHF-then-PRF approach, since the output transformation is *unkeyed*. Actually, they can be seen as block cipher-based instantiations of a new paradigm (which, to the best of our knowledge, has not been formally explored yet), which could be dubbed *UHF-then-RO*: the tag is computed as $T = G(H_K(M))$, where H is a (keyed) uniform and universal hash function, and G is a (keyless) cryptographic hash function. It is easy to prove (and we do so in Appendix B) that this construction is a secure MAC (in fact, a variable-input-length PRF) in the random oracle (RO) model for G.[2] Obviously, the output transformation must be hard to invert (as otherwise the adversary can compute the output of the UHF from the tag), which for the nonce-based construction implies that we must use the block cipher in Davies-Meyer mode. The resulting variants of NaT and HaT, called respectively NaK (*Nonce-as-Key*) and HaK (*Hash-as-Key*), are depicted in Figure 1 (bottom). They are provably secure in the ideal cipher model [BRS02].

We provide a comparison of our new constructions with existing UHF-based MAC constructions in Table 1.

Proof Technique. Our proofs rely on the H-coefficients technique, which has been introduced by Patarin [Pat08b], and has recently been highlighted by Chen and Steinberger for analyzing the iterated Even-Mansour cipher [CS14]. This method is typically used

[2] A natural question is whether standard security assumptions on G are sufficient to prove MAC/PRF security.

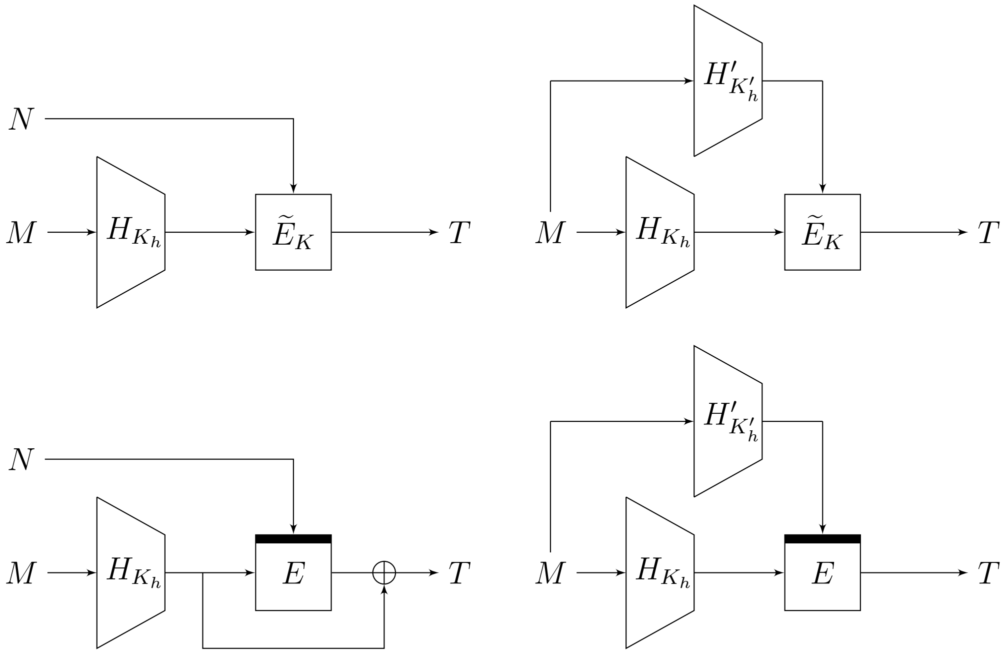

Figure 1: Top: the TBC-based constructions NaT (left) and HaT (right) based on a TBC $\widetilde{E}$ and AU hash functions H and H'. The tweak input is on top of the $\widetilde{E}_K$ box, while the block input is on the left. Bottom: the block cipher-based variants NaK (left) and HaK (right) based on a block cipher E and AU and almost uniform hash functions H and H'. The solid line materializes the key input for E.

to prove information-theoretic pseudorandomness of constructions such as Feistel networks [Pat90, Pat91, Pat10], the XOR of permutations [Pat08a, Pat13] or Even-Mansour constructions [CLS15, CS15b, CS15a, Men15, HT16]. The use of the H-coefficients technique to study the security of MAC constructions (in particular, to directly handle verification queries rather than appealing to generic results resulting in looser bounds) was previously introduced by Cogliati and Seurin [CS16].

More Related Work. Several other MAC constructions based on tweakable block ciphers have been proposed. For example, TBC-MAC [LRW02] and TBC-MAC2 [LST12] are two such constructions. They are similar in design to CBC-MAC, however the chaining in these constructions is done through the tweak. Because of their structure, these two constructions require as many calls to the tweakable block cipher as the number of blocks in the message, whereas our constructions only require one call to the TBC and one to the universal hash function, which can be much more efficient depending on the choice of the UHF. Moreover, the proven security of TBC-MAC is still birthday bound and, while TBC-MAC2 has a security bound comparable to our bounds, it requires the underlying TBC to have a tweak length much larger than the block length, which is not the case in our constructions.

Black and Cochran [BC09] also proposed a nonce-based MAC construction, called WMAC, which is based on a PRF and a universal hash function. Our NaT construction can actually be seen as a particular case of WMAC, where the PRF is instantiated by a tweakable block cipher. However, our security bound is actually tighter than what would be achieved by simply applying [BC09, Theorem 6] to our construction.

Naito [Nai15] and more recently List and Nandi [LN17] have proposed TBC-based

Table 1: Comparison of our new constructions with prominent existing MAC constructions based on an arbitrary (xor-)universal hash function. WC stands for Wegman-Carter; SD stands for stateless deterministic; “prim.” indicates which primitive is used in addition to the UHF; “# calls” gives the number of calls to the underlying primitive (in addition to the UHF call); “BBB” indicates whether the construction is secure beyond the birthday bound (when nonces are not repeated for nonce-based ones); “NMR” indicates whether nonce-based constructions are nonce-misuse resistant; “proof” indicates whether the security proof is in the standard model (SM) or the ideal cipher model (ICM).

construction	type	prim.	# calls	BBB	NMR	proof	ref.
UHF-then-BC	SD	BC	1	×	—	SM	[BS, Section 7]
HaT	SD	TBC	1	✓	—	SM	this paper
HaK	SD	BC	1	✓	—	ICM	this paper
PRF-based WC	nonce	PRF	1	✓	×	SM	[WC81]
BC-based WC	nonce	BC	1	×	×	SM	[Sho96, Ber05]
EWCDM	nonce	BC	2	✓	✓	SM	[CS16]
NaT	nonce	TBC	1	✓	✓	SM	this paper
NaK	nonce	BC	1	✓	✓	ICM	this paper
EHtM	rand.	PRF	2	✓	—	SM	[Min10]
NaT	rand.	TBC	1	✓	—	SM	this paper
NaK	rand.	BC	1	✓	—	ICM	this paper

constructions of stateless deterministic MACs that do not use a generic UHF, but instead construct the UHF from the underlying TBC (so that the resulting constructions are entirely TBC-based). In principle, the two UHFs of our construction HaT can be instantiated for example with PMAC1 [Rog04] (the TBC-based generalization of PMAC [BR02]) to obtain a similarly “purely” TBC-based construction, however the resulting construction makes two TBC calls per message block, whereas Naito’s and List-Nandi’s constructions only make one, so that we do not claim to compete with them in terms of efficiency.

Organization. We first establish the notation and recall standard security definitions in Section 2. We also give a general lemma allowing to translate any security bound for a nonce-based MAC that provides graceful security degradation with respect to the number of nonce repetitions to the corresponding randomized scheme (where “nonces” are chosen uniformly at random). We then describe and prove the security of our TBC-based constructions in Section 3 and of our block cipher-based constructions in Section 4. In Section 5, we give a simple generic result about the security loss induced by tag truncation that might be of independent interest.

2 Preliminaries

2.1 General Definitions

Basic Notation. Given a non-empty set $\mathcal{X}$, we let $X \leftarrow_\$ \mathcal{X}$ denote the draw of an element X from $\mathcal{X}$ uniformly at random. The set of all functions from $\mathcal{X}$ to $\mathcal{Y}$ is denoted $\mathsf{Func}(\mathcal{X},\mathcal{Y})$, and the set of all permutations of $\mathcal{X}$ is denoted $\mathsf{Perm}(\mathcal{X})$. The set of binary strings of length n is denoted $\{0,1\}^n$. The set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$ is simply denoted $\mathsf{Func}(n)$, and the set of all permutations of $\{0,1\}^n$ is simply denoted $\mathsf{Perm}(n)$. For integers $1 \le b \le a$, we will write $(a)_b = a(a-1)\cdots(a-b+1)$ and $(a)_0 = 1$ by convention. Remark that, using our notation, the probability that a random permutation

$P \leftarrow_\$ \mathsf{Perm}(n)$ satisfies q equations $P(X_i) = Y_i$ for distinct X_i's and distinct Y_i's is exactly $1/(2^n)_q$.

PRFs. A keyed function with key space $\mathcal{K}$, domain $\mathcal{X}$, and range $\mathcal{Y}$ is a function $F : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$. We write $F_K(X)$ for $F(K, X)$. A (q, t)-adversary against F is an algorithm A with oracle access to a function from $\mathcal{X}$ to $\mathcal{Y}$, making at most q oracle queries, running in time at most t, and outputting a single bit. The advantage of A in breaking the PRF-security of F, i.e., in distinguishing F from a uniformly randomly chosen function $R \leftarrow_\$ \mathsf{Func}(\mathcal{X}, \mathcal{Y})$, is defined as

$$\mathbf{Adv}_F^{\mathrm{PRF}}(\mathsf{A}) = \left|\Pr\left[K \leftarrow_\$ \mathcal{K} : \mathsf{A}^{F_K} = 1\right] - \Pr\left[R \leftarrow_\$ \mathsf{Func}(\mathcal{X}, \mathcal{Y}) : \mathsf{A}^R = 1\right]\right|.$$

Block Ciphers and Tweakable Block Ciphers. A block cipher with key space $\mathcal{K}$ and message space $\mathcal{X}$ is a mapping $E : \mathcal{K} \times \mathcal{X} \to \mathcal{X}$ such that for any key $K \in \mathcal{K}$, $X \mapsto E(K, X)$ is a permutation of $\mathcal{X}$. We write $E_K(X)$ for $E(K, X)$. The security proofs for block cipher-based constructions studied in this paper will be done in the ideal cipher model (ICM): this means that a block cipher E is drawn uniformly at random from the set of all block ciphers with key space $\mathcal{K}$ and message space $\mathcal{X}$, and given as an oracle (both in the encryption and decryption directions) to the adversary.

A tweakable permutation with tweak space $\mathcal{W}$ and message space $\mathcal{X}$ is a mapping $\widetilde{P} : \mathcal{W} \times \mathcal{X} \to \mathcal{X}$ such that, for any tweak $W \in \mathcal{W}$, $X \mapsto \widetilde{P}(W, X)$ is a permutation of $\mathcal{X}$. We let $\mathsf{Perm}(\mathcal{W}, \mathcal{X})$ denote the set of all tweakable permutations with tweak space $\mathcal{W}$ and message space $\mathcal{X}$. As in the case of simple permutations, we let $\mathsf{Perm}(\mathcal{W}, n)$ denote the set of all tweakable permutations with tweak space $\mathcal{W}$ and message space $\{0, 1\}^n$.

A tweakable block cipher $\widetilde{E}$ with key space $\mathcal{K}$, tweak space $\mathcal{W}$ and message space $\mathcal{X}$ is a mapping $\widetilde{E} : \mathcal{K} \times \mathcal{W} \times \mathcal{X} \to \mathcal{X}$ such that for any key $K \in \mathcal{K}$, $(W, X) \mapsto \widetilde{E}(K, W, X)$ is a tweakable permutation with tweak space $\mathcal{W}$ and message space $\mathcal{X}$. We write $E_K(T, X)$ or $E_K^T(X)$ for $E(K, T, X)$. A (q, t)-adversary against the security of $\widetilde{E}$ as a tweakable pseudorandom permutation (TPRP-security) is an algorithm A with oracle access to a tweakable permutation with tweak space $\mathcal{W}$ and message space $\mathcal{X}$, making at most q oracle queries, running in time at most t, and outputting a single bit. The advantage of A in breaking the TPRP-security of $\widetilde{E}$ is defined as

$$\mathbf{Adv}_{\widetilde{E}}^{\mathrm{TPRP}}(\mathsf{A}) = \left|\Pr\left[K \leftarrow_\$ \mathcal{K} : \mathsf{A}^{\widetilde{E}_K} = 1\right] - \Pr\left[\widetilde{P} \leftarrow_\$ \mathsf{Perm}(\mathcal{W}, \mathcal{X}) : \mathsf{A}^{\widetilde{P}} = 1\right]\right|.$$

Note that we do not require the "strong TPRP"-security for E, i.e., when the adversary is allowed to adaptively query an encryption and a decryption oracle, since the underlying tweakable block cipher in our construction will only be queried in one direction.

MACs. We define three security notions for MACs: stateless and deterministic MACs (SD-MACs), nonce-based MACs, and randomized MACs.

Definition 1 (SD-MAC)**.** Let $\mathcal{K}$, $\mathcal{M}$, and $\mathcal{T}$ be non-empty sets. Let $F : \mathcal{K} \times \mathcal{M} \to \mathcal{T}$ be a keyed function. For $K \in \mathcal{K}$, let Ver_K be the *verification* oracle which takes as input a pair $(M, T) \in \mathcal{M} \times \mathcal{T}$ and returns 1 ("accept") if $F_K(K, M) = T$, and 0 ("reject") otherwise. A (q_m, q_v, t)-adversary against the sdMAC-security of F is an adversary A with oracle access to the two oracles F_K and Ver_K for $K \in \mathcal{K}$, making at most q_m "MAC" queries to its first oracle and at most q_v "verification" queries to its second oracle, and running in time at most t. We say that A forges if any of its queries to Ver_K returns 1. The advantage of A against the sdMAC-security of F is defined as

$$\mathbf{Adv}_F^{\mathrm{sdMAC}}(\mathsf{A}) = \Pr\left[K \leftarrow_\$ \mathcal{K} : \mathsf{A}^{F_K, \mathsf{Ver}_K} \text{ forges}\right],$$

where the probability is also taken over the random coins of A, if any. The adversary is not allowed to ask a verification query (M, T) if a previous query M to F_K returned T.

Given four non-empty sets $\mathcal{K}$, $\mathcal{N}$, $\mathcal{M}$, and $\mathcal{T}$, a nonce-based keyed function with key space $\mathcal{K}$, nonce space $\mathcal{N}$, message space $\mathcal{M}$ and range $\mathcal{T}$ is simply a function $F : \mathcal{K} \times \mathcal{N} \times \mathcal{M} \to \mathcal{T}$. Stated otherwise, it is a keyed function whose domain is a cartesian product $\mathcal{N} \times \mathcal{M}$. We write $F_K(N, M)$ for $F(K, N, M)$. Given an adversary with oracle access to F_K for some key K, the *multiplicity* μ of a nonce N in an attack is the number of times it is used in oracle queries to F_K (e.g., $\mu = 1$ for all nonces for a nonce-respecting adversary).

Definition 2 (Nonce-Based/Randomized MAC)**.** Let $\mathcal{K}$, $\mathcal{N}$, $\mathcal{M}$, and $\mathcal{T}$ be non-empty sets. Let $F : \mathcal{K} \times \mathcal{N} \times \mathcal{M} \to \mathcal{T}$ be a nonce-based keyed function. For $K \in \mathcal{K}$, let Ver_K be the *verification* oracle which takes as input a triple $(N, M, T) \in \mathcal{N} \times \mathcal{M} \times \mathcal{T}$ and returns 1 ("accept") if $F_K(N, M) = T$, and 0 ("reject") otherwise.

