New Bounds for PMAC, TMAC, and XCBC

Kazuhiko Minematsu[1,2] and Toshiyasu Matsushima[2]

[1] NEC Corporation, 1753 Shimonumabe, Nakahara-Ku, Kawasaki, Japan,
k-minematsu@ah.jp.nec.com
[2] Waseda University, 3-4-1 Okubo Shinjuku-ku Tokyo, Japan

Abstract. We provide new security proofs for PMAC, TMAC, and XCBC message authentication modes. The previous security bounds for these modes were $\sigma^2/2^n$, where n is the block size in bits and σ is the total number of queried message blocks. Our new bounds are $\ell q^2/2^n$ for PMAC and $\ell q^2/2^n + \ell^4 q^2/2^{2n}$ for TMAC and XCBC, where q is the number of queries and ℓ is the maximum message length in n-bit blocks. This improves the previous results under most practical cases, e.g., when no message is exceptionally long compared to other messages.

1 Introduction

Message authentication code (MAC) is a symmetric-keyed function used for ensuring the authenticity of messages. Many studies have been done on MACs built using blockciphers (i.e., MAC modes of operation) including the CBC-MAC and its variants. The theoretical security of stateless (i.e., no counter or nonce is used) MAC mode $F[E_K]$ using blockcipher E_K can be measured using the maximum advantage of an adversary trying to distinguish $F[E_K]$ from the random oracle, which provides an independent and uniform output for any distinct input, using a chosen-plaintext attack (CPA). Typically, the key task in proving the maximum advantage is to prove the maximum information-theoretic (IT) advantage for the target MAC, where the adversary has infinite computational power and the MAC is built using the uniform random permutation (URP), which is the ideal functionality of a blockcipher. Improving the maximum IT-advantage is important, because it will contribute to better understanding of the target function and to expanding the scope of application.

Bellare, Pietrzak, and Rogaway [5] analyzed the IT-advantage for the CBC-MAC and the encrypted CBC-MAC called EMAC [2]. Neglecting constants, the previous EMAC bound using the n-bit URP was $\ell^2 q^2/2^n$ [7] for any (q,ℓ)-CPA, which uses q chosen messages with lengths less than $n\ell$ bits. Bellare et al. investigated whether this could be improved, particularly with respect to ℓ. They proved the improved bound $\mathsf{d}(\ell)q^2/2^n + \ell^4 q^2/2^{2n}$, where $\mathsf{d}(\ell)$ is a function that grows very slowly with ℓ (see Sect. 4). A similar result was obtained for CBC-MAC for prefix-free messages. Recently, Pietrzak [18] proved EMAC bound $q^2/2^n$ for a range of (q,ℓ) (in fact, the result was $q^2/2^n + \ell^8 q^2/2^{2n}$ for any $q \geq \ell^2$).

Given these findings, it is quite natural to ask if similar improvements can be obtained for modes other than EMAC, especially more sophisticated ones.

EMAC uses two blockcipher keys, and only messages with a length multiple of n are supported. In this paper, we describe several MAC modes and provide new security bounds for them. Our first target is PMAC, which was proposed by Black and Rogaway [7] and Rogaway [19]. It is a one-key MAC; i.e., the MAC function uses one blockcipher key, and messages of any lengths are supported, and is fully parallelizable. The original security bound was $\sigma^2/2^n$ [7, 19], where σ is the total number of queried message blocks, which immediately implies $\ell^2 q^2/2^n$ for any (q, ℓ)-CPA, as $\sigma \leq \ell q$ holds. Here, we demonstrate a new bound $\ell q^2/2^n$ by taking an approach different from that of the previous proof.

We also provide new bounds for two successors of EMAC called TMAC [13] and XCBC [7]. Like EMAC, they are based on CBC-MAC. However, they do not use two blockcipher keys, and can efficiently handle messages of arbitrary length. Our bounds are obtained by combining our PMAC proof technique and the CBC-MAC collision analysis provided by Bellare et al.[5]. For TMAC and XCBC, the previous bounds are $\sigma^2/2^n$ shown by Iwata and Kurosawa [10], and our bound is $\ell q^2/2^n + \ell^4 q^2/2^{2n}$. We also investigated OMAC [11] (i.e., CMAC [1]), which is an optimized version of TMAC and XCBC. Although some part of TMAC proof can also be applied to OMAC, we could not obtain a new bound at this moment. The analysis of OMAC is briefly described in Sect. 5.

We have to emphasize that our results are not always better than the previous ones. Since all of our targets have $\sigma^2/2^n$ bounds, ours are worse if message length distribution is heavily biased to the left, e.g., one ℓ-block message and $(q-1)$ one-block messages. For other cases, ours are better. A detailed comparison is given in Sect. 5.

2 Preliminaries

Notation. $\{0,1\}$ and $\{0,1\}^n$ are denoted by Σ and Σ^n. The set of i-bit sequences for all $i = 1, \ldots, n$ is denoted by $\Sigma^{\leq n} \stackrel{\text{def}}{=} \bigcup_{i=1,\ldots n} \Sigma^i$. $(\Sigma^n)^{\leq m}$ is the set of binary sequences with lengths that are a multiple of n and at most nm. Σ^* is the set of all finite-length bit sequences. The bit length of x is denoted by $|x|$.

The n-bit uniform random permutation (URP), denoted by P_n, is a random permutation with a uniform distribution over all permutations on Σ^n. The random oracle (RO), which has n-bit output and is denoted by O_n, is a random function that accepts any $x \in \Sigma^*$ and outputs independent and uniformly random n-bit values for any distinct inputs. For any two colliding inputs, RO outputs the same value.

Field with 2^n points. We consider the elements of field GF(2^n) as n-bit coefficient vectors of the polynomials in the field. We represent n-bit coefficient vectors by integers $0, 1, \ldots, 2^n - 1$, e.g., 2 corresponds to coefficient vector $(00\ldots010)$, which corresponds to $\mathtt{x}$ in the polynomial representation, and 3 denotes $(00\ldots011)$, which corresponds to $\mathtt{x}+1$. For any $x, y \in \Sigma^n$, xy denotes the field multiplication of two elements represented by x and y. For simplicity, we assume $n = 128$ throughout the paper.

SECURITY NOTIONS. We used the standard security notion for symmetric cryptography [3, 4, 9].

Definition 1. *Let F and G be two random (here, random means it is probabilistic) functions. The oracle has implemented H, which is identical to one of F or G. An adversary, A, guesses if H is F or G using a θ-chosen-plaintext attack (θ-CPA), where θ is a list of parameters, such as the number of queries. The maximum advantage in distinguishing F from G is defined as*

$$\mathtt{Adv}^{\mathtt{cpa}}_{F,G}(\theta) \stackrel{\text{def}}{=} \max_{A:\theta\text{-CPA}} \big| \Pr[A^F = 1] - \Pr[A^G = 1] \big|,$$

where $A^F = 1$ denotes that A's guess is 1, which indicates one of F or G. The probabilities are determined by the randomness of F or G and A.

THE GOAL OF OUR ANALYSIS. In this paper, we consider only the information-theoretic security, where the adversary has infinite computational power (thus θ contains no computational restrictions), and the target is realized by the ideal n-bit blockcipher, i.e., P_n. In many cases, including ours, once the information-theoretic security is proved, the computational counterpart, where the adversary is computationally restricted and a real blockcipher is used, is quite easy.

Our target modes are stateless and variable-input-length (VIL) functions with n-bit output (VIL means that the domain is Σ^*). Therefore, for mode $F[E_K]$, where E_K is a blockcipher, we evaluate $\mathtt{Adv}^{\mathtt{vilqrf}}_{F[\mathsf{P}_n]}(\theta) \stackrel{\text{def}}{=} \mathtt{Adv}^{\mathtt{cpa}}_{F[\mathsf{P}_n],\mathsf{O}_n}(\theta)$. `vilqrf` denotes a VIL quasi-random function [15] that cannot be information-theoretically distinguished from RO without a negligibly small success probability. If $\mathtt{Adv}^{\mathtt{vilqrf}}_{F[\mathsf{P}_n]}(\theta)$ is small, the maximum success probability of a MAC forgery for all θ-CPAs against $F[\mathsf{P}_n]$ is also small (e.g., see Proposition 2.7 of [3]). In this paper, θ contains one of two additional parameters in addition to the number of queries, q: the total number of n-bit blocks for all q queries, σ, and the maximum length of a query (in n-bit blocks), ℓ. We focus on the $\theta = (q, \ell)$ case.

3 PMAC

3.1 Description and Previous Security Proof

PMAC has two versions; we focus on the later version [12, 19]. We call it simply "PMAC". The main idea of PMAC is as follows.

Lemma 1. *(Proposition 5 of [19]) Assume that the representation of* $\mathrm{GF}(2^n)$ *($n = 128$) is based on the lexicographically first primitive polynomial (see [19] for details). Let $\mathbb{I} = \{1, .., 2^{n/2}\}$ and $\mathbb{J} = \{0, 1, 2\}$ be the set of integers used as indices for distinct elements ("bases") of* $\mathrm{GF}(2^n)$*. Then, for any $(\alpha, \beta), (\alpha', \beta') \in \mathbb{I} \times \mathbb{J}$, $2^\alpha 3^\beta \neq 2^{\alpha'} 3^{\beta'}$ holds if $(\alpha, \beta) \neq (\alpha', \beta')$ holds*[3].

[3] Actually, Proposition 5 of [19] proved this for a wider range of indices. The original PMAC uses $\mathbb{J} = \{2, 3, 4\}$ instead of $\{0, 1, 2\}$, so our PMAC definition is slightly different from the original. However, the security proofs are essentially the same.

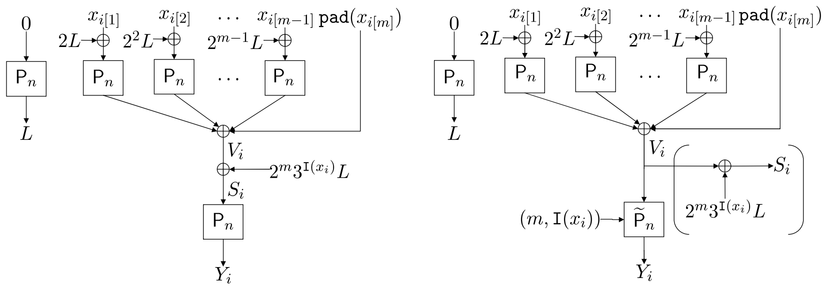

Fig. 1. PMAC[P_n] (left) and the modified PMAC (MPMAC) (right).

