Suprita Talnikar

Design, Analysis of Security and Cryptanalysis of Message Authentication Codes

Doctoral Thesis
to achieve the degree of
Doctor of Philosophy in Computer Science

submitted to

Indian Statistical Institute, Kolkata

Supervisor
Prof. Mridul Nandi

Applied Statistics Unit, Applied Statistics Division

December 2022

Related Papers

The following published/submitted papers are related to the work presented in this thesis:

1. Dutta, A., Nandi, M., Talnikar, S. (2019). *Beyond Birthday Bound Secure MAC in Faulty Nonce Model*. In: Ishai, Y., Rijmen, V. (eds) Advances in Cryptology – EUROCRYPT 2019. EUROCRYPT 2019. Lecture Notes in Computer Science(), vol 11476. Springer, Cham.
   https://doi.org/10.1007/978-3-030-17653-2_15

2. Chakraborti, A., Nandi, M., Talnikar, S., Yasuda, K. (2020). *On the Composition of Single-Keyed Tweakable Even-Mansour for Achieving BBB Security*. IACR Transactions on Symmetric Cryptology, 2020(2), 1–39.
   https://doi.org/10.13154/tosc.v2020.i2.1-39

3. Dutta, A., Nandi, M., Talnikar, S. (2021). *Permutation Based EDM: An Inverse Free BBB Secure PRF*. IACR Trans. Symmetric Cryptol. 2021 (2): 31-70.
   https://doi.org/10.46586/tosc.v2021.i2.31-70

4. Dutta, A., Nandi, M., Talnikar, S. *Tight Security Analysis of the Public Permutation-Based PMAC_Plus*. eprint 2022/905.
   https://eprint.iacr.org/2022/905.pdf
   Submitted to Advances in Mathematics of Communications, 2022

5. Datta, N., Dutta, A., Nandi, M., Talnikar, S. (2022). *Tight Multi-User Security Bound of DbHtS*. eprint 2022/689.
   https://eprint.iacr.org/2022/689.pdf
   Submitted to IACR Transactions on Symmetric Cryptology, 2023(1)

Acknowledgment

First and foremost, I would like to express my deepest gratitude to my Ph.D. supervisor, Professor Mridul Nandi for his invaluable guidance throughout my Ph.D duration, and for his sustained patience and feedback. I would also like to express my sincere gratitude to other professors of our unit — Prof. Bimal Kumar Roy, Prof. Subhamoy Maitra, Prof. Rana Barua, Prof. Palash Sarkar and Prof. Kishan Chand Gupta — for their support and useful feedback through annual evaluations on my research. My special thanks to my senior Dr. Avijit Dutta for his unwavering patience and continued guidance throughout the duration of my Ph.D. I could not have undertaken this journey without him or my other seniors Dr. Avik Chakraborti, Dr. Nilanjan Datta, Dr. Ashwin Jha and Dr. Ritam Bhaumik. I must also deeply thank Dr. Kan Yasuda for collaborating with me on one part of my research.

I am also grateful to my colleagues and office mates Mr. Anandarup Roy, Mr. Soumya Chattopadhyay, Mr. Chandranan Dhar, Mr. Anik Raychaudhuri, Mr. Bishwajit Chakraborty, Ms. Snehal Mitragotri, Mr. Sayantan Paul, Mr. Abishanka Saha, Mr. Arghya Bhattacharjee, Mr. Avishek Majumder, Mr. Samir Kundu and others for our numerous discussion sessions, tea breaks and for their moral support. Thanks should also go to all the staff of the Applied Statistics Unit, ISI Kolkata for their constant help with all official, administrative and other relevant tasks.

I must certainly mention my sister, Dr. Anushree Talnikar and my friends, especially Mrs. Richa Gokhale, who provided me with much-needed constant moral support and inspiration. Lastly, I would be remiss in not mentioning my parents, Mr. Subhash Talnikar and Mrs. Priti Talnikar for believing in my decision to undertake this huge endeavor and succeed despite numerous roadblocks.

