Generic Security of NMAC and HMAC with Input Whitening

Peter Gaži[1(✉)], Krzysztof Pietrzak[1], and Stefano Tessaro[2]

[1] IST Austria, Klosterneuburg, Austria
{peter.gazi,pietrzak}@ist.ac.at
[2] UC Santa Barbara, Santa Barbara, USA
tessaro@cs.ucsb.edu

Abstract. HMAC and its variant NMAC are the most popular approaches to deriving a MAC (and more generally, a PRF) from a cryptographic hash function. Despite nearly two decades of research, their exact security still remains far from understood in many different contexts. Indeed, recent works have re-surfaced interest for *generic* attacks, i.e., attacks that treat the compression function of the underlying hash function as a black box.

Generic security can be proved in a model where the underlying compression function is modeled as a random function – yet, to date, the question of proving tight, non-trivial bounds on the generic security of HMAC/NMAC even as a PRF remains a challenging open question.

In this paper, we ask the question of whether a small modification to HMAC and NMAC can allow us to exactly characterize the security of the resulting constructions, while only incurring little penalty with respect to efficiency. To this end, we present simple variants of NMAC and HMAC, for which we prove tight bounds on the generic PRF security, expressed in terms of numbers of construction and compression function queries necessary to break the construction. All of our constructions are obtained via a (near) *black-box* modification of NMAC and HMAC, which can be interpreted as an initial step of key-dependent message pre-processing.

While our focus is on PRF security, a further attractive feature of our new constructions is that they clearly defeat all recent generic attacks against properties such as state recovery and universal forgery. These exploit properties of the so-called "functional graph" which are not directly accessible in our new constructions.

Keywords: Message authentication codes · HMAC · Generic attacks · Provable security

1 Introduction

This paper presents new variants of the HMAC/NMAC constructions of message authentication codes which enjoy *provable* security as a pseudorandom function (PRF) against generic distinguishing attacks, i.e., attacks which treat the compression function of the underlying hash function as a black-box. In particular,

T. Iwata and J.H. Cheon (Eds.): ASIACRYPT 2015, Part II, LNCS 9453, pp. 85–109, 2015.
DOI: 10.1007/978-3-662-48800-3_4

we prove concrete *tight* bounds in terms of the number of queries to the construction *and* to the compression function necessary to distinguishing our construction from a random function. Our constructions are the first HMAC/NMAC variants to enjoy such a tight analysis, and we see this as an important stepping stone towards the understanding of the generic security of such constructions.

Hash-Based MACs. HMAC [3] is the most widely used approach to key a hash function H to obtain a PRF or a MAC. It computes the output on message M and a key K as

$$\mathsf{HMAC}(K, M) = H(K \oplus \mathsf{opad} \,\|\, H(K \oplus \mathsf{ipad} \,\|\, M)),$$

where $\mathsf{opad} \neq \mathsf{ipad}$ are constants.[1] Usually, H is a hash function like SHA-1, SHA-256 or MD5, in particular following the Merkle-Damgård paradigm [4,16]. That is, it extends a compression function $\mathsf{f} : \{0,1\}^c \times \{0,1\}^b \to \{0,1\}^c$ into a hash function $\mathsf{MD}^{\mathsf{f}}_{\mathsf{IV}}$ by first padding M into b-bit blocks $M[1], \ldots, M[\ell]$, and then producing the output $H(M) = S_\ell$, where

$$S_0 \leftarrow \mathsf{IV}\ , \quad S_i \leftarrow \mathsf{f}(S_{i-1} \,\|\, M[i]) \text{ for all } i = 1, \ldots, \ell. \tag{1}$$

starting with the c-bit initialization value IV. A cleaner yet slightly less practical variant of HMAC is NMAC, which instead outputs

$$\mathsf{NMAC}_{K_{\mathsf{in}}, K_{\mathsf{out}}}(M) = \mathsf{MD}^{\mathsf{f}}_{K_{\mathsf{out}}}(\mathsf{MD}^{\mathsf{f}}_{K_{\mathsf{in}}}(M)),$$

where $K_{\mathsf{in}}, K_{\mathsf{out}} \in \{0,1\}^c$ are key values.

Security of HMAC/NMAC. The security of both constructions has been studied extensively, both by obtaining security proofs and proposing attacks. On the former side, NMAC and HMAC were proven to be secure *pseudorandom functions* (PRFs) in the standard model [3], later also using weaker assumptions [2] and via a tight bound in the uniform setting [7]. However, as argued in [7], this standard-model bound might be overly pessimistic, covering also very unnatural constructions of the underlying compression function f (for example the one used in their tightness proof). The authors hence argue for the need of an analysis of the PRF security of HMAC in the so-called *ideal compression function model* where the compression function is modelled as an ideal random function and the adversary is allowed to query it. This model was previously used by Dodis *et al.* [6] to study *indifferentiability* of HMAC, which however only holds for certain key lengths.

This is also the model implicitly underlying many of the recently proposed attacks on hash-based MACs [5,10,15,17,19,20,22]. These attacks are termed *generic*, meaning they can be mounted for any underlying hash function as long as it follows the Merkle-Damgård (MD) paradigm. The complexity of such a generic attack is then expressed in the number of key-dependent queries to the construction (denoted q_{C}) as well as the number of queries to the underlying compression function (denoted q_{f}). These two classes of queries are also often referred to as *online* and *offline*, respectively.

[1] Some details such as padding and arbitrary key length are addressed in Sect. 2.

All iterated MACs are subject to the long-known Preneel and van Oorschot's attack [21] which implies a forgery (and hence also distinguishing) attack against HMAC/NMAC making $q_{\mathrm{C}} = 2^{c/2}$ construction queries (consisting of constant-length messages) and no direct compression function queries (i.e., $q_{\mathsf{f}} = 0$). This immediately raises two questions:

> *How does the security of* HMAC and NMAC degrade *(in terms of tolerable q_{C}) by increasing (1) the length ℓ of the messages and (2) the number q_{f} of compression-function evaluations?*

The first question has been partially addressed in [7]. Their result[2] can be interpreted as giving tight bounds on the PRF security of NMAC against an attacker making q_{C} key-dependent construction queries (of length at most $\ell < 2^{c/3}$ b-bit blocks) but *no* queries to the compression function. They show that both constructions can only be distinguished from random function with advantage roughly $\epsilon(q_{\mathrm{C}}, \ell) \approx \ell^{1+o(1)} {q_{\mathrm{C}}}^2/2^c$, improving significantly on the bound $\epsilon(q_{\mathrm{C}}, \ell) \approx \ell^2 {q_{\mathrm{C}}}^2/2^c$ provable using standard folklore techniques. From our perspective, this bound can be read as a smooth trade-off: with increasing maximum allowed query length ℓ it tells us how many queries q_{C} can be tolerated for any acceptable upper bound on advantage.

Still, it is not clear how this trade-off changes when allowing extremely long messages ($\ell > 2^{c/3}$) and/or some queries to the compression function ($q_{\mathsf{f}} > 0$). Note that while huge ℓ can be prevented by standards, in practical settings q_{f} is very likely to be much higher than q_{C}, as it represents cheap local (offline) computation of the attacker. We therefore focus on capturing the trade-off between q_{C} and q_{f} for values of q_{C} that do not allow to mount the attack from [21]. However, as we argue below, getting such a tight trade-off for NMAC/HMAC seems to be out of reach for now, we hence relax the problem by allowing for slight modifications to the vanilla NMAC/HMAC construction.

Our Contributions. We ask the following question here, and answer it positively:

> *Can we devise variants of* HMAC/NMAC *whose security provably degrades gracefully with an increasing number of compression function queries q_{f}, possibly retaining security for q_{f} being much larger than 2^c?*

The main contribution of this paper is the introduction and analysis of a variant of NMAC (which we then adapt to the HMAC setting, as described below) which uses additional key material to "whiten" message blocks before being processed by the compression function. Concretely, our construction – termed WNMAC (for "whitened NMAC") uses an additional extra b-bit key K_{w}, and given a message M padded as $M[1], \ldots, M[\ell]$, operates as NMAC on input padded to blocks $M'[i] = M[i] \oplus K_b$, i.e., every message block is whitened with the *same* key (see also Fig. 1).

[2] Here we refer to Theorem 2 in [7] that formally considers a related construction NI in the standard model. However, its proof starts by a transition to the ideal-model analysis of a construction very closely related to NMAC, while disallowing compression-function queries.

The rationale behind WNMAC is two-fold. First, from the security viewpoint, the justification comes from the rich line of research on generic attacks on hash-based MACs. Most recent attacks [10,15,19,20] exploit the so-called "functional graph" of the compression function f, i.e., the graph capturing the structure of f when repeatedly invoked with its b-bit input fixed to some constant (say 0^b). Since our whitening denies the adversary the knowledge of b-bit inputs on which f is invoked during construction queries, intuitively it seems to be the right way to foil such attacks. Moreover, a recent work by Sasaki and Wang [22] suggests that keying *every* invocation of f is necessary in order to prevent suboptimal security against generic state recovery attacks. WNMAC arguably provides the simplest and most natural such keying. Second, from the practical perspective, WNMAC can be implemented on top of an existing implementation of NMAC, using it as a black-box.

PRF-Security of WNMAC. Our main result shows that WNMAC is a secure PRF; more precisely, no attacker making at most q_{C} construction queries (for messages padded into at most ℓ blocks) and q_{f} primitive queries can distinguish WNMAC from a random function, except with distinguishing advantage

$$\epsilon_{\mathsf{WNMAC}}(q_{\mathsf{C}}, q_{\mathsf{f}}, \ell) \leq \frac{q_{\mathsf{f}} q_{\mathsf{C}}}{2^{2c}} + 2 \cdot \frac{\ell {q_{\mathsf{C}}}^2 q_{\mathsf{f}}}{2^{b+c}} + \frac{\ell {q_{\mathsf{C}}}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c} + 1 \right).$$

Here, $d'(\ell)$ is the maximum, over all positive integers $\ell' \leq \ell$, of the number of positive divisors of ℓ', and grows very slowly, i.e., $d'(\ell) \approx \ell^{1/\ln\ln \ell}$. We also prove that this bound is essentially tight. Namely, we give an attack that achieves advantage roughly $q_{\mathsf{C}} q_{\mathsf{f}} / 2^{2c}$, showing the first term above to be necessary. Additionally, we know from [7] that the third term is tight for $\ell \leq 2^{c/3}$.

Note that in the case of $q_{\mathsf{f}} = 0$, the bound matches exactly the bound from [7]. Moreover, observe that under the realistic assumption that $\ell < \min\{2^{c/3}, 2^{b-c}\}$, the bound simplifies to

$$\epsilon_{\mathsf{WNMAC}}(q_{\mathsf{C}}, q_{\mathsf{f}}, \ell) \leq 3\frac{q_{\mathsf{f}} q_{\mathsf{C}}}{2^{2c}} + (d'(\ell) + 2) \cdot \frac{\ell {q_{\mathsf{C}}}^2}{2^c}.$$

Ignoring $d'(\ell)$ for simplicity, we see that we can tolerate up to $q_{\mathsf{C}} \approx 2^{c/2}/\sqrt{\ell}$ construction queries and up to $q_{\mathsf{f}} \approx 2^{1.5c}$ primitive queries. This corresponds to the security threshold ranging from 2^{192} f-queries for MD5 up to 2^{768} f-queries for SHA-512. The first term also clearly characterizes the complete trade-off curve between $q_{\mathsf{C}} < 2^{c/2}/\sqrt{\ell}$ and q_{f} for any reasonable upper bound on the message length and acceptable distinguishing advantage.