Multiplication of a constant and a variable is generally much simpler than multiplication of variables. For example, multiplication with 2 (i.e., a doubling operation) requires a bit shift followed by a conditional xor. Therefore, computation of $2^\alpha 3^\beta$ from $2^{\alpha-1}3^\beta$ or $2^\alpha 3^{\beta-1}$ is significantly faster than one blockcipher invocation. The idea of PMAC [19], called the "powering-up construction", is to use $2^\alpha 3^\beta$ as a masking value for every blockcipher input, incrementing α or β.

For blockcipher E_K, PMAC is defined as follows. We say "partition $x \in \Sigma^*$ into $(x_{[1]}, \ldots, x_{[m]})$" to set $m = \|x\|_n \overset{\text{def}}{=} \max\{\lceil |x|/n \rceil, 1\}$ and $x = (x_{[1]}, \ldots, x_{[m]})$ with $x_{[i]} \in \Sigma^n$ for $i = 1, \ldots, m-1$ and $x_{[m]} \in \Sigma^{\leq n}$. First, compute $L = E_K(0)$, where 0 corresponds to the all-zero n-bit sequence, in the preprocessing. Then, for input $x \in \Sigma^*$, partition it into $(x_{[1]}, \ldots, x_{[m]})$. The tag for x is $Y = E_K(\mathsf{Psum} \oplus \mathtt{pad}(x_{[m]}) \oplus 2^m 3^{\mathtt{I}(x)})$, where $\mathsf{Psum} = \bigoplus_{\alpha=1}^{m-1} E_K(x_{[\alpha]} \oplus 2^\alpha L)$ if $m > 1$, and if $m = 1$, $\mathsf{Psum} = 0$. Here, $\mathtt{I}(x) = 1$ if $|x|$ is a multiple of n and $\mathtt{I}(x) = 2$ otherwise, and $\mathtt{pad}(x_{[m]}) = x_{[m]}$ if $|x_{[m]}| = n$ and $\mathtt{pad}(x_{[m]}) = x_{[m]} \| 10^*$ otherwise, where $x_{[m]} \| 10^*$ is a concatenation of $x_{[m]}$ and the $(n - |x_{[m]}|)$-bit sequence $(100 \ldots 0)$.

Rogaway [19] proved the security of PMAC, which is as follows.

Theorem 1. *(Corollary 17 of [19]) Let* PMAC[P_n] *be the PMAC using* P_n *(see the left of Fig. 1). We then have* $\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{PMAC}[\mathsf{P}_n]}(q, \sigma) \leq 5.5\sigma^2/2^n$ *and* $\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{PMAC}[\mathsf{P}_n]}(q, \ell) \leq 5.5\ell^2 q^2/2^n$*, where* q*,* σ*, and* ℓ *are as defined in Sect. 2.*

Corollary 17 of [19] only proved the first claim. The second follows from the first and $\sigma \leq \ell q$.

3.2 New Security Bound for PMAC

Our security bound of PMAC is the following. The proof will be provided later.

Theorem 2. *Let* PMAC[P_n] *be the PMAC using* P_n*. We then have*

$$\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{PMAC}[\mathsf{P}_n]}(q, \ell) \leq \frac{5\ell q^2}{2^n - 2\ell}.$$

From this theorem, we have $\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{PMAC}[\mathsf{P}_n]}(q, \ell) \leq 10\ell q^2/2^n$ if $\ell \leq 2^{n-2}$.

Notation for Proof. Since we use Maurer's methodology[4] (e.g., see [15]) to make our proofs intuitive and simple, we briefly describe his notation. For completeness, part of his results that we used for our proof is cited in Appendix A. Consider event a_i defined for i input/output pairs, and possibly some internal variables, of random function F. Let $\overline{a_i}$ be the negation of a_i. We assume a_i is monotone; i.e., a_i never occurs if $\overline{a_{i-1}}$ occurs. For instance, a_i is monotone if it indicates that all i outputs are distinct. An infinite sequence of monotone events, $\mathcal{A} = a_0 a_1 \ldots$, is called a *monotone event sequence* (MES) [15]. Here, a_0 denotes some tautological event. Note that $\mathcal{A} \wedge \mathcal{B} = (a_0 \wedge b_0)(a_1 \wedge b_1)\ldots$ is an MES if $\mathcal{A} = a_0 a_1 \ldots$ and $\mathcal{B} = b_0 b_1 \ldots$ are both MESs. For any sequence of variables, $X_1, X_2, \ldots$, let X^i denote $(X_1, \ldots, X_i)$. We use $\mathrm{dist}(X^i)$ (or, equivalently, $\mathrm{dist}(\mathbf{X}^{(i)})$, where $\mathbf{X}^{(i)}$ is set $\{X_j\}_{j=1,\ldots,i}$) to denote an event where $X_1, X_2, \ldots, X_i$ are distinct.

Let MESs $\mathcal{A}$ and $\mathcal{B}$ be defined for two random functions, F and G, respectively. Let X_i and Y_i be the i-th input and output. Let P^F be the probability space defined by F. For example, $P^F_{Y_i|X^iY^{i-1}}(y^i, x^i)$ means $\Pr[Y_i = y_i | X^i = x^i, Y^{i-1} = y^{i-1}]$, where $Y_j = F(X_j)$ for $j \geq 1$.

Definition 2. *Let θ contain q. For MES $\mathcal{A}$ defined for F, $\nu_\theta(F, \overline{a_q})$ denotes the maximum probability of $\overline{a_q}$ for any θ-CPA that interacts with F. Similarly, $\mu_\theta(F, \overline{a_q})$ denotes the maximum probability of $\overline{a_q}$ for any non-adaptive θ-CPA. For $\theta = (q, \ell)$, they are abbreviated to $\nu_\ell(F, \overline{a_q})$ and $\mu_\ell(F, \overline{a_q})$. If $\theta = q$, the subscript is omitted, e.g., we write $\nu(F, \overline{a_q})$.*

Here, $\mu_\theta(F, \overline{a_q})$ can be rewritten as $\max_{x^q} P^F_{\overline{a_q}|X^q}(x^q)$, where the maximum is taken for all (non-adaptively chosen) x^q satisfying θ (e.g., if $\theta = (q, \ell)$, $|x_i| \leq n\ell$ for all $i \leq q$), hereafter abbreviated to $\max_{X^q} P^F_{\overline{a_q}|X^q}$.

Analysis of PHASH. Proving Theorem 2 requires an analysis of the message-hashing part of PMAC[P_n], which we call PHASH. For $x = (x_{[1]}, \ldots, x_{[m]}) \in (\Sigma^n)^m$, it is defined as:

$$\mathrm{PHASH}(x) \stackrel{\mathrm{def}}{=} \bigoplus_{i=1,\ldots,m} \mathsf{P}_n(x_{[i]} \oplus 2^i L), \text{ where } L = \mathsf{P}_n(0)\ .$$

Lemma 2. *For any $x = (x_{[1]}, \ldots, x_{[m]}) \in (\Sigma^n)^m$ and $x' = (x'_{[1]}, \ldots, x'_{[m']}) \in (\Sigma^n)^{m'}$, $x \neq x'$, and for any $f : \Sigma^n \to \Sigma^n$, we have*

$$\Pr[\mathrm{PHASH}(x) \oplus \mathrm{PHASH}(x') = f(L)] \leq \frac{m+m'}{2^n} + \frac{1}{2^n - (m+m')}, \text{ and} \quad (1)$$

$$\Pr[\mathrm{PHASH}(x) = f(L)] \leq \frac{m}{2^n} + \frac{1}{2^n - m}, \text{ where } L = \mathsf{P}_n(0)\ . \quad (2)$$

[4] It is known that some information-theoretic results obtained by Maurer's methodology can not be converted into computational ones (for instance, see [16, 17]). However, we do not encounter such difficulties in this paper.

Proof. We only prove Eq. (1) as Eq. (2) can be similarly proved. Fix x and x'. Let $U_i = x_{[i]} \oplus 2^i L$ for $i = 1, \ldots, m$, and $U_i = x'_{[i-m]} \oplus 2^{i-m} L$ for $i = m+1, \ldots, m+m'$. Then, $\mathrm{PHASH}(x) \oplus \mathrm{PHASH}(x')$ equals $\mathsf{Sum} \stackrel{\text{def}}{=} \mathsf{P}_n(U_1) \oplus \ldots \oplus \mathsf{P}_n(U_{m+m'})$. Let $\mathbf{U} = \{U_1, \ldots, U_{m+m'}\} \setminus \mathbf{U}_{\text{coll}}$, where $\mathbf{U}_{\text{coll}}$ is the set of all *trivial collisions*, e.g., U_1 and U_{1+m} when $x_{[1]} = x'_{[1]}$. Note that $\mathbf{U}$ can not be the empty set as $x \neq x'$. For simplicity, we assume no trivial collision (thus $\mathbf{U} = \{U_1, \ldots, U_{m+m'}\}$), however the following analysis works even if some trivial collisions exist. For index subset $\{i_1, \ldots, i_k\} \subseteq \{1, \ldots, q\}$, $\mathbf{U}_{\text{sub}} = \{U_{i_j}\}_{j=1,\ldots,k}$ is an *equivalent set* if $U_{i_1} = U_{i_2} = \cdots = U_{i_k}$ and $U_{i_1} \neq U_h$ for all $h \notin \{i_1, \ldots, i_k\}$. Here, the sum of all equivalent sets is a decomposition of $\mathbf{U}$. Whether $\mathbf{U}_{\text{sub}}$ is an equivalent set or not depends on the value of L. If k is odd (even), we say $\mathbf{U}_{\text{sub}}$ is an odd (even) equivalent set. Let odd_k be the event such that there are k odd equivalent sets having non-zero values (the value of an equivalent set is the value of its members). We have

$$\begin{aligned} &\Pr[\mathsf{Sum} = f(L) | \mathsf{odd}_k] \\ &\leq \max_{c \text{ satisfies } \mathsf{odd}_k} \Pr[\mathsf{P}_n(u_1(c)) \oplus \ldots \oplus \mathsf{P}_n(u_{m+m'}(c)) = f(c) | \mathsf{odd}_k, L = c], \end{aligned} \quad (3)$$