Abstract

This thesis is a compilation of various message authentication codes having beyond the birthday bound (BBB) security. Kicking off with preliminary development in chapter 1, it proceeds to introduce the nEHtM (nonce-based Enhanced Hash-then-Mask) MAC in chapter 2, which is BBB-secure when nonce misuse occurs, through the concept of faulty nonces. The construction is based on a single block cipher, used on the inputs after they undergo a domain-separation. Next, chapter 3 tackles the security and cryptanalysis of MAC constructions that use pseudorandom permutations as primitives by introducing the construction PDMMAC (Permutation-based Davies-Meyer MAC) and its variants. The work on obtaining pseudorandom functions from PRPs by [53] lead to our exploration of PRP-based MACs, and one of our constructions was inspired by the DWCDM of [62]. This was instrumental in the search for an inverse-free permutation-based MAC with a single instance of permutation. This is addressed in chapter 4 through the p-EDM (permutation-based Encrypted Davies-Meyer), which follows the trend of constructing n-bit to n-bit PRFs by summing smaller constructions such as the Even-Mansour and the Davies-Meyer, like the SoEM and SoKAC constructions of [53] and the PDMMAC and variant constructions of [47] before it. The BBB security is again tight.

Two interesting treatments of the DbHtS construction [61] can be found in chapters 5 and 6. A permutation-based version, dubbed p-DbHtS (permutation-based Double-block Hash-then-Sum) is proven to possess BBB security and a matching attack provided. Finally, a block cipher-based version of the original construction is shown to have BBB security in the multi-user setting for underlying hash functions that are constructed without the use of block ciphers.

Furthermore, each chapter extends Patarin's Mirror Theory to provide partial bounds for solutions to a system of affine bivariate equations and non-equations satisfying certain conditions.

Contents

1. Introduction

1.1. Background

A Pseudo-Random Function (PRF) is a fundamental primitive in symmetric key cryptography. It is useful in providing solutions like authentication of messages, encryption of any arbitrary-length messages, etc. Most PRFs are built on top of a block cipher in some mode of operation. Some commonly used block cipher-based PRFs are CBC-MAC [13], PMAC [35], OMAC [89], LightMAC [100], etc. However, all of these block cipher-based PRF constructions only provide security up to around $2^{n/2}$ adversarial queries (where n is the block size of the block cipher). This bound is typically known as the **birthday bound**.

Birthday bound-secure constructions are often acceptable in practice when they are instantiated with block ciphers having a large block size (e.g., AES-128). For example, consider PMAC, whose PRF advantage is roughly $5\ell q^2/2^n$ [117], where ℓ is the upper limit on message size in terms of the number of blocks. When it is instantiated with AES-128, it gives a security of roughly up to 2^{48} adversarial queries, provided the longest message size is 2^{16} blocks and the success probability of breaking the scheme is restricted to 2^{-10}. However, with the growing trend of designing and standardizing lightweight block ciphers (NIST lightweight competition) like PRESENT [38], GIFT [7], LED [82, 81], etc. that are particularly suitable for a resource-constrained environment, birthday bound-secure constructions are no longer as suitable for use in practice. For example, PMAC instantiated with the PRESENT block cipher (a 64-bit block cipher) gives security up to 2^{16} adversarial queries when the longest message size is 2^{16} blocks and the success probability of breaking the scheme is 2^{-10}. Thus, it is not safe to use birthday bound-secure PRFs when they are instantiated with lightweight block ciphers. Although using AES-128 in a birthday bound-secure mode provides 64-bit security (which is adequate for the present-day), it may not be so in the future due to technological advancement. In such a situation, the feasible option would be to use a mode that gives higher security instead of replacing the underlying cipher with a larger block size.

An authenticated encryption (AE) mode is a cryptographic scheme that guarantees the privacy and authenticity of a message concurrently. Authenticated encryption has received much attention from the cryptographic community mostly due to its application to TLS and many other protocols. The recently concluded CAESAR competition [43] which aimed to identify a portfolio of authenticated encryption schemes drafted three use cases, namely *lightweight*, *high-performance*, and *defense-in-depth*. The competition considered GCM [101] as the baseline algorithm as it is widely adopted (e.g. in TLS 1.2 and in its variant RGCM [18], which shall soon be considered in TLS 1.3 [5]) and standardized. ChaCha20+Poly1305 [19] is a popular alternative for settings where AES-NI is not implemented.