Other Security Properties. Additionally, we also analyze the security level WNMAC achieves with respect to other security notions frequently considered in the attacks literature. By a series of reductions, we show that, roughly speaking, $\epsilon_{\mathsf{WNMAC}}$ also upper-bounds the adversary's advantage for *distinguishing-H* and *state recovery*. We believe that addressing these cryptanalytic notions also using the traditional toolbox of provable security is important and see this paper as taking the first step on that path.

Lifting to HMAC. We then move our attention from NMAC to HMAC and propose two analogous modifications to it. The first one, called WHMAC, is obtained from HMAC in the same way WNMAC is obtained from NMAC: by whitening the padded message blocks with an independent key, The second one, termed WHMAC$^+$, additionally processes a fresh key K^+ instead of the first block of the message. Both variants can be implemented given only black-box access to HMAC, and we prove that they maintain the same security level as WNMAC as long as the parameters b, c of f satisfy $b \gg 2c$ (for WHMAC) or $b \gg c$ (for WHMAC$^+$). Note that for existing hash functions, the former condition is satisfied for both MD5 and SHA-1, while the latter holds also for SHA-256 and SHA-512.

The Dual Construction. Motivated by the most restrictive term $q_C q_f/2^{2c}$ in $\epsilon_{\mathsf{WNMAC}}$, the final construction we propose in this paper is a "dual" version of WNMAC denoted DWNMAC, that differs in the final, outer f-call. Instead of $\mathsf{f}(K_2, s \,\|\, 0^{b-c})$ for a c-bit key K_2 and a c-bit state s padded with zeroes, the outer call in DWNMAC computes $\mathsf{f}(s, K_2)$ for a longer, b-bit key. As expected, we prove that this tweak removes the need for the $q_C q_f/2^{2c}$ term and replaces it by the strictly favourable term $q_C q_f/2^{b+c}$, proving that the zero-padding in the outer call of WNMAC was actually responsible for the "bottle-neck" term in its security bound.

Our Techniques. In our information-theoretic analysis of WNMAC we employ the H-coefficient technique by Patarin [18], partially inheriting the notational framework from the recent analysis of keyed sponges by Gaži, Pietrzak, and Tessaro [8]. On a high level, the heart of our proof is a careful analysis of the probability that two sets intersect in the ideal experiment: (1) the set of adversarial queries to f, and (2) the set of inputs on which f is invoked when answering the adversary's queries to WNMAC. Obtaining a bound on the probability of this event then allows us to exclude it and use the result from [7] that considers $q_f = 0$, properly adapted to the WNMAC setting.

Related Work. As mentioned above, the motivation for our work partially stems from the recent line of work on generic attacks against iterated hash-based MACs [5,10,15,17,19,20,22]. While our security bound for WNMAC does not exclude attacks of the complexity (in terms of numbers of queries and message lengths) considered in these papers, the design of WNMAC was partially guided by the structure of these attacks and seems to prevent them. We find in particular the work [22] to be a good justification for investigating the security of WNMAC and related constructions. Iterated MAC that uses keying in every f-invocation was already considered by An and Bellare [1], their construction NI was later subject to analysis [7] that we adapt and reuse. One can see WNMAC as a conceptual simplification of NI where the key is simply used to whiten the b-bit input to the compression function. Finally, our dual construction considered in Sect. 5 bears resemblance to the Sandwich MAC analyzed by Yasuda [23], we believe that our methods could be easily adapted to cover this construction as well.

Perspective and Open Problems. We stress that the reader should not conclude from this work that NMAC and HMAC are necessarily less secure than the

constructions proposed in this paper, specifically with respect to PRF security. In fact, we are not aware of any attacks showing a separation between the PRF security of our constructions and that of the original NMAC/HMAC constructions, finding one is an interesting open problem.

While obtaining a non-tight birthday-type bound for NMAC/HMAC is feasible (for most key-length values, a bound follow directly from the indifferentiability analysis of [6]), proving *tight* bounds in terms of compression function and construction queries on the generic PRF security of NMAC/HMAC is a challenging open problem, on which little progress has been made. The main challenge is to understand how partial information in form of f-queries can help the attacker to break security (i.e., distinguish) in settings with $q_{\mathrm{C}} \ll 2^{c/2}/\sqrt{\ell}$, when the attack from [7] does not apply. This will require in particular developing a better understanding of the functional graph defined by queries to the function f. Some of its properties have been indeed exploited in existing generic attacks, but proving security appears to require a much deeper understanding: Most of the recent attacks, which are probably still not tight, do not come with rigorous proofs but instead rely on conjectures on the structure of these graphs [10]. The difficulty of this question for NMAC/HMAC is also well documented by the fact that even proving security of the whitened constructions presented in this paper required some novel tricks and considerable effort.

Similarly, it remains equally challenging to prove that for the properties considered by the recent HMAC/NMAC attacks (such as distinguishing-H, state recovery or various types of forgeries), the security of WNMAC/WHMAC is provably superior. Yet, we note that our construction invalidates direct application of all existing attacks, and hence we feel confident conjecturing that its security is much higher.

Black-box Instantiations. Throughout the paper we implicitly assume we can add a key to each b-bit input block, even though we aim for a black-box instantiation. For many MD-based hash functions, such fine-grained control of the input to the compression function is generally not possible via a black-box message pre-processing. Concretely, the functions from the SHA-family with 512-bit blocks only allow to effectively control (via alterations of the message) the first 447 bits of the last block, since the remaining 65 bits are reserved for the 64-bit length, and an additional 1-bit. Our analysis can be easily modified to take this into account. The resulting bound will change very little, and will result in the term $\ell q_{\mathrm{C}} q_{\mathsf{f}}/2^{b+c}$ being replaced by the term $(\ell-1+2^d)\cdot q_{\mathrm{C}}\cdot q_{\mathsf{f}}/2^{b+c}$, where d is the length of the non-controllable part of the input (for SHA-functions, $d = 65$). Note that since $d \ll b - c$, this will not affect the tightness of the bounds for concrete parameters.

2 Preliminaries

Basic Notation. We denote $[n] := \{1, \ldots, n\}$. Moreover, for a finite set $\mathcal{S}$ (e.g., $\mathcal{S} = \{0, 1\}$), we let $\mathcal{S}^n$, $\mathcal{S}^+$ and $\mathcal{S}^*$ be the sets of sequences of elements of $\mathcal{S}$ of length n, of arbitrary (but non-zero) length, and of arbitrary length,

respectively (with ε denoting the empty sequence, as opposed to ϵ which is a small quantity). As a shorthand, let $\{0,1\}^{b*}$ denote $\left(\{0,1\}^b\right)^*$. We denote by $S[i]$ the i-th element of $S \in \mathcal{S}^n$ for all $i \in [n]$. Similarly, we denote by $S[i \dots j]$, for every $1 \le i \le j \le n$, the sub-sequence consisting of $S[i], S[i+1], \dots, S[j]$, with the convention that $S[i \dots i] = S[i]$. Moreover, we denote by $S \,\|\, S'$ the concatenation of two sequences in $\mathcal{S}^*$, and also, we let $S \,|\, T$ be the usual prefix-of relation: $S \,|\, T :\Leftrightarrow (\exists S' \in \mathcal{S}^* \colon S \,\|\, S' = T)$.

For an integer n, $d(n) = |\{i \in \mathbb{N} : i \mid n\}|$ is the number of its positive divisors and

$$d'(n) := \max_{n' \in \{1,\dots,n\}} |\{d \in \mathbb{N} : d \mid n'\}| \approx n^{1/\ln\ln n}$$

is the maximum, over all positive integers $n' \le n$, of the number of positive divisors of n'. More precisely, we have $\forall \varepsilon > 0\ \exists n_0\ \forall n > n_0 \colon d(n) < n^{(1+\varepsilon)/\ln\ln n}$ [11].

We also let $\mathcal{F}(\mathcal{D}, \mathcal{R})$ be the set of all functions from $\mathcal{D}$ to $\mathcal{R}$; and with a slight abuse of notation we sometimes write $\mathcal{F}(m, n)$ (resp. $\mathcal{F}(*, n)$) to denote the set of functions mapping m-bit strings to n-bit strings (resp. from $\{0,1\}^*$ to $\{0,1\}^n$). We denote by $x \xleftarrow{\$} \mathcal{X}$ the act of sampling x uniformly at random from $\mathcal{X}$. Finally, we denote the event that an adversary A, given access to an oracle O, outputs a value y, as $\mathsf{A}^{\mathsf{O}} \Rightarrow y$. To emphasize the random experiment considered, we sometimes denote the probability of an event A in a random experiment E by $\mathsf{P}^{\mathsf{E}}[A]$. Finally, the min-entropy $\mathsf{H}_\infty(X)$ of a random variable X with range $\mathcal{X}$ is defined as $-\log\left(\max_{x \in \mathcal{X}} \mathsf{P}_X(x)\right)$.

Pseudorandom Functions. We consider *keyed* functions $\mathsf{F} \colon \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ taking a κ-bit key (i.e., $\mathcal{K} = \{0,1\}^\kappa$), a message $M \in \mathcal{D}$ as input, and returning an output from $\mathcal{R}$. For a keyed function F under a key $k \in \mathcal{K}$ we often write $\mathsf{F}_k(\cdot)$ instead of $\mathsf{F}(k, \cdot)$. One often considers the security of F as a *pseudorandom function* (or PRF, for short) [9]. This is defined via the following advantage measure, involving an adversary A:

$$\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathsf{A}) := \left|\mathsf{P}\left[K \xleftarrow{\$} \{0,1\}^\kappa : \ \mathsf{A}^{\mathsf{F}_K} \Rightarrow 1\right] - \mathsf{P}\left[f \xleftarrow{\$} \mathcal{F}(\mathcal{D}, \mathcal{R}) : \ \mathsf{A}^{f} \Rightarrow 1\right]\right|.$$

Informally, we say that F is a PRF if this advantage is "negligible" for all "efficient" adversaries A.

PRFs in the Ideal Compression Function Model. For our analysis below, we are going to consider keyed constructions $\mathsf{C}[\mathsf{f}] \colon \{0,1\}^\kappa \times \mathcal{D} \to \mathcal{R}$ which make queries to a randomly chosen compression function $\mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b, c)$ which can also be evaluated by the adversary (we sometimes write C^{f} instead of $\mathsf{C}[\mathsf{f}]$). For this case, we use the following notation to express the PRF advantage of A:

$$\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{C}[\mathsf{f}]}(\mathsf{A}) := \Big|\mathsf{P}\left[K \xleftarrow{\$} \{0,1\}^\kappa, \mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b, c) : \ \mathsf{A}^{\mathsf{C}^{\mathsf{f}}_K, \mathsf{f}} \Rightarrow 1\right] - \mathsf{P}\left[\mathsf{R} \xleftarrow{\$} \mathcal{F}(\mathcal{D}, \mathcal{R}), \mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b, c) : \ \mathsf{A}^{\mathsf{R}, \mathsf{f}} \Rightarrow 1\right]\Big|.$$

We call A's queries to its first oracle *construction queries* (or C-queries) and its queries to the second oracle as *primitive queries* (or f-queries).