- A (μ, q_m, q_v, t)-adversary against the nonce-based MAC-security of F is an adversary A with oracle access to the two oracles F_K and Ver_K for $K \in \mathcal{K}$, making at most q_m MAC queries to its first oracle with maximal nonce multiplicity at most μ and at most q_v verification queries to its second oracle, and running in time at most t. We say that A forges if any of its queries to Ver_K returns 1. The advantage of A against the nonce-based MAC-security of F is defined as

  $$\mathbf{Adv}_F^{\mathrm{nMAC}}(\mathsf{A}) = \Pr\left[K \leftarrow_\$ \mathcal{K} : \mathsf{A}^{F_K, \mathsf{Ver}_K} \text{ forges}\right],$$

  where the probability is also taken over the random coins of A, if any. The adversary is not allowed to ask a verification query (N, M, T) if a previous query (N, M) to F_K returned T. When $\mu = 1$, we say that A is nonce-respecting, otherwise A is said nonce-misusing.

- For $K \in \mathcal{K}$, let $F_K^\$$ be the probabilistic algorithm which takes as input $M \in \mathcal{M}$, internally generates a uniformly random $N \leftarrow_\$ \mathcal{N}$, computes $T = F_K(N, M)$, and outputs (N, T). A (q_m, q_v, t)-adversary against the randomized MAC-security of F is an adversary A with oracle access to the two oracles $F_K^\$$ and Ver_K for $K \in \mathcal{K}$, making at most q_m MAC queries to its first oracle and at most q_v verification queries to its second oracle, and running in time at most t. We say that A forges if any of its queries to Ver_K returns 1. The advantage of A against the randomized MAC-security of F is defined as

  $$\mathbf{Adv}_F^{\mathrm{rMAC}}(\mathsf{A}) = \Pr\left[K \leftarrow_\$ \mathcal{K} : \mathsf{A}^{F_K^\$, \mathsf{Ver}_K} \text{ forges}\right],$$

  where the probability is also taken over the random coins of $F_K^\$$ and of A, if any. The adversary is not allowed to ask a verification query (N, M, T) if a previous query M to $F_K^\$$ returned (N, T).

For the three notions above, in case the function F is built from a block cipher and the security proof is done in the ideal cipher model, the advantage additionally depends on the number q_e of ideal cipher queries made by the adversary. The notation is modified in the natural way (e.g., we will talk of a (μ, q_e, q_m, q_v, t)-adversary against the nonce-based MAC security of F).

Almost Uniform and AU Hash Functions. We will need the following definitions of almost uniform and almost universal (AU) hash functions.

Definition 3 (Almost Uniform and AU Hash Functions)**.** Let $\varepsilon > 0$, and let $H : \mathcal{K}_h \times \mathcal{X} \to \mathcal{Y}$ be a keyed hash function for three non-empty sets $\mathcal{K}_h$, $\mathcal{X}$, and $\mathcal{Y}$.

- H is said to be ε*-almost uniform* if for any $X \in \mathcal{X}$ and any $Y \in \mathcal{Y}$,

  $$\Pr\left[K_h \leftarrow_\$ \mathcal{K}_h : H_{K_h}(X) = Y\right] \le \varepsilon;$$

- H is said to be *ε-almost universal (ε-AU)* if for any distinct X and $X' \in \mathcal{X}$,

$$\Pr\left[K_h \leftarrow_{\$} \mathcal{K}_h : H_{K_h}(X) = H_{K_h}(X')\right] \le \varepsilon.$$

Remark 1. Recall that for an ε-AU hash function with n-bit outputs one has $\varepsilon \gtrsim 2^{-n}$ [Sti96]. (In fact, ε can be slightly *less* than 2^{-n}, but when the domain $\mathcal{X}$ is much larger than the range $\mathcal{Y}$ it can only be negligibly smaller.) In order to simplify our bounds, we will always assume that the ε-AU hash functions used in our constructions are such that $\varepsilon \ge 2^{-n}$.

2.2 From Nonce-Based to Randomized MACs

Let F be a nonce-based MAC with nonce space $\mathcal{N}$. In some situations, it can be cumbersome to maintain a state on the MAC generation side to avoid repeating nonces. However, as suggested by Definition 2, any nonce-based MAC can easily be turned into a randomized MAC by letting the MAC generation algorithm choose "nonces" uniformly at random. Of course, these are no longer real nonces since they will start to repeat after roughly $|\mathcal{N}|^{1/2}$ adversarial MAC queries. For some schemes (e.g., Wegman-Carter MACs), security might completely collapse as soon as a single nonce is repeated. However, if the original nonce-based scheme is sufficiently resilient to nonce repetition (in particular, if security only degrades linearly with the maximal nonce multiplicity), the resulting randomized scheme will still enjoy good security bounds. This is captured by the following lemma, which holds for MACs provably secure in the standard or ideal cipher model. (Note that for $\mu_0 = 1$, the first term is exactly a birthday term.)

Lemma 1. *Let $F : \mathcal{K} \times \mathcal{N} \times \mathcal{M} \to \mathcal{T}$ be a nonce-based keyed function (potentially constructed from some underlying block cipher E). Then, for any $((q_e), q_m, q_v, t)$-adversary A against the rMAC security of F, and for any integer $\mu_0 \le q_m$, one has*

$$\mathbf{Adv}_F^{\mathrm{rMAC}}(\mathsf{A}) \le \frac{q_m^{\mu_0+1}}{(2|\mathcal{N}|)^{\mu_0}} + \max_{\mathsf{A}'}\left\{\mathbf{Adv}_F^{\mathrm{nMAC}}(\mathsf{A}')\right\},$$

where the maximum is taken over all $(\mu_0, (q_e), q_m, q_v, t)$-adversaries against the nMAC security of F.

Proof. Let us fix a $((q_e), q_m, q_v, t)$-adversary A against the rMAC-security of F. We make the randomness of the MAC oracle $F_K^{\$}$ explicit through a random function $R : \{1, \ldots, q_m\} \to \mathcal{N}$. Let $\mathcal{F}(q_m, \mathcal{N})$ denote the set of every such function. For every function $R \in \mathcal{F}(q_m, \mathcal{N})$, let

$$\mu(R) \stackrel{\text{def}}{=} \max_{i \in \{1,\ldots,q_m\}} |\{j \in \{1, \ldots, q_m\} \ : \ R(j) = R(i)\}|$$

be the maximal multiplicity of any element in the image of R.

We define a $(\mu(R), (q_e), q_m, q_v, t)$-adversary A_R against the nMAC security of F as follows: A_R runs A, answering its verification queries (and potentially its ideal cipher queries) using his own oracles, and answering A's i-th MAC query by querying his own MAC oracle F_K using the same message and nonce $R(i)$, for $i = 1, \ldots, q_m$. Then, one has

$$\begin{aligned}
\mathbf{Adv}_F^{\mathrm{rMAC}}(\mathsf{A}) &= \sum_{R' \in \mathcal{F}(q_m, \mathcal{N})} \Pr\left[R \leftarrow_{\$} \mathcal{F}(q_m, \mathcal{N}) \ : \ R = R' \text{ and } \mathsf{A}_{R'}^{(E), F_K, \mathsf{Ver}_K} \text{ forges}\right] \\
&= \sum_{R' \in \mathcal{F}(q_m, \mathcal{N})} \Pr\left[R = R'\right] \cdot \Pr\left[\mathsf{A}_{R'}^{(E), F_K, \mathsf{Ver}_K} \text{ forges}\right] \\
&\le \Pr\left[\mu(R) \ge \mu_0 + 1\right] + \frac{1}{|\mathcal{N}|^{q_m}} \sum_{\substack{R' \in \mathcal{F}(q_m, \mathcal{N}) \\ \mu(R') \le \mu_0}} \Pr\left[\mathsf{A}_{R'}^{(E), F_K, \mathsf{Ver}_K} \text{ forges}\right]
\end{aligned}$$

$$\leq \Pr\left[\mu(R) \geq \mu_0 + 1\right] + \max_{\substack{R' \in \mathcal{F}(q_m, \mathcal{K}) \\ \mu(R') \leq \mu_0}} \left\{ \mathbf{Adv}_F^{\mathrm{nMAC}}(\mathsf{A}_{R'}) \right\}$$
$$\leq \Pr\left[\mu(R) \geq \mu_0 + 1\right] + \max_{\mathsf{A}'} \left\{ \mathbf{Adv}_F^{\mathrm{nMAC}}(\mathsf{A}') \right\},$$

where the maximum is taken over all $(\mu_0, (q_e), q_m, q_v, t)$-adversaries A' against the nMAC security of F (since for every function R' such that $\mu(R') \leq \mu_0$, $\mathsf{A}_{R'}$ is a $(\mu_0, (q_e), q_m, q_v, t)$-adversary against the nMAC security of F).

We now upper bound the first term. Assume first that $q_m \geq \mu_0 + 1$. Then, one has

$$\Pr\left[\mu(R) \geq \mu_0 + 1\right] = \Pr\left[\exists\, 1 \leq i_1 < \ldots < i_{\mu_0+1} \leq q_m \;:\; F(i_1) = \ldots = F(i_{\mu_0+1})\right]$$
$$\leq \frac{(q_m)_{\mu_0+1}}{(\mu_0+1)! \cdot |\mathcal{N}|^{\mu_0}} \leq \left(\frac{q_m}{|\mathcal{N}|}\right)^{\mu_0} \cdot \frac{q_m}{(\mu_0+1)!} \leq \frac{q_m^{\mu_0+1}}{(2|\mathcal{N}|)^{\mu_0}},$$

where, for the last inequality, we used the fact that $(\mu_0 + 1)! \geq 2^{\mu_0}$ for every integer $\mu_0 \geq 0$. Note that this upper bound also holds when $q_m < \mu_0 + 1$, since in that case $\Pr\left[\mu(R) \geq \mu_0 + 1\right] = 0$. This concludes the proof. $\square$

2.3 The H-Coefficients Technique

In this work, we prove the security of our stateless deterministic and nonce-based MAC constructions using the H-coefficients technique [Pat08b, CS14], which we explain here. The details will be slightly different depending on whether the construction is proven secure in the ideal cipher model or the standard model. We start by describing the formalism for block cipher-based constructions proven secure in the ideal cipher model.

Let $\mathsf{MAC}[E]$ denote a SD or nonce-based MAC construction based on a block cipher $E \in \mathsf{Perm}(\mathcal{K}, n)$. In all the following, nonces N and multiplicity μ will be written in parenthesis to indicate that they are omitted for a SD-MAC. Let $\mathcal{K}'$ denote the key space for $\mathsf{MAC}[E]$, and let $\mathsf{Ver}[E]_{K'}$ be the verification oracle for key $K' \in \mathcal{K}'$. Let A be a $((\mu), q_e, q_m, q_v, t)$-adversary against the sd/nMAC security of $\mathsf{MAC}[E]$ and recall that

$$\mathbf{Adv}_{\mathsf{MAC}[E]}^{\mathrm{(sd/n)MAC}}(\mathsf{A}) = \Pr\left[E \leftarrow_\$ \mathsf{Perm}(\mathcal{K}, n), K' \leftarrow_\$ \mathcal{K}' : \mathsf{A}^{E, \mathsf{MAC}[E]_{K'}, \mathsf{Ver}[E]_{K'}} \text{ forges}\right].$$

It will be more convenient to express this quantity as the advantage of a *distinguisher* trying to distinguish the real world $(E, \mathsf{MAC}[E]_{K'}, \mathsf{Ver}[E]_{K'})$ and an ideal world defined as follows. Let Rand denote a perfectly random oracle with the same domain and range as $\mathsf{MAC}[E]_{K'}$, and let Rej denote an oracle with the same domain as $\mathsf{Ver}[E]_{K'}$ which always returns 0 ("reject"). Since the adversary cannot have the right oracle return 1 in the ideal world (i.e., when interacting with $(E, \mathsf{Rand}, \mathsf{Rej})$), we have

$$\mathbf{Adv}_{\mathsf{MAC}[E]}^{\mathrm{(sd/n)MAC}}(\mathsf{A}) = \Pr\left[\mathsf{A}^{E, \mathsf{MAC}[E]_{K'}, \mathsf{Ver}[E]_{K'}} \text{ forges}\right] - \Pr\left[\mathsf{A}^{E, \mathsf{Rand}, \mathsf{Rej}} \text{ forges}\right].$$

Consider now an adversary D which queries a triplet of oracles $(\mathcal{O}_1, \mathcal{O}_2, \mathcal{O}_3)$ and outputs a bit β, which we write $\mathsf{D}^{\mathcal{O}_1, \mathcal{O}_2, \mathcal{O}_3} = \beta$. (We will refer to such an adversary as a *distinguisher.*) Say that such an adversary is *non-trivial* if it never makes a query $((N), M, T)$ to its right (verification) oracle if a previous query $((N), M)$ to its middle (MAC) oracle returned T. Then

$$\mathbf{Adv}_{\mathsf{MAC}[E]}^{\mathrm{(sd/n)MAC}}(\mathsf{A}) \leq \max_{\mathsf{D}} \left\{ \Pr\left[\mathsf{D}^{E, \mathsf{MAC}[E]_{K'}, \mathsf{Ver}[E]_{K'}} = 1\right] - \Pr\left[\mathsf{D}^{E, \mathsf{Rand}, \mathsf{Rej}} = 1\right] \right\}, \quad (1)$$

where the maximum is taken over non-trivial distinguishers making q_e queries to $\mathcal{O}_1$, q_m queries to $\mathcal{O}_2$ (with maximal nonce multiplicity μ in the case of a nonce-based MAC), and q_v queries to $\mathcal{O}_3$. (This follows easily by considering the particular D which runs A and outputs 1 *iff* A successfully forges.) This formulation of the problem now allows us to use the H-coefficients technique [Pat08b, CS14], as we explain in more details below.

We assume that D is computationally unbounded (and hence *wlog* deterministic) and that it never repeats a query. Let

$$\tau_e = \big((K_1, X_1, Y_1), \ldots, (K_{q_e}, X_{q_e}, Y_{q_e})\big)$$

be the list of ideal cipher queries of D and corresponding answers (i.e., for $1 \le i \le q_e$, D either made a query $E(K_i, X_i)$ and received answer Y_i, or a query $E^{-1}(K_i, Y_i)$ and received answer X_i). Let

$$\tau_m = \big(((N_1), M_1, T_1), \ldots, ((N_{q_m}), M_{q_m}, T_{q_m})\big)$$

be the list of MAC queries of D and corresponding answers. Let also

$$\tau_v = \big(((N'_1), M'_1, T'_1, b_1), \ldots, ((N'_{q_v}), M'_{q_v}, T'_{q_v}, b_{q_v})\big)$$

be the list of verification queries of D and corresponding answers (with $b_i \in \{0,1\}$). The triple (τ_e, τ_m, τ_v) constitutes the *queries transcript* of the attack. In order to have a simple description of bad transcripts, we slightly modify the security experiment by revealing to the distinguisher (after the interaction but before it outputs its decision bit) the secret key K' if we are in the real world, or a uniformly random "dummy" key K' if we are in the ideal world (this is obviously *wlog* since the distinguisher can ignore this additional piece of information). All in all, the *transcript* of the attack is the tuple $\tau = (\tau_e, \tau_m, \tau_v, K')$.

A transcript τ is said *attainable* (with respect to distinguisher D) if the probability to obtain this transcript in the ideal world is non-zero. Let Θ denote the set of attainable transcripts. We also let X_{re}, resp. X_{id}, denote the probability distribution of the transcript τ induced by the real world, resp. the ideal world. Then the main lemma of the H-coefficients technique allows one to upper bound D's distinguishing advantage as follows (see for example [CS14] or [CLL+14] for the proof).