Encrypt-then-MAC. Both ChaCha20+Poly1305 and GCM follow the Encrypt-then-MAC (EtM) paradigm [15]. Some other popular AE designs following the same paradigm are CWC [93], OGCM2 [3], CHM [88], CIP [87], GCM-RUP [4], OGCM1 [137], OGCM2 [137] etc. The authenticated encryption of this paradigm is described as follows. Let $\mathcal{E}$ be a nonce-based encryption scheme and $\mathcal{I}$ be a message authentication code. Given a nonce N, a message M and an associated data A, the ciphertext $C = \mathcal{E}^N(M)$ is first computed, which is then used to compute the tag $T = \mathcal{I}(N, A, C)$. All the aforementioned algorithms can be described by an encryption $\mathcal{E}^N$ (involving stream cipher encryptions) and a MAC $\mathcal{I}$ (all constructions are algebraic hash function-based and most of them uses Wegman-Carter MAC (WC MAC) [133]). EtM is a popular design paradigm due to its generic security guarantee. Authors of [44] showed that (stating informally) if $\mathcal{E}$ is a secure symmetric encryption scheme and $\mathcal{I}$ is a secure MAC family then this method of implementing EtM results in secure channels. This was later also analyzed by [15, 116].

Chapter 2 introduces the block-cipher based n-EHtM (nonce-based Enhanced Hash-then-Mask) MAC, which follows the EtM paradigm. It is shown to have beyond the birthday bound security that degrades gracefully with increasing number of faulty nonces. An AE instantiation of n-EHtM with CWC is also described. Chapter 3 tackles the security and cryptanalysis of MAC constructions that use pseudorandom permutations as primitives by introducing the construction PDMMAC (Permutation-based Davies-Meyer MAC) and its variants, while chapter 4 follows through with the inverse-free p-EDM (permutation-based Encrypted Davies-Meyer) — both shown to possess tight BBB security.

Hash-then-PRF [129]. HtP is a well-known paradigm for designing variable input-length PRFs, in which an input message of arbitrary length is hashed and the hash value is encrypted through a PRF to obtain a short tag. Most popular MACs including the CBC-MAC [13], PMAC [35], OMAC [89] and LightMAC [100] are designed using the HtP paradigm. Although the method is simple, in particular being deterministic and stateless, the security of MACs following the HtP paradigm is capped at the birthday bound due to the collision probability of the hash function. Birthday bound-secure constructions are acceptable in practice when any of these MACs are instantiated with a block cipher of moderately large block size. For example, instantiating PMAC with AES-128 permits roughly 2^{48} queries (using $5\ell q^2/2^n$ [117] bound) when the longest message size is 2^{16} blocks, and the success probability of breaking the scheme is restricted to 2^{-10}. However, the same construction becomes vulnerable if instantiated with some lightweight (smaller block size) block ciphers, whose number has grown tremendously in recent years, e.g. PRESENT [38], GIFT [7], LED [81], etc. For example, PMAC, when instantiated with the PRESENT block cipher (a 64-bit block cipher), permits

only about 2^{16} queries when the longest message size is 2^{16} blocks, and the probability of breaking the scheme is 2^{-10}. Therefore, it becomes risky to use birthday bound-secure constructions instantiated with lightweight block ciphers. In fact, in a large number of financial sectors, web browsers still widely use 64-bit block ciphers 3-DES instead of AES in their legacy applications with backward compatibility feature, as using the latter in corporate mainframe computers is more expensive. However, it does not give adequate security if the mode in which 3-DES is used provides only birthday bound security, and hence a beyond birthday secure mode solves the issue. Although many secure practical applications use the standard AES-128, which provides 64-bit security in a birthday bound-secure mode, which is adequate for the current technology, it may not remain so in the near future. In such a situation, using a mode with beyond the birthday bound security instead of replacing the cipher with a larger block size is a better option. [1]