Note that the notion of PRF-security is identical to the notion of *distinguishing-R*, first defined in [13] and often used in the cryptanalytic literature on hash-based MACs.

Distinguishing-H. A further security notion defined in [13] is the so-called *distinguishing-H* security. Here, the goal of the adversary is to distinguish the hash-based MAC construction $\mathsf{C}_K[\mathsf{f}]$ using its underlying compression function f (say SHA-1) and a random key K, from the same construction $\mathsf{C}_K[\mathsf{g}]$ built on top of an independent random compression function g. In the ideal compression function model, where we model already the initial compression function f as ideal, this corresponds to distinguishing a pair of oracles $(\mathsf{C}_K[\mathsf{f}], \mathsf{f})$ from $(\mathsf{C}_K[\mathsf{f}], \mathsf{g})$. Formally,

$$\mathsf{Adv}^{\mathsf{dist\text{-}H}}_{\mathsf{C}}(\mathsf{A}) := \Big|\mathsf{P}\left[K \xleftarrow{\$} \{0,1\}^\kappa, \mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b,c) : \ \mathsf{A}^{\mathsf{C}^{\mathsf{f}}_K,\mathsf{f}} \Rightarrow 1\right] - \mathsf{P}\left[K \xleftarrow{\$} \{0,1\}^\kappa, \mathsf{f},\mathsf{g} \xleftarrow{\$} \mathcal{F}(c+b,c) : \ \mathsf{A}^{\mathsf{C}^{\mathsf{f}}_K,\mathsf{g}} \Rightarrow 1\right]\Big|.$$

State Recovery. An additional notion considered in the literature is security against *state recovery*. Since the definition of this notion needs to be tailored for the concrete construction it is applied to, we postpone the formal definition of security against state recovery to Sect. 3.10.

MACs and Unpredictability. It is well known that a good PRF also yields a good message-authentication code (MAC). A concrete security bound for unforgeability can be obtained from the PRF bound via a standard argument.

Iterated MACs. For a keyed function $\mathsf{f} : \{0,1\}^c \times \{0,1\}^b \to \{0,1\}^c$ we denote with $\mathsf{Casc}^{\mathsf{f}} : \{0,1\}^c \times \{0,1\}^{b*} \to \{0,1\}^c$ the cascade construction (also known as Merkle-Damgård) built from f as

$$\mathsf{Casc}^{\mathsf{f}}(K, m_1 \| \ldots \| m_\ell) := y_\ell \text{ where } y_0 := K \text{ and for } i \geq 1 \ : \ y_i := \mathsf{f}(y_{i-1}, m_i),$$

in particular $\mathsf{Casc}^{\mathsf{f}}(K, \varepsilon) := K$.

The construction $\mathsf{NMAC}^{\mathsf{f}} : (\{0,1\}^c)^2 \times \{0,1\}^{b*} \to \{0,1\}^c$ is derived from $\mathsf{Casc}^{\mathsf{f}}$ by adding an additional, independently keyed application of f at the end. It assumes that the domain sizes of f satisfy $b \geq c$ and the output of the cascade is padded with zeroes before the last f-call. Formally,

$$\mathsf{NMAC}^{\mathsf{f}}((K_1, K_2), M) := \mathsf{f}(K_2, \mathsf{Casc}^{\mathsf{f}}(K_1, M) \| 0^{b-c}).$$

Note that practical MD-based hash functions take as input arbitrary-length bit-strings and then pad them to a multiple of the block length, often including the message length in the so-called MD-strengthening. This padding then also appears in NMAC (and HMAC) but here we take the customary shortcut and our definition of NMAC above (resp. HMAC below) actually corresponds to the

generalized constructions denoted as GNMAC (resp. GHMAC) in [2] where this step is also justified in detail.

$\mathsf{HMAC}^{\mathsf{f}}$ is a practice-oriented version of $\mathsf{NMAC}^{\mathsf{f}}$, where the two keys (K_1, K_2) are derived from a single key $K \in \{0,1\}^b$ by xor-ing it with two fixed b-bit strings ipad and opad. In addition, the keys are not given through the key-input of the compression function f, but are prepended to the message instead. This allows for the usage of existing implementations of hash functions that contain a hard-coded initialization vector IV. Formally:

$$\begin{aligned}\mathsf{HMAC}^{\mathsf{f}}(K, m) &:= \mathsf{Casc}^{\mathsf{f}}(\mathsf{IV}, K_2 \| \mathsf{Casc}^{\mathsf{f}}(\mathsf{IV}, K_1 \| m) \| \mathsf{fpad}) \\ &\quad \text{where } (K_1, K_2) := (K \oplus \mathsf{ipad}, K \oplus \mathsf{opad})\end{aligned}$$

and fpad is a fixed $(b-c)$-bit padding not affecting the security analysis. (Technically, [14] allows for arbitrary length of the key K: a key shorter than b bits is padded with zeroes before applying the xor transformations, a longer key is first hashed.)

3 The Whitened NMAC Construction

We now present our main construction called *Whitened NMAC* (or WNMAC for short). To that end, let us first consider a modification of the cascade construction Casc called *whitened cascade* and denoted WCasc. For a keyed function $\mathsf{f} : \{0,1\}^c \times \{0,1\}^b \to \{0,1\}^c$ we denote with $\mathsf{WCasc}^{\mathsf{f}} : (\{0,1\}^c \times \{0,1\}^b) \times \{0,1\}^{b*} \to \{0,1\}^c$ the whitened cascade construction built from f as

$$\begin{aligned}&\mathsf{WCasc}^{\mathsf{f}}((K_1, K_{\mathrm{w}}), m_1 \| \ldots \| m_\ell) := y_\ell \\ &\text{where } y_0 := K_1 \text{ and for } i \geq 1 \ : \ y_i := \mathsf{f}(y_{i-1}, m_i \oplus K_{\mathrm{w}}),\end{aligned}$$

in particular $\mathsf{WCasc}^{\mathsf{f}}((K_1, K_{\mathrm{w}}), \varepsilon) := K_1$.

The construction WNMAC is derived from NMAC, the only difference being that the inner cascade Casc is replaced by the whitened cascade WCasc. More precisely,

$$\mathsf{WNMAC}^{\mathsf{f}}((K_1, K_2, K_{\mathrm{w}}), M) := \mathsf{f}(K_2, \mathsf{WCasc}^{\mathsf{f}}((K_1, K_{\mathrm{w}}), M) \| 0^{b-c}).$$

For a graphical depiction of WNMAC, see Fig. 1. We devote most of this section to the proof of the following theorem that quantifies the PRF-security of WNMAC.

Theorem 1 (PRF-Security of WNMAC). *Let* A *be an adversary making at most* q_{f} *queries to the compression function* f *and at most* q_{C} *construction queries, each of length at most* ℓ *b-bit blocks. Let* $K = (K_1, K_2, K_{\mathrm{w}}) \in \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$ *be a tuple of random keys. Then we have*

$$\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{WNMAC}^{\mathsf{f}}_K}(\mathsf{A}) \leq \frac{q_{\mathsf{f}} q_{\mathrm{C}}}{2^{2c}} + 2 \cdot \frac{\ell q_{\mathrm{C}} q_{\mathsf{f}}}{2^{b+c}} + \frac{\ell {q_{\mathrm{C}}}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c} + 1 \right). \qquad (2)$$

Note that as observed in Sect. 2, this also covers the so-called distinguishing-R security of WNMAC. Moreover, our analysis also implies security bounds for distinguishing-H and state recovery, as we discuss later.

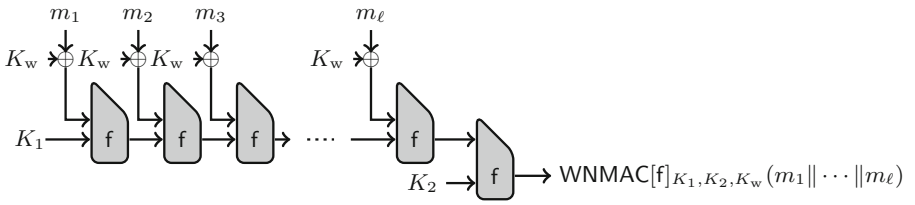

Fig. 1. The construction $\mathsf{WNMAC}[\mathsf{f}]_{K_1,K_2,K_\mathrm{w}}$.

3.1 Basic Notation, Message Trees and Repetition Patterns

Let us fix an adversary A. We assume that A is deterministic, it makes *exactly* q_f queries to f and q_C construction queries, and it never repeats the same query twice. All these assumptions are without loss of generality for an information-theoretic indistinguishability analysis, since an arbitrary (possibly randomized) adversary making at most this many queries can be transformed into one satisfying the above constraints and achieving advantage which is at least as large.

Let $\mathcal{Q}_C \subseteq \left(\{0,1\}^b\right)^*$ be any non-empty set of messages (later this will represent the set of A's C-queries). Based on it, we now introduce the *message tree* and its labeled version, which capture the inherent combinatorial structure of the messages $\mathcal{Q}_C$, as well as the internal values computed while these messages are processed by $\mathsf{WCasc}^\mathsf{f}$ inside of $\mathsf{WNMAC}^\mathsf{f}$. The message tree $T(\mathcal{Q}_C) = (V, E)$ for $\mathcal{Q}_C$ is defined as follows:

- The vertex set is $V := \left\{ M' \in \left(\{0,1\}^b\right)^* : \exists M \in \mathcal{Q}_C : M' \mid M \right\}$, where $\mid$ is the prefix-of partial ordering of strings. In particular, note that the empty string ε is a vertex and that $\mathcal{Q}_C \subseteq V$.
- The set $E \subseteq V \times V$ of (directed) edges is

$$E := \left\{ (M, M') : \exists m \in \{0,1\}^b : M' = M \,\|\, m \right\}.$$

To simplify our exposition, we also define the following two mappings based on $T(\mathcal{Q}_C)$.

- The mapping $\pi(v)\colon V \setminus \{\varepsilon\} \to V$ returns the unique parent node of $v \in V \setminus \{\varepsilon\}$; i.e., the unique node u such that $(u, v) \in E$.
- The mapping $\mu(v)\colon V \setminus \{\varepsilon\} \to \{0,1\}^b$ returns the unique message block $m \in \{0,1\}^b$ such that $\pi(v) \,\|\, \mu(v) = v$ (intuitively, this will be the message block that is processed when "arriving" in vertex v).

Alternatively, with a slight abuse of notation we will also refer to the vertices in V as $v_1, \ldots, v_{|V|}$ which is an arbitrary ordering of them such that for all $1 \le i, j \le |V|$ it satisfies $v_i | v_j \Rightarrow i \le j$. Note that one obtains such an ordering for example if one, intuitively speaking, processes the messages in $\mathcal{Q}_C$ block-wise and labels the vertices by their "first appearance": in particular $v_1 = \varepsilon$ is the tree root.