Lemma 2 ([Pat08b])**.** *Fix a distinguisher* D*. Let* $\Theta = \Theta_{\mathrm{good}} \sqcup \Theta_{\mathrm{bad}}$ *be a partition of the set of attainable transcripts. Assume that there exists* ε_1 *such that for any* $\tau \in \Theta_{\mathrm{good}}$*, one has*[3]

$$\frac{\Pr[X_{\mathrm{re}} = \tau]}{\Pr[X_{\mathrm{id}} = \tau]} \ge 1 - \varepsilon_1,$$

and that there exists ε_2 *such that* $\Pr[X_{\mathrm{id}} \in \Theta_{\mathrm{bad}}] \le \varepsilon_2$*. Then* $\mathbf{Adv}(\mathsf{D}) \le \varepsilon_1 + \varepsilon_2$*.*

Note that for an attainable transcript $\tau = (\tau_e, \tau_m, \tau_v, K')$, any verification query $((N'_i), M'_i, T'_i, b_i) \in \tau_v$ is such that $b_i = 0$. Hence, some transcripts are attainable in the real world but not in the ideal world, which is unusual as, in most H-coefficients-based proofs, the set of transcripts attainable in the real world is a subset of those attainable in the ideal world. However, the standard proof of Lemma 2 can be trivially extended to handle this peculiarity (see Appendix A). In order to simplify the notation, in all the following, since we only deal with attainable transcripts, we omit decision bits b_i from the verification queries transcript and simply write

$$\tau_v = \big(((N'_1), M'_1, T'_1), \ldots, ((N'_{q_v}), M'_{q_v}, T'_{q_v})\big).$$

Since security in the ideal cipher model for block cipher-based constructions hold against computationally unbounded adversaries, we omit parameter t from theorem statements. For TBC-based constructions proven secure in the standard model, the formalism is identical, except that there is no ideal cipher queries transcript τ_e and hence parameter q_e is irrelevant.

[3] Recall that for an attainable transcript, one has $\Pr[X_{\mathrm{id}} = \tau] > 0$.

Note that in our proofs we see the forgery game for a MAC construction as a particular case of a distinguishing game against the construction. Then, we lower bound the advantage of *any* distinguisher against the construction. Hence, when the number q_v of verification queries is fixed to 0, our security bounds correspond to the PRF security of our constructions.

2.4 Permutation (In)equalities List

Fix any non-empty set $S = \{s_1, \ldots, s_r\}$. At some point in our security proofs, we will need to evaluate the probability that a certain family $(P_s)_{s\in S} \in \mathsf{Perm}(n)^S$ of uniformly random and independent permutations[4] satisfies some sets of equalities and inequalities. To this end, we introduce the notion of *permutation equalities list* and *permutation inequalities list.* A permutation equalities list is a set λ_{eq} of triples $(s, x, y) \in S \times \{0,1\}^n \times \{0,1\}^n$ such that, for any pair of distinct triple $(s, x, y), (s', x', y') \in \lambda_{\mathrm{eq}}$, if $s = s'$, then $x \neq x'$ and $y \neq y'$. A permutation inequalities list is a set λ_{ineq} of triples $(s', x', y') \in S \times \{0,1\}^n \times \{0,1\}^n$, and it is said *compatible with the permutations equalities list* λ_{eq} if $\lambda_{\mathrm{eq}} \cap \lambda_{\mathrm{ineq}} = \emptyset$.

Fix any permutation equalities list λ_{eq} and any permutation inequalities list λ_{ineq} which is compatible with λ_{eq}. A family of permutations $(P_s)_{s\in S}$ is said compatible with λ_{eq} if, for any $(s, x, y) \in \lambda_{\mathrm{eq}}$, one has $P_s(x) = y$. It is said compatible with λ_{ineq} if, for every $(s, x, y) \in \lambda_{\mathrm{ineq}}$, one has $P_s(x) \neq y$. Finally, we say that $(P_s)_{s\in S}$ is compatible with $\lambda = (\lambda_{\mathrm{eq}}, \lambda_{\mathrm{ineq}})$ if it is both compatible with λ_{eq} and λ_{ineq}. We let $\mathsf{Comp}(\lambda_{\mathrm{eq}})$, $\mathsf{Comp}(\lambda_{\mathrm{ineq}})$ and $\mathsf{Comp}(\lambda)$ denote the set of families of permutations that are compatible with respectively λ_{eq}, λ_{ineq} and λ. Then one has the following lemma.

Lemma 3. *Let $S = \{s_1, \ldots, s_r\}$, λ_{eq} be a permutation equalities list, and λ_{ineq} be a permutation inequalities list compatible with λ_{eq}. Let $q = |\lambda_{\mathrm{eq}}|$ and $q' = |\lambda_{\mathrm{ineq}}|$. Assume that $q < 2^n$ and $q' < 2^n$. For $i = 1, \ldots, r$, let q_i be the number of $(s, x, y) \in \lambda_{\mathrm{eq}}$ such that $s = s_i$. Then, one has*

$$\Pr\left[(P_s) \leftarrow_{\$} \mathsf{Perm}(n)^S \;:\; (P_s) \in \mathsf{Comp}(\lambda)\right] \geq \frac{1}{\prod_{i=1}^{r} (2^n)_{q_i}} \left(1 - \frac{q'}{2^n - \max\{q_1, \ldots, q_r\}}\right).$$

Proof. We are going to consider the permutation equalities list and the permutation inequalities list in turn.

First, we lower bound the probability that a random family $(P_s)_{s\in S}$ of permutations satisfies

$$\forall (s, x, y) \in \lambda_{\mathrm{eq}}, P_s(x) = y.$$

Since λ_{eq} is a permutation equalities list, each permutation P_{s_i} must satisfy exactly q_i equalities. Thus one has

$$\Pr\left[(P_s)_{s\in S} \in \mathsf{Comp}(\lambda_{\mathrm{eq}})\right] = \frac{1}{\prod\limits_{i=1}^{r} (2^n)_{q_i}}. \tag{2}$$

We will now lower bound the probability that a random family $(P_s)_{s\in S}$ of permutations is compatible with λ_{ineq}, conditioned on $(P_s)_{s\in S}$ being compatible with λ_{eq}. It will actually be easier to upper bound the probability that $(P_s)_{s\in S}$ is *not* compatible with λ_{ineq}, i.e., that there exists $(s', x', y') \in \lambda_{\mathrm{ineq}}$ such that

$$P_{s'}(x') = y'. \tag{3}$$

Fix any permutation inequality $(s', x', y') \in \lambda_{\mathrm{ineq}}$. We consider two possible cases:

[4] For block cipher-based constructions, $(P_s)_{s\in S}$ will be a block cipher, and for TBC-based constructions, $(P_s)_{s\in S}$ will be a tweakable permutation.

1. if there exists $(s, x, y) \in \lambda_{\text{eq}}$ such that $s = s'$ and $x = x'$ or $y = y'$, then, since by definition of λ_{ineq} being compatible with λ_{eq} one has $(s, x, y) \neq (s', x', y')$, Equation (3) cannot hold;

2. otherwise, Equation (3) holds with probability $1/(2^n - m)$, where m is the number of times s' appears in λ_{eq}, which cannot be larger than $\max\{q_1, \dots, q_r\}$.

Hence, we see that for any $(s', x', y') \in \lambda_{\text{ineq}}$, Equation (3) is satisfied with probability at most $1/(2^n - \max\{q_1, \dots, q_r\})$. By a union bound over the q' permutation inequalities, we obtain that

$$\Pr\left[(P_s)_{s\in S} \in \mathsf{Comp}(\lambda_{\text{ineq}}) \,|\, (P_s)_{s\in S} \in \mathsf{Comp}(\lambda_{\text{eq}})\right] \geq 1 - \frac{q'}{2^n - \max\{q_1, \dots, q_r\}}. \quad (4)$$

Using Equation (2) and Equation (4), we have

$$\Pr\left[(P_s)_{s\in S} \in \mathsf{Comp}(\lambda)\right] \geq \frac{1}{\prod_{i=1}^{r} (2^n)_{q_i}} \left(1 - \frac{q'}{2^n - \max\{q_1, \dots, q_r\}}\right). \qquad \square$$

3 Tweakable Block Cipher-Based Constructions

In this section, we describe and analyze two TBC-based MAC constructions: a nonce-based one called NaT and a stateless deterministic one called HaT. The security proofs are done in the standard model (i.e., they do not require to idealize the underlying TBC).

3.1 The Nonce-Based Construction NaT

We start with a nonce-based construction named NaT (*Nonce-as-Tweak*). Given a TBC $\widetilde{E}$ with key space $\mathcal{K}$, tweak space $\mathcal{W}$, and message space $\{0,1\}^n$ and a keyed hash function H with key space $\mathcal{K}_h$, domain $\mathcal{M}$, and range $\{0,1\}^n$, we define a MAC with key space $\mathcal{K} \times \mathcal{K}_h$, nonce space $\mathcal{W}$, and message space $\mathcal{M}$ as

$$\mathsf{NaT}[\widetilde{E}, H]_{K,K_h}(N, M) = \widetilde{E}_K^N(H_{K_h}(M)).$$

Our security result is the following one.

Theorem 1. *Let $\mathcal{M}$, $\mathcal{K}$, $\mathcal{W}$ and $\mathcal{K}_h$ be non-empty sets. Let $\widetilde{E} : \mathcal{K} \times \mathcal{W} \times \{0,1\}^n \to \{0,1\}^n$ be a tweakable block cipher and $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ be an ε-AU hash function. Let μ, q_m, q_v, and t be integers such that $q_m, q_v \leq 2^n$ and $\mu \leq \min\{q_m, 2^n - 1\}$. Then for any (μ, q_m, q_v, t)-adversary A against the nMAC-security of $\mathsf{NaT}[\widetilde{E}, H]$, there exists a $(q_m + q_v, t')$-adversary A' against the* TPRP*-security of $\widetilde{E}$, where $t' = O(t + (q_m + q_v)t_H)$ and t_H is an upper bound on the time to compute H on any message, such that*

$$\mathbf{Adv}^{\mathsf{MAC}}_{\mathsf{NaT}[\widetilde{E},H]}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{TPRP}}_{\widetilde{E}}(\mathsf{A}') + 2(\mu - 1)q_m\varepsilon + \frac{q_v}{2^n - \mu} + \mu q_v \varepsilon.$$

Recall that for an ε-AU hash function with n-bit outputs one has $\varepsilon \gtrsim 2^{-n}$ [Sti96] (see Remark 1). Hence, in the nonce-respecting case (i.e., $\mu = 1$), the NaT construction is secure up to roughly ε^{-1} verification queries, irrespectively of the number of MAC queries (neglecting the effect of the TPRP-advantage term). When A can freely choose nonces (i.e., $\mu = q_m$), then the NaT construction is secure up to the birthday bound. The security bound degrades linearly with the maximal multiplicity μ of nonces.

As a corollary, we obtain the following for the security of the NaT construction as a randomized MAC, which shows that it is secure up to roughly ε^{-1}/n MAC and verification queries under the additional assumption that $|\mathcal{W}| \geq 2^n$.

Corollary 1. *Let $\mathcal{M}$, $\mathcal{K}$, $\mathcal{W}$ and $\mathcal{K}_h$ be non-empty sets. Let $\widetilde{E} : \mathcal{K} \times \mathcal{W} \times \{0,1\}^n \to \{0,1\}^n$ be a tweakable block cipher and $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ be an ε-AU hash function. Let q_m, q_v, and t be integers such that $q_m, q_v \leq 2^n$. Then for any (q_m, q_v, t)-adversary A against the rMAC-security of $\mathsf{NaT}[\widetilde{E}, H]$, there exists a $(q_m + q_v, t')$-adversary A' against the* TPRP*-security of $\widetilde{E}$, where $t' = O(t + (q_m + q_v)t_H)$ and t_H is an upper bound on the time to compute H on any message, such that*

$$\mathbf{Adv}^{\mathrm{rMAC}}_{\mathsf{NaT}[\widetilde{E},H]}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{TPRP}}_{\widetilde{E}}(\mathsf{A}') + 2(n-1)q_m\varepsilon + \frac{q_v}{2^n - n} + nq_v\varepsilon + \frac{q_m^{n+1}}{(2|\mathcal{W}|)^n}.$$

Proof. This follows by combining Lemma 1 with $\mu_0 = n$ and Theorem 1. □

The remaining of this section is devoted to the proof of Theorem 1. Let us fix a (μ, q_m, q_v, t)-adversary A against the nMAC-security of $\mathsf{NaT}[\widetilde{E}, H]$.

The first step of the proof is standard and consists in replacing $\widetilde{E}_K$ by a uniformly random tweakable permutation $\widetilde{P}$, both in the MAC and in the verification oracles (in other words, we replace the tweakable block cipher $\widetilde{E}$ by the *perfect tweakable block cipher cipher* $\widetilde{E}^*$ whose key space is the set of all tweakable permutations of $\{0,1\}^n$ with tweak space $\mathcal{W}$). Let $\mathsf{NaT}[\widetilde{E}^*, H]$ denote the resulting construction. It is easy to show that there exists an adversary A' against the TPRP-security of $\widetilde{E}$, making at most $q_m + q_v$ oracle queries and running in time at most $O(t + (q_m + q_v)t_H)$, such that

$$\mathbf{Adv}^{\mathrm{nMAC}}_{\mathsf{NaT}[\widetilde{E},H]}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{TPRP}}_{\widetilde{E}}(\mathsf{A}') + \mathbf{Adv}^{\mathrm{nMAC}}_{\mathsf{NaT}[\widetilde{E}^*,H]}(\mathsf{A}). \tag{5}$$

The next step is to find an upper bound for

$$\mathbf{Adv}^{\mathrm{nMAC}}_{\mathsf{NaT}[\widetilde{E}^*,H]}(\mathsf{A}) = \Pr\left[\widetilde{P} \leftarrow_\$ \mathsf{Perm}(\mathcal{W}, n), K_h \leftarrow_\$ \mathcal{K}_h : \mathsf{A}^{\mathsf{NaT}[\widetilde{P},H]_{K_h}, \mathsf{Ver}[\widetilde{P},H]_{K_h}} \text{ forges}\right],$$

where, overloading the notation, $\mathsf{NaT}[\widetilde{P}, H]_{K_h}$ denotes construction $\mathsf{NaT}[\widetilde{E}^*, H]$ instantiated with tweakable permutation $\widetilde{P}$ and hashing key K_h and $\mathsf{Ver}[\widetilde{P}, H]_{K_h}$ denotes the corresponding verification oracle. This is now a purely information-theoretic problem, and we can follow the H-coefficients technique as explained in Section 2.3.

Let us fix a non-trivial (μ, q_m, q_v)-distinguisher D interacting either with the real world $(\mathsf{NaT}[\widetilde{P}, H]_{K_h}, \mathsf{Ver}[\widetilde{P}, H]_{K_h})$ or with the ideal world $(\mathsf{Rand}, \mathsf{Rej})$. We let

$$\mathbf{Adv}(\mathsf{D}) = \Pr\left[\mathsf{D}^{\mathsf{NaT}[\widetilde{P},H]_{K_h}, \mathsf{Ver}[\widetilde{P},H]_{K_h}} = 1\right] - \Pr\left[\mathsf{D}^{\mathsf{Rand},\mathsf{Rej}} = 1\right].$$

Let $\tau = (\tau_m, \tau_v, K_h)$ denote the transcript of the attack, with

$$\tau_m = \big((N_1, M_1, T_1), \ldots, (N_{q_m}, M_{q_m}, T_{q_m})\big)$$
$$\tau_v = \big((N_1', M_1', T_1'), \ldots, (N_{q_v}', M_{q_v}', T_{q_v}')\big).$$

Recall that Θ denotes the set of attainable transcripts and X_{re}, resp. X_{id}, the probability distribution of the transcript τ induced by the real world, resp. the ideal world.