where $u_i(c)$ is $x_{[i]} \oplus 2^i c$ for $i = 1, \ldots, m$ and $x'_{[i-m]} \oplus 2^{i-m} c$ for $i = m+1, \ldots, m+m'$. In Eq. (3), note that $\mathsf{P}_n(u_i(c))$ is canceled out if $u_i(c)$ is in an even equivalent set. Therefore, given $L = c$ and odd_k for some $k > 0$, $\mathsf{P}_n(u_1(c)) \oplus \ldots \oplus \mathsf{P}_n(u_{m+m'}(c))$ is either the sum of k URP outputs for k non-zero distinct inputs or the sum of c and k URP outputs for k non-zero distinct inputs (note that odd_k does not exclude an odd equivalent set with value 0). Then, the property of P_n shows that, for any non-zero distinct k inputs, $z_1, \ldots, z_k$,

$$\begin{aligned} &\Pr[\mathsf{P}_n(z_1) \oplus \mathsf{P}_n(z_2) \oplus \ldots \oplus \mathsf{P}_n(z_k) = f(c) | \mathsf{P}_n(0) = c] \\ &= \sum_{c_1, \ldots, c_k, \mathrm{dist}(\{c_1, \ldots, c_k, c\}), c_1 \oplus \ldots \oplus c_k = f(c)} \Pr[\mathsf{P}_n(z_1) = c_1, \ldots, \mathsf{P}_n(z_k) = c_k | \mathsf{P}_n(0) = c] \\ &= \frac{|\{(c_1, \ldots, c_k) : \mathrm{dist}(\{c_1, \ldots, c_k, c\}), c_1 \oplus \ldots \oplus c_k = f(c)\}|}{(2^n - 1) \cdot \cdots \cdot (2^n - k)} \leq \frac{1}{2^n - k} \end{aligned} \quad (4)$$

holds, where the inequality holds since c_k is uniquely determined (or does not exist) if $c_1, \ldots, c_{k-1}$ are fixed. From Eqs. (3) and (4), we obtain

$$\Pr[\mathsf{Sum} = f(L) | \mathsf{odd}_k] \leq \frac{1}{2^n - k} \text{ for any } 0 < k \leq m + m' \text{ and for any } f. \quad (5)$$

Next, we analyze $\Pr[\mathsf{odd}_0]$. We have

$$\begin{aligned} \Pr[\mathsf{odd}_0] &= \Pr[\mathsf{odd}_0, U_1 \notin \{U_2, \ldots, U_{m+m'}\}] \\ &\quad + \Pr[\mathsf{odd}_0,\ U_1 = U_j \text{ for some } j = 2, \ldots, m+m'\ ] \\ &\leq \Pr[U_1 = 0] + \sum_{j=2,\ldots,m+m'} \Pr[U_1 = U_j] \leq (m + m') \frac{1}{2^n}, \end{aligned} \quad (6)$$

where the first inequality holds since if U_1 is unique (i.e., U_1 is in an odd equivalent set) and $\mathtt{odd}_0$ holds, U_1 must be 0. The second holds since both U_1 and $U_1 \oplus U_j$ for any $j \neq 1$ are permutations of L from Lemma 1. From Eqs. (5) and (6), we obtain

$$\begin{aligned}
\Pr[\mathsf{Sum} = f(L)] &= \sum_{k=0,\ldots,m+m'} \Pr[\mathsf{Sum} = f(L) | \mathtt{odd}_k] \cdot \Pr[\mathtt{odd}_k] \\
&\leq \Pr[\mathtt{odd}_0] + \sum_{k=1}^{m+m'-1} \frac{\Pr[\mathtt{odd}_k]}{2^n - k} + \frac{1 - \sum_{k=0}^{m+m'-1} \Pr[\mathtt{odd}_k]}{2^n - (m+m')} \\
&\leq \Pr[\mathtt{odd}_0] + \frac{1}{2^n - (m+m')} \leq \frac{m+m'}{2^n} + \frac{1}{2^n - (m+m')}.
\end{aligned}$$

This concludes the proof of Lemma 2.

Proof of Theorem 2. First, we introduce the tweakable[14] n-bit URP, $\widetilde{\mathsf{P}}_n$. It has tweak space $\mathcal{T} = \mathbb{I} \times \mathbb{J}'$, where $\mathbb{I} = \{1, \ldots, 2^{n/2}\}$ and $\mathbb{J}' = \{1, 2\}$. It consists of $|\mathcal{T}|$ independent n-bit URPs; $\widetilde{\mathsf{P}}_n(t, x)$ is the output of an n-bit URP indexed by $t \in \mathcal{T}$ and having input $x \in \Sigma^n$. Using $\widetilde{\mathsf{P}}_n$ and P_n, independent of $\widetilde{\mathsf{P}}_n$, we define the modified PMAC (MPMAC) as follows. First, compute $L = \mathsf{P}_n(0)$. For input $x \in \Sigma^* = (x_{[1]}, \ldots, x_{[m]})$, compute Psum using PHASH (i.e., $\mathsf{Psum} = \mathrm{PHASH}(\widehat{x})$, where $\widehat{x} = (x_{[1]}, \ldots, x_{[m-1]})$, if $m > 1$, and $\mathsf{Psum} = 0$ otherwise). The tag is $Y = \widetilde{\mathsf{P}}_n((m, \mathtt{I}(x)), \mathsf{Psum} \oplus \mathtt{pad}(x_{[m]}))$. Here, $(m, \mathtt{I}(x))$ is the tweak. Note that a tweak is a function of x.
Proof Idea. Since the advantage is the absolute difference between two probabilities, we can use a triangle inequality, $\mathtt{Adv}^{\mathsf{cpa}}_{\mathrm{PMAC}[\mathsf{P}_n], \mathsf{O}_n}(\theta)$ is not larger than $\mathtt{Adv}^{\mathsf{cpa}}_{\mathrm{PMAC}[\mathsf{P}_n], H}(\theta) + \mathtt{Adv}^{\mathsf{cpa}}_{H, \mathsf{O}_n}(\theta)$ for any VIL function H, and for any θ. Here, H is an intermediate function. Theorem 1 was derived using "PMAC with an ideal tweakable blockcipher", which invokes an independent URP for each message block in the message-hashing part as well as in the finalization, as the intermediate function. Here, our proof uses MPMAC as the intermediate function.

We start by proving the advantage between PMAC[P_n] and MPMAC, which requires defining some random variables. Let $X_i \in \Sigma^*$ be the i-th query of the adversary. If $m = \|X_i\|_n$, we write $X_i = (X_{i[1]}, \ldots, X_{i[m]})$. Note that X_i is a random variable, and its distribution is determined by the adversary and the target MAC. Fixed queries (and other random variables) are written in lower case, e.g., $x_i = (x_{i[1]}, \ldots, x_{i[m]})$. For PMAC[$\mathsf{P}_n$], let $\mathbf{M}^{(q)}$ ($\mathbf{C}^{(q)}$) be the set of inputs (outputs) to P_n generated in the PHASH for all q queries. We do not include the result of preprocessing, i.e., $L = \mathsf{P}_n(0)$, in $\mathbf{M}^{(q)}$ and $\mathbf{C}^{(q)}$. We also define $Y_i \in \Sigma^n$ as the i-th tag, and $\mathbf{Y}^{(q)} \stackrel{\mathrm{def}}{=} \{Y_i\}_{i=1,\ldots,q}$. If $m = \|X_i\|_n > 1$, we define V_i as the XOR of the i-th PHASH output and $\mathtt{pad}(X_{i[m]})$. If $m = 1$, $V_i = \mathtt{pad}(X_i)$. Moreover, $S_i \stackrel{\mathrm{def}}{=} V_i \oplus 2^m 3^{\mathtt{I}(X_{i[m]})} L$, and $\mathbf{S}^{(q)} \stackrel{\mathrm{def}}{=} \{S_i\}_{i=1,\ldots,q}$. Thus, in PMAC[$\mathsf{P}_n$], $Y_i = \mathsf{P}_n(S_i)$. These variables are similarly defined for MPMAC except Y_i; in MPMAC, Y_i is $\widetilde{\mathsf{P}}_n((m, \mathtt{I}(X_{i[m]})), V_i)$ when the i-th query has m blocks. Also, S_i is defined as a dummy variable in MPMAC. See Fig. 1 for reference.

Lemma 3. *Let event* $a_q \stackrel{\text{def}}{=} [\mathbf{M}^{(q)} \cap \mathbf{S}^{(q)} = \emptyset] \wedge [\text{dist}(\mathbf{S}^{(q)})] \wedge [0 \notin \mathbf{M}^{(q)} \cup \mathbf{S}^{(q)}]$. *Moreover,* $b_q \stackrel{\text{def}}{=} [\text{dist}(\mathbf{Y}^{(q)})]$, $d_q \stackrel{\text{def}}{=} [\mathbf{C}^{(q)} \cap \mathbf{Y}^{(q)} = \emptyset]$, *and* $e_q \stackrel{\text{def}}{=} [L \notin \mathbf{Y}^{(q)}]$, *where* $L = \mathsf{P}_n(0)$. *We then have*

$$\begin{aligned}\mathtt{Adv}^{\mathtt{cpa}}_{\mathrm{PMAC}[\mathsf{P}_n],\mathrm{MPMAC}}(q,\ell) &\leq \nu_\ell(\mathrm{MPMAC}, \overline{a_q \wedge b_q \wedge d_q \wedge e_q}) \\ &\leq \nu_\ell(\mathrm{MPMAC}, \overline{a_q \wedge b_q}) + \nu_\ell(\mathrm{MPMAC}, \overline{d_q \wedge e_q}) \\ &\leq \frac{1}{2^n - 2\ell}\left((4\ell - 2.5)q^2 + 1.5q\right). \end{aligned} \tag{7}$$

Proof. (of Lemma 3) The first and second inequalities are derived from Maurer's methodology. See Appendix B for the proof. In the following, we prove the third.
Analysis for $\nu_\ell(\mathrm{MPMAC}, \overline{a_q \wedge b_q})$. We use the following lemma. The proof is in Appendix C.

Lemma 4.

$$\nu_\ell(\mathrm{MPMAC}, \overline{a_q \wedge b_q}) = \mu_\ell(\mathrm{MPMAC}, \overline{a_q \wedge b_q}) = \max_{X^q} P^{\mathrm{MPMAC}}_{\overline{a_q}|X^q} + \max_{X^q} P^{\mathrm{MPMAC}}_{\overline{b_q}|a_q X^q},$$

where the maximums are taken for all $X^q = x^q$ *with* $|x_i| \leq n\ell$ *for all* i.