Double-Block Hash-then-Sum. Many studies tried to tweak the HtP design paradigm to obtain beyond the birthday bound-secure MACs; while they possess a similar structural design, the internal state of the hash function is doubled and the two n-bit hash values are first encrypted and then XORed together to produce the output. In [135], Yasuda proposed a beyond the birthday bound-secure deterministic MAC called SUM-ECBC, a rate-1/2 sequential mode of construction with four block cipher keys. Followed by this work, Yasuda [134] came up with another deterministic MAC called PMAC_Plus, but unlike SUM-ECBC, PMAC_Plus is a rate-1 parallel mode of construction with three block cipher keys. Zhang et al. [136] proposed another rate-1 beyond the birthday bound-secure deterministic MAC called 3kf9 with three block cipher keys. In [114], Naito proposed LightMAC_Plus, a rate $(1 - s/n)$ parallel mode of operation, where s is the size of the block counter. The structural design of all these constructions first applies a $2n$-bit hash function on the message, then the two n-bit output values are encrypted and XORed together to produce the tag, where n is the block size of the block cipher. Moreover, all of them also give $2n/3$-bit security. In FSE 2019, Datta et al. [61] proposed a generic design paradigm dubbed as the double-block hash-then-sum or DbHtS, defined as follows:

$$\mathsf{DbHtS}(M) := \mathsf{E}_{K_1}(\Sigma) \oplus \mathsf{E}_{K_2}(\Theta), \quad (\Sigma, \Theta) \leftarrow \mathsf{H}_{K_h}(M),$$

where H_{K_h} is a double-block hash function that maps an arbitrary-length string to a $2n$-bit string. Within this unified framework, they revisited the security proof of existing DbHtS constructions, including PolyMAC [92], SUM-ECBC [135], PMAC_Plus [134], 3kf9 [136] and LightMAC_Plus [114] and also their two-keyed versions [61] and confirmed that all the constructions are

[1] Note that there are no standard block ciphers of size higher than 128 bits.

secure up to $2^{2n/3}$ queries when they are instantiated with an n-bit block cipher.

In CRYPTO 2018, Leurent et al. [95] proposed a generic attack on all these constructions using $2^{3n/4}$ (short message) queries, leaving a gap between the upper and the lower bounds for the provable security of DbHtS constructions. Recently, Kim et al. [92] have improved the bound of DbHtS constructions from $2^{2n/3}$ to $2^{3n/4}$. They have shown that if the underlying $2n$-bit hash function is the concatenation of two independent n-bit-universal hash functions [2], then the resulting DbHtS paradigm is secure up to $2^{3n/4}$ queries. They have also improved the security bound of PMAC_Plus, 3kf9 and LightMAC_Plus from $2^{2n/3}$ to $2^{3n/4}$ and hence closed the gap between the upper and the lower bounds of the provable security of DbHtS constructions.

Chapters 5 and 6 give tight security bounds for two different versions of the DbHtS construction. First, chapter 5 proposes the p-DbHtS (permutation-based Double-block Hash-then-Sum), which is proven BBB secure in the random oracle model, along with a matching attack that proves tightness of the bound. Next, chapter 6 considers a version of the original (block cipher-based) construction in the ideal cipher model and proves tight BBB multi-user security (with matching attack) for cases when the underlying hash function is not constructed from block ciphers.

1.2. Preliminaries and Notation

The set of all n-bit binary strings is denoted by $\{0,1\}^n$, for an integer $n \in \mathbb{N}$. The empty set shall be denoted by ϕ. For a set $\mathcal{X}$, $X \xleftarrow{\$} \mathcal{X}$ means X is sampled uniformly at random from $\mathcal{X}$, independently of all other random variables defined so far. The set of all functions from $\mathcal{X}$ to $\mathcal{Y}$ is denoted as $\mathsf{Func}(\mathcal{X},\mathcal{Y})$ and the set of all permutations over $\mathcal{X}$ is denoted as $\mathsf{Perm}(\mathcal{X})$. $\mathsf{Func}(\mathcal{X})$ denotes the set of all functions from $\mathcal{X}$ to $\{0,1\}^n$ and Perm denotes the set of all permutations over $\{0,1\}^n$. We often write Func instead of $\mathsf{Func}(\mathcal{X})$ when the domain of functions is understood from the context. Let $\mathcal{Z}_1 = (z_1^1,\ldots,z_q^1)$ and $\mathcal{Z}_2 = (z_1^2,\ldots,z_q^2)$ be two finite q-tuples containing n-bit strings z_i^b $(b \in \{1,2\}, i \in [q])$. When an n-bit permutation $\pi \in \mathsf{Perm}(n)$ maps $\mathcal{Z}_1$ to $\mathcal{Z}_2$, we shall write $\mathcal{Z}_1 \xrightarrow{\pi} \mathcal{Z}_2$ if $\forall\ i \in [q]$, $\pi(z_i^1) = z_i^2$. We say $\mathcal{Z}_1$ is *permutation compatible* with $\mathcal{Z}_2$ if there exists at least one $\pi \in \mathsf{Perm}(n)$ such that $\mathcal{Z}_1 \xrightarrow{\pi} \mathcal{Z}_2$. For integers $1 \le b \le a$, the notation $(a)_b$ means $a(a-1)\ldots(a-b+1)$, and $(a)_0 := 1$. $[q]$ refers to the set $\{1,\ldots,q\}$ and $[q_1,q_2]$ to the set $\{q_1,q_1+1\ldots,q_2-1,q_2\}$.