The remaining of the proof of Theorem 1 is structured as follows. First, we define bad transcripts and upper bound their probability in the ideal world (Lemma 4). Then, we analyze good transcripts and prove that they are almost as likely in the real and the ideal world (Lemma 5). Combining Lemma 2 from Section 2.3 with Lemma 4 and Lemma 5 gives us an upper bound on D's advantage, which by Equation (1) from Section 2.3 gives us an upper bound on

$$\mathbf{Adv}^{\mathrm{nMAC}}_{\mathsf{NaT}[\widetilde{E}^*,H]}(\mathsf{A}).$$

Theorem 1 follows easily by combining this upper bound with Equation (5).

We start by defining bad transcripts.

Definition 4. We say that an attainable transcript $\tau = (\tau_m, \tau_v, K_h)$ is *bad* if one of these conditions is fulfilled:

(C-1) there exists two distinct MAC queries (N_i, M_i, T_i) and (N_j, M_j, T_j) such that $N_i = N_j$ and either $H_{K_h}(M_i) = H_{K_h}(M_j)$ or $T_i = T_j$;

(C-2) there exists a MAC query $(N_i, M_i, T_i) \in \tau_m$ and a verification query $(N'_j, M'_j, T'_j) \in \tau_v$ such that

$$\begin{cases} N_i = N'_j \\ H_{K_h}(M_i) = H_{K_h}(M'_j) \\ T_i = T'_j. \end{cases}$$

We let Θ_{bad}, resp. Θ_{good} denote the set of bad, respectively good transcripts.

Note that the second condition can only happen in the ideal world since in the real world, if $N_i = N'_j$, $H_{K_h}(M_i) = H_{K_h}(M'_j)$, and $T_i = T'_j$, the verification oracle should return 1 on query (N'_j, M'_j, T'_j) (which is impossible for an attainable transcript).

We now upper bound the probability to get a bad transcript in the ideal world.

Lemma 4. *For any integers q_m and q_v, one has*

$$\Pr[X_{\text{id}} \in \Theta_{\text{bad}}] \leq 2(\mu - 1)q_m\varepsilon + \mu q_v \varepsilon.$$

Proof. We let Θ_i denote the set of attainable transcripts satisfying condition (C-i). Recall that, in the ideal world, K_h is drawn independently from the queries transcript. We are going to consider both conditions in turn.

Condition (C-1). Fix a MAC query (N_i, M_i, T_i). There are exactly q_m possible choices for this query. Then we fix another MAC query (N_j, M_j, T_j) such that $N_i = N_j$ (there are at most $\mu - 1$ possible choices). The probability, over the random draw of T_i and T_j that $T_i = T_j$ is 2^{-n}, and the probability, over the random draw of K_h, that $H_{K_h}(M_i) = H_{K_h}(M_j)$, is lower than ε. Summing over every possible choice of (N_i, M_i, T_i) and (N_j, M_j, T_j), we get

$$\Pr[X_{\text{id}} \in \Theta_1] \leq \frac{(\mu - 1)q_m}{2^n} + (\mu - 1)q_m\varepsilon \leq 2(\mu - 1)q_m\varepsilon,$$

where we used that $\varepsilon \geq 2^{-n}$ (see Remark 1).

Condition (C-2). We consider any verification query $(N'_j, M'_j, T'_j) \in \tau_v$ and upper bound the probability that this condition is satisfied for this particular query. By definition of the multiplicity, there are at most μ MAC queries (N_i, M_i, T_i) such that $N_i = N'_j$. Fix any of these queries. We distinguish two cases:

- If the verification query comes after the MAC query, then since the distinguisher is non-trivial, either $T_i \neq T'_j$, or $M_i \neq M'_j$. In the former case, the condition cannot be satisfied, while in the latter case, the probability over the random draw of K_h that $H_{K_h}(M_i) = H_{K_h}(M'_j)$ is at most ε.
- If the MAC query comes after the verification query, then T_i is random and independent from T'_j and the probability that $T_i = T'_j$ is 2^{-n}.

Since $\varepsilon \geq 2^{-n}$ (see Remark 1), we see that in all cases the condition is met with probability at most ε. Thus, by summing over every verification query, and every MAC query using the same nonce as the verification query, one has

$$\Pr[X_{\text{id}} \in \Theta_2] \leq \mu q_v \varepsilon.$$

The result follows by a union bound over these conditions. □

We now analyze good transcripts and prove the following lemma.

Lemma 5. *For any good transcript τ, one has*

$$\frac{\Pr\left[X_{\mathrm{re}}=\tau\right]}{\Pr\left[X_{\mathrm{id}}=\tau\right]} \geq 1-\frac{q_v}{2^n-\mu}.$$

Proof. Let $\tau=(\tau_m,\tau_v,K_h)$ be a good transcript. Let $\mathcal{L}=\{N_1,\ldots,N_{q_m}\}$ be the set of all nonces used in MAC queries. Using any arbitrary order, we rewrite the set $\mathcal{L}$ as

$$\mathcal{L}=\{L_1,\ldots,L_r\},$$

where r is the total number of distinct values in $\mathcal{L}$. For $i=1,\ldots,r$, we let q_i denote the multiplicity of nonce N_i in τ_m. Note that $q_i \leq \mu$ for $i=1,\ldots,r$.

Since in the ideal world the MAC oracle is perfectly random and the verification always rejects, one simply has

$$\Pr[X_{\mathrm{id}}=\tau]=\frac{1}{|\mathcal{K}_h|\cdot(2^n)^{q_m}}. \tag{6}$$

We must now lower bound the probability of getting τ in the real world. We say that a tweakable permutation $\widetilde{P}$ is compatible with τ_m if

$$\forall i\in\{1,\ldots,q_m\},\ \mathsf{NaT}[\widetilde{P},H]_{K_h}(N_i,M_i)=T_i,$$

and compatible with τ_v if

$$\forall i\in\{1,\ldots,q_v\},\ \mathsf{NaT}[\widetilde{P},H]_{K_h}(N'_i,M'_i)\neq T'_i.$$

We simply say that $\widetilde{P}$ is compatible with τ if it is compatible with τ_m and τ_v. We let $\mathsf{Comp}(\tau_m)$, $\mathsf{Comp}(\tau_v)$, and $\mathsf{Comp}(\tau)$ denote the set of tweakable permutations that are compatible with respectively τ_m, τ_v, and τ. Then one can easily check (see for example [CS14] for a detailed explanation) that

$$\Pr[X_{\mathrm{re}}=\tau]=\frac{1}{|\mathcal{K}_h|}\cdot\Pr\left[\widetilde{P}\leftarrow_{\$}\mathsf{Perm}(\mathcal{W},n):\widetilde{P}\in\mathsf{Comp}(\tau)\right]. \tag{7}$$

We now define

$$\begin{aligned}\lambda_{\mathrm{eq}}&=\{(N_1,H_{K_h}(M_1),T_1),\ldots,(N_{q_m},H_{K_h}(M_{q_m}),T_{q_m})\}\\ \lambda_{\mathrm{ineq}}&=\{(N'_1,H_{K_h}(M'_1),T'_1),\ldots,(N'_{q_v},H_{K_h}(M'_{q_v}),T'_{q_v})\}.\end{aligned}$$

Then, since τ is a good transcript, λ_{eq} is a permutation equalities list[5] (otherwise condition (C-1) defining bad transcripts would be met), and λ_{ineq} is a permutation inequalities list which is compatible with λ_{eq} (otherwise condition (C-2) would be met). Moreover, $|\lambda_{\mathrm{eq}}|=q_m$ and $|\lambda_{\mathrm{ineq}}|=q_v$. Note that the event $\widetilde{P}\in\mathsf{Comp}(\tau)$ is actually equivalent to the event $\widetilde{P}\in\mathsf{Comp}(\lambda)$ where $\lambda=(\lambda_{\mathrm{eq}},\lambda_{\mathrm{ineq}})$. Using Lemma 3, one has

$$\Pr\left[\widetilde{P}\in\mathsf{Comp}(\tau)\right]\geq\frac{1}{\prod_{i=1}^r(2^n)_{q_i}}\left(1-\frac{q_v}{2^n-\mu}\right).$$

Combining this equation with Equation (6) and Equation (7), and using the fact that $q_m=\sum_{i=1}^r q_i$, we get

$$\frac{\Pr\left[X_{\mathrm{re}}=\tau\right]}{\Pr\left[X_{\mathrm{id}}=\tau\right]}\geq\left(1-\frac{q_v}{2^n-\mu}\right)\cdot\prod_{i=1}^r\underbrace{\frac{(2^n)^{q_i}}{(2^n)_{q_i}}}_{\geq 1}\geq 1-\frac{q_v}{2^n-\mu}.$$ □

[5]Refer to Section 2.4 for the definition of permutation (in)equalities lists.

3.2 The Stateless Deterministic Construction HaT

Our second TBC-based construction is a stateless deterministic MAC called HaT (*Hash-as-Tweak*). Given a TBC $\widetilde{E} : \mathcal{K} \times \mathcal{W} \times \{0,1\}^n \to \{0,1\}^n$ and two keyed hash functions $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ and $H' : \mathcal{K}'_h \times \mathcal{M} \to \mathcal{W}$, we define a MAC with key space $\mathcal{K} \times \mathcal{K}_h \times \mathcal{K}'_h$ and message space $\mathcal{M}$ as

$$\mathsf{HaT}[\widetilde{E}, H, H']_{K,K_h,K'_h}(M) = \widetilde{E}_K^{H'_{K'_h}(M)}(H_{K_h}(M)).$$

Then the following result holds.

Theorem 2. *Let $\mathcal{M}$, $\mathcal{W}$, $\mathcal{K}$, $\mathcal{K}_h$, and $\mathcal{K}'_h$ be non-empty sets. Let $\widetilde{E} : \mathcal{K} \times \mathcal{W} \times \{0,1\}^n \to \{0,1\}^n$ be a tweakable block cipher and let $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ and $H' : \mathcal{K}'_h \times \mathcal{M} \to \mathcal{W}$ be two ε-AU hash functions. Let q_m, q_v, and t be integers such that $q_m < 2^n$. Then for any (q_m, q_v, t)-adversary A against the sdMAC-security of $\mathsf{HaT}[\widetilde{E}, H, H']$, there exists a $(q_m + q_v, t')$-adversary A' against the* TPRP*-security of $\widetilde{E}$, where $t' = O(t + (q_m + q_v)t_H)$ and t_H is an upper bound on the time to compute H or H' on any message, such that*

$$\mathbf{Adv}^{\mathrm{sdMAC}}_{\mathsf{HaT}[\widetilde{E},H]}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{TPRP}}_{\widetilde{E}}(\mathsf{A}') + q_m^2\varepsilon^2 + q_m q_v \varepsilon^2 + \frac{q_v}{2^n - q_m}.$$

Hence, the HaT construction is secure up to roughly $q_m \simeq \varepsilon^{-1}$ MAC queries and $q_v \simeq \min\{2^n, \varepsilon^{-2}/q_m\}$ verification queries.

The remaining of this section is devoted to the proof of Theorem 2. Let us fix a (q_m, q_v, t)-adversary A against the MAC-security of $\mathsf{HaT}[\widetilde{E}, H, H']$.

The first step of the proof is standard and consists in replacing $\widetilde{E}_K$ by a tweakable permutation $\widetilde{P}$ both in the MAC and in the verification oracles (in other words, we replace the tweakable block cipher $\widetilde{E}$ by the *perfect tweakable block cipher cipher* $\widetilde{E}^*$ whose key space is the set of all tweakable permutations of $\{0,1\}^n$ with tweak space $\mathcal{N}$). Let $\mathsf{HaT}[\widetilde{E}^*, H, H']$ denote the resulting construction. It is easy to show that there exists an adversary A' against the TPRP-security of $\widetilde{E}$, making at most $q_m + q_v$ oracle queries and running in time at most $O(t + (q_m + q_v)t_H)$, such that

$$\mathbf{Adv}^{\mathrm{sdMAC}}_{\mathsf{HaT}[\widetilde{E},H,H']}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{TPRP}}_{\widetilde{E}}(\mathsf{A}') + \mathbf{Adv}^{\mathrm{sdMAC}}_{\mathsf{HaT}[\widetilde{E}^*,H,H']}(\mathsf{A}). \tag{8}$$

The next step is to find an upper bound for

$$\mathbf{Adv}^{\mathrm{sdMAC}}_{\mathsf{HaT}[\widetilde{E}^*,H,H']}(\mathsf{A}) = \Pr\Big[\widetilde{P} \leftarrow_\$ \mathsf{Perm}(\mathcal{N}, n), (K_h, K'_h) \leftarrow_\$ \mathcal{K}_h \times \mathcal{K}'_h : \\ \mathsf{A}^{\mathsf{HaT}[\widetilde{P},H,H']_{K_h,K'_h}, \mathsf{Ver}[\widetilde{P},H,H']_{K_h,K'_h}} \text{ forges}\Big],$$

where, slightly overloading the notation, $\mathsf{HaT}[\widetilde{P}, H, H']_{K_h,K'_h}$ denotes the construction $\mathsf{HaT}[\widetilde{E}^*, H, H']$ instantiated with tweakable permutation $\widetilde{P}$ and hashing keys (K_h, K'_h) and $\mathsf{Ver}[\widetilde{P}, H, H']_{K_h,K'_h}$ denotes the corresponding verification oracle. This is now a purely information-theoretic problem, and we can follow the H-coefficients technique as explained in Section 2.3.

Let us fix a non-trivial (q_m, q_v)-distinguisher D interacting either with the real world $(\mathsf{HaT}[\widetilde{P}, H, H']_{K_h,K'_h}, \mathsf{Ver}[\widetilde{P}, H, H']_{K_h,K'_h})$ or with the ideal world $(\mathsf{Rand}, \mathsf{Rej})$. We let

$$\mathbf{Adv}(\mathsf{D}) = \Pr\left[\mathsf{D}^{\mathsf{HaT}[\widetilde{P},H,H']_{K_h,K'_h}, \mathsf{Ver}[\widetilde{P},H,H']_{K_h,K'_h}} = 1\right] - \Pr\left[\mathsf{D}^{\mathsf{Rand},\mathsf{Rej}} = 1\right].$$

Let $\tau = (\tau_m, \tau_v, K_h, K'_h)$ denote the transcript of the attack, with

$$\tau_m = \big((M_1, T_1), \ldots, (M_{q_m}, T_{q_m})\big)$$
$$\tau_v = \big((M'_1, T'_1), \ldots, (M'_{q_v}, T'_{q_v})\big).$$

Recall that Θ denotes the set of attainable transcripts and X_{re}, resp. X_{id}, the probability distribution of the transcript τ induced by the real world, resp. the ideal world.

The remaining of the proof of Theorem 2 is structured as follows. First, we define bad transcripts and upper bound their probability in the ideal world (Lemma 6). Then, we analyze good transcripts and prove that they are almost as likely in the real and the ideal world (Lemma 7). Combining Lemma 2 from Section 2.3 with Lemma 6 and Lemma 7 gives us an upper bound on D's advantage, which by Equation (1) from Section 2.3 gives us an upper bound on

$$\mathbf{Adv}^{\mathrm{sdMAC}}_{\mathsf{HaT}[\widetilde{E}^*,H,H']}(\mathsf{A}).$$

Theorem 2 follows easily by combining this upper bound with Equation (8).

We start by defining bad transcripts.