Note that $\max_{X^q} P^{\mathrm{MPMAC}}_{\overline{a_q}|X^q}$ denotes $\max_{x^q} P^{\mathrm{MPMAC}}_{\overline{a_q}|X^q}(x^q)$. Let $\mathbf{M}_i$ denote the input set to P_n that occur in the i-th PHASH call, except the all-zero input used to obtain L. Note that $\mathbf{M}^{(q)} = \mathbf{M}_1 \cup \cdots \cup \mathbf{M}_q$. For $i = 1, \ldots, q$, we have

$$\chi_1 \stackrel{\text{def}}{=} \text{dist}(\mathbf{S}^{(q)}), \quad \chi_{2,i} \stackrel{\text{def}}{=} [S_i \notin \mathbf{M}^{(q)}], \quad \chi_{3,i} \stackrel{\text{def}}{=} [S_i \neq 0], \text{ and } \quad \chi_{4,i} \stackrel{\text{def}}{=} [0 \notin \mathbf{M}_i].$$

Note that $\overline{a_q} \equiv \overline{\chi_1 \wedge \chi_2 \wedge \chi_3 \wedge \chi_4}$ where $\chi_i \stackrel{\text{def}}{=} \chi_{i,1} \wedge \cdots \wedge \chi_{i,q}$ for $i = 2, 3, 4$. Using the union bound and its variant, we have

$$\begin{aligned}\max_{X^q} P^{\mathrm{MPMAC}}_{\overline{a_q}|X^q} &\leq \max_{X^q} P^{\mathrm{MPMAC}}_{\overline{\chi_1}|X^q} \\ &+ \sum_{i=1,\ldots,q}\left(\max_{X^q} P^{\mathrm{MPMAC}}_{\overline{\chi_{2,i}}|\chi_{4,i},X^q} + \max_{X^q} P^{\mathrm{MPMAC}}_{\overline{\chi_{3,i}}|\chi_{4,i},X^q} + \max_{X^q} P^{\mathrm{MPMAC}}_{\overline{\chi_{4,i}}|X^q}\right).\end{aligned} \tag{8}$$

Now we analyze each term in Eq. (8). For this analysis, for some $i \neq j$, we fix the i-th and j-th queries to $x_i = (x_{i[1]}, \ldots, x_{i[m]})$ and $x_j = (x_{j[1]}, \ldots, x_{j[m']})$ with $x_i \neq x_j$. We start with the first term. Collision $S_i = S_j$ is equivalent to

$$V_i \oplus V_j \oplus 2^m 3^{\mathtt{I}(x_i)} L \oplus 2^{m'} 3^{\mathtt{I}(x_j)} L = 0. \tag{9}$$

To prove the maximum probability of Eq. (9), we need to use a case analysis.

Case 1: $m = m' = 1$. In this case, $V_i \oplus V_j = \mathtt{pad}(x_i) \oplus \mathtt{pad}(x_j)$. If $\mathtt{I}(x_i) \neq \mathtt{I}(x_j)$, the L.H.S. of Eq. (9) is a permutation of L from Lemma 1. Thus, the probability of Eq. (9) is $1/2^n$. If $\mathtt{I}(x_i) = \mathtt{I}(x_j)$, the probability is zero as $\mathtt{pad}(x_i) \neq \mathtt{pad}(x_j)$.

Case 2: $m > 1, m' = 1$. In this case, the probability of Eq. (9) is obviously at most $(m-1)/2^n + 1/(2^n - (m-1)) \leq (\ell - 1)/2^n + 1/(2^n - (\ell - 1))$ from the second claim of Lemma 2.

Case 3: $m = m' > 1$. If the first $m-1$ blocks of x_i and x_j are the same, the probability is at most $1/2^n$, which is the same as in Case 1. Otherwise, Eq. (9) occurs with a probability of at most $(2\ell-2)/2^n + 1/(2^n - (2\ell-2))$ from the first claim of Lemma 2.

Case 4: $m > 1, m' > 1, m \neq m'$. The bound of Case 3 also holds true.

Thus, we have

$$\max_{X^q} P^{\mathrm{MPMAC}}_{\overline{\chi_1}|X^q} \leq \sum_{i<j} \max_{X^q} P^{\mathrm{MPMAC}}_{[S_i=S_j]|X^q} \leq \binom{q}{2}\left(\frac{2\ell-2}{2^n} + \frac{1}{2^n-(2\ell-2)}\right). \quad (10)$$

For the second term, observe that $\overline{\chi_{2,i}}$ is the logical sum of events such that $S_i = 2^h L \oplus x_{i'[h]}$ for some i' including i, and $1 \leq h \leq \ell-1$. As x_i has m blocks, this is equivalent to $V_i = x_{i'[h]} \oplus 2^m 3^{\mathtt{I}(x_i)} L \oplus 2^h L$. From Lemma 1, we have $2^m 3^{\mathtt{I}(x_i)} \neq 2^h$. Thus, it is enough to evaluate the maximum of $\Pr[V_i = \mathtt{u}_1 L \oplus \mathtt{u}_2 | \chi_{4,i}]$ for all $\mathtt{u}_1 \in \Sigma^n \setminus \{0\}, \mathtt{u}_2 \in \Sigma^n$. We fix $\mathtt{u}_1 \neq 0$ and $\mathtt{u}_2$, and let $f(z) = \mathtt{u}_1 z \oplus \mathtt{u}_2$ and $\mathtt{u}_3 \stackrel{\mathrm{def}}{=} (\mathtt{u}_2 \oplus \mathsf{pad}(x_{i[m]}))/\mathtt{u}_1$, where / denotes field division. We assume that $m > 1$ and $L = \mathtt{u}_3$ satisfies $\chi_{4,i}$ with x_i. Note that $\Pr[V_i = f(L)|\chi_{4,i}]$ equals:

$$\begin{aligned}
&\sum_c \Pr[V_i = f(L)|L = c]\Pr[L = c|\chi_{4,i}] + \Pr[V_i = 0|L = \mathtt{u}_3]\Pr[L = \mathtt{u}_3|\chi_{4,i}], \\
&\leq \max_c \Pr[V_i = f(c)|L = c] + \Pr[L = \mathtt{u}_3|\chi_{4,i}], && (11) \\
&= \max_c \Pr[\mathsf{Sum}(c) = f(c) \oplus \mathsf{pad}(x_{i[m]})|L = c] + \Pr[L = \mathtt{u}_3|\chi_{4,i}], && (12)
\end{aligned}$$

where the sum and maximums are taken for all $c \neq \mathtt{u}_3$ that satisfies $\chi_{4,i}$, and $\mathsf{Sum}(c) = \mathsf{P}_n(x_{i[1]} \oplus 2c) \oplus \ldots \oplus \mathsf{P}_n(x_{i[m-1]} \oplus 2^{m-1}c)$. If every element in $\{(x_{i[1]} \oplus 2c), \ldots, (x_{i[m-1]} \oplus 2^{m-1}c)\}$ is in an even equivalent set, $\mathsf{Sum}(c)$ is 0 while $f(c) \oplus \mathsf{pad}(x_{i[m]}) \neq 0$ from $c \neq \mathtt{u}_3$. If there exists any element which is in an odd equivalent set, $\mathsf{Sum}(c)$ is the sum of k URP outputs for distinct inputs, for some $1 \leq k \leq m-1$. These inputs are not 0 as c satisfies $\chi_{4,i}$. Therefore, the first term of the R.H.S. of Eq. (12) is at most $1/(2^n - (\ell-1))$ from Eq. (4). Also, the second term of the R.H.S. of Eq. (12) is at most $1/(2^n - (\ell-1))$. From these observations, $\max_{\mathtt{u}_1 \neq 0, \mathtt{u}_2} \Pr[V_i = \mathtt{u}_1 L \oplus \mathtt{u}_2 | \chi_{4,i}]$ is at most $2/(2^n - (\ell-1))$ if $m > 1$ and $L = \mathtt{u}_3$ satisfies $\chi_{4,i}$. For other cases (i.e., when $m = 1$ or $m > 1$ and $L = \mathtt{u}_3$ does not satisfy $\chi_{4,i}$), this bound also holds true. Therefore,

$$\max_{X^q} P^{\mathrm{MPMAC}}_{\overline{\chi_{2,i}}|\chi_{4,i},X^q} \leq (\ell-1)q \cdot \max_{\mathtt{u}_1 \neq 0, \mathtt{u}_2} \Pr[V_i = \mathtt{u}_1 L \oplus \mathtt{u}_2 | \chi_{4,i}] = \frac{2(\ell-1)q}{2^n - (\ell-1)} \quad (13)$$

holds for any $1 \leq i \leq q$, where the inequality holds since $\mathbf{M}^{(q)}$ contains at most $(\ell-1)q$ distinct elements. For the third and fourth terms of Eq. (8), we have

$$\max_{X^q} P^{\mathrm{MPMAC}}_{\overline{\chi_{3,i}}|\chi_{4,i},X^q} \leq \frac{2}{2^n - (\ell-1)}, \text{ and } \max_{X^q} P^{\mathrm{MPMAC}}_{\overline{\chi_{4,i}}|X^q} \leq \frac{(\ell-1)}{2^n}, \quad (14)$$

where the first inequality follows from the same analysis as for the second term, and the second inequality holds since $\overline{\chi_{4,i}}$ occurs if L takes one of (at most) $\ell-1$

values defined by x_i. Combining Eqs. (10),(13), and (14), we get

$$\max_{X^q} P^{\mathrm{MPMAC}}_{\overline{a_q}|X^q} \le \binom{q}{2}\left(\frac{2\ell-2}{2^n} + \frac{1}{2^n-(2\ell-2)}\right) + \frac{2(\ell-1)q^2+2q}{2^n-(\ell-1)} + \frac{(\ell-1)q}{2^n}$$
$$\le \frac{1}{2^n-(2\ell-2)}((3\ell-2.5)q^2+1.5q). \tag{15}$$

Note that a_q implies $V_i \neq V_j$ if i-th and j-th tweaks are the same, for all $1 \le i < j \le q$. Therefore, if a_q is given, the collision probability between Y_i and Y_j is at most $1/2^n$ for all fixed q queries. Thus we have

$$\max_{X^q} P^{\mathrm{MPMAC}}_{\overline{b_q}|a_q X^q} \le \sum_{i<j} \max_{X^q} P^{\mathrm{MPMAC}}_{[Y_i=Y_j]|a_q X^q} \le \binom{q}{2}\frac{1}{2^n}. \tag{16}$$