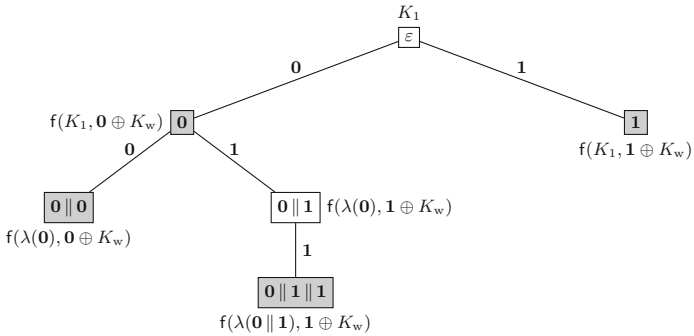

Fig. 2. Labeled message tree. Example of a labeled message tree $T^{\mathsf{f}}_K(\mathcal{Q}_C)$ for four messages $\mathcal{Q}_C = \{\mathbf{0}, \mathbf{0} \,\|\, \mathbf{0}, \mathbf{0} \,\|\, \mathbf{1} \,\|\, \mathbf{1}, \mathbf{1}\}$, where $\boldsymbol{r} = r^b$ for $r \in \{0,1\}$. The gray vertices correspond to these four messages. Next to each vertex v and edge (u, v), we give the label $\lambda(v)$ and the value $\mu(v)$, respectively.

Additionally, for a mapping $\mathsf{f}\colon \{0,1\}^c \times \{0,1\}^b \to \{0,1\}^c$ and a key tuple $K = (K_1, K_2, K_w) \in \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$ we also consider an extended version of $T(\mathcal{Q}_C)$ which we call the *labeled message tree* and denote $T^{\mathsf{f}}_K(\mathcal{Q}_C) = (V, E, \lambda)$, and which is defined as follows:

- The set of vertices V and edges E are defined exactly as for $T(\mathcal{Q}_C)$ above.
- The vertex-labeling function $\lambda\colon V \to \{0,1\}^c$ is defined iteratively: $\lambda(\varepsilon) := K_1$ and for each non-root vertex $v \in V \setminus \{\varepsilon\}$ we put $\lambda(v) := \mathsf{f}(\lambda(\pi(v)), \mu(v) \oplus K_w)$.

An example of a labeled message tree is given in Fig. 2. Note that each vertex label $\lambda(v)$ is exactly the output of the inner, whitened cascade $\mathsf{WCasc}^{\mathsf{f}}_{K_1,K_w}(v)$ in $\mathsf{WNMAC}^{\mathsf{f}}_K$ (recall that v is actually a message from $\{0,1\}^{b*}$).

For any message tree $T(\mathcal{Q}_C) = (V, E)$, a *repetition pattern* is any equivalence relation ρ on V. For a labeled message tree $T^{\mathsf{f}}_K(\mathcal{Q}_C) = (V, E, \lambda)$ we say that a repetition pattern ρ is *induced* by it if it satisfies

$$\forall u, v \in V : \lambda(u) = \lambda(v) \Leftrightarrow \rho(u, v).$$

3.2 Interactions and Transcripts

Let $\mathcal{QR}_C$ denote the set of q_C pairs (x, r) such that $x \in \{0,1\}^{b*}$ is a construction query and $r \in \{0,1\}^c$ is a potential response to it (what we mean by "potential" will be clear from below). Similarly let $\mathcal{QR}_{\mathsf{f}}$ denote the set of q_{f} pairs (x, r) such that $x \in \{0,1\}^c \times \{0,1\}^b$ is an f-query and $r \in \{0,1\}^c$ is a potential response to it. Let $\mathcal{Q}_C \subseteq \{0,1\}^{b*}$ and $\mathcal{Q}_{\mathsf{f}} \subseteq \{0,1\}^c \times \{0,1\}^b$ denote the sets of first coordinates (i.e., the queries) in $\mathcal{QR}_C$ and $\mathcal{QR}_{\mathsf{f}}$, respectively; we have $|\mathcal{Q}_C| = q_C$ and $|\mathcal{Q}_{\mathsf{f}}| = q_{\mathsf{f}}$.

We call the pair of sets $(\mathcal{QR}_C, \mathcal{QR}_{\mathsf{f}})$ *valid* if the adversary A would indeed ask these queries throughout the experiment, assuming that each of her queries

would be replied by the respective response in $\mathcal{QR}_C$ or $\mathcal{QR}_\mathsf{f}$ (note that once a deterministic A is fixed, this determines whether a given pair $(\mathcal{QR}_C, \mathcal{QR}_\mathsf{f})$ is valid).

We then define a *valid transcript* to be of the form

$$\tau = \left(\mathcal{QR}_C, \mathcal{QR}_\mathsf{f}, K = (K_1, K_2, K_\mathrm{w}), T_K^\mathsf{f}(\mathcal{Q}_C)\right),$$

where $(\mathcal{QR}_C, \mathcal{QR}_\mathsf{f})$ is valid, $\mathsf{f}\colon \{0,1\}^c \times \{0,1\}^b \to \{0,1\}^c$ is a function and $K = (K_1, K_2, K_\mathrm{w}) \in \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$ is a key tuple.

We differentiate between the ways in which such valid transcripts are generated in the real and in the ideal worlds (or experiments), respectively, by defining corresponding distributions T_real and $\mathsf{T}_\mathsf{ideal}$ over the set of valid transcripts:

Real World. The transcript T_real for the adversary A is obtained by sampling $\mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b, c)$ and $K = (K_1, K_2, K_\mathrm{w}) \leftarrow \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$, and letting T_real denote

$$\left(\mathcal{QR}_C = \{(M_i, Y_i)\}_{i=1}^{q_\mathrm{C}}, \mathcal{QR}_\mathsf{f} = \{(X_i, R_i)\}_{i=1}^{q_\mathsf{f}}, K = (K_1, K_2, K_\mathrm{w}), T_K^\mathsf{f}(\mathcal{Q}_C)\right),$$

where we execute A, which asks construction queries $M_1, \ldots, M_{q_\mathrm{C}}$ answered with $Y_i := \mathsf{WNMAC}[\mathsf{f}]_K(M_i)$ for all $i \in [q_\mathrm{C}]$; and f-queries $X_1, \ldots, X_{q_\mathsf{f}}$ answered with $R_i := \mathsf{f}(X_i)$ for all $i \in [q_\mathsf{f}]$ (note that the C-queries and f-queries may in general be interleaved adaptively, depending on A). Finally, we let $T_K^\mathsf{f}(\mathcal{Q}_C)$ be the labeled message tree corresponding to $\mathcal{Q}_C$, f and K.

Ideal World. The transcript $\mathsf{T}_\mathsf{ideal}$ for the adversary A is obtained similarly to the above, but here, together with the random function $\mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b, c)$ and the key tuple $K = (K_1, K_2, K_\mathrm{w}) \leftarrow \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$, we also sample q_C independent random values $Y_1, \ldots, Y_{q_\mathrm{C}} \in \{0,1\}^r$. Then we let $\mathsf{T}_\mathsf{ideal}$ denote

$$\left(\mathcal{QR}_C = \{(M_i, Y_i)\}_{i=1}^{q_\mathrm{C}}, \mathcal{QR}_\mathsf{f} = \{(X_i, R_i)\}_{i=1}^{q_\mathsf{f}}, K = (K_1, K_2, K_\mathrm{w}), T_K^\mathsf{f}(\mathcal{Q}_C)\right),$$

where we execute A, answer each its C-query M_i with Y_i for all $i \in [q_\mathrm{C}]$ and each its f-query X_i with $R_i := \mathsf{f}(X_i)$ for all $i \in [q_\mathsf{f}]$. Then we let $T_K^\mathsf{f}(\mathcal{Q}_C)$ be the labeled message tree corresponding to $\mathcal{Q}_C$, f and K.

Later we refer to the above two random experiments as real and ideal, respectively. Note that the range of T_real is included in the range of $\mathsf{T}_\mathsf{ideal}$ by definition, and that the range of $\mathsf{T}_\mathsf{ideal}$ is easily seen to contain all valid transcripts.

3.3 The H-Coefficient Method

We upper-bound the advantage A in distinguishing $\mathsf{WNMAC}[\mathsf{f}]_K$ for $\mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b, c)$ from a random function in terms of the statistical distance of the transcripts, i.e.,

$$\mathsf{Adv}^\mathsf{prf}_\mathsf{WNMAC}(\mathsf{A}) \le \mathsf{SD}(\mathsf{T}_\mathsf{real}, \mathsf{T}_\mathsf{ideal}) = \frac{1}{2} \sum_\tau \left|\mathsf{P}\left[\mathsf{T}_\mathsf{real} = \tau\right] - \mathsf{P}\left[\mathsf{T}_\mathsf{ideal} = \tau\right]\right|, \qquad (3)$$

where the sum is over all valid transcripts. This is because an adversary for $\mathsf{T}_{\mathsf{real}}$ and $\mathsf{T}_{\mathsf{ideal}}$, whose optimal advantage is exactly $\mathsf{SD}(\mathsf{T}_{\mathsf{real}}, \mathsf{T}_{\mathsf{ideal}})$, can always output the same decision bit as A, ignoring any extra information provided by the transcript.

We are going to use Patarin's H-coefficient method [18]. This means that we need to partition the set of valid transcripts into *good* transcripts GT and *bad* transcripts BT and then apply the following lemma.

Lemma 1 (The H-Coefficient Method [18]). *Let $\delta, \epsilon \in [0,1]$ be such that:*

(a) $\mathsf{P}[\mathsf{T}_{\mathsf{ideal}} \in \mathsf{BT}] \leq \delta$.
(b) *For all $\tau \in \mathsf{GT}$,*

$$\frac{\mathsf{P}[\mathsf{T}_{\mathsf{real}} = \tau]}{\mathsf{P}[\mathsf{T}_{\mathsf{ideal}} = \tau]} \geq 1 - \epsilon.$$

Then,

$$\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{WNMAC}}(\mathsf{A}) \leq \mathsf{SD}(\mathsf{T}_{\mathsf{real}}, \mathsf{T}_{\mathsf{ideal}}) \leq \epsilon + \delta.$$

More verbally, we want a set of good transcripts GT such that with very high probability (i.e., $1 - \delta$) a generated transcript *in the ideal world* is going to be in this set, and moreover, for each such good transcript, the probabilities that it occurs in the real and in the ideal worlds are *roughly* the same, i.e., at most a multiplicative factor $1 - \epsilon$ apart.

3.4 Good and Bad Transcripts

Given a valid transcript τ we define the sets $\mathcal{L}_{\mathrm{in}}, \mathcal{L}_{\mathrm{out}} \subseteq \{0,1\}^c \times \{0,1\}^b$ as

$$\mathcal{L}_{\mathrm{in}} := \{(\lambda(\pi(v)), \mu(v) \oplus K_{\mathrm{w}}) \,:\, v \in V \setminus \{\varepsilon\}\}$$
$$\mathcal{L}_{\mathrm{out}} := \left\{\left(K_2, \lambda(v) \,\|\, 0^{b-c}\right) \,:\, v \in \mathcal{Q}_C\right\},$$

and let $\mathcal{L} = \mathcal{L}_{\mathrm{in}} \cup \mathcal{L}_{\mathrm{out}}$. Intuitively, $\mathcal{L}$ represents the set of inputs on which f is evaluated while processing A's construction queries in the real experiment. This set is also well-defined in the ideal experiment by the above equations, and in both experiments it is determined by the transcript. We refer to $\mathcal{L}_{\mathrm{in}}$ as the set of *inner* f*-invocations*, i.e., those invocations of f that were required to evaluate the inner, whitened cascade $\mathsf{WCasc}^{\mathsf{f}}$ in WNMAC; and similarly, $\mathcal{L}_{\mathrm{out}}$ denotes the *outer invocations*.