Definition 5. We say that an attainable transcript $\tau = (\tau_m, \tau_v, K_h, K'_h)$ is *bad* if one of these conditions is fulfilled:

(C-1) there exists two distinct MAC queries (M_i, T_i) and (M_j, T_j) such that $H'_{K'_h}(M_i) = H'_{K'_h}(M_j)$ and either $H_{K_h}(M_i) = H_{K_h}(M_j)$ or $T_i = T_j$;

(C-2) there exist a MAC query $(M_i, T_i) \in \tau_m$ and a verification query $(M'_j, T'_j) \in \tau_v$ such that

$$\begin{cases} H'_{K'_h}(M_i) = H'_{K'_h}(M'_j) \\ H_{K_h}(M_i) = H_{K_h}(M'_j) \\ T_i = T'_j. \end{cases}$$

We let Θ_{bad}, resp. Θ_{good} denote the set of bad, respectively good transcripts.

We now upper bound the probability to get a bad transcript in the ideal world.

Lemma 6. *For any integers q_m and q_v, one has*

$$\Pr\left[X_{\mathrm{id}} \in \Theta_{\mathrm{bad}}\right] \le q_m^2 \varepsilon^2 + q_m q_v \varepsilon^2.$$

Proof. Let Θ_i denote the set of attainable transcripts satisfying condition (C-i). Recall that, in the ideal world, (K_h, K'_h) is drawn independently from the queries transcript. We are going to consider both conditions in turn.

Condition (C-1). Fix two distinct MAC queries (M_i, T_i) and (M_j, T_j). Then the probability that $H'_{K'_h}(M_i) = H'_{K'_h}(M_j)$ (over the draw of K'_h) is at most ε, the probability that $H_{K_h}(M_i) = H_{K_h}(M_j)$ (over the draw of K_h) is at most ε, and the probability that $T_i = T_j$ is at most 2^{-n}. Summing over all pairs of distinct MAC queries,

$$\Pr\left[X_{\mathrm{id}} \in \Theta_1\right] \le \frac{q_m^2 \varepsilon}{2 \cdot 2^n} + \frac{q_m^2 \varepsilon^2}{2} \le q_m^2 \varepsilon^2,$$

where we used that $\varepsilon \ge 2^{-n}$ (see Remark 1).

Condition (C-2). In order to upper bound the probability of obtaining bad transcripts satisfying condition (C-2) in the ideal world, fix a MAC query $(M_i, T_i) \in \tau_m$ and a verification query $(M'_j, T'_j) \in \tau_v$. Since K'_h is drawn independently from the queries transcript and H' is ε-AU, the probability that $H'_{K'_h}(M_i) = H'_{K'_h}(M'_j)$ is upper bounded by ε. We now distinguish two cases:

- If the verification query comes after the MAC query, then since the distinguisher is non-trivial, either $T_i \neq T'_j$, or $M_i \neq M'_j$. In the former case, the condition cannot be satisfied, while in the latter case, the probability over the random draw of K_h that $H_{K_h}(M_i) = H_{K_h}(M'_j)$ is at most ε.

- If the MAC query comes after the verification query, then T_i is random and independent from T'_j and the probability that $T_i = T'_j$ is 2^{-n}.

Since $\varepsilon \geq 2^{-n}$ (see Remark 1), we see that in all cases the condition is met with probability at most ε^2. Thus, by summing over every verification query, and every MAC query using the same nonce as the verification query, one has

$$\Pr[X_{\mathrm{id}} \in \Theta_2] \leq q_m q_v \varepsilon^2.$$

The result follows by a union bound over the two conditions. □

We now analyze good transcripts and prove the following lemma.

Lemma 7. *For any good transcript τ, one has*

$$\frac{\Pr[X_{\mathrm{re}} = \tau]}{\Pr[X_{\mathrm{id}} = \tau]} \geq 1 - \frac{q_v}{2^n - q_m}.$$

Proof. Let $\tau = (\tau_m, \tau_v, K_h, K'_h)$ be a good transcript. Let $\mathcal{L} = \{H_{K'_h}(M_1), \ldots, H_{K'_h}(M_{q_m})\}$ be the set of all the tweaks used in the MAC queries. Using an arbitrary order, we rewrite the set $\mathcal{L}$ as

$$\mathcal{L} = \{L_1, \ldots, L_r\},$$

where r is the total number of distinct values of $\mathcal{L}$. For $i = 1, \ldots, r$, we let q_i denote the number of MAC queries (M, T) in τ_m such that $H_{K'_h}(M) = L_i$.

Since in the ideal world the MAC oracle is perfectly random and the verification always rejects, one simply has

$$\Pr[X_{\mathrm{id}} = \tau] = \frac{1}{|\mathcal{K}_h| \cdot |\mathcal{K}'_h| \cdot (2^n)^{q_m}}. \tag{9}$$

We must now lower bound the probability of getting τ in the real world. We say that a tweakable permutation $\widetilde{P}$ is compatible with τ_m if

$$\forall i \in \{1, \ldots, q_m\},\ \widetilde{P}(H_{K'_h}(M_i), H_{K_h}(M_i)) = T_i,$$

and compatible with τ_v if

$$\forall i \in \{1, \ldots, q_v\},\ \widetilde{P}(H_{K'_h}(M'_i), H_{K_h}(M'_i)) \neq T'_i.$$

We simply say that $\widetilde{P}$ is compatible with τ if it is compatible with τ_m and τ_v. We let $\mathsf{Comp}(\tau_m)$, $\mathsf{Comp}(\tau_v)$, and $\mathsf{Comp}(\tau)$ denote the set of tweakable permutations that are compatible with respectively τ_m, τ_v, and τ. Then one can easily check (see for example [CS14] for a detailed explanation) that

$$\Pr[X_{\mathrm{re}} = \tau] = \frac{1}{|\mathcal{K}_h| \cdot |\mathcal{K}'_h|} \cdot \Pr\left[\widetilde{P} \leftarrow_\$ \mathsf{Perm}(\mathcal{W}, n) : \widetilde{P} \in \mathsf{Comp}(\tau)\right]. \tag{10}$$

We now define

$$\begin{aligned} \lambda_{\mathrm{eq}} &= \{(H'_{K'_h}(M_1), H_{K_h}(M_1), T_1), \ldots, (H'_{K'_h}(M_{q_m}), H_{K_h}(M_{q_m}), T_{q_m})\} \\ \lambda_{\mathrm{ineq}} &= \{(H'_{K'_h}(M'_1), H_{K_h}(M'_1), T'_1), \ldots, (H'_{K'_h}(M'_{q_v}), H_{K_h}(M'_{q_v}), T'_{q_v})\}. \end{aligned}$$

Then, since τ is a good transcript, λ_{eq} is a permutation equalities list (otherwise condition (C-1) defining bad transcripts would be met), and λ_{ineq} is a permutation inequalities list which is compatible with λ_{eq} (otherwise condition (C-2) would be met). Moreover,

$|\lambda_{\text{eq}}| = q_m$ and $|\lambda_{\text{ineq}}| = q_v$. Note that the event $\widetilde{P} \in \mathsf{Comp}(\tau)$ is actually equivalent to the event $\widetilde{P} \in \mathsf{Comp}(\lambda)$ where $\lambda = (\lambda_{\text{eq}}, \lambda_{\text{ineq}})$. Using Lemma 3, one has

$$\Pr\left[\widetilde{P} \in \mathsf{Comp}(\tau)\right] \geq \frac{1}{\prod_{i=1}^{r}(2^n)_{q_i}}\left(1 - \frac{q_v}{2^n - q_m}\right).$$

Combining this equation with Equation (9) and Equation (10), and using the fact that $q_m = \sum_{i=1}^{r} q_i$, we get

$$\frac{\Pr\left[X_{\text{re}} = \tau\right]}{\Pr\left[X_{\text{id}} = \tau\right]} \geq \left(1 - \frac{q_v}{2^n - q_m}\right) \cdot \prod_{i=1}^{r} \underbrace{\frac{(2^n)^{q_i}}{(2^n)_{q_i}}}_{\geq 1} \geq 1 - \frac{q_v}{2^n - q_m}. \qquad \square$$

4 Block Cipher-Based Constructions

The two constructions NaT and HaT of Section 3 follow the traditional UHF-then-PRF paradigm: first, the message is hashed with a UHF, and then a *keyed* transformation based on a TBC is applied. In this section, we give variants of these constructions where the final transformation uses a *keyless* transformation based on a block cipher. As a result, the MAC key of these construction only consists of the key(s) of the underlying universal hash function(s). The keyless final mapping must obviously be pre-image resistant (otherwise the adversary can recover the output of the UHF from the tag, which might reveal the hashing key, e.g., for polynomial-based universal hashing). For the nonce-based construction NaK, since the nonce is known from the adversary, this implies that we need to use the block cipher in Davies-Meyer mode. Technically, we also need to require that the universal hash functions be almost uniform. The security proofs for these two variants are done in the ideal cipher model.

As a warm-up, the reader might want to check that the construction

$$F_{K_h}(M) = G(H_{K_h}(M)),$$

where $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^m$ is ε-AU and ε'-almost uniform and $G : \{0,1\}^m \to \{0,1\}^n$ is some fixed function, is a PRF provably secure in the Random Oracle model for G, as proved in Appendix B.

4.1 The Nonce-Based Construction NaK

We start with the block-cipher-based variant of NaT named NaK (*Nonce-as-Key*). Given a block cipher E with key space $\mathcal{K}$ and message space $\{0,1\}^n$ and a keyed hash function H with key space $\mathcal{K}_h$, domain $\mathcal{M}$, and range $\{0,1\}^n$, we define a MAC with key space $\mathcal{K}_h$, nonce space $\mathcal{K}$, and message space $\mathcal{M}$ as

$$\mathsf{NaK}[E, H]_{K_h}(N, M) = E_N(H_{K_h}(M)) \oplus H_{K_h}(M).$$

Our security result is the following one.

Theorem 3. *Let $\mathcal{M}$, $\mathcal{K}$, and $\mathcal{K}_h$ be non-empty sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ be an ε-AU and ε'-almost uniform hash function. Let μ, q_e, q_m, and q_v be integers such that $q_e, q_m, q_v \leq 2^n$, and $\mu \leq \min\{q_m, 2^n - 1 - q_e\}$. Then, in the ideal cipher model for E, for any (μ, q_e, q_m, q_v)-adversary A against the nMAC-security of $\mathsf{NaK}[E, H]$, one has*

$$\mathbf{Adv}^{\text{nMAC}}_{\mathsf{NaK}[E,H]}(\mathsf{A}) \leq 2(\mu - 1)q_m\varepsilon + \frac{q_v}{2^n - \mu - q_e} + \mu q_v \varepsilon + n q_v \varepsilon' + \left(\frac{q_e}{2^n - q_e + 1}\right)^{n+1} + 2\mu q_e \varepsilon'.$$

Proof. Deferred to Appendix C. □

Hence, in the nonce-respecting case ($\mu = 1$), the NaK construction is secure up to $q_v \simeq \min\{\varepsilon^{-1}, (\varepsilon')^{-1}/n\}$ verification queries and $q_e \simeq (\varepsilon')^{-1}$ ideal cipher queries, irrespectively of the number of MAC queries. When A can freely choose nonces (i.e., $\mu = q_m$), then the NaK construction is secure up to the birthday bound. The security bound degrades linearly with the maximal multiplicity μ of nonces.

As a corollary, we obtain the following for the security of NaK as a randomized MAC.

Corollary 2. *Let $\mathcal{M}$, $\mathcal{K}$, and $\mathcal{K}_h$ be non-empty sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ be an ε-AU and ε'-almost uniform hash function. Let q_e, q_m, and q_v be integers such that $q_e, q_m, q_v \le 2^n$ and $q_e < 2^n - n$. Then, in the ideal cipher model for E, for any (q_e, q_m, q_v)-adversary A against the rMAC-security of $\mathsf{NaK}[E, H]$, one has*

$$\mathbf{Adv}^{\mathrm{rMAC}}_{\mathsf{NaK}[E,H]}(\mathsf{A}) \le 2(n-1)q_m\varepsilon + \frac{q_v}{2^n - n - q_e} + nq_v\varepsilon + nq_v\varepsilon' + \left(\frac{q_e}{2^n - q_e + 1}\right)^{n+1} + 2nq_e\varepsilon' + \frac{q_m^{n+1}}{(2|\mathcal{K}|)^n}.$$

Proof. This follows by combining Lemma 1 with $\mu_0 = n$ and Theorem 3. □

4.2 The Stateless Deterministic Construction HaK

The block cipher-based variant of HaT is a stateless deterministic MAC called HaK (*Hash-as-Key*). Given a block cipher $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ and two keyed hash functions $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ and $H' : \mathcal{K}'_h \times \mathcal{M} \to \mathcal{K}$, we define a MAC with key space $\mathcal{K}_h \times \mathcal{K}'_h$ and message space $\mathcal{M}$ as

$$\mathsf{HaK}[E, H, H']_{K_h, K'_h}(M) = E_{H'_{K'_h}(M)}(H_{K_h}(M)).$$

Our security result is the following one.

Theorem 4. *Let $\mathcal{M}$, $\mathcal{K}$, $\mathcal{K}_h$, and $\mathcal{K}'_h$ be non-empty sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and let $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ and $H' : \mathcal{K}'_h \times \mathcal{M} \to \mathcal{K}$ be two ε-AU and ε'-almost uniform hash functions. Let q_e, q_m, and q_v be integers such that $q_m + q_e < 2^n$. Then, in the ideal cipher model for E, for any (q_e, q_m, q_v)-adversary A against the sdMAC-security of $\mathsf{HaK}[E, H, H']$, one has*

$$\mathbf{Adv}^{\mathrm{sdMAC}}_{\mathsf{HaK}[E,H]}(\mathsf{A}) \le q_m^2\varepsilon^2 + q_mq_v\varepsilon^2 + q_mq_e(\varepsilon')^2 + \left(\frac{q_m}{2^n}\right)^{n+1} + nq_e\varepsilon' + q_vq_e(\varepsilon')^2 + \frac{q_v}{2^n - q_m - q_e}.$$

Proof. Deferred to Appendix D. □

5 Security of Truncated MACs

In this section, we analyze how tag truncation affects the security of MACs. Let $F : \mathcal{K} \times (\mathcal{N}\times)\mathcal{M} \to \{0,1\}^n$ be a keyed function with key space $\mathcal{K}$, message space $\mathcal{M}$, range $\mathcal{T} = \{0,1\}^n$ and potentially nonce space $\mathcal{N}$ (the reasoning below applies both to SD-MACs and nonce-based MACs). For any $1 \le s \le n-1$, let $\mathsf{trunc}_s : \{0,1\}^n \to \{0,1\}^s$ be a function that takes s bits of the input in any way (e.g., the leftmost s bits of an n-bit input). Let

$$F_s \stackrel{\text{def}}{=} \mathsf{trunc}_s \circ F$$

denote a truncated variant of F that returns only s bits of the original tag.

Lemma 8. *If there exists a function δ of q_m, q_v, t, and potentially μ, such that, for any $((\mu), q_m, q_v, t)$-adversary A against F,*

$$\mathbf{Adv}_F^{\mathrm{sd/nMAC}}(\mathsf{A}) \le \delta((\mu), q_m, q_v, t),$$

then, for any $((\mu), q_m, q_v, t)$-adversary A' against F_s, one has

$$\mathbf{Adv}_{F_s}^{\mathrm{sd/nMAC}}(\mathsf{A}') \le \delta((\mu), q_m, 2^{n-s} q_v, t).$$

Proof. Given a $((\mu), q_m, q_v, t)$-adversary A' against F_s, one can use it as a subroutine to construct a $((\mu), q_m, 2^{n-s} q_v, t)$-adversary A against F as follows:

- A faithfully relays each MAC query made by A' to its MAC oracle; if A receives T from the oracle as the answer to this query, then A sends $T' = \mathsf{trunc}_s(T)$ to A';

- If A' makes a verification query $((N'), M', T')$, then A makes 2^{n-s} verification queries $((N'), M', T)$ for all n-bit T such that $\mathsf{trunc}_s(T) = T'$.