Analysis for $\nu_\ell(\mathrm{MPMAC}, \overline{d_q \wedge e_q})$. We consider a tweakable function, G, having n-bit input and output and tweak space $\mathcal{T} = \mathbb{I}\times\mathbb{J}$, where $\mathbb{I} = \{1,..,2^{n/2}\}$ and $\mathbb{J} = \{0,1,2\}$. For any input $x \in \Sigma^n$ and tweak $t = (t_{[1]}, t_{[2]}) \in \mathcal{T}$, it is defined as $G(t,x) \overset{\mathrm{def}}{=} \mathsf{P}_n(x)$ if $t_{[2]} = 0$, otherwise $G(t,x) \overset{\mathrm{def}}{=} \widetilde{\mathsf{P}}_n((t_{[1]}, t_{[2]}), x)$, where P_n and $\widetilde{\mathsf{P}}_n$ are independent. If we allow an adversary against G to make $(\ell-1)q$ queries for P_n and q queries for $\widetilde{\mathsf{P}}_n$ (the order of query is arbitrary), he can *simulate* any (q,ℓ)-CPA against MPMAC. Here, we assume that $L = \mathsf{P}_n(0)$ is publicly available, so that ℓq queries are enough to simulate an attack. Moreover, if a G-based simulation generates distinct ℓq outputs of G, this implies the occurrence of d_q in MPMAC[5]. From these observations, $\nu_\ell(\mathrm{MPMAC}, \overline{d_q})$ is at most

$$\nu_{\widetilde{\ell q}}(G, \overline{\mathrm{dist}(\mathbf{Y}^{(\ell q)})}) = \mu_{\widetilde{\ell q}}(G, \overline{\mathrm{dist}(\mathbf{Y}^{(\ell q)})}) \le \frac{(\ell-1)q^2}{2^n} + \binom{q}{2}\frac{1}{2^n},$$

where $\mathbf{Y}^{(\ell q)}$ is the set of ℓq outputs, and $\widetilde{\ell q}$ means that the adversary can make $(\ell-1)q$ queries for P_n and q queries for $\widetilde{\mathsf{P}}_n$, and the equality follows from an analysis similar to the one used for the proof of Lemma 4. The last inequality is trivial. Similarly, we can prove $\nu_\ell(\mathrm{MPMAC}, \overline{e_q}) \le q/2^n$ using G. Thus we have

$$\nu_\ell(\mathrm{MPMAC}, \overline{d_q \wedge e_q}) \le \frac{(\ell-1)q^2}{2^n} + \binom{q}{2}\frac{1}{2^n} + \frac{q}{2^n} \tag{17}$$

using Lemma 9. Combining Eqs. (15),(16), and (17) and Lemma 4, Lemma 3 is proved.

Proving Theorem 2. Deriving an upper bound of $\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{MPMAC}}(q,\ell)$ is easy since MPMAC can be seen as an instance of the Carter-Wegman MAC [20](CW-MAC). Since the following lemma is almost the same as previous CW-MAC lemmas (e.g., Lemma 4 of [5]), we omit the proof here.

[5] We assume that the adversary never makes colliding queries and a pair of queries such as $((t_{[1]}, t_{[2]}), x)$ and $((t'_{[1]}, t'_{[2]}), x')$ with $t_{[2]} = t'_{[2]} = 0$, $x = x'$, and $t_{[1]} \neq t'_{[1]}$. These queries are obviously useless for simulation.

Lemma 5. $\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{MPMAC}}(q,\ell) \le \binom{q}{2}\mathrm{dp}(\ell-1) + \binom{q}{2}/2^n$*, where* $\mathrm{dp}(m)$ *denotes* $\max_{x,x' \in (\Sigma^n)^{\le m}, x \ne x', \mathtt{u} \in \Sigma^n} \Pr[\mathrm{PHASH}(x) \oplus \mathrm{PHASH}(x') = \mathtt{u}]$.

Finally, combining Lemmas 2, 3, and 5, we obtain

$$\begin{aligned}
\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{PMAC}[\mathsf{P}_n]}(q,\ell) &\le \mathtt{Adv}^{\mathtt{cpa}}_{\mathrm{PMAC}[\mathsf{P}_n],\mathrm{MPMAC}}(q,\ell) + \mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{MPMAC}}(q,\ell), \text{ and} && (18)\\
&\le \frac{(4\ell-2.5)q^2+1.5q}{2^n-2\ell} + \binom{q}{2}\left(\frac{2\ell-2}{2^n} + \frac{1}{2^n-(2\ell-2)} + \frac{1}{2^n}\right)\\
&\le \frac{(5\ell-2.5)q^2+(1.5-\ell)q}{2^n-2\ell} \le \frac{5\ell q^2}{2^n-2\ell}, && (19)
\end{aligned}$$

where the last inequality holds since $q, \ell \ge 1$. This concludes the proof of Theorem 2.

4 TMAC and XCBC

4.1 New Security Bounds for TMAC and XCBC

Since CBC-MAC provides no security if two messages with the same prefix are processed, a number of modifications have been proposed to make CBC-MAC secure for any message. EMAC, an early attempt, uses two blockcipher keys; TMAC [13] and XCBC [7] were later proposed as better solutions: they use one blockcipher key and some additional keys, and thus avoid two blockcipher key schedulings. TMAC and XCBC are defined as follows. Let CBC be the CBC-MAC function using P_n; that is, for input $x = (x_{[1]}, \dots, x_{[m]}) \in (\Sigma^n)^m$, $\mathrm{CBC}(x) = C_m$, where $C_i = \mathsf{P}_n(x_{[i]} \oplus C_{i-1})$ and $C_0 = 0$. Let $\mathrm{TMAC}[\mathsf{P}_n]$ denote the TMAC using P_n. For input $x \in \Sigma^*$, $\mathrm{TMAC}[\mathsf{P}_n]$ works as follows. First, we partition x into $x = (x_{[1]}, \dots, x_{[m]})$, where $m = \|x\|_n$. If $m > 1$, the tag for x is $Y = \mathsf{P}_n(\mathrm{CBC}(\widehat{x}) \oplus \mathtt{pad}(x_{[m]}) \oplus 2^{\mathtt{I}(x)-1}L)$, where $\widehat{x} = (x_{[1]}, \dots, x_{[m-1]})$ and L is independent and uniform over Σ^n. If $m = 1$, Y is $\mathsf{P}_n(\mathtt{pad}(x) \oplus 2^{\mathtt{I}(x)-1}L)$. Note that the P_n used in CBC and the one used in the finalization are identical. Therefore, in practice, TMAC has one blockcipher key and an additional n-bit key L. XCBC is similar to TMAC, but uses two n-bit keys, L_1 and L_2, as masking values instead of L and $2L$. The previous bound of $\mathrm{TMAC}[\mathsf{P}_n]$ is $(3\ell^2+1)q^2/2^n$ [13] against (q,ℓ)-CPA, and $3\sigma^2/2^n$ against (q,σ)-CPA [10]. Almost the same results are obtained for XCBC [10, 13]. However, using our proof approach in Sect. 3 and Bellare et al.'s analysis of the CBC function [5], we obtain the following.

Theorem 3. *Let* $\mathrm{TMAC}[\mathsf{P}_n]$ *be the TMAC using* P_n*. We then have*

$$\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{TMAC}[\mathsf{P}_n]}(q,\ell) \le \frac{4\ell q^2}{2^n} + \frac{64\ell^4 q^2}{2^{2n}}.$$

The proof of Theorem 3 is in the next section. The bound of Theorem 3 is also applicable to XCBC. The proof for XCBC is the same as the proof of Theorem 3, thus we omit it here.

4.2 Proof of Theorem 3

Since the proof structure is the same as that of Theorem 2, we give only a sketch of the proof. We define a modified TMAC, denoted by MTMAC, that uses an independent tweakable URP for its finalization. In MTMAC, we partition message x into $(x_{[1]},\ldots,x_{[m]})$, where $m = \|x\|_n$, and when $m > 1$, the tag is $Y = \widetilde{\mathsf{P}}_n(\mathtt{I}(x), \mathrm{CBC}(\widehat{x}) \oplus \mathtt{pad}(x_{[m]}))$, where $\mathtt{I}(x) \in \{1,2\}$ is a tweak. When $m = 1$, $Y = \widetilde{\mathsf{P}}_n(\mathtt{I}(x), \mathtt{pad}(x))$. For both TMAC[$\mathsf{P}_n$] and MTMAC, let $X_i \in \Sigma^*$ be the i-th query and $\mathbf{M}^{(q)}$ ($\mathbf{C}^{(q)}$) be the set of inputs (outputs) to P_n generated in the CBC function for all q queries. We also define Y_i as the i-th tag, and $\mathbf{Y}^{(q)} \stackrel{\text{def}}{=} \{Y_i\}_{i=1,\ldots,q}$. When $\|X_i\|_n = m > 1$, we define V_i as the XOR of the i-th CBC output and $\mathtt{pad}(X_{i[m]})$, and when $m = 1$, $V_i = \mathtt{pad}(X_i)$. Moreover, $S_i \stackrel{\text{def}}{=} V_i \oplus 2^{\mathtt{I}(X_i)-1}L$, and $\mathbf{S}^{(q)} \stackrel{\text{def}}{=} \{S_i\}_{i=1,\ldots,q}$. In MTMAC, S_i is a dummy variable. Note that $Y_i = \mathsf{P}_n(S_i)$ in TMAC[P_n] and that $Y_i = \widetilde{\mathsf{P}}_n(\mathtt{I}(X_i), V_i)$ in MTMAC, where $m = \|X_i\|_n$. We define $a_q \stackrel{\text{def}}{=} \mathrm{dist}(\mathbf{S}^{(q)}) \wedge [\mathbf{M}^{(q)} \cap \mathbf{S}^{(q)} = \emptyset]$ and $b_q \stackrel{\text{def}}{=} \mathrm{dist}(\mathbf{Y}^{(q)})$, $d_q \stackrel{\text{def}}{=} [\mathbf{C}^{(q)} \cap \mathbf{Y}^{(q)} = \emptyset]$. We then obtain

$$\mathtt{Adv}^{\mathtt{cpa}}_{\mathrm{TMAC}[\mathsf{P}_n],\mathrm{MTMAC}}(q,\ell) \le \nu_\ell(\mathrm{MTMAC}, \overline{a_q \wedge b_q}) + \nu_\ell(\mathrm{MTMAC}, \overline{d_q}) \tag{20}$$

for any (q,ℓ) using an argument similar to that used for Lemma 3. Note that a_q does not contain $[0 \notin \mathbf{M}^{(q)} \cup \mathbf{S}^{(q)}]$, as we do not have to care about 0 being an input to P_n. Since Lemma 4 does not depend on the structure of message-hashing part, it also applies to MTMAC and we have

$$\nu_\ell(\mathrm{MTMAC}, \overline{a_q \wedge b_q}) = \mu_\ell(\mathrm{MTMAC}, \overline{a_q \wedge b_q}) \le \max_{X^q} P^{\mathrm{MTMAC}}_{\overline{a_q}|X^q} + \max_{X^q} P^{\mathrm{MTMAC}}_{\overline{b_q}|a_q X^q}. \tag{21}$$

To obtain bounds of last two terms of Eq. (21), we need the following lemma[6]. It generalizes a lemma of Bellare et al.[5].