If there is an intersection between the adversary's f-queries and the inputs in $\mathcal{L}_{\mathrm{in}}$ (resp. $\mathcal{L}_{\mathrm{out}}$), we call this an *inner (resp., outer) C-f-collision*. We then denote by $\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}$ (resp., $\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}$) the event that any inner (resp., outer) C-f-collision occurs. Formally,

$$\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}} :\Leftrightarrow (\mathcal{Q}_{\mathsf{f}} \cap \mathcal{L}_{\mathrm{in}} \neq \emptyset) \quad \text{and} \quad \mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}} :\Leftrightarrow (\mathcal{Q}_{\mathsf{f}} \cap \mathcal{L}_{\mathrm{out}} \neq \emptyset)$$

and let $\mathsf{C\text{-}f\text{-}coll} := \mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}} \cup \mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}$. Furthermore, if the vertex labels $\lambda(M)$ collide for two messages $M, M' \in \mathcal{Q}_C$, we call this a C-collision and denote such an event by

$$\mathsf{C\text{-}coll} :\Leftrightarrow (\exists M, M' \in \mathcal{Q}_C \colon \lambda(M) = \lambda(M')).$$

Definition 1 (Good Transcripts). *Let*

$$\tau = \left(\mathcal{QR}_C, \mathcal{QR}_{\mathsf{f}}, K = (K_1, K_2, K_{\mathrm{w}}), T_K^{\mathsf{f}}(\mathcal{Q}_C) = (V, E, \lambda)\right)$$

be a valid transcript. We say that the transcript is good *(and thus* $\tau \in \mathsf{GT}$*) if the following properties are true:*

(1) *The event* $\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}$ *has not occurred.*
(2) *The event* $\mathsf{C\text{-}coll}$ *has not occurred.*
(3) *For any* $v \in V$ *we have* $\lambda(v) \neq K_2$*.*

We denote as GT the set of all good transcripts, and BT the set of all *bad* transcripts, i.e., transcripts which can possibly occur (i.e., they are in the range of $\mathsf{T}_{\mathsf{ideal}}$) and are not good. More specifically, we denote by BT_i the set of all bad transcripts that do not satisfy the i-th property in the definition of a good transcript above, hence we have $\mathsf{BT} = \bigcup_{i=1}^{3} \mathsf{BT}_i$.

3.5 Probability of a C-f-collision

In this section we upper-bound the probability of $\mathsf{C\text{-}f\text{-}coll}$ by considering inner and outer C-f-collisions separately.

Lemma 2. *We have* $\mathsf{P}^{\mathsf{ideal}}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}] \leq \ell q_C q_{\mathsf{f}} / 2^{b+c}$*.*

Proof. We start by modifying the ideal experiment to obtain an experiment denoted ideal' and the corresponding transcript distribution $\mathsf{T}_{\mathsf{ideal}'}$. The experiment ideal' is given in Fig. 3. Clearly, ideal' differs from the ideal experiment only in the way the vertex labeling function $\lambda(\cdot)$ is determined.

We now argue that $\mathsf{P}^{\mathsf{ideal}}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}] = \mathsf{P}^{\mathsf{ideal}'}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}]$. To see this, consider an intermediate experiment ideal'' that is defined exactly as ideal except that it uses a separate ideal compression function g to generate the vertex labels of the tree contained in the transcript, where g is completely independent of f queried by the adversary (i.e., the adversary queries f and the transcript contains $\mathcal{QR}_{\mathsf{f}}$ and $T_K^{\mathsf{g}}(\mathcal{Q}_C)$). It is now clear that $\mathsf{P}^{\mathsf{ideal}}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}] = \mathsf{P}^{\mathsf{ideal}''}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}]$ since as long as no inner C-f-collision happens, the experiments are identical.

The remaining equality $\mathsf{P}^{\mathsf{ideal}''}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}] = \mathsf{P}^{\mathsf{ideal}'}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}]$ follows from the definition of ideal'. It is easy to see that the distribution of vertex labels sampled in steps 2 and 3 of ideal' and by labeling the tree $T_K^{\mathsf{g}}(\mathcal{Q}_C)$ in ideal'' are the same. In both cases, repeated inputs to the compression function lead to consistent outputs, while fresh inputs lead to independent random outputs. The two experiments only differ in the order of sampling: ideal'' first samples g and then performs the labeling, while ideal' starts by sampling the repetition pattern, and then chooses the actual labels correspondingly. The same distribution of vertex labels in these two experiments then implies the same probability of $\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}$ occurring.

1. **The adversary asks its C-queries and f-queries and these are answered by independent random values.** Once the q_C queries in $\mathcal{Q}_C$ are fixed, they also determine the message tree $T(\mathcal{Q}_C)$ and mappings μ and π as defined in Section 3.1 (the labeling λ is so far undefined).
2. **Sample a repetition pattern ρ.** The equivalence relation ρ is determined indirectly by first iteratively defining a mapping $\hat{\rho}\colon V \to [|V|]$. Recall the vertex ordering $v_1, \ldots, v_{|V|}$ defined in Section 3.1. First, set $\hat{\rho}(v_1) := 1$. Then, for i taking values $2, \ldots, |V|$, determine $\hat{\rho}(v_i)$ as follows. If there exists $j \in [i-1]$ such that $\mu(v_j) = \mu(v_i)$ and $\hat{\rho}(\pi(v_j)) = \hat{\rho}(\pi(v_i))$ then let $\hat{\rho}(v_i) := \hat{\rho}(v_j)$ for the minimal such j. Otherwise let $z := \max_{j\in[i-1]}\{\hat{\rho}(v_j)\}$ and sample $\hat{\rho}(v_i)$ as

$$\hat{\rho}(v_i) := \begin{cases} 1 & \text{with probability } 2^{-c} \\ \vdots & \vdots \\ z & \text{with probability } 2^{-c} \\ z+1 & \text{with probability } 1 - z \cdot 2^{-c}. \end{cases}$$

   Finally, for all $i, j \in [|V|]$ let $\rho(v_i, v_j) :\Leftrightarrow (\hat{\rho}(v_i) = \hat{\rho}(v_j))$.
3. **Sample a vertex labeling $\lambda(\cdot)$ according to ρ.** Namely, sample $|\rho|$ distinct uniformly random values $s_1, \ldots, s_{|\rho|} \in \{0,1\}^c$ where $|\rho|$ is the number of equivalence classes of ρ, and let $\lambda(v_i) := s_{\hat{\rho}(v_i)}$ for all $i \in [|V|]$. Also let $K_1 := \lambda(\varepsilon)$.
4. **Sample random keys $(K_2, K_w) \in \{0,1\}^c \times \{0,1\}^b$.**

Fig. 3. The random experiment ideal' for the proofs of Lemmas 2 and 3.

Finally, we upper-bound the probability $\mathsf{P}^{\mathsf{ideal}'}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}]$. Conditioned on the repetition pattern ρ taking some fixed value rp, in step 2, we have

$$\begin{aligned} \mathsf{P}^{\mathsf{ideal}'}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}} \mid \rho = rp] &\le \sum_{v\in V\setminus\{\varepsilon\}} \mathsf{P}^{\mathsf{ideal}'}\left[(\lambda(\pi(v)), \mu(v) \oplus K_w) \in \mathcal{Q}_f \mid \rho = rp\right] \\ &= \sum_{v\in V\setminus\{\varepsilon\}} \mathsf{P}^{\mathsf{ideal}'}\left[(s_{\hat{\rho}(\pi(v))}, \mu(v) \oplus K_w) \in \mathcal{Q}_f \mid \rho = rp\right] \\ &= \sum_{v\in V\setminus\{\varepsilon\}} q_f/2^{b+c} \le \ell q_C q_f/2^{b+c} \end{aligned}$$

because the random variables s_i and K_w sampled in steps 3 and 4 are uniformly distributed and independent of $\mathcal{Q}_f$. Since this bound holds conditioned on ρ being any fixed repetition pattern rp, it remains valid also without conditioning on it, hence concluding the proof. □

We proceed by upper-bounding the probability of an outer C-f-collision.

Lemma 3. *We have*

$$\mathsf{P}^{\mathsf{ideal}}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}] \le \frac{\ell q_C q_f}{2^{b+c}} + \frac{q_C q_f}{2^{2c}}.$$

Proof. Let us again consider the experiments ideal' and ideal'' defined in the proof of Lemma 2. We start by the simple observation that for any event A we have

$$\begin{aligned}\mathsf{P}^{\mathsf{ideal}}[A] &= \mathsf{P}^{\mathsf{ideal}}[A \wedge \mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}] + \mathsf{P}^{\mathsf{ideal}}[A \wedge \neg\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}] \\ &\leq \frac{\ell q_C q_{\mathsf{f}}}{2^{b+c}} + \mathsf{P}^{\mathsf{ideal}''}[A \wedge \neg\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}] \leq \frac{\ell q_C q_{\mathsf{f}}}{2^{b+c}} + \mathsf{P}^{\mathsf{ideal}''}[A]\,, \end{aligned} \tag{4}$$

which follows from Lemma 2 and the observation that ideal and ideal'' only differ if $\mathsf{C\text{-}f\text{-}coll}_{\mathsf{in}}$ occurs.

Applying (4) to the event $\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}$ as A, it remains to bound the probability $\mathsf{P}^{\mathsf{ideal}''}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}]$; for this we observe that $\mathsf{P}^{\mathsf{ideal}''}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}] = \mathsf{P}^{\mathsf{ideal}'}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}]$ similarly as before: the repetition pattern ρ sampled in step 2 of ideal' has the same distribution as the repetition pattern induced by the tree $T_K^{\mathsf{g}}(\mathcal{Q}_C)$ in ideal'', and this together with the sampling performed in step 3 results in the same distribution of vertex labels in ideal'' and ideal' and hence also in the same probability of $\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}$ in both experiments.

Finally, to upper-bound the probability $\mathsf{P}^{\mathsf{ideal}'}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}}]$, again conditioned on the repetition pattern ρ sampled in step 2 taking some fixed value rp, we have

$$\begin{aligned}\mathsf{P}^{\mathsf{ideal}'}[\mathsf{C\text{-}f\text{-}coll}_{\mathsf{out}} \mid \rho = rp] &\leq \sum_{v \in \mathcal{Q}_C} \mathsf{P}^{\mathsf{ideal}'}\left[(K_2, \lambda(v) \,\|\, 0^{b-c}) \in \mathcal{Q}_{\mathsf{f}} \mid \rho = rp\right] \\ &\leq \sum_{v \in \mathcal{Q}_C} \mathsf{P}^{\mathsf{ideal}'}\left[(K_2, s_{\hat{\rho}(v)} \,\|\, 0^{b-c}) \in \mathcal{Q}_{\mathsf{f}} \mid \rho = rp\right] \\ &= \sum_{v \in \mathcal{Q}_C} q_{\mathsf{f}}/2^{2c} \leq q_C q_{\mathsf{f}}/2^{2c}\end{aligned}$$

because the random variables s_i and K_2 sampled in steps 3 and 4 are uniformly distributed and independent of $\mathcal{Q}_{\mathsf{f}}$. Since this bound holds conditioned on ρ being any fixed repetition pattern rp, it remains valid also without conditioning on it. □

3.6 Probability of Repeated Outer Invocations

In this section we analyze the probability that any of the outer f-invocations in the ideal experiment will not be fresh, in particular we upper-bound both $\mathsf{P}[\mathsf{T}_{\mathsf{ideal}} \in \mathsf{BT}_2]$ and $\mathsf{P}[\mathsf{T}_{\mathsf{ideal}} \in \mathsf{BT}_3]$.