Clearly, A is successful at least as often as A, hence one has

$$\mathbf{Adv}_{F_s}^{\mathrm{sd/nMAC}}(\mathsf{A}') \le \mathbf{Adv}_F^{\mathrm{sd/nMAC}}(\mathsf{A}). \qquad \square$$

As an example, applying this analysis to HaT, we obtain the following theorem.

Theorem 5. *For any $1 \le s \le n-1$, let*

$$\mathsf{HaT}_s[\widetilde{E}, H, H'] = \mathsf{trunc}_s \circ \mathsf{HaT}[\widetilde{E}, H, H']$$

denote an s-bit truncated variant of $\mathsf{HaT}[\widetilde{E}, H, H']$, where $\widetilde{E} : \mathcal{K} \times \mathcal{W} \times \{0,1\}^n \to \{0,1\}^n$ is a tweakable block cipher and $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ and $H' : \mathcal{K}'_h \times \mathcal{M} \to \mathcal{W}$ are ε-AU hash functions. Let q_m, q_v, and t be integers such that $q_m < 2^n$. Then for any (q_m, q_v, t)-adversary A against the sdMAC-security of $\mathsf{HaT}_s[\widetilde{E}, H, H']$, there exists a $(q_m + q_v, t')$-adversary A' against the TPRP*-security of $\widetilde{E}$, where $t' = O(t + (q_m + q_v)t_H)$ and t_H is an upper bound on the time to compute H or H' on any message, such that*

$$\mathbf{Adv}_{\mathsf{HaT}_s[\widetilde{E},H,H']}^{\mathrm{sdMAC}}(\mathsf{A}) \le \mathbf{Adv}_{\widetilde{E}}^{\mathrm{TPRP}}(\mathsf{A}') + 2q_m^2\varepsilon^2 + 2^{n-s} q_m q_v \varepsilon^2 + \frac{2^{n-s} q_v}{2^n - q_m}.$$

Assuming $\varepsilon \simeq \ell 2^{-n}$, where ℓ is the maximal length of messages in n-bit blocks, the HaT_s construction is secure up to $q_m \ell \simeq 2^n$ blocks in MAC queries and $q_v \simeq 2^s$ verification queries, as long as $q_m q_v \ell^2$ is small compared to 2^{n+s}. Similar results can be obtained for nonce-based, randomized and/or ideal cipher-based MACs.

Acknowledgments

We thank the anonymous reviewers of TOSC for their useful feedback and suggestions. The first author has been partially supported by the European Union's H2020 Programme under grant agreement number ICT-644209 and the third author has been partially supported by the French Agence Nationale de la Recherche through the BRUTUS project under Contract ANR-14-CE28-0015.

References

[BC09] John Black and Martin Cochran. MAC Reforgeability. In Orr Dunkelman, editor, *Fast Software Encryption - FSE 2009*, volume 5665 of *LNCS*, pages 345–362. Springer, 2009.

[BCK96] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. Keying Hash Functions for Message Authentication. In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO '96*, volume 1109 of *LNCS*, pages 1–15. Springer, 1996.

[Ber05] Daniel J. Bernstein. Stronger Security Bounds for Wegman-Carter-Shoup Authenticators. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 164–180. Springer, 2005.

[Ber07] Daniel J. Bernstein. Polynomial Evaluation and Message Authentication. Unpublished manuscript, 2007. Available at http://cr.yp.to/papers.html#pema.

[BGK99] Mihir Bellare, Oded Goldreich, and Hugo Krawczyk. Stateless Evaluation of Pseudorandom Functions: Security beyond the Birthday Barrier. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99*, volume 1666 of *LNCS*, pages 270–287. Springer, 1999.

[BKR00] Mihir Bellare, Joe Kilian, and Phillip Rogaway. The Security of the Cipher Block Chaining Message Authentication Code. *Journal of Computer and System Sciences*, 61(3):362–399, 2000.

[BR02] John Black and Phillip Rogaway. A Block-Cipher Mode of Operation for Parallelizable Message Authentication. In Lars R. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 384–397. Springer, 2002.

[BRS02] John Black, Phillip Rogaway, and Thomas Shrimpton. Black-Box Analysis of the Block-Cipher-Based Hash-Function Constructions from PGV. In Moti Yung, editor, *Advances in Cryptology - CRYPTO 2002*, volume 2442 of *LNCS*, pages 320–335. Springer, 2002.

[BS] Dan Boneh and Victor Shoup. A Graduate Course in Applied Cryptography. Available at http://toc.cryptobook.us.

[CLL+14] Shan Chen, Rodolphe Lampe, Jooyoung Lee, Yannick Seurin, and John P. Steinberger. Minimizing the Two-Round Even-Mansour Cipher. In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology - CRYPTO 2014 (Proceedings, Part I)*, volume 8616 of *LNCS*, pages 39–56. Springer, 2014. Full version available at http://eprint.iacr.org/2014/443.

[CLS15] Benoît Cogliati, Rodolphe Lampe, and Yannick Seurin. Tweaking Even-Mansour Ciphers. In Rosario Gennaro and Matthew Robshaw, editors, *Advances in Cryptology - CRYPTO 2015 (Proceedings, Part I)*, volume 9215 of *LNCS*, pages 189–208. Springer, 2015. Full version available at http://eprint.iacr.org/2015/539.

[CS14] Shan Chen and John Steinberger. Tight Security Bounds for Key-Alternating Ciphers. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology - EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 327–350. Springer, 2014. Full version available at http://eprint.iacr.org/2013/222.

[CS15a] Benoît Cogliati and Yannick Seurin. Beyond-Birthday-Bound Security for Tweakable Even-Mansour Ciphers with Linear Tweak and Key Mixing. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 (Proceedings, Part II)*, volume 9453 of *LNCS*, pages 134–158. Springer, 2015.

[CS15b] Benoît Cogliati and Yannick Seurin. On the Provable Security of the Iterated Even-Mansour Cipher against Related-Key and Chosen-Key Attacks. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 (Proceedings, Part I)*, volume 9056 of *LNCS*, pages 584–613. Springer, 2015. Full version available at http://eprint.iacr.org/2015/069.

[CS16] Benoît Cogliati and Yannick Seurin. EWCDM: An Efficient, Beyond-Birthday Secure, Nonce-Misuse Resistant MAC. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 (Proceedings, Part I)*, volume 9814 of *LNCS*, pages 121–149. Springer, 2016.

[HT16] Viet Tung Hoang and Stefano Tessaro. Key-Alternating Ciphers and Key-Length Extension: Exact Bounds and Multi-user Security. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 (Proceedings, Part I)*, volume 9814 of *LNCS*, pages 3–32. Springer, 2016.

[IK03] Tetsu Iwata and Kaoru Kurosawa. OMAC: One-Key CBC MAC. In Thomas Johansson, editor, *Fast Software Encryption - FSE 2003*, volume 2887 of *LNCS*, pages 129–153. Springer, 2003.

[JJV02] Éliane Jaulmes, Antoine Joux, and Frédéric Valette. On the Security of Randomized CBC-MAC Beyond the Birthday Paradox Limit: A New Construction. In Joan Daemen and Vincent Rijmen, editors, *Fast Software Encryption - FSE 2002*, volume 2365 of *LNCS*, pages 237–251. Springer, 2002.

[JL04] Éliane Jaulmes and Reynald Lercier. FRMAC, a Fast Randomized Message Authentication Code. 2004. Available at http://eprint.iacr.org/2004/166.

[JNPS16] Jérémy Jean, Ivica Nikolic, Thomas Peyrin, and Yannick Seurin. Deoxys v1.41. Submitted to the CAESAR competition, 2016.

[LN17] Eik List and Mridul Nandi. Revisiting Full-PRF-Secure PMAC and Using It for Beyond-Birthday Authenticated Encryption. In Helena Handschuh, editor, *Topics in Cryptology - CT-RSA 2017*, volume 10159 of *LNCS*, pages 258–274. Springer, 2017. Full version at http://eprint.iacr.org/2016/1174.

[LRW02] Moses Liskov, Ronald L. Rivest, and David Wagner. Tweakable Block Ciphers. In Moti Yung, editor, *Advances in Cryptology - CRYPTO 2002*, volume 2442 of *LNCS*, pages 31–46. Springer, 2002.

[LST12] Will Landecker, Thomas Shrimpton, and R. Seth Terashima. Tweakable Blockciphers with Beyond Birthday-Bound Security. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology - CRYPTO 2012*, volume 7417 of *LNCS*, pages 14–30. Springer, 2012. Full version available at http://eprint.iacr.org/2012/450.

[Men15] Bart Mennink. Optimally Secure Tweakable Blockciphers. In Gregor Leander, editor, *Fast Software Encryption - FSE 2015*, volume 9054 of *LNCS*, pages 428–448. Springer, 2015. Full version available at http://eprint.iacr.org/2015/363.

[Min10] Kazuhiko Minematsu. How to Thwart Birthday Attacks against MACs via Small Randomness. In Seokhie Hong and Tetsu Iwata, editors, *Fast Software Encryption - FSE 2010*, volume 6147 of *LNCS*, pages 230–249. Springer, 2010.

[Nai15] Yusuke Naito. Full PRF-Secure Message Authentication Code Based on Tweakable Block Cipher. In Man Ho Au and Atsuko Miyaji, editors, *ProvSec 2015*, volume 9451 of *LNCS*, pages 167–182. Springer, 2015.

[Pat90] Jacques Patarin. Pseudorandom Permutations Based on the DES Scheme. In Gérard D. Cohen and Pascale Charpin, editors, *EUROCODE '90*, volume 514 of *LNCS*, pages 193–204. Springer, 1990.

[Pat91] Jacques Patarin. New Results on Pseudorandom Permutation Generators Based on the DES Scheme. In Joan Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91*, volume 576 of *LNCS*, pages 301–312. Springer, 1991.

[Pat08a] Jacques Patarin. A Proof of Security in $O(2^n)$ for the Xor of Two Random Permutations. In Reihaneh Safavi-Naini, editor, *Information Theoretic Security - ICITS 2008*, volume 5155 of *LNCS*, pages 232–248. Springer, 2008. Full version available at http://eprint.iacr.org/2008/010.

[Pat08b] Jacques Patarin. The "Coefficients H" Technique. In Roberto Maria Avanzi, Liam Keliher, and Francesco Sica, editors, *Selected Areas in Cryptography - SAC 2008*, volume 5381 of *LNCS*, pages 328–345. Springer, 2008.

[Pat10] Jacques Patarin. Security of balanced and unbalanced Feistel Schemes with Linear Non Equalities. 2010. Available at http://eprint.iacr.org/2010/293.

[Pat13] Jacques Patarin. Security in $O(2^n)$ for the Xor of Two Random Permutations: Proof with the Standard H Technique. IACR Cryptology ePrint Archive, Report 2013/368, 2013. Available at http://eprint.iacr.org/2013/368.

[Rog04] Phillip Rogaway. Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In Pil Joong Lee, editor, *Advances in Cryptology - ASIACRYPT 2004*, volume 3329 of *LNCS*, pages 16–31. Springer, 2004.

[Sho96] Victor Shoup. On Fast and Provably Secure Message Authentication Based on Universal Hashing. In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO '96*, volume 1109 of *LNCS*, pages 313–328. Springer, 1996.

[Sti96] Douglas R. Stinson. On the Connections Between Universal Hashing, Combinatorial Designs and Error-Correcting Codes. In *Congressus Numerantium 114*, pages 7–27, 1996.

[WC81] Mark N. Wegman and Larry Carter. New Hash Functions and Their Use in Authentication and Set Equality. *J. Comput. Syst. Sci.*, 22(3):265–279, 1981.

A Proof of Lemma 2

Let Θ' be the set of all transcripts τ such that

$$\max\{\Pr[X_{\mathrm{id}} = \tau], \Pr[X_{\mathrm{re}} = \tau]\} > 0.$$

Remark that the set Θ of attainable transcripts is included in (and in our case, different from) Θ'.

Let $\mathcal{O}_{\mathrm{id}}$, resp. $\mathcal{O}_{\mathrm{re}}$ denote the oracle from the ideal, resp. the real world. Recall that

$$\begin{aligned}\mathbf{Adv}(\mathsf{D}) &= \left|\Pr\left[\mathsf{D}^{\mathcal{O}_{\mathrm{id}}} = 1\right] - \Pr\left[\mathsf{D}^{\mathcal{O}_{\mathrm{re}}} = 1\right]\right| \\ &= \left|\Pr\left[\mathsf{D}^{\mathcal{O}_{\mathrm{id}}} = 0\right] - \Pr\left[\mathsf{D}^{\mathcal{O}_{\mathrm{re}}} = 0\right]\right|.\end{aligned}$$

Moreover, the distinguisher's output is a deterministic function of the transcript. If we let Θ'_i denote the subset of Θ' such that D outputs i, for $i = 0, 1$, it is easy to see that

$$\begin{aligned}\Pr\left[\mathsf{D}^{\mathcal{O}_{\mathrm{id}}} = i\right] &= \sum_{\tau \in \Theta'_i} \Pr[X_{\mathrm{id}} = \tau] \qquad \text{and} \\ \Pr\left[\mathsf{D}^{\mathcal{O}_{\mathrm{re}}} = i\right] &= \sum_{\tau \in \Theta'_i} \Pr[X_{\mathrm{re}} = \tau]\end{aligned}$$

for $i = 0, 1$. Thus

$$\begin{aligned}\mathbf{Adv}(\mathsf{D}) &= \left|\sum_{\tau \in \Theta'_1} \left(\Pr[X_{\mathrm{re}} = \tau] - \Pr[X_{\mathrm{id}} = \tau]\right)\right| \\ &\leq \sum_{\tau \in \Theta'_1} |\Pr[X_{\mathrm{re}} = \tau] - \Pr[X_{\mathrm{id}} = \tau]|.\end{aligned}$$

Similarly,

$$\mathbf{Adv}(\mathsf{D}) \leq \sum_{\tau \in \Theta'_0} |\Pr[X_{\mathrm{re}} = \tau] - \Pr[X_{\mathrm{id}} = \tau]|,$$

which implies that

$$\mathbf{Adv}(\mathsf{D}) \leq \frac{1}{2} \sum_{\tau \in \Theta'} |\Pr[X_{\mathrm{re}} = \tau] - \Pr[X_{\mathrm{id}} = \tau]| = \|X_{\mathrm{re}} - X_{\mathrm{id}}\|$$

since $\Theta' = \Theta'_0 \sqcup \Theta'_1$. Moreover, one has

$$\|X_{\mathrm{re}} - X_{\mathrm{id}}\| = \sum_{\substack{\tau \in \Theta' \\ \Pr[X_{\mathrm{id}} = \tau] > \Pr[X_{\mathrm{re}} = \tau]}} \left(\Pr[X_{\mathrm{id}} = \tau] - \Pr[X_{\mathrm{re}} = \tau]\right).$$

For every transcript τ appearing in this sum, one has $\Pr[X_{\mathrm{id}} = \tau] > \Pr[X_{\mathrm{re}} = \tau]$, which means, in particular, that τ is an attainable transcript. Thus one has

$$\begin{aligned}\|X_{\mathrm{re}} - X_{\mathrm{id}}\| &= \sum_{\substack{\tau \in \Theta \\ \Pr[X_{\mathrm{id}} = \tau] > \Pr[X_{\mathrm{re}} = \tau]}} \left(\Pr[X_{\mathrm{id}} = \tau] - \Pr[X_{\mathrm{re}} = \tau]\right) \\ &= \sum_{\substack{\tau \in \Theta \\ \Pr[X_{\mathrm{id}} = \tau] > \Pr[X_{\mathrm{re}} = \tau]}} \Pr[X_{\mathrm{id}} = \tau] \left(1 - \frac{\Pr[X_{\mathrm{re}} = \tau]}{\Pr[X_{\mathrm{id}} = \tau]}\right) \\ &\leq \sum_{\tau \in \Theta_{\mathrm{good}}} \Pr[X_{\mathrm{id}} = \tau]\, \epsilon_1 + \sum_{\tau \in \Theta_{\mathrm{bad}}} \Pr[X_{\mathrm{id}} = \tau] \\ &\leq \varepsilon_1 + \varepsilon_2.\end{aligned}$$

B The UHF-then-RO Construction

We prove the following theorem.