Lemma 6.

$$\max_{x\in(\Sigma^n)^m, x'\in(\Sigma^n)^{m'}, x\neq x', \mathtt{u}\in\Sigma^n} \Pr[\mathrm{CBC}(x) \oplus \mathrm{CBC}(x') = \mathtt{u}] \le \frac{2\mathsf{d}(m^*)}{2^n} + \frac{64(m^*)^4}{2^{2n}},$$

$$\max_{x\in(\Sigma^n)^m, \mathtt{u}\in\Sigma^n} \Pr[\mathrm{CBC}(x) = \mathtt{u}] \le \frac{2\mathsf{d}(m+1)}{2^n} + \frac{64(m+1)^4}{2^{2n}},$$

where $\mathsf{d}(m)$ is the maximum number of positive integers that divide h, for all $h \le m$, and $m^* = \max\{m, m'\} + 1$.

Proof. (of Lemma 6) For any $z \in \Sigma^n$, $\mathrm{CBC}(x) \oplus \mathrm{CBC}(x') = \mathtt{u}$ is equivalent to $\mathsf{P}_n(\mathrm{CBC}(x) \oplus z) = \mathsf{P}_n(\mathrm{CBC}(x') \oplus z \oplus \mathtt{u})$, which is equivalent to $\mathrm{CBC}(x\|z) =$

[6] Pietrzak [18] proved that the collision probability of CBC among q messages could be smaller than the union bound applied to Lemma 6 for some (q,ℓ). Since our analysis is based on the union bound, we do not know if the result of [18] can be combined into our proof to obtain other proofs.

CBC($x' \| (z \oplus \mathtt{u})$). From Lemma 5 of [5], we see that the collision probability of CBC($x \| z$) and CBC($x' \| (z \oplus \mathtt{u})$) is at most $2\mathsf{d}(m^*)/2^n + 64(m^*)^4/2^{2n}$ for any z (note that $x \| z \neq x' \| (z \oplus \mathtt{u})$ holds for any z and $\mathtt{u}$ as we assumed $x \neq x'$). Therefore, the first claim is proved. The second can be similarly proved.

We analyze $\max_{X^q} P^{\mathrm{MTMAC}}_{\overline{a_q}|X^q}$. If the i-th and j-th queries are fixed to x_i and x_j with $x_i \neq x_j$, collision $S_i = S_j$ is equivalent to $V_i \oplus 2^{\mathtt{I}(x_i)}L = V_j \oplus 2^{\mathtt{I}(x_j)}L$. If $\mathtt{I}(x_i) \neq \mathtt{I}(x_j)$, the collision occurs with probability $1/2^n$ since L is independent of V_i and V_j and $2^{\mathtt{I}(x_i)}L \oplus 2^{\mathtt{I}(x_j)}L$ is a permutation of L from Lemma 1. If $\mathtt{I}(x_i) = \mathtt{I}(x_j)$, $S_i = S_j$ implies $V_i = V_j$, which has a probability of at most $2\mathsf{d}(\ell)/2^n + 64\ell^4/2^{2n}$ from Lemma 6 and a case analysis similar to the one used to derive Eq. (10). Therefore, the probability of $\overline{\mathrm{dist}(\mathbf{S}^{(q)})}$ is at most $\binom{q}{2}(2\mathsf{d}(\ell)/2^n + 64\ell^4/2^{2n})$. Note that any collision event consisting of $\overline{[\mathbf{M}^{(q)} \cap \mathbf{S}^{(q)} = \emptyset]}$ has probability $1/2^n$ since L is independent of all members of $\mathbf{M}^{(q)}$. From these observations, we have

$$\begin{aligned}
\max_{X^q} P^{\mathrm{MTMAC}}_{\overline{a_q}|X^q} &\leq \max_{X^q} P^{\mathrm{MTMAC}}_{\overline{\mathrm{dist}(\mathbf{S}^{(q)})}|X^q} + \max_{X^q} P^{\mathrm{MTMAC}}_{\overline{[\mathbf{M}^{(q)} \cap \mathbf{S}^{(q)} = \emptyset]}|X^q} \\
&\leq \binom{q}{2}\left(\frac{2\mathsf{d}(\ell)}{2^n} + \frac{64\ell^4}{2^{2n}}\right) + \frac{(\ell-1)q^2}{2^n}.
\end{aligned} \tag{22}$$

The analyses of $\max_{X^q} P^{\mathrm{MTMAC}}_{\overline{b_q}|a_q X^q}$ and $\nu_\ell(\mathrm{MTMAC}, \overline{d_q})$ are the same as those used for the proof of Lemma 3. We obtain

$$\max_{X^q} P^{\mathrm{MTMAC}}_{\overline{b_q}|a_q X^q} \leq \binom{q}{2}\frac{1}{2^n}, \text{ and } \nu_\ell(\mathrm{MTMAC}, \overline{d_q}) \leq \frac{(\ell-1)q^2}{2^n} + \binom{q}{2}\frac{1}{2^n}. \tag{23}$$

As with MPMAC, MTMAC is an instance of CW-MAC. Thus, we have

$$\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{MTMAC}}(q,\ell) \leq \binom{q}{2}\left(\frac{2\mathsf{d}(\ell)}{2^n} + \frac{64\ell^4}{2^{2n}} + \frac{1}{2^n}\right). \tag{24}$$

Combining the bound of $\mathtt{Adv}^{\mathtt{cpa}}_{\mathrm{TMAC}[\mathsf{P}_n],\mathrm{MTMAC}}(q,\ell)$, which can be derived from Eqs. (20),(21),(22), and (23), and the bound of $\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{MTMAC}}(q,\ell)$ by Eq. (24), $\mathtt{Adv}^{\mathtt{vilqrf}}_{\mathrm{TMAC}[\mathsf{P}_n]}(q,\ell)$ is at most $((2\mathsf{d}(\ell) + 2\ell)q^2)/2^n + 64\ell^4 q^2/2^{2n}$. Since $\mathsf{d}(\ell) \leq \ell$, this concludes the proof of Theorem 3.

5 Conclusion and Future Work

In this paper, we have provided new security bounds for PMAC, TMAC, and XCBC. Our result demonstrates that the security degradation with respect to the maximum length of a message is linear for PMAC and almost linear (unless message is impractically long) for TMAC and XCBC, while previous analyses of these modes proved quadratic security degradation.

A Comparison of Bounds. As we mentioned, our new bounds improve the old ones under most (but not all) cases. Here, we give a detailed comparison between new and old bounds. For simplicity, we ignore the constants. Thus, the new PMAC bound is $\ell q^2/2^n$, the new TMAC (and XCBC) bound is $\ell q^2/2^n + \ell^4 q^2/2^{2n}$, and the old bounds are $\sigma^2/2^n$ for all. For PMAC, the new bound is better if and only if $\sqrt{\ell} q < \sigma$, i.e., the mean message block length (σ/q) is larger than $\sqrt{\ell}$. Similarly, for TMAC and XCBC, the new bound is better if and only if the mean message block length is larger than $\sqrt{\ell(1+c)}$, where $c = \ell^3/2^n$, which can be small in practice. Thus, the criterion for choosing a bound is the distance between the mean block length and the square root of the maximum block length.

As a concrete example, let $n = 128$, $q = 2^{40}$, and $\ell = 2^{16}$. Then the new PMAC bound is 2^{-32} (the new TMAC and XCBC bounds are almost 2^{-32}), while the old bound ranges from 2^{-48} to 2^{-16}. The old bound is better if 99.9% of the messages are one-block, as $\sigma^2/2^n \leq (1 \cdot 0.999q + \ell \cdot 0.001q)^2/2^n < 2^{-35}$. In this case, the mean block length is smaller than 2^6, which is smaller than $\sqrt{\ell} = 2^8$. In contrast, if 1% of the messages are ℓ-block, the new bound is better since $\sigma^2/2^n \geq (\ell \cdot 0.01q + 1 \cdot 0.99q)^2/2^n > 2^{-30}$ and the mean block length is at least 2^9. Generally, the new bounds are better when only a tiny fraction of the message length distribution is concentrated on the right.

On The Security of OMAC. OMAC [11], i.e., CMAC [1], is similar to TMAC, but uses a different finalization. In OMAC using P_n, denoted by OMAC[P_n], L is $\mathsf{P}_n(0)$, and, instead of using $2^{\mathtt{I}(x)-1}L$, it uses $2^{\mathtt{I}(x)}L$ as the masking value. Thus OMAC has only one blockcipher key. The known security bound of OMAC[P_n] is $(5\ell^2+1)q^2/2^n$[11] against (q,ℓ)-CPA, and $4\sigma^2/2^n$ against (q,σ)-CPA [10]. Unfortunately, we have not yet succeeded in showing new bounds. In a manner similar to that for TMAC, we define a modified[7] OMAC (MOMAC), using P_n and $\widetilde{\mathsf{P}}_n$, and define sets of variables, ($\mathbf{M}^{(q)}$, $\mathbf{C}^{(q)}$, $\mathbf{S}^{(q)}$, and $\mathbf{Y}^{(q)}$), for both OMAC[P_n] and MOMAC. By defining events $a_q \stackrel{\text{def}}{=} \text{dist}(\mathbf{S}^{(q)}) \wedge [\mathbf{M}^{(q)} \cap \mathbf{S}^{(q)} = \emptyset] \wedge [0 \notin \mathbf{S}^{(q)}]$, $b_q \stackrel{\text{def}}{=} \text{dist}(\mathbf{Y}^{(q)})$, $d_q \stackrel{\text{def}}{=} [\mathbf{C}^{(q)} \cap \mathbf{Y}^{(q)} = \emptyset]$, and $e_q \stackrel{\text{def}}{=} [L \notin \mathbf{Y}^{(q)}]$, we can prove that $\mathtt{Adv}^{\mathtt{cpa}}_{\text{OMAC}[\mathsf{P}_n],\text{MOMAC}}(q,\ell)$ is at most $\nu_\ell(\text{MOMAC}, \overline{a_q \wedge b_q \wedge d_q \wedge e_q})$. However, to obtain a bound of $\nu_\ell(\text{MOMAC}, \overline{a_q \wedge b_q})$, we need the maximum probability of $[\text{CBC}(x) \oplus \text{CBC}(x') = \mathtt{u}_1 L \oplus \mathtt{u}_2]$ for $\mathtt{u}_1 = (2 \oplus 2^2)$, which corresponds to the sum of two distinct masking values, and for all $\mathtt{u}_2$, i.e., we need a generalization of Lemma 6. We think that this is an interesting open problem.