[2] A family of keyed hash functions is said to be universal if for any distinct x and x', the probability of a collision in their hash values for a randomly sampled hash function from the family is negligible.

Given a tuple of ordered pairs $\mathcal{Q} = ((x_1, y_1), \ldots, (x_q, y_q))$ with pairwise distinct n-bit strings $\{x_i\}_{i=1}^{q}$ and $\{y_i\}_{i=1}^{q}$, $\mathsf{Dom}(\mathcal{Q}) := \{x_i \in \{0,1\}^n : (x_i, y_i) \in \mathcal{Q}\}$ and $\mathsf{Ran}(\mathcal{Q}) := \{y_i \in \{0,1\}^n : (x_i, y_i) \in \mathcal{Q}\}$. Clearly, $|\mathsf{Dom}(\mathcal{Q})| = |\mathsf{Ran}(\mathcal{Q})| = q$. We say that an n-bit permutation $\pi \in \mathsf{Perm}(n)$ *extends* $\mathcal{Q}$ (i.e. $\pi \longrightarrow \mathcal{Q}$) if $\forall\ i \in [q], \pi(x_i) = y_i$. We say that $\mathcal{Q}$ is *extendable* if there exists at least one $\pi \in \mathsf{Perm}(n)$ such that $\pi \longrightarrow \mathcal{Q}$.

We generalize this notion for more than one tuple of ordered pairs. Let $\widetilde{\mathcal{Q}} = (\mathcal{Q}_1, \ldots, \mathcal{Q}_s)$ such that for each $j \in [s]$, $\mathcal{Q}_j = ((x_1^j, y_1^j), \ldots, (x_{q_j}^j, y_{q_j}^j))$, where each x_i^j and each y_i^j is an n-bit string pairwise distinct from all others. Now for each $j \in [s]$, we define the following two sets: $\mathsf{Dom}(\mathcal{Q}_j) = \{x_i^j : (x_i^j, y_i^j) \in \mathcal{Q}_j\}$ and $\mathsf{Ran}(\mathcal{Q}_j) = \{y_i^j : (x_i^j, y_i^j) \in \mathcal{Q}_j\}$. Clearly, $|\mathsf{Dom}(\mathcal{Q}_j)| = |\mathsf{Ran}(\mathcal{Q}_j)| = q_j\ \forall\ j \in [s]$. Moreover, for all $j \neq j' \in [s]$, $\mathsf{Dom}(\mathcal{Q}_j)$ is disjoint from $\mathsf{Dom}(\mathcal{Q}_{j'})$ and $\mathsf{Ran}(\mathcal{Q}_j)$ is disjoint from $\mathsf{Ran}(\mathcal{Q}_{j'})$. Hence, $\mathfrak{X} = (\mathsf{Dom}(\mathcal{Q}_1), \ldots, \mathsf{Dom}(\mathcal{Q}_s))$ and $\mathfrak{Y} = (\mathsf{Ran}(\mathcal{Q}_1), \ldots, \mathsf{Ran}(\mathcal{Q}_s))$ are two disjoint collections of finite sets. An n-bit permutation $\pi \in \mathsf{Perm}(n)$ thus *extends* $\widetilde{\mathcal{Q}}$, denoted $\pi \longrightarrow \widetilde{\mathcal{Q}}$, if $\pi \longrightarrow \mathcal{Q}_j\ \forall\ j \in [s]$. As an alternative notation of $\pi \longrightarrow \widetilde{\mathcal{Q}}$, we also write $\mathfrak{X} \xrightarrow{\pi} \mathfrak{Y}$.