Theorem 6. *Let $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^m$ be a ε-AU and ε'-almost uniform hash function and $G : \{0,1\}^m \to \{0,1\}^n$. Let $F : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ be the keyed function defined as*

$$F_{K_h}(M) = G(H_{K_h}(M)).$$

Then, in the random oracle for G, for any adversary A making at most q queries to F and q' queries to G, one has

$$\mathbf{Adv}_F^{\mathrm{PRF}}(\mathsf{A}) \leq \frac{q^2\varepsilon}{2} + qq'\varepsilon'.$$

Proof. The adversary is trying to distinguish F_{K_h} for a random key K_h from a uniformly random function $\mathsf{Rand} : \mathcal{M} \to \{0,1\}^n$. Let $\tau = (\tau_f, \tau_g, K_h)$ be the transcript of the attack, where

$$\tau_f = \big((M_1, T_1), \ldots, (M_q, T_q)\big)$$
$$\tau_g = \big((X_1, Y_1), \ldots, (X_{q'}, Y_{q'})\big)$$

are respectively the queries of the adversary to F and G. (As usual, we provide the real or a dummy key to the distinguisher at the end of the attack, depending on which oracle it is interacting with.)

We say that a transcript is bad if there exists two distinct queries $(M_i, T_i), (M_j, T_j) \in \tau_f$ such that $H_{K_h}(M_i) = H_{K_h}(M_j)$, or if there exists $(M, T) \in \tau_f$ and $(X, Y) \in \tau_g$ such that $H_{K_h}(M) = X$. By respectively the ε-AU and ε'-almost uniformity of H, and since in the ideal world K_h is drawn independently from (τ_f, τ_g), the probability to obtain a bad transcript in the ideal world is at most

$$\frac{q^2\varepsilon}{2} + qq'\varepsilon'.$$

Fix now any good transcript $\tau = (\tau_f, \tau_g, K_h)$. The probability to obtain τ in the ideal world is

$$\frac{1}{|\mathcal{K}_h|} \cdot \Pr_F[F(M_i) = T_i, i \in \{1, \ldots, q\}] \cdot \Pr_G[G(X_j) = Y_j, j \in \{1, \ldots, q'\}]$$
$$= \frac{1}{|\mathcal{K}_h| \cdot (2^n)^{q+q'}},$$

while in the real world it is

$$\frac{1}{|\mathcal{K}_h|} \cdot \Pr_G[G(H_{K_h}(M_i)) = T_i, i \in \{1, \ldots, q\} \text{ and } G(X_j) = Y_j, j \in \{1, \ldots, q'\}]$$
$$= \frac{1}{|\mathcal{K}_h| \cdot (2^n)^{q+q'}},$$

since by definition of a good transcript, all values $H_{K_h}(M_i)$, $i = 1, \ldots, q$ and X_j, $j = 1, \ldots, q'$ are distinct. Hence the ratio is 1 and Lemma 2 allows to conclude. $\square$

C Proof of Theorem 3

Following Section 2.3, let us fix a non-trivial (μ, q_e, q_m, q_v)-distinguisher D interacting either with the real world $(E, \mathsf{NaK}[E,H]_{K_h}, \mathsf{Ver}[E,H]_{K_h})$ for a uniformly random block cipher E and a random hashing key K_h, or with the ideal world $(E, \mathsf{Rand}, \mathsf{Rej})$. We let

$$\mathbf{Adv}(\mathsf{D}) = \Pr\left[\mathsf{D}^{E, \mathsf{NaK}[E,H]_{K_h}, \mathsf{Ver}[E,H]_{K_h}} = 1\right] - \Pr\left[\mathsf{D}^{E, \mathsf{Rand}, \mathsf{Rej}} = 1\right].$$

Let $\tau = (\tau_e, \tau_m \tau_v, K_h)$ denote the transcript of the attack, with

$$\begin{aligned}\tau_e &= \big((K_1, X_1, Y_1), \ldots, (K_{q_e}, X_{q_e}, Y_{q_e})\big)\\ \tau_m &= \big((N_1, M_1, T_1), \ldots, (N_{q_m}, M_{q_m}, T_{q_m})\big)\\ \tau_v &= \big((N'_1, M'_1, T'_1), \ldots, (N'_{q_v}, M'_{q_v}, T'_{q_v})\big).\end{aligned}$$

Recall that Θ denotes the set of attainable transcripts and X_{re}, resp. X_{id}, the probability distribution of the transcript τ induced by the real world, resp. the ideal world.

The remaining of the proof of Theorem 3 is structured as follows. First, we define bad transcripts and upper bound their probability in the ideal world (Lemma 9). Then, we analyze good transcripts and prove that they are almost as likely in the real and the ideal world (Lemma 10). Theorem 3 follows easily by combining Equation (1) and Lemma 2 from Section 2.3 with Lemma 9 and Lemma 10.

Definition 6. We say that an attainable transcript $\tau = (\tau_e, \tau_m, \tau_v, K_h)$ is *bad* if one of these conditions is fulfilled:

(C-1) there exists two distinct MAC queries (N_i, M_i, T_i) and (N_j, M_j, T_j) such that $N_i = N_j$ and either $H_{K_h}(M_i) = H_{K_h}(M_j)$ or $T_i \oplus H_{K_h}(M_i) = T_j \oplus H_{K_h}(M_j)$;

(C-2) there exists an IC query $(K_i, X_i, Y_i) \in \tau_e$ and a MAC query $(N_j, M_j, T_j) \in \tau_m$ such that $K_i = N_j$ and either $X_i = H_{K_h}(M_j)$ or $Y_i = T_j \oplus H_{K_h}(M_j)$;

(C-3) there exists a MAC query $(N_i, M_i, T_i) \in \tau_m$ and a verification query $(N'_j, M'_j, T'_j) \in \tau_v$ such that

$$\begin{cases} N_i = N'_j \\ H_{K_h}(M_i) = H_{K_h}(M'_j) \\ T_i = T'_j; \end{cases}$$

(C-4) there exists an IC query $(K_i, X_i, Y_i) \in \tau_e$ and a verification query $(N'_j, M'_j, T'_j) \in \tau_v$ such that

$$\begin{cases} K_i = N'_j \\ X_i = H_{K_h}(M'_j) \\ Y_i = T'_j \oplus H_{K_h}(M'_j). \end{cases}$$

We let Θ_{bad}, resp. Θ_{good} denote the set of bad, respectively good transcripts.

Note that the third and fourth conditions can only happen in the ideal world since in the real world, if e.g., $N = N'$, $T = T'$, and $H_{K_h}(M) = H_{K_h}(M')$, the verification oracle should return 1 on query (N', M', T') (which is impossible for any attainable transcript). We now upper bound the probability to get a bad transcript in the ideal world.

Lemma 9. *For any integers q_e, q_m and q_v such that $q_e \leq 2^n$, one has*

$$\Pr\left[X_{\mathrm{id}} \in \Theta_{\mathrm{bad}}\right] \leq 2(\mu - 1) q_m \varepsilon + \mu q_v \varepsilon + n q_v \varepsilon' + \left(\frac{q_e}{2^n - q_e + 1}\right)^{n+1} + 2\mu q_e \varepsilon'.$$

Proof. We let Θ_i denote the set of attainable transcripts satisfying condition (C-i). Recall that, in the ideal world, K_h is drawn independently from the queries transcript, and that E is independent from Rand. We are going to consider the four conditions in turn.

Condition (C-1). Fix a MAC query (N_i, M_i, T_i). There are exactly q_m possible choices for this query. Then we fix another MAC query (N_j, M_j, T_j) such that $N_i = N_j$ (there are at most $\mu - 1$ possible choices). The probability, over the random draw of T_i and T_j that $T_i \oplus H_{K_h}(M_i) = T_j \oplus H_{K_h}(M_j)$ is 2^{-n}, and the probability, over the random draw of K_h, that $H_{K_h}(M_i) = H_{K_h}(M_j)$, is at most ε. Summing over every possible choice of (N_i, M_i, T_i) and (N_j, M_j, T_j), we get

$$\Pr[X_{\text{id}} \in \Theta_1] \leq \frac{(\mu-1)q_m}{2^n} + (\mu-1)q_m\varepsilon \leq 2(\mu-1)q_m\varepsilon,$$

where we used that $\varepsilon \geq 2^{-n}$ (see Remark 1).

Condition (C-2). Fix an ideal cipher query $(K_i, X_i, Y_i) \in \tau_e$. Then, since D cannot repeat a nonce in its MAC queries more than μ times, there are at most μ MAC queries $(N_j, M_j, T_j) \in \tau_m$ such that $K_i = N_j$. Fix any of these queries. Then, the probability, over the random draw of K_h, that either $X_i = H_{K_h}(M_j)$ or $Y_i = T_j \oplus H_{K_h}(M_j)$ is lower than $2\varepsilon'$ thanks to the ε'-almost uniformity of H. Summing over every possible choice of queries, we get

$$\Pr[X_{\text{id}} \in \Theta_2] \leq 2\mu q_e \varepsilon'.$$

Condition (C-3). This condition is exactly the same as condition (C-2) in Lemma 4, hence by exactly the same proof one has

$$\Pr[X_{\text{id}} \in \Theta_3] \leq \mu q_v \varepsilon.$$

Condition (C-4). In order to upper bound the probability that condition (C-4) is fulfilled, we need to upper bound the number of ideal cipher queries $(K_i, X_i, Y_i) \in \tau_e$ satisfying $K_i = N'_j$ and $H_{K_h}(M'_j) = X_i = Y_i \oplus T'_j$, for a verification query $(N'_j, M'_j, T'_j) \in \tau_v$. In particular, this means that such a query must satisfy $X_i \oplus Y_i = T'_j$. The first step of our proof is to upper bound the probability, over the random draw of E, that there exists $n+1$ distinct ideal cipher queries $(K_{i_1}, X_{i_1}, Y_{i_1}), \ldots, (K_{i_{n+1}}, X_{i_{n+1}}, Y_{i_{n+1}})$ such that $X_{i_l} \oplus Y_{i_l}$ is constant for $l = 1, \ldots, n+1$. Let us define

$$\alpha(E) = \max_{a \in \{0,1\}^n} \left|\left\{ i \in \{1, \ldots, q_e\} \,:\, X_i \oplus Y_i = a \right\}\right|.$$

We are going to upper bound the probability, over the random choice of E, that $\alpha(E) \geq n+1$. Fix any $n+1$-tuple of indexes $(i_1, \ldots, i_{n+1})$ such that $1 \leq i_1 < \cdots < i_{n+1} \leq q_e$. Then one has

$$\Pr\left[X_{i_1} \oplus Y_{i_1} = \cdots = X_{i_{n+1}} \oplus Y_{i_{n+1}}\right] \leq \frac{1}{(2^n - q_e + 1)^n}.$$

This can easily be seen as follows: if query $(K_{i_j}, X_{i_j}, Y_{i_j})$ is an encryption (resp. decryption) query, then Y_{i_j} (resp. X_{i_j}) is chosen uniformly at random in a set of size at least $2^n - q_e + 1$, and the probability to have $X_{i_j} \oplus Y_{i_j} = X_{i_1} \oplus Y_{i_1}$ is lower than $1/(2^n - q_e + 1)$, for every $j = 2, \ldots, n+1$. Summing over every such possible tuple of queries, one has

$$\begin{aligned}\Pr[\alpha(E) \geq n+1] &= \Pr\left[\exists\, 1 \leq i_1 < \cdots < i_{n+1} \leq q_e \,:\, X_{i_1} \oplus Y_{i_1} = \cdots = X_{i_{n+1}} \oplus Y_{i_{n+1}}\right] \\ &\leq \frac{(q_e)_{n+1}}{(n+1)!(2^n - q_e + 1)^n} \leq \left(\frac{q_e}{2^n - q_e + 1}\right)^{n+1},\end{aligned}$$

where we used that $(n+1)! \geq 2^n \geq 2^n - q_e + 1$. Now assume that $\alpha(E) \leq n$ and fix any verification query $(N'_j, M'_j, T'_j) \in \tau_v$. There are at most n ideal cipher queries $(K_i, X_i, Y_i) \in \tau_e$ satisfying $X_i \oplus Y_i = T'_j$. Fix any of these queries. The probability, over the random choice of K_h, that $H_{K_h}(M'_j) = X_i$, is lower than ε'. Thus, by summing over every possible choice of queries, one has

$$\begin{aligned}\Pr\left[X_{\mathrm{id}} \in \Theta_4\right] &\leq \Pr\left[\alpha(E) \geq n+1\right] + \Pr\left[\alpha(E) \leq n\right](nq_v\varepsilon') \\ &\leq \left(\frac{q_e}{2^n - q_e + 1}\right)^{n+1} + nq_v\varepsilon'.\end{aligned}$$

Note that while this reasoning assumes that $q_e \geq n+1$, the bound still holds when $q_e \leq n$.

The result follows by a union bound over these conditions. □

We now analyze good transcripts and prove the following lemma.

Lemma 10. *For any good transcript τ, one has*

$$\frac{\Pr\left[X_{\mathrm{re}} = \tau\right]}{\Pr\left[X_{\mathrm{id}} = \tau\right]} \geq 1 - \frac{q_v}{2^n - \mu - q_e}.$$

Proof. Let $\tau = (\tau_e, \tau_m, \tau_v, K_h)$ be a good transcript. Let $\mathcal{L} = \{K_1, \ldots, K_{q_e}, N_1, \ldots, N_{q_m}\}$ be the set of every key or nonce used in the ideal cipher or MAC queries. Using an arbitrary order, we rewrite the set $\mathcal{L}$ as

$$\mathcal{L} = \{L_1, \ldots, L_r\},$$

where r is the total number of distinct values in $\mathcal{L}$. For $i = 1, \ldots, r$, we let q_i denote the number of ideal cipher queries in τ_e using L_i as a key and q'_i the number of MAC queries using L_i as a nonce.