Lemma 4. *We have*

$$\mathsf{P}^{\mathsf{ideal}}[\mathsf{C\text{-}coll}] \leq \frac{\ell q_C q_{\mathsf{f}}}{2^{b+c}} + \frac{\ell {q_C}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c}\right).$$

Proof. Applying (4) to the event C-coll, we have $\mathsf{P}^{\mathsf{ideal}}[\mathsf{C\text{-}coll}] \le \ell q_{\mathrm{C}} q_{\mathrm{f}}/2^{b+c} + \mathsf{P}^{\mathsf{ideal}''}[\mathsf{C\text{-}coll}]$. Since the queries $\mathcal{Q}_C$ in the experiment ideal'' are chosen non-adaptively (with respect to the keys K_1, K_{w} and the function g used to later compute the tree labeling), we can obtain via a union bound that

$$\mathsf{P}^{\mathsf{ideal}''}[\mathsf{C\text{-}coll}] \le {q_{\mathrm{C}}}^2 \cdot \max_{\substack{M_1 \neq M_2 \\ |M_1|,|M_2| \le \ell b}} \mathsf{P}^{\mathsf{g},K_1,K_{\mathrm{w}}}\left[\mathsf{WCasc}^{\mathsf{g}}_{K_1,K_{\mathrm{w}}}(M_1) = \mathsf{WCasc}^{\mathsf{g}}_{K_1,K_{\mathrm{w}}}(M_2)\right].$$

Moreover, we have

$$\begin{aligned}
&\max_{\substack{M_1 \neq M_2 \\ |M_1|,|M_2| \le \ell b}} \mathsf{P}^{\mathsf{g},K_1,K_{\mathrm{w}}}\left[\mathsf{WCasc}^{\mathsf{g}}_{K_1,K_{\mathrm{w}}}(M_1) = \mathsf{WCasc}^{\mathsf{g}}_{K_1,K_{\mathrm{w}}}(M_2)\right] \\
&= \max_{\substack{M_1 \neq M_2 \\ |M_1|,|M_2| \le \ell b}} \sum_{\substack{K_1 \in \{0,1\}^c \\ K_{\mathrm{w}} \in \{0,1\}^b}} \frac{1}{2^{c+b}} \cdot \mathsf{P}^{\mathsf{g}}\left[\mathsf{WCasc}^{\mathsf{g}}_{K_1,K_{\mathrm{w}}}(M_1) = \mathsf{WCasc}^{\mathsf{g}}_{K_1,K_{\mathrm{w}}}(M_2)\right] \\
&\le \sum_{\substack{K_1 \in \{0,1\}^c \\ K_{\mathrm{w}} \in \{0,1\}^b}} \frac{1}{2^{c+b}} \cdot \max_{\substack{M_1 \neq M_2 \\ |M_1|,|M_2| \le \ell b}} \mathsf{P}^{\mathsf{g}}\left[\mathsf{WCasc}^{\mathsf{g}}_{K_1,K_{\mathrm{w}}}(M_1) = \mathsf{WCasc}^{\mathsf{g}}_{K_1,K_{\mathrm{w}}}(M_2)\right] \\
&= \sum_{\substack{K_1 \in \{0,1\}^c \\ K_{\mathrm{w}} \in \{0,1\}^b}} \frac{1}{2^{c+b}} \cdot \max_{\substack{M_1 \neq M_2 \\ |M_1|,|M_2| \le \ell b}} \mathsf{P}^{\mathsf{g}}\left[\mathsf{Casc}^{\mathsf{g}}_{K_1}(M_1 \oplus K_{\mathrm{w}}) = \mathsf{Casc}^{\mathsf{g}}_{K_1}(M_2 \oplus K_{\mathrm{w}})\right] \\
&= \sum_{\substack{K_1 \in \{0,1\}^c \\ K_{\mathrm{w}} \in \{0,1\}^b}} \frac{1}{2^{c+b}} \cdot \underbrace{\max_{\substack{M_1 \neq M_2 \\ |M_1|,|M_2| \le \ell b}} \mathsf{P}^{\mathsf{g}}\left[\mathsf{Casc}^{\mathsf{g}}_{K_1}(M_1) = \mathsf{Casc}^{\mathsf{g}}_{K_1}(M_2)\right]}_{\mathsf{CascColl}(\ell)},
\end{aligned}$$

where the notation $M_i \oplus K_{\mathrm{w}}$ denotes XOR-ing the key K_{w} to each of the blocks of M_i.

The last maximization term above was already studied in the context of the construction NI2 in [7], where it was denoted as $\mathsf{CColl}(\ell)$, but we will refer to it as $\mathsf{CascColl}(\ell)$ to avoid confusion with the event C-coll considered here. It was shown in [7] that

$$\mathsf{CascColl}(\ell) \le \frac{\ell \cdot d'(\ell)}{2^c} + \frac{64\ell^4}{2^{2c}}. \tag{5}$$

Putting all the above bounds together concludes the proof of Lemma 4. □

Lemma 5. *We have*

$$\mathsf{P}^{\mathsf{ideal}}[\exists v \in V \colon \lambda(v) = K_2] \le \frac{\ell q_{\mathrm{C}}}{2^c}.$$

Proof. As is clear from the description of the ideal experiment, the key K_2 is chosen uniformly at random and independently of the rest of the experiment, in particular of the labels $\lambda(v)$. The lemma hence follows by a simple union bound over all ℓq_{C} vertices $v \in V$. □

3.7 Good Transcripts and Putting Pieces Together

Let us consider a good transcript τ. First, since $\tau \notin \mathsf{BT}_1$, there is no overlap between the outer f-invocations and the f-queries issued by the adversary. Second, since $\tau \notin \mathsf{BT}_2$, there is also no repetition between the outer f-invocations themselves. Finally, since $\tau \notin \mathsf{BT}_3$, there is also no overlap between the outer f-invocations and the inner f-invocations (all the outer invocations contain K_2 as their first component). Altogether, this means that each outer f-invocation in real is fresh and hence its outcome can be seen as freshly uniformly sampled (since f is an ideal random function). Therefore, the distribution of these outcomes will be the same as in ideal, where they correspond to the independent random values Y_i. Hence, for all $\tau \in \mathsf{GT}$, we have

$$\frac{\mathsf{P}\left[\mathsf{T}_{\mathsf{real}} = \tau\right]}{\mathsf{P}\left[\mathsf{T}_{\mathsf{ideal}} = \tau\right]} = 1.$$

Plugging this into Lemma 1, together with the bounds from Lemmas 3, 4 and 5, we obtain

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{WNMAC}}(\mathsf{A}) &\leq \sum_{i=1}^{3} \mathsf{P}\left[\mathsf{T}_{\mathsf{ideal}} \in \mathsf{BT}_i\right] \\
&\leq \frac{q_{\mathsf{f}} q_{\mathsf{C}}}{2^{2c}} + 2 \cdot \frac{\ell {q_{\mathsf{C}}} q_{\mathsf{f}}}{2^{b+c}} + \frac{\ell {q_{\mathsf{C}}}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c}\right) + \frac{\ell q_{\mathsf{C}}}{2^c} \\
&\leq \frac{q_{\mathsf{f}} q_{\mathsf{C}}}{2^{2c}} + 2 \cdot \frac{\ell {q_{\mathsf{C}}} q_{\mathsf{f}}}{2^{b+c}} + \frac{\ell {q_{\mathsf{C}}}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c} + 1\right),
\end{aligned}$$

which concludes the proof of Theorem 1. □

3.8 Tightness

We now argue that the $q_{\mathsf{C}} q_{\mathsf{f}}/2^{2c}$ term in our bound on the security of WNMAC as given in (2) is tight, by giving a matching attack (up to a linear factor $O(c)$). For most practical parameters, this will be the dominating term in (2), and thus for those parameters Theorem 1 gives a tight bound. Here we only describe an attack for the case where $q_{\mathsf{C}} = \Theta(c)$ is very small, and defer the general case to the full version.

The $q_{\mathsf{C}} = \Theta(c)$ Case. We must define an adversary $\mathsf{A}^{\mathcal{O},\mathsf{f}}$ who can distinguish the case where the first oracle $\mathcal{O}$ implements a random function R from the case where it implements $\mathsf{WNMAC}^{\mathsf{f}}((K_1, K_2, K_{\mathrm{w}}), \cdot)$ with random keys K_1, K_2, K_{w} using the random function $\mathsf{f} : \{0,1\}^{b+c} \to \{0,1\}^c$ which is given as the second oracle.

$\mathsf{A}^{\mathcal{O},\mathsf{f}}$ first picks $t := q_{\mathsf{f}}/2^c$ keys $\widetilde{K}_1, \ldots, \widetilde{K}_t$ arbitrarily, and then uses its q_{f} function queries to learn the outputs

$$\mathcal{Z}_i = \{\mathsf{f}(\widetilde{K}_i, x \| 0^{b-c}) \ : \ x \in \{0,1\}^c\}$$

for all the keys. When throwing 2^c balls randomly into 2^c bins, we expect a $1 - 1/e \approx 0.63$ fraction of the bins to be non-empty (and the value is strongly concentrated around this expectation). We can think of evaluating the random function $\mathsf{f}(\widetilde{K}_i, \cdot \| 0^{b-c}) : \{0,1\}^c \to \{0,1\}^c$ as throwing 2^c balls (the inputs) to random bins (the outputs), and thus have $|\mathcal{Z}_i| \approx 0.63 \cdot 2^c$. Then $\mathsf{A}^{\mathcal{O},\mathsf{f}}$ queries $\mathcal{O}$ on $\Theta(c)$ random inputs, let $\mathcal{Q}_c$ denote the corresponding outputs. Now $\mathsf{A}^{\mathcal{O},\mathsf{f}}$ outputs 1 if and only if for some i we have $\mathcal{Q}_c \subset \mathcal{Z}_i$. If $\mathcal{O}(\cdot) = \mathsf{WNMAC}^{\mathsf{f}}((K_1, K_2, K_{\mathrm{w}}), \cdot) = \mathsf{f}(K_2, \mathsf{WCasc}^{\mathsf{f}}((K_1, K_{\mathrm{w}}), \cdot) \| 0^{b-c})$ and moreover $K_2 = \widetilde{K}_i$ for some i – which happens with probability $t/2^c$ – then all the outputs of $\mathcal{O}(\cdot)$ are in the range of $\mathsf{f}(\widetilde{K}_i, . \| 0^{b-c})$ and thus $\mathsf{A}^{\mathcal{O},\mathsf{f}}$ outputs 1.