Acknowledgments

We would like to thank Tetsu Iwata and the anonymous referees for very useful comments and suggestions.

References

1. Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication. *NIST Special Publication* 800-38B, available from `http://csrc.nist.gov/CryptoToolkit/modes/`

[7] The "modified OMAC" was also described in [11]; however, our definition is different.

2. B. den Boer, J.P. Boly, A. Bosselaers, J. Brandt, D. Chaum, I. Damgård, M. Dichtl, W. Fumy, M. van der Ham, C.J.A. Jansen, P. Landrock, B. Preneel, G. Roelofsen, P. de Rooij, and J. Vandewalle, *RIPE Integrity Primitives*, final report of RACE Integrity Primitives Evaluation. 1995.
3. M. Bellare, J. Kilian, and P. Rogaway. "The Security of the Cipher Block Chaining Message Authentication Code." *Journal of Computer and System Science*, Vol. 61, No. 3, 2000.
4. M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. " A Concrete Security Treatment of Symmetric Encryption." *Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, FOCS '97, pp. 394-403, 1997.
5. M. Bellare, K. Pietrzak, and P. Rogaway. "Improved Security Analyses for CBC MACs." *Advances in Cryptology* - CRYPTO '05, LNCS 3621, pp. 527-541, 2005.
6. D. J. Bernstein. "Stronger Security Bounds for Wegman-Carter-Shoup Authenticators." *Advances in Cryptology*- EUROCRYPT '05, LNCS 3494, pp. 164-180, 2005.
7. J. Black and P. Rogaway. "CBC MACs for Arbitrary-Length Messages: The Three-Key Constructions." *Advances in Cryptology*- CRYPTO '00, LNCS 1880, pp. 197-215, 2000.
8. L. Carter and M. Wegman. "Universal Classes of Hash Functions." *Journal of Computer and System Science*, Vol. 18, pp. 143-154, 1979.
9. O. Goldreich. "Modern Cryptography, Probabilistic Proofs and Pseudorandomness." Springer-Verlag, Algorithms and Combinatorics, Vol. 17, 1998.
10. T. Iwata and K. Kurosawa. "Stronger Security Bounds for OMAC, TMAC, and XCBC." *Progress in Cryptology*- INDOCRYPT'03, LNCS 2904, pp. 402-415, 2003.
11. T. Iwata and K. Kurosawa. "OMAC: One-Key CBC MAC." *Fast Software Encryption*- FSE'03, LNCS 2887, pp. 129-153, 2003.
12. T. Krovetz and P. Rogaway. "The OCB Authenticated-Encryption Algorithm. " *Internet Draft*, 2005.
13. K. Kurosawa and T. Iwata. "TMAC: Two-Key CBC MAC." *Topics in Cryptology*- CT-RSA 2003, LNCS 2612, pp. 33-49, 2003.
14. M. Liskov, R. L. Rivest, and D. Wagner. "Tweakable Block Ciphers." *Advances in Cryptology*- CRYPTO'02, LNSC 2442, pp. 31-46, 2002.
15. U. Maurer. "Indistinguishability of Random Systems." *Advances in Cryptology*- EUROCRYPT'02, LNCS 2332, pp. 110-132, 2002.
16. U. Maurer and K. Pietrzak. "Composition of Random Systems: When Two Weak Make One Strong." *Theory of Cryptography* - TCC'04, LNCS 2951, pp. 410-427, 2004.
17. K. Pietrzak. "Composition Does Not Imply Adaptive Security." *Advances in Cryptology* - CRYPTO'05, LNCS 3621, pp. 55-65, 2005.
18. K. Pietrzak. "A Tight Bound for EMAC." *Automata, Languages and Programming, 33rd International Colloquium*, ICALP 2006, Proceedings, Part II. LNCS 4052, pp. 168-179, 2006.
19. P. Rogaway. "Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC." Full version of *Advances in Cryptology*- ASIACRYPT'04. LNCS 3329, pp. 16-31, 2004, `http://www.cs.ucdavis.edu/~rogaway/papers/offsets.pdf`, Sep.24, 2006.
20. M. Wegman and L. Carter. "New Hash Functions and Their Use in Authentication and Set Equality." *Journal of Computer and System Sciences*, Vol. 22, pp. 265-279, 1981.

A Lemmas from Maurer's Methodology

We describe some lemmas developed by Maurer (e.g., [15]) that we used. We assume that F and G are two random functions with the same input/output size; we define MESs $\mathcal{A} = a_0 a_1 \ldots$ and $\mathcal{B} = b_0 b_1 \ldots$ for F and G. The i-th input and output are denoted by X_i and Y_i for F (or G), respectively. Equality of (possibly conditional) probability distributions means equality as functions, i.e., equality holds for all possible arguments. For example, we write $P^F_{Y^i|X^i a_i} = P^G_{Y^i|X^i b_i}$ to mean $P^F_{Y^i|X^i a_i}(y^i, x^i) = P^G_{Y^i|X^i b_i}(y^i, x^i)$ for all (x^i, y^i), where $P^F_{a_i|X^i}(x^i)$ and $P^G_{b_i|X^i}(x^i)$ are positive. Inequalities, such as $P^F_{Y^i|X^i a_i} \le P^G_{Y^i|X^i b_i}$, are similarly defined.

Lemma 7. *(A corollary from Theorem* 1 (i)*, Lemma* 1 (iv)*, and Lemma* 4 (ii) *of [15]) Let $\mathbb{F}$ be the function of F or G (i.e., $\mathbb{F}[F]$ is a function that internally invokes F, possibly several times, to process its inputs). Here, $\mathbb{F}$ can be probabilistic, and, if so, $\mathbb{F}$ is independent of F or G. Suppose that $P^F_{Y_i|X^i Y^{i-1} a_i} = P^G_{Y_i|X^i Y^{i-1} b_i}$ and $P^F_{a_i|X^i Y^{i-1} a_{i-1}} \le P^G_{b_i|X^i Y^{i-1} b_{i-1}}$ holds for $i \ge 1$. We then have*

$$\mathtt{Adv}^{\mathtt{cpa}}_{F,G}(q) \le \nu(F, \overline{a_q}), \text{ and } \quad \mathtt{Adv}^{\mathtt{cpa}}_{\mathbb{F}[F],\mathbb{F}[G]}(q) \le \nu(\mathbb{F}[F], \overline{a^*_q}).$$

Here, MES $\mathcal{A}^* = a^*_0 a^*_1 \ldots$ is defined such that a^*_i denotes $\mathcal{A}$-event is satisfied for time period i. For example, if $\mathbb{F}[F]$ always invokes F k times for any input, then $a^*_i \equiv a_{ki}$.

Lemma 8. *(Theorem* 2 *of [15]) If $P^F_{a_i|X^i Y^{i-1} a_{i-1}} = P^F_{a_i|X^i a_{i-1}}$ holds for $i \ge 1$, the maximum probabilities of $\overline{a_q}$ for all adaptive and non-adaptive attacks are the same, i.e., $\nu(F, \overline{a_q}) = \mu(F, \overline{a_q})$.*

Lemma 9. *(Lemma* 6 (iii) *of [15]) If MESs $\mathcal{A} = a_0 a_1 \ldots$ and $\mathcal{B} = b_0 b_1 \ldots$ are defined for F, we have $\nu(F, \overline{a_q \wedge b_q}) \le \nu(F, \overline{a_q}) + \nu(F, \overline{b_q})$.*

These lemmas are easily extended even if the adversary's parameter θ contains ℓ (or σ) in addition to q.

B Proof of The First and Second Inequalities of Lemma 3

We define two tweakable functions having n-bit input/output and tweak space $\mathcal{T} = \{1, .., 2^{n/2}\} \times \{0, 1, 2\}$. For any input $x \in \Sigma^n$ and tweak $t = (t_{[1]}, t_{[2]}) \in \mathcal{T}$,

$$\mathrm{XE}(t, x) \stackrel{\text{def}}{=} \mathsf{P}_n(x \oplus 2^{t_{[1]}} 3^{t_{[2]}} L), \text{ where } L = \mathsf{P}_n(0), \text{ and}$$

$$\widetilde{\mathrm{XE}}(t, x) \stackrel{\text{def}}{=} \begin{cases} \mathsf{P}_n(x \oplus 2^{t_{[1]}} 3^{t_{[2]}} L), & \text{if } t_{[2]} = 0, \text{ where } L = \mathsf{P}_n(0); \\ \widetilde{\mathsf{P}}_n((t_{[1]}, t_{[2]}), x), & \text{otherwise.} \end{cases}$$

In the definition of $\widetilde{\mathrm{XE}}$, P_n and $\widetilde{\mathsf{P}}_n$ are assumed to be independent. It is obvious that PMAC[P_n] and MPMAC can be realized by using XE and $\widetilde{\mathrm{XE}}$ in a black-box manner. We consider a game in which an adversary tries to distinguish XE from $\widetilde{\mathrm{XE}}$ using q queries. Note that a query is in $\mathcal{T}\times\Sigma^n$. Let $(T_i,X_i)\in\mathcal{T}\times\Sigma^n$ be the i-th query, and $Y_i\in\Sigma^n$ be the i-th output. In addition, let S_i be $X_i\oplus 2^{T_{i[1]}}3^{T_{i[2]}}L$, where $T_i=(T_{i[1]},T_{i[2]})$. For $\widetilde{\mathrm{XE}}$, S_i is defined as a dummy variable when $T_{i[2]}\neq 0$. We define the following two events:

$$
\begin{aligned}
a_i^* &\overset{\text{def}}{=} [S_j\neq S_k \text{ for all } (j,k)\in\xi(i)]\wedge[S_j\neq 0 \text{ for all } j=1,\ldots,i].\\
b_i^* &\overset{\text{def}}{=} [Y_j\neq Y_k \text{ for all } (j,k)\in\xi(i)]\wedge[Y_j\neq L \text{ for all } j=1,\ldots,i],\\
&\quad \text{where } \psi(i)\overset{\text{def}}{=}\{j:1\le j\le i, T_{j[2]}\in\{1,2\}\}, \text{ and}\\
&\quad \xi(i)\overset{\text{def}}{=}\{(j,k)\in\{1,\ldots,i\}^2: j\neq k, \text{ at least one of } j \text{ or } k \text{ is in } \psi(i)\}.
\end{aligned}
$$