Since in the ideal world the ideal cipher is perfectly random and independent from the other oracles, the MAC oracle is perfectly random, and the verification oracle always rejects, one simply has

$$\Pr[X_{\mathrm{id}} = \tau] = \frac{1}{|\mathcal{K}_h| \cdot (2^n)^{q_m} \cdot \prod_{i=1}^{r} (2^n)_{q_i}} = \frac{1}{|\mathcal{K}_h|} \prod_{i=1}^{r} \frac{1}{(2^n)^{q'_i} (2^n)_{q_i}}, \tag{11}$$

since $q_e = \sum_{i=1}^{r} q_i$ and $q_m = \sum_{i=1}^{r} q'_i$. We must now lower bound the probability of getting τ in the real world. We say that a block cipher E is compatible with τ_e if

$$\forall i \in \{1, \ldots, q_e\},\ E_{K_i}(X_i) = Y_i,$$

compatible with τ_m if

$$\forall i \in \{1, \ldots, q_m\},\ \mathsf{NaK}[E, H]_{K_h}(N_i, M_i) = T_i,$$

and compatible with τ_v if

$$\forall i \in \{1, \ldots, q_v\},\ \mathsf{NaK}[E, H]_{K_h}(N'_i, M'_i) \neq T'_i.$$

We simply say that E is compatible with τ if it is compatible with τ_e, τ_m and τ_v. We let $\mathsf{Comp}(\tau_e, \tau_m)$, $\mathsf{Comp}(\tau_v)$, and $\mathsf{Comp}(\tau)$ denote the set of block ciphers that are compatible with respectively τ_e and τ_m, τ_v, and τ. Then one can easily check (see for example [CS14] for a detailed explanation) that

$$\Pr[X_{\mathrm{re}} = \tau] = \frac{1}{|\mathcal{K}_h|} \cdot \Pr\left[E \leftarrow_{\$} \mathsf{Perm}(\mathcal{K}, n) : E \in \mathsf{Comp}(\tau)\right]. \tag{12}$$

We now define

$$\begin{aligned}\lambda_{\mathrm{eq}} = \{&(K_1, X_1, Y_1), \ldots, (K_{q_e}, X_{q_e}, Y_{q_e})\} \\ &\cup \{(N_1, H_{K_h}(M_1), T_1 \oplus H_{K_h}(M_1)), \ldots, (N_{q_m}, H_{K_h}(M_{q_m}), T_{q_m} \oplus H_{K_h}(M_{q_m}))\},\end{aligned}$$

and

$$\lambda_{\text{ineq}} = \{(N'_1, H_{K_h}(M'_1), T'_1 \oplus H_{K_h}(M'_1)), \ldots, (N'_{q_v}, H_{K_h}(M'_{q_v}), T'_{q_v} \oplus H_{K_h}(M'_{q_v}))\}.$$

Then, since τ is a good transcript, λ_{eq} is a permutation equalities list (as otherwise condition (C-1) or (C-2) would be fulfilled), and λ_{ineq} is a permutation inequalities list which is compatible with λ_{eq} (as otherwise condition (C-3) or (C-4) would be fulfilled). Moreover $|\lambda_{\text{eq}}| = q_m + q_e$, $|\lambda_{\text{ineq}}| = q_v$, and for $i = 1, \ldots, r$, L_i appears in λ_{eq} exactly $q_i + q'_i \leq q_e + \mu$ times. Note that the event $E \in \mathsf{Comp}(\tau)$ is actually equivalent to the event $E \in \mathsf{Comp}(\lambda)$ where $\lambda = (\lambda_{\text{eq}}, \lambda_{\text{ineq}})$. Using Lemma 3, one has

$$\Pr\left[E \in \mathsf{Comp}(\tau)\right] \geq \frac{1}{\prod_{i=1}^{r} (2^n)_{q_i+q'_i}} \left(1 - \frac{q_v}{2^n - \mu - q_e}\right).$$

Combining this equation with Equation (11) and Equation (12), we get

$$\frac{\Pr\left[X_{\text{re}} = \tau\right]}{\Pr\left[X_{\text{id}} = \tau\right]} \geq \left(1 - \frac{q_v}{2^n - \mu - q_e}\right) \cdot \prod_{i=1}^{r} \underbrace{\frac{(2^n)^{q'_i} (2^n)_{q_i}}{(2^n)_{q_i+q'_i}}}_{\geq 1} \geq 1 - \frac{q_v}{2^n - \mu - q_e}. \qquad \square$$

D Proof of Theorem 4

Following Section 2.3, let us fix a non-trivial (q_e, q_m, q_v)-distinguisher D interacting either with the real world $(E, \mathsf{HaK}[E, H, H']_{K_h, K'_h}, \mathsf{Ver}[E, H, H']_{K_h, K'_h})$ for a uniformly random block cipher E and independent random hashing keys K_h and K'_h, or with the *ideal world* $(E, \mathsf{Rand}, \mathsf{Rej})$, making at most q_e queries to its left (ideal cipher) oracle, at most q_m queries to its middle (MAC) oracle and at most q_v queries to its right (verification) oracle, and outputting a single bit. We let

$$\mathbf{Adv}(\mathsf{D}) = \Pr\left[\mathsf{D}^{E, \mathsf{HaK}[E,H,H']_{K_h,K'_h}, \mathsf{Ver}[E,H,H']_{K_h,K'_h}} = 1\right] - \Pr\left[\mathsf{D}^{E, \mathsf{Rand}, \mathsf{Rej}} = 1\right].$$

Let $\tau = (\tau_e, \tau_m, \tau_v, K_h, K'_h)$ be the transcript of the attack, where

$$\begin{aligned}
\tau_e &= \big((K_1, X_1, Y_1), \ldots, (K_{q_e}, X_{q_e}, Y_{q_e})\big) \\
\tau_m &= \big((M_1, T_1), \ldots, (M_{q_m}, T_{q_m})\big) \\
\tau_v &= \big((M'_1, T'_1), \ldots, (M'_{q_v}, T'_{q_v})\big).
\end{aligned}$$

As usual, we let Θ denote the set of attainable transcripts, and X_{re}, resp. X_{id}, the probability distribution of the transcript τ induced by the real world, resp. the ideal world. As in Appendix C, Theorem 4 follows easily by combining Equation (1) and Lemma 2 from Section 2.3 with Lemma 11 and Lemma 12 proven below.

We start by defining bad transcripts.

Definition 7. We say that an attainable transcript $\tau = (\tau_e, \tau_m, \tau_v, K_h, K'_h)$ is *bad* if one of these conditions is fulfilled:

(C-1) there exists two distinct MAC queries (M_i, T_i) and (M_j, T_j) such that $H'_{K'_h}(M_i) = H'_{K'_h}(M_j)$ and either $H_{K_h}(M_i) = H_{K_h}(M_j)$ or $T_i = T_j$;

(C-2) there exists an IC query $(K_i, X_i, Y_i) \in \tau_e$ and a MAC query $(M_j, T_j) \in \tau_m$ such that $K_i = H'_{K'_h}(M_j)$ and either $X_i = H_{K_h}(M_j)$ or $Y_i = T_j$;

(C-3) there exist a MAC query $(M_i, T_i) \in \tau_m$ and a verification query $(M'_j, T'_j) \in \tau_v$ such that

$$\begin{cases} H'_{K'_h}(M_i) = H'_{K'_h}(M'_j) \\ H_{K_h}(M_i) = H_{K_h}(M'_j) \\ T_i = T'_j. \end{cases}$$

(C-4) there exist an IC query $(K_i, X_i, Y_i) \in \tau_e$ and a verification query $(M'_j, T'_j) \in \tau_v$ such that

$$\begin{cases} K_i = H'_{K'_h}(M'_j) \\ X_i = H_{K_h}(M'_j) \\ Y_i = T'_j; \end{cases}$$

We let Θ_{bad}, resp. Θ_{good} denote the set of bad, respectively good transcripts.

We now upper bound the probability to get a bad transcript in the ideal world.

Lemma 11. *For any integers q_e, q_m and q_v, one has*

$$\Pr\left[X_{\mathrm{id}} \in \Theta_{\mathrm{bad}}\right] \leq q_m^2 \varepsilon^2 + q_m q_e (\varepsilon')^2 + \left(\frac{q_m}{2^n}\right)^{n+1} + n q_e \varepsilon' + q_m q_v \varepsilon^2 + q_v q_e (\varepsilon')^2.$$

Proof. Let Θ_i denote the set of attainable transcripts satisfying condition (C-i). Recall that, in the ideal world, (K_h, K'_h) is drawn independently from the queries transcript. We are going to consider each condition in turn.

Condition (C-1). This condition is exactly the same as condition (C-1) in Lemma 6, hence by exactly the same proof one has

$$\Pr\left[X_{\mathrm{id}} \in \Theta_1\right] \leq q_m^2 \varepsilon^2.$$

Condition (C-2). The probability that there exists an IC query $(K_i, X_i, Y_i) \in \tau_e$ and a MAC query $(M_j, T_j) \in \tau_m$ such that $K_i = H'_{K'_h}(M_j)$ and $X_i = H_{K_h}(M_j)$ (over the draw of K_h and K'_h) is at most $q_m q_e (\varepsilon')^2$. We now upper bound the probability that there exists an IC query $(K_i, X_i, Y_i) \in \tau_e$ and a MAC query $(M_j, T_j) \in \tau_m$ such that $K_i = H'_{K'_h}(M_j)$ and $Y_i = T_j$. Let us denote $\alpha(\tau_m)$ the maximal multiplicity of any tag in the MAC queries transcript, i.e.,

$$\alpha(\tau_m) = \max_{T \in \{0,1\}^n} \left|\left\{ j \in \{1, \ldots, q_m\} \,:\, T_j = T \right\}\right|.$$

Then, over the random draw of the T_j's, one has

$$\begin{aligned} \Pr\left[\alpha(\tau_m) \geq n+1\right] &= \Pr\left[\exists\, 1 \leq i_1 < \cdots < i_{n+1} \leq q_m \,:\, T_{i_1} = \cdots = T_{i_{n+1}}\right] \\ &\leq \frac{(q_m)_{n+1}}{(n+1)!(2^n)^n} \leq \left(\frac{q_m}{2^n}\right)^{n+1}, \end{aligned}$$

where we used that $(n+1)! \geq 2^n$. Now assume that $\alpha(\tau_m) \leq n$. Then there are at most $n q_e$ pairs of ideal cipher/MAC queries $((K_i, X_i, Y_i), (M_j, T_j))$ such that $Y_i = T_j$ and for each such pair, $K_i = H'_{K'_h}(M_j)$ with probability at most ε' over the random choice of K'_h. Hence,

$$\begin{aligned} \Pr\left[X_{\mathrm{id}} \in \Theta_2\right] &\leq q_m q_e (\varepsilon')^2 + \Pr\left[\alpha(\tau_m) \geq n+1\right] + \Pr\left[\alpha(\tau_m) \leq n\right] (n q_e \varepsilon') \\ &\leq q_m q_e (\varepsilon')^2 + \left(\frac{q_m}{2^n}\right)^{n+1} + n q_e \varepsilon'. \end{aligned}$$

Condition (C-3). This condition is exactly the same as condition (C-2) in Lemma 6, hence by exactly the same proof one has

$$\Pr\left[X_{\mathrm{id}} \in \Theta_3\right] \leq q_m q_v \varepsilon^2.$$

Condition (C-4). Fix an ideal cipher query $(K_i, X_i, Y_i) \in \tau_e$ and a verification query $(M'_j, T'_j) \in \tau_v$. Since in the ideal world K_h and K'_h are drawn independently from the queries transcript and H and H' are ε'-almost uniform, the probability that $K_i = H'_{K'_h}(M'_j)$ and $X_i = H_{K_h}(M'_j)$ is upper bounded by $(\varepsilon')^2$ (just ignoring the condition $Y_i = T'_j$), and hence

$$\Pr\left[X_{\mathrm{id}} \in \Theta_4\right] \leq q_v q_e (\varepsilon')^2.$$

The result follows by a union bound over these conditions. □

We now analyze good transcripts and prove the following lemma.

Lemma 12. *For any good transcript τ, one has*

$$\frac{\Pr\left[X_{\mathrm{re}} = \tau\right]}{\Pr\left[X_{\mathrm{id}} = \tau\right]} \geq 1 - \frac{q_v}{2^n - q_m - q_e}.$$

Proof. Let $\tau = (\tau_e, \tau_m, \tau_v, K_h, K'_h)$ be a good transcript. Let

$$\mathcal{L} = \{K_1, \ldots, K_{q_e}, H'_{K'_h}(M_1), \ldots, H'_{K'_h}(M_{q_m})\}$$

be the set of all the keys used in the ideal cipher or MAC queries. Using an arbitrary order, we rewrite the set $\mathcal{L}$ as

$$\mathcal{L} = \{L_1, \ldots, L_r\},$$

where r is the total number of distinct values in $\mathcal{L}$. For $i = 1, \ldots, r$, we let q_i denote the number of ideal cipher queries (K, X, Y) in τ_e such that $K = L_i$ and q'_i the number of MAC queries (M, T) in τ_m such that $H_{K'_h}(M) = L_i$.

Since in the ideal world the ideal cipher is perfectly random and independent from the other oracles, the MAC oracle is perfectly random, and the verification always rejects, one simply has

$$\Pr[X_{\mathrm{id}} = \tau] = \frac{1}{|\mathcal{K}_h| \cdot |\mathcal{K}'_h| \cdot (2^n)^{q_m} \cdot \prod_{i=1}^{r} (2^n)_{q_i}} = \frac{1}{|\mathcal{K}_h| \cdot |\mathcal{K}'_h|} \prod_{i=1}^{r} \frac{1}{(2^n)^{q'_i} (2^n)_{q_i}}, \tag{13}$$

since $q_m = \sum_{i=1}^{r} q'_i$. We must now lower bound the probability of getting τ in the real world. We say that a block cipher E is compatible with τ_m if

$$\forall i \in \{1, \ldots, q_m\},\ \mathsf{HaK}[E, H]_{K_h, K'_h}(M_i) = T_i,$$

compatible with τ_e if

$$\forall i \in \{1, \ldots, q_e\},\ E_{K_i}(X_i) = Y_i,$$

and compatible with τ_v if

$$\forall i \in \{1, \ldots, q_v\},\ \mathsf{HaK}[E, H]_{K_h, K'_h}(M'_i) \neq T'_i.$$

We simply say that E is compatible with τ if it is compatible with τ_e, τ_m and τ_v. We let $\mathsf{Comp}(\tau_e, \tau_m)$, $\mathsf{Comp}(\tau_v)$, and $\mathsf{Comp}(\tau)$ denote the set of block ciphers that are compatible with respectively τ_e and τ_m, τ_v, and τ. Then one can easily check that

$$\Pr[X_{\mathrm{re}} = \tau] = \frac{1}{|\mathcal{K}_h| \cdot |\mathcal{K}'_h|} \cdot \Pr\left[E \leftarrow_{\$} \mathsf{Perm}(\mathcal{K}, n) : E \in \mathsf{Comp}(\tau)\right]. \tag{14}$$

We now define

$$\lambda_{\mathrm{eq}} = \{(K_1, X_1, Y_1), \ldots, (K_{q_e}, X_{q_e}, Y_{q_e})\} \cup \bigcup_{i=1}^{q_m} \{(H'_{K'_h}(M_i), H_{K_h}(M_i), T_i)\},$$

and

$$\lambda_{\mathrm{ineq}} = \bigcup_{i=1}^{q_v} \{(H'_{K'_h}(M'_i), H_{K_h}(M'_i), T'_i)\}.$$

Then, since τ is a good transcript, λ_{eq} is a permutation equalities list (as otherwise condition (C-1) or (C-2) would be fulfilled) and λ_{ineq} is a permutation inequalities list which is compatible with λ_{eq} (as otherwise condition (C-3) or (C-4) would be fulfilled). Moreover $|\lambda_{\mathrm{eq}}| = q_m + q_e$, $|\lambda_{\mathrm{ineq}}| = q_v$, and for $i = 1 \ldots, r$, key L_i appears in λ_{eq} exactly $q_i + q'_i \le q_e + q_m$ times. Note that the event $E \in \mathsf{Comp}(\tau)$ is actually equivalent to the event $E \in \mathsf{Comp}(\lambda)$ where $\lambda = (\lambda_{\mathrm{eq}}, \lambda_{\mathrm{ineq}})$. Using Lemma 3, one has

$$\Pr\left[E \in \mathsf{Comp}(\tau)\right] \ge \frac{1}{\prod_{i=1}^{r} (2^n)_{q_i + q'_i}} \left(1 - \frac{q_v}{2^n - q_m - q_e}\right).$$

Combining this with Equation (13) and Equation (14), we get

$$\frac{\Pr\left[X_{\mathrm{re}} = \tau\right]}{\Pr\left[X_{\mathrm{id}} = \tau\right]} \ge \left(1 - \frac{q_v}{2^n - q_m - q_e}\right) \cdot \prod_{i=1}^{r} \underbrace{\frac{(2^n)^{q'_i} (2^n)_{q_i}}{(2^n)_{q_i + q'_i}}}_{\ge 1} \ge 1 - \frac{q_v}{2^n - q_m - q_e}. \qquad \square$$