On the other hand, if $\mathcal{O}(\cdot)$ is a random function, then every single query will miss the set $\mathcal{Z}_i$ with constant probability 0.37. Using this, we get by a Chernoff bound (and the union bound over all t keys) that

$$\mathsf{P}[\exists i \ : \ \mathcal{Q}_c \subset \mathcal{Z}_i] \leq \frac{t}{2^{\Theta(q_{\mathrm{C}})}} \ .$$

Summing up we get for $q_{\mathrm{C}} = \Theta(c)$ and $t = q_{\mathsf{f}}/2^c$

$$\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{WNMAC}}(\mathsf{A}_{q_{\mathrm{C}},t}) \geq \left| \frac{t}{2^c} - \frac{t}{2^{\Theta(q_{\mathrm{C}})}} \right| \geq \frac{t}{2^{c-1}} \geq \frac{q_{\mathsf{f}}}{2^{2c-1}} = \frac{q_{\mathsf{f}} q_{\mathrm{C}}}{2^{2c} \cdot \Theta(c)}$$

which matches our term $q_{\mathsf{f}} q_{\mathrm{C}}/2^{2c}$ from the lower bound up to a $\Theta(c)$ factor.

3.9 Distinguishing-H Security of WNMAC

The above results also imply a bound on the distinguishing-H security of WNMAC. To capture this, we first introduce the notion of distinguishing-C, which corresponds to PRF-security with the restriction that the distinguisher only uses construction queries.

Definition 2 (Distinguishing-C). *Let* $\mathsf{C}[\mathsf{f}] \colon \{0,1\}^\kappa \times \mathcal{D} \to \mathcal{R}$ *be a keyed construction making queries to a randomly chosen compression function* $\mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b, c)$. *The* distinguishing-C *advantage of an adversary* A *is defined as*

$$\mathsf{Adv}^{\mathsf{dist\text{-}C}}_{\mathsf{C}[\mathsf{f}]}(\mathsf{A}) := \Big| \mathsf{P}\left[K \xleftarrow{\$} \{0,1\}^\kappa, \mathsf{f} \xleftarrow{\$} \mathcal{F}(c+b,c) : \ \mathsf{A}^{\mathsf{C}^{\mathsf{f}}_K} \Rightarrow 1\right] - \mathsf{P}\left[\mathsf{R} \xleftarrow{\$} \mathcal{F}(\mathcal{D},\mathcal{R}) : \ \mathsf{A}^{\mathsf{R}} \Rightarrow 1\right] \Big| .$$

The notion of distinguishing-C is useful for bridging distinguishing-H and PRF-security, as the following lemma shows (we omit its simple proof).

Lemma 6. *For every adversary* A *asking* q_{C} *and* q_{f} *construction and primitive queries, respectively, there exists an adversary* A' *asking* q_{C} *queries to its single oracle such that*

$$\mathsf{Adv}^{\mathsf{dist\text{-}H}}_{\mathsf{C}}(\mathsf{A}) \leq \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{C}[\mathsf{f}]}(\mathsf{A}) + \mathsf{Adv}^{\mathsf{dist\text{-}C}}_{\mathsf{C}[\mathsf{f}]}(\mathsf{A}')$$

and

$$\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{C[f]}}(\mathsf{A}) \le \mathsf{Adv}^{\mathsf{dist\text{-}H}}_{\mathsf{C}}(\mathsf{A}) + \mathsf{Adv}^{\mathsf{dist\text{-}C}}_{\mathsf{C[f]}}(\mathsf{A}').$$

One can readily obtain a bound on the distinguishing-C security of WNMAC using Theorem 1 with $q_{\mathsf{f}} = 0$.

Lemma 7 (Distinguishing-C Security of WNMAC). *Let* A *be an adversary making at most q_{C} construction queries, each of length at most ℓ b-bit blocks. Let $K = (K_1, K_2, K_{\mathrm{w}}) \in \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$ be a tuple of random keys. Then we have*

$$\mathsf{Adv}^{\mathsf{dist\text{-}C}}_{\mathsf{WNMAC}_K}(\mathsf{A}) \le \frac{\ell {q_{\mathrm{C}}}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c} + 1 \right).$$

By combining Theorem 1 and Lemmas 6 and 7, we get the following theorem.

Theorem 2 (Distinguishing-H Security of WNMAC). *Let* A *be an adversary making at most q_{f} queries to the compression function and at most q_{C} construction queries, each of length at most ℓ b-bit blocks. Let $K = (K_1, K_2, K_{\mathrm{w}}) \in \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$ be a tuple of random keys. Then we have*

$$\mathsf{Adv}^{\mathsf{dist\text{-}H}}_{\mathsf{WNMAC}_K}(\mathsf{A}) \le \frac{q_{\mathsf{f}} q_{\mathrm{C}}}{2^{2c}} + 2 \cdot \frac{\ell q_{\mathrm{C}} q_{\mathsf{f}}}{2^{b+c}} + 2 \cdot \frac{\ell {q_{\mathrm{C}}}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c} + 1 \right).$$

3.10 State Recovery for WNMAC

We now formally define the notion of security against state recovery for WNMAC. We consider the strong notion where the goal of the adversary is to output a pair (M, s) such that the state s occurs *at any point* during the evaluation of WCasc on M. Formally, we define $\mathsf{Adv}^{\mathsf{sr}}_{\mathsf{WNMAC[f]}}(\mathsf{A})$ to be

$$\mathsf{P}\Big[K \xleftarrow{\$} \mathcal{K}, \mathsf{f} \xleftarrow{\$} \mathcal{F}, \mathsf{A}^{\mathsf{WNMAC}^{\mathsf{f}}_K, \mathsf{f}} \Rightarrow (M, s) : \exists M' \in \{0,1\}^{b*} \text{ s.t. } M' \,|\, M \wedge \mathsf{WCasc}^{\mathsf{f}}_{K_1, K_{\mathrm{w}}}(M') = s\Big]$$

where $\mathcal{K} = \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$, $K = (K_1, K_2, K_{\mathrm{w}})$ and $\mathcal{F} := \mathcal{F}(c+b, c)$.

Theorem 3 (State-Recovery Security of WNMAC). *Let* A *be an adversary making at most q_{f} queries to the compression function and at most q_{C} construction queries, each of length at most ℓ b-bit blocks. Let $K = (K_1, K_2, K_{\mathrm{w}}) \in \{0,1\}^c \times \{0,1\}^c \times \{0,1\}^b$ be a tuple of random keys. Then we have*

$$\mathsf{Adv}^{\mathsf{sr}}_{\mathsf{WNMAC}^{\mathsf{f}}_K}(\mathsf{A}) \le \frac{q_{\mathsf{f}} q_{\mathrm{C}}}{2^{2c}} + 2 \cdot \frac{\ell q_{\mathrm{C}} q_{\mathsf{f}}}{2^{b+c}} + 2 \cdot \frac{\ell {q_{\mathrm{C}}}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c} + 2 \right).$$

Proof (sketch). First, we replace the compression function oracle f by an independent random function g completely unrelated to $\mathsf{WNMAC}^{\mathsf{f}}$. The error introduced by this is upper-bounded by Theorem 2 and now, compression-function queries are useless to the adversary, hence we can disregard them.

Let us denote by $\mathcal{E}$ the experiment where A interacts with $\mathsf{WNMAC}^{\mathsf{f}}$ (without direct access to f). Consider an alternative experiment $\mathcal{E}'$ given in Fig. 4. As long as the key K_2 chosen in step 4 does not hit any of the internal states that occurred during the query evaluation, the experiment $\mathcal{E}'$ is identical to $\mathcal{E}$. Moreover, since K_2 is chosen independently at random, such a hit can only occur with probability at most $\ell q_{\mathrm{C}}/2^c$. Since the vertex labels are only sampled after the adversary makes its guess for the state, the probability that the guess will be correct is at most $\ell/2^c$. □

1. **The adversary asks its C-queries.** For each of them, only the repetition pattern for the state values belonging to this query is sampled (as in the experiment ideal' in Figure 3) and the query is answered with a fresh random value, unless the outer f-invocation happens on a repeated value, in which case the query is answered consistently. After answering all queries, we have a complete repetition pattern ρ for all state values.
2. **Let A output its guess** (M, s)**.**
3. **Sample a vertex labeling** $\lambda(\cdot)$ **according to** ρ**, let** $K_1 := \lambda(\varepsilon)$**.**
4. **Sample random keys** $(K_2, K_{\mathrm{w}}) \in \{0,1\}^c \times \{0,1\}^b$**.**

Fig. 4. The random experiment $\mathcal{E}'$ for the proof of Theorem 3.

4 Whitening HMAC

HMAC is a "practice-oriented" variant of NMAC, see Sect. 2 for its definition. In this section we consider a "whitened" variant WHMAC of HMAC which is derived from HMAC in the same way as WNMAC was derived from NMAC, i.e., by XORing a random key K_{w} to every message block. We also consider a variant WHMAC^{+} where the first message block is a fresh key $K^{+} \in \{0,1\}^b$. More precisely,

$$\mathsf{WHMAC}_{K,K_{\mathrm{w}}}[\mathsf{f}](m) := \mathsf{f}\left(K_2', \mathsf{WCasc}^{\mathsf{f}}_{K_1',K_{\mathrm{w}}}(m) \| \mathsf{fpad}\right)$$

where

$$K_1' := \mathsf{f}(\mathsf{IV}, K \oplus \mathsf{ipad}) \quad \text{and} \quad K_2' := \mathsf{f}(\mathsf{IV}, K \oplus \mathsf{opad}) \tag{6}$$

and fpad is some fixed padding; and

$$\mathsf{WHMAC}^{+}_{K,K_{\mathrm{w}},K^{+}}[\mathsf{f}](m) := \mathsf{f}\left(K_2', \mathsf{WCasc}^{\mathsf{f}}_{K_1',K_{\mathrm{w}}}(m) \| \mathsf{fpad}\right),$$

where this time

$$Z := \mathsf{f}(\mathsf{IV}, K \oplus \mathsf{ipad}) \quad \text{and} \quad K_1' := \mathsf{f}(Z, K^+) \quad \text{and} \quad K_2' := \mathsf{f}(\mathsf{IV}, K \oplus \mathsf{opad})$$

and fpad is again some padding. Note that both variants, WHMAC and WHMAC$^+$, can be implemented given just black-box access to an implementation of HMAC.

The theorem below relates the security of WHMAC and WHMAC$^+$ to the security of WNMAC.