Note that $\psi(i)$ and $\xi(i)$ depend on $\{T_{j[2]}\}_{j=1,\ldots,i}$. Clearly, $\mathcal{A}^*=a_0^*a_1^*\ldots$ and $\mathcal{B}^*=b_0^*b_1^*\ldots$ are MESs and they are equivalent in XE (but not in $\widetilde{\mathrm{XE}}$). Also, note that $\mathcal{A}^*$ and $\mathcal{B}^*$ defined for XE ($\widetilde{\mathrm{XE}}$) are compatible with MESs defined for PMAC[P_n] (MPMAC). For example, if one uses XE to simulate an attack against PMAC[P_n] and observes a_i^* in time period i, $a_{i'}$ is occurring for the i'-th query to PMAC[P_n], for some $i'\le i$. We then have

$$
P^{\mathrm{XE}}_{Y_i|X^iT^iY^{i-1}a_i^*}=\sum P^{\mathrm{XE}}_{Y_i|LX^iT^iY^{i-1}}\cdot P^{\mathrm{XE}}_{L|X^iT^iY^{i-1}a_i^*}, \tag{25}
$$

where the summation is taken for all $L\in\Gamma(x^i,t^i,y^{i-1})$, which is the set of $L=c$ such that the rightmost term is non-zero. The equality of Eq. (25) holds since S^i is completely determined if X^i and L are fixed.

We focus on the rightmost two terms of Eq. (25) for some fixed $X^i=x^i$, $T^i=t^i$, $Y^{i-1}=y^{i-1}$ satisfying b_{i-1}^*, and $L=c\in\Gamma(x^i,t^i,y^{i-1})$ (thus $S^i=s^i$ is also fixed). It is clear that $P^{\mathrm{XE}}_{L|X^iT^iY^{i-1}a_i^*}(c,x^i,t^i,y^{i-1})$ is the uniform distribution over $\Gamma(x^i,t^i,y^{i-1})$. The conditional probability $P^{\mathrm{XE}}_{Y_i|LX^iT^iY^{i-1}}(y_i,c,x^i,t^i,y^{i-1})$ is 1 if $y_i=y_j$, and $i\notin\psi(i)$ and $\exists j\notin\psi(i)$ such that $s_i=s_j$. If $i\in\psi(i)$ or $i\notin\psi(i)$ but $s_i\neq s_j$ for $j\le i-1$, Y_i is uniform over $\Sigma^n\setminus\{y_1,\ldots,y_{i-1},c\}$.

Similarly, for $\widetilde{\mathrm{XE}}$, we have

$$
P^{\widetilde{\mathrm{XE}}}_{Y_i|X^iT^iY^{i-1}a_i^*b_i^*}=\sum P^{\widetilde{\mathrm{XE}}}_{Y_i|LX^iT^iY^{i-1}b_i^*}\cdot P^{\widetilde{\mathrm{XE}}}_{L|X^iT^iY^{i-1}a_i^*b_{i-1}^*}, \tag{26}
$$

where the summation is taken for all $L\in\Gamma(x^i,t^i,y^{i-1})$. Then, a simple case analysis shows that

$$
P^{\mathrm{XE}}_{Y_i|X^iT^iY^{i-1}a_i^*}=P^{\widetilde{\mathrm{XE}}}_{Y_i|X^iT^iY^{i-1}a_i^*b_i^*}. \tag{27}
$$

Moreover, we have

$$
P^{\mathrm{XE}}_{a_i^*|X^iT^iY^{i-1}a_{i-1}^*}=\sum P^{\mathrm{XE}}_{a_i^*|LX^iT^iY^{i-1}a_{i-1}^*}\cdot P^{\mathrm{XE}}_{L|X^iT^iY^{i-1}a_{i-1}^*}, \text{ and} \tag{28}
$$

$$
P^{\widetilde{\mathrm{XE}}}_{a_i^*b_i^*|X^iT^iY^{i-1}a_{i-1}^*b_{i-1}^*}=\sum P^{\widetilde{\mathrm{XE}}}_{a_i^*b_i^*|LX^iT^iY^{i-1}a_{i-1}^*b_{i-1}^*}\cdot P^{\widetilde{\mathrm{XE}}}_{L|X^iT^iY^{i-1}a_{i-1}^*b_{i-1}^*}, \tag{29}
$$

where the summations are taken for all $L \in \Gamma'(x^{i-1}, t^{i-1}, y^{i-1})$, which is the set of $L = c$ such that the last term of Eq. (28) (or Eq. (29)) is non-zero. It is easy to find that the last terms of Eqs. (28) and (29) are the same conditional distributions. However, we have $P^{\widetilde{\mathrm{XE}}}_{a_i^* b_i^* | L X^i T^i Y^{i-1} a_{i-1}^* b_{i-1}^*} \leq P^{\mathrm{XE}}_{a_i^* | L X^i T^i Y^{i-1} a_{i-1}^*}$ since both sides are 0 if $L \notin \Gamma(x^i, t^i, y^{i-1})$, and otherwise the R.H.S. is 1. Thus we have

$$P^{\widetilde{\mathrm{XE}}}_{a_i^* b_i^* | X^i T^i Y^{i-1} a_{i-1}^* b_{i-1}^*} \leq P^{\mathrm{XE}}_{a_i^* | X^i T^i Y^{i-1} a_{i-1}^*}. \tag{30}$$

From Eqs. (27) and (30) and the second claim of Lemma 7, the first inequality of Lemma 3 is proved. The second follows from the first and Lemma 9.

C Proof of Lemma 4

Note that $\mathbf{M}^{(i)}$ ($\mathbf{C}^{(i)}$) denotes the set of P_n inputs (outputs) generated in PHASH up to the i-th query. Let $\mathbf{Z}^{(i)}$ be the set of random variables $(L, \mathbf{C}^{(i)})$. If $\mathbf{Z}^{(i)}$ and X^i are fixed, $\mathbf{M}^{(i)}$, V^i, and S^i are uniquely determined. We have

$$\begin{aligned} P^{\mathrm{MPMAC}}_{a_i b_i | X^i Y^{i-1} a_{i-1} b_{i-1}} = \sum_{\mathbf{Z}^{(i)}} & P^{\mathrm{MPMAC}}_{b_i | \mathbf{Z}^{(i)} X^i Y^{i-1} a_i b_{i-1}} \\ & \cdot P^{\mathrm{MPMAC}}_{a_i | \mathbf{Z}^{(i)} X^i Y^{i-1} a_{i-1} b_{i-1}} \cdot P^{\mathrm{MPMAC}}_{\mathbf{Z}^{(i)} | X^i Y^{i-1} a_{i-1} b_{i-1}}, \end{aligned} \tag{31}$$

where the summations are taken for all $\mathbf{Z}^{(i)} = \mathbf{z}^{(i)}$ such that $(\mathbf{z}^{(i)}, x^i)$ satisfies a_{i-1}. Note that a_i implies that, if the j-th and j'-th tweaks (recall that the i-th tweak is a function of X_i) are the same, $V_j \neq V_{j'}$ holds for all $j, j' \leq i$, $j \neq j'$. From this, $P^{\mathrm{MPMAC}}_{b_i | \mathbf{Z}^{(i)} X^i Y^{i-1} a_i b_{i-1}}(\mathbf{z}^{(i)}, x^i, y^{i-1})$ does not depend on y^{i-1}, and it is $(2^n - (i-1))/(2^n - \pi(x^i))$, where $\pi(x^i)$ is the number of indices $j \in \{1, \ldots, i-1\}$ such that the j-th and i-th tweaks are the same. Moreover, $P^{\mathrm{MPMAC}}_{a_i | \mathbf{Z}^{(i)} X^i Y^{i-1} a_{i-1} b_{i-1}}(\mathbf{z}^{(i)}, x^i, y^{i-1})$ does not depend on y^{i-1} as it is 1 if $(\mathbf{z}^{(i)}, x^i)$ satisfies a_i, and otherwise 0. Finally, it is easy to see that $P^{\mathrm{MPMAC}}_{\mathbf{Z}^{(i)} | X^i Y^{i-1} a_{i-1} b_{i-1}}$ equals $P^{\mathrm{MPMAC}}_{\mathbf{Z}^{(i)} | X^i a_{i-1}}$ (here, b_{i-1} implies $V_j \neq V_{j'}$ whenever j-th and j'-th tweaks are the same, however, this is already implied by a_{i-1}). Thus, $P^{\mathrm{MPMAC}}_{a_i b_i | X^i Y^{i-1} a_{i-1} b_{i-1}}$ does not depend on y^{i-1}, and, for any x^i and $\widehat{y}^{i-1}$ satisfying b_{i-1}, we have

$$\begin{aligned} P^{\mathrm{MPMAC}}_{a_i b_i | X^i a_{i-1} b_{i-1}}(x^i) &= \sum P^{\mathrm{MPMAC}}_{a_i b_i | X^i Y^{i-1} a_{i-1} b_{i-1}}(x^i, y^{i-1}) \cdot P^{\mathrm{MPMAC}}_{Y^{i-1} | X^i a_{i-1} b_{i-1}}(y^{i-1}, x^i), \\ &= P^{\mathrm{MPMAC}}_{a_i b_i | X^i Y^{i-1} a_{i-1} b_{i-1}}(x^i, \widehat{y}^{i-1}) \sum P^{\mathrm{MPMAC}}_{Y^{i-1} | X^i a_{i-1} b_{i-1}}(y^{i-1}, x^i), \\ &= P^{\mathrm{MPMAC}}_{a_i b_i | X^i Y^{i-1} a_{i-1} b_{i-1}}(x^i, \widehat{y}^{i-1}), \end{aligned} \tag{32}$$

where the summations are taken for all y^{i-1} satisfying b_{i-1}. From this and Lemma 8, we prove the first claim of Lemma 4. The second follows from the first and the union bound.