Theorem 4 (Relating Security of WHMAC to WNMAC). *Consider any* $xxx \in \{prf, dist\text{-}H, sr\}$. *Assume that for every adversary* A *making at most* q_{f} *queries to the compression function* f *and at most* q_{C} *construction queries, each of length at most* ℓ *b-bit blocks, we have*

$$\mathsf{Adv}^{xxx}_{\mathsf{WNMAC}_{K_1,K_2,K_{\mathrm{w}}}[\mathsf{f}]}(\mathsf{A}) \leq \epsilon,$$

where here and below, $K_1, K_2 \in \{0,1\}^c$ *and* $K, K_{\mathrm{w}}, K^+ \in \{0,1\}^b$ *are uniformly random keys. Then for every such adversary* A *we have*

$$\mathsf{Adv}^{xxx}_{\mathsf{WHMAC}_{K,K_{\mathrm{w}}}[\mathsf{f}]}(\mathsf{A}) \leq \epsilon + 2^{-\frac{b-2c}{2}} \tag{7}$$

and

$$\mathsf{Adv}^{xxx}_{\mathsf{WHMAC}^+_{K,K_{\mathrm{w}},K^+}[\mathsf{f}]}(\mathsf{A}) \leq \epsilon + 2 \cdot 2^{-\frac{b-c}{2}} + 2^{-c}. \tag{8}$$

Proof. Intuitively, for WHMAC one can think of f as an extractor which extracts keys K_1', K_2' from K, and the bound then readily follows by the leftover hash lemma. For WNMAC$^+$ one can roughly think of K_1' and K_2' as being extracted from independent keys K^+ and K, respectively. For the latter it is thus sufficient that b (which is the length, and thus also the entropy of the uniform K and K^+) is sufficiently larger than c (the length of K_1', K_2'), whereas for the former we need b to be sufficiently larger than $2c$. We now give the details of the proof for WHMAC and postpone the treatment of WNMAC$^+$ to the full version.

In order to prove the bound (7) it is sufficient to show that the statistical distance between the transcripts (as seen by the adversary) when interacting with WNMAC or WHMAC is at most $2^{-\frac{b-2c}{2}}$. As the only difference between WNMAC and WHMAC is that we replace the uniform keys K_1, K_2 with keys K_1', K_2' derived according to (6), to bound the distance between the transcripts, it is sufficient to bound the distance between the random and derived keys. As K_1', K_2' are not independent of f, it is important to bound the distance when given f, concretely, we must show that

$$\mathsf{SD}\left((K_1', K_2', \mathsf{f}), (K_1, K_2, \mathsf{f})\right) \leq 2^{-\frac{b-2c}{2}}.$$

We will use the leftover hash lemma [12] which states that for any random variable $X \in \{0,1\}^m$ with min-entropy at least $H_\infty(X) \geq k$ and a hash function

$h : \{0,1\}^m \to \{0,1\}^\ell$ chosen from a family of pairwise independent hash functions we have (with U_ℓ being uniform over $\{0,1\}^\ell$)

$$\mathsf{SD}\left((h(X), h), (U_\ell, h)\right) \leq 2^{\frac{\ell - H_\infty(X)}{2}} \leq 2^{\frac{\ell-k}{2}}.$$

Since $\mathsf{f} : \{0,1\}^{b+c} \to \{0,1\}^c$ is uniformly random, also the function

$$\mathsf{f}'(K) = (\mathsf{f}(\mathsf{IV}, K \oplus \mathsf{ipad}), \mathsf{f}(\mathsf{IV}, K \oplus \mathsf{opad}))$$

is uniformly random, and thus also pairwise independent. Using $H_\infty(K) = H_\infty(K \oplus \mathsf{ipad}) = b$ and $(K_1', K_2') = \mathsf{f}'(K)$ we thus get

$$\mathsf{SD}\left((K_1', K_2', \mathsf{f}'), (K_1, K_2, \mathsf{f}')\right) = \mathsf{SD}\left((K_1', K_2', \mathsf{f}), (K_1, K_2, \mathsf{f})\right) \leq 2^{-\frac{b-2c}{2}}$$

as required. The first equality above holds as f defines all of f' and vice versa. □

5 The Dual WNMAC Construction

Looking at the security bounds for WNMAC given in Sect. 3 from a distance, it seems that under reasonable assumptions the most restrictive term in the bounds is $q_\mathsf{f} q_\mathrm{C}/2^{2c}$. Intuitively speaking, the reason for this term is the outer f-call in WNMAC that only takes $2c$ bits of actual inputs and adds $b - c$ padding zeroes.

In an attempt to overcome this limitation, we propose a variant of the WNMAC construction that we call *Dual WNMAC* (DWNMAC). We prove the PRF-security of DWNMAC that goes beyond the restrictive term $q_\mathsf{f} q_\mathrm{C}/2^{2c}$ and our proof again extends also to distinguishing-H and state-recovery security. The price we pay for this improvement is a slight increase in the key length and the fact that DWNMAC cannot be implemented using only black-box access to NMAC. Similarly, if we apply the same modification to WHMAC, the resulting construction can no longer be implemented using black-box access to HMAC.

The construction DWNMAC is derived from WNMAC, the only difference being that the outer f-call is performed on the c-bit state and a b-bit key K_2. More precisely, for a key tuple $(K_1, K_2, K_\mathrm{w}) \in \{0,1\}^c \times \{0,1\}^b \times \{0,1\}^b$ and a message $M \in \{0,1\}^{b*}$, we define

$$\mathsf{DWNMAC}^\mathsf{f}((K_1, K_2, K_\mathrm{w}), M) := \mathsf{f}(\mathsf{WCasc}^\mathsf{f}_{K_1, K_\mathrm{w}}(M), K_2).$$

Note that DWNMAC is slightly similar to what we would obtain by whitening from the Sandwich MAC construction [23].

We now summarize the security of DWNMAC.

Theorem 5. (Security of DWNMAC). *Let* A *be an adversary making at most* q_f *queries to the compression function* f *and at most* q_C *construction queries, each of length at most* ℓ *b-bit blocks. Let* $K = (K_1, K_2, K_\mathrm{w}) \in \{0,1\}^c \times \{0,1\}^b \times \{0,1\}^b$ *be a tuple of random keys. Then we have*

$$\mathsf{Adv}^{\mathsf{xxx}}_{\mathsf{DWNMAC}^\mathsf{f}_K}(\mathsf{A}) \leq 3 \cdot \frac{\ell q_\mathrm{C} q_\mathsf{f}}{2^{b+c}} + 2 \cdot \frac{\ell {q_\mathrm{C}}^2}{2^c} \cdot \left(d'(\ell) + \frac{64\ell^3}{2^c} + 2\right)$$

for all $\mathsf{xxx} \in \{\mathsf{prf}, \mathsf{dist\text{-}H}, \mathsf{sr}\}$.

Proof (sketch). The proofs are analogous to the proofs for WNMAC given in Sect. 3, with the main modification needed in Lemma 3 where the probability of an outer C-f-collision can be upper-bounded by $q_C q_f / 2^{b+c}$. Roughly speaking, this is because the outer call in DWNMAC does not contain the 0^{b-c} padding and instead processes $b+c$ bits of input that are hard to predict for the attacker. □

Acknowledgments. We thank the anonymous reviewers for their helpful comments. Gaži and Pietrzak's work was partly funded by the European Research Council under an ERC Starting Grant (259668-PSPC). Tessaro's research was partially supported by NSF grant CNS-1423566 and by the Glen and Susanne Culler Chair.

References

1. An, J.H., Bellare, M.: Constructing VIL-MACs from FIL-MACs: message authentication under weakened assumptions. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 252–269. Springer, Heidelberg (1999)
2. Bellare, M.: New proofs for NMAC and HMAC: security without collision-resistance. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 602–619. Springer, Heidelberg (2006)
3. Bellare, M., Canetti, R., Krawczyk, H.: Keying hash functions for message authentication. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 1–15. Springer, Heidelberg (1996)
4. Damgård, I.B.: A design principle for hash functions. In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 416–427. Springer, Heidelberg (1990)
5. Dinur, I., Leurent, G.: Improved generic attacks against hash-based MACs and HAIFA. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 149–168. Springer, Heidelberg (2014)
6. Dodis, Y., Ristenpart, T., Steinberger, J., Tessaro, S.: To hash or not to hash again? (In)differentiability results for H^2 and HMAC. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 348–366. Springer, Heidelberg (2012)
7. Gaži, P., Pietrzak, K., Rybár, M.: The exact PRF-security of NMAC and HMAC. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 113–130. Springer, Heidelberg (2014)
8. Gaži, P., Pietrzak, K., Tessaro, S.: The exact PRF security of truncation: tight bounds for keyed sponges and truncated CBC. In: Gennaro, R., Robshaw, M.J.B. (eds.) CRYPTO 2015, Part I. LNCS, vol. 9215, pp. 368–387. Springer, Heidelberg (2015)
9. Goldreich, O., Goldwasser, S., Micali, S.: On the cryptographic applications of random functions. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 276–288. Springer, Heidelberg (1985)
10. Guo, J., Peyrin, T., Sasaki, Y., Wang, L.: Updates on generic attacks against HMAC and NMAC. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 131–148. Springer, Heidelberg (2014)
11. Hardy, G.H., Wright, E.M.: An Introduction to the Theory of Numbers, 6th edn. Oxford University Press, Oxford (2008)
12. Håstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM J. Comput. **28**(4), 1364–1396 (1999)

13. Kim, J.-S., Biryukov, A., Preneel, B., Hong, S.H.: On the security of HMAC and NMAC based on HAVAL, MD4, MD5, SHA-0 and SHA-1 (Extended abstract). In: De Prisco, R., Yung, M. (eds.) SCN 2006. LNCS, vol. 4116, pp. 242–256. Springer, Heidelberg (2006)
14. Krawczyk, H., Bellare, M., Canetti, R.: HMAC: keyed-hashing for message authentication. In: IETF Internet Request for Comments 2104, February 1997
15. Leurent, G., Peyrin, T., Wang, L.: New generic attacks against hash-based MACs. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part II. LNCS, vol. 8270, pp. 1–20. Springer, Heidelberg (2013)
16. Merkle, R.C.: One way hash functions and DES. In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 428–446. Springer, Heidelberg (1990)
17. Naito, Y., Sasaki, Y., Wang, L., Yasuda, K.: Generic state-recovery and forgery attacks on ChopMD-MAC and on NMAC/HMAC. In: Sakiyama, K., Terada, M. (eds.) IWSEC 2013. LNCS, vol. 8231, pp. 83–98. Springer, Heidelberg (2013)
18. Patarin, J.: The "Coefficients H" technique. In: Avanzi, R.M., Keliher, L., Sica, F. (eds.) SAC 2008. LNCS, vol. 5381, pp. 328–345. Springer, Heidelberg (2009)
19. Peyrin, T., Sasaki, Y., Wang, L.: Generic related-key attacks for HMAC. In: Wang, X., Sako, K. (eds.) ASIACRYPT 2012. LNCS, vol. 7658, pp. 580–597. Springer, Heidelberg (2012)
20. Peyrin, T., Wang, L.: Generic universal forgery attack on iterative hash-based MACs. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 147–164. Springer, Heidelberg (2014)
21. Preneel, B., van Oorschot, P.C.: MDx-MAC and building fast MACs from hash functions. In: Coppersmith, D. (ed.) CRYPTO 1995. LNCS, vol. 963, pp. 1–14. Springer, Heidelberg (1995)
22. Sasaki, Y., Wang, L.: Generic attacks on strengthened HMAC: n-bit secure HMAC requires key in all blocks. In: Abdalla, M., De Prisco, R. (eds.) SCN 2014. LNCS, vol. 8642, pp. 324–339. Springer, Heidelberg (2014)
23. Yasuda, K.: "Sandwich" Is indeed secure: how to authenticate a message with just one hashing. In: Pieprzyk, J., Ghodosi, H., Dawson, E. (eds.) ACISP 2007. LNCS, vol. 4586, pp. 355–369. Springer, Heidelberg (2007)