We write $x \leftarrow y$ to represent the value y being assigned to the variable x. We say two sets $\mathcal{X}$ and $\mathcal{Y}$ are disjoint if $\mathcal{X} \cap \mathcal{Y} = \phi$. We denote their union as $\mathcal{X} \sqcup \mathcal{Y}$ (i.e. the *disjoint union* of $\mathcal{X}$ and $\mathcal{Y}$). Let $\mathfrak{X} = (\mathcal{X}_1, \ldots, \mathcal{X}_s)$ be a finite collection of finite sets. $\mathfrak{X}$ is called a *disjoint collection* if for each $j \neq j' \in [s]$, $\mathcal{X}_j$ and $\mathcal{X}_{j'}$ are disjoint. The size of $\mathfrak{X}$, denoted as $|\mathfrak{X}|$, is $|\mathcal{X}_1| + \ldots + |\mathcal{X}_s|$. For a disjoint collection $\mathfrak{X} = (\mathcal{X}_1, \ldots, \mathcal{X}_s, \mathcal{X}_{s+1})$, we write $\mathfrak{X} \setminus \mathcal{X}_{s+1}$ to denote the collection $(\mathcal{X}_1, \ldots, \mathcal{X}_s)$. For two disjoint collections $\mathfrak{X} = (\mathcal{X}_1, \ldots, \mathcal{X}_s)$ and $\mathfrak{Y} = (\mathcal{Y}_1, \ldots, \mathcal{Y}_{s'})$, we say $\mathfrak{X}$ is *inter-disjoint* with $\mathfrak{Y}$ if for all $j \in [s], j' \in [s']$, $\mathcal{X}_j$ is disjoint from $\mathcal{Y}_{j'}$. If $\mathfrak{X}$ is inter-disjoint from $\mathfrak{Y}$, then we denote their union as $\mathfrak{X} \sqcup \mathfrak{Y}$. Moreover, $|\mathfrak{X} \sqcup \mathfrak{Y}| = |\mathfrak{X}| + |\mathfrak{Y}|$. For a set $\mathcal{S}$ and for a finite disjoint collection of finite sets $\mathfrak{X} = (\mathcal{X}_1, \ldots, \mathcal{X}_s)$, $\mathcal{S} \setminus \mathfrak{X}$ means $\mathcal{S} \setminus (\mathcal{X}_1 \sqcup \ldots \sqcup \mathcal{X}_s)$. For a finite subset $\mathcal{S}$ of $\mathbb{N}$, $\max \mathcal{S}$ denotes the maximum-valued element of $\mathcal{S}$. For a finite set $\mathcal{X} \subseteq \{0,1\}^n$ and for an arbitrary non-zero element $a \in \{0,1\}^n$, $\mathcal{X} \oplus a$ denotes the set $\{x \oplus a : x \in \mathcal{X}\}$.

A function Φ is said to be a *block function* if it maps elements from an arbitrary domain to $\{0,1\}^n$. The set of all block functions with domain $\mathcal{X}$ is denoted as $\mathsf{Func}(\mathcal{X})$. [3] We call Φ to be a *double-block function* if it maps elements from an arbitrary set $\mathcal{X}$ to $(\{0,1\}^n)^2$. For a given double-block function $\Phi : \mathcal{X} \to \{0,1\}^{2n}$, we write $\Phi_\ell : \mathcal{D} \to \{0,1\}^n$ such that for every $x \in \mathcal{X}$, $\Phi_\ell(x) = \mathsf{left}(\Phi(x))$. Similarly, we write $\Phi_{\mathbf{r}} : \mathcal{X} \to \{0,1\}^n$ such that for every $x \in \mathcal{X}$, $\Phi_{\mathbf{r}}(x) = \mathsf{right}(\Phi(x))$. For two block functions $\Phi_\ell : \mathcal{X} \to \{0,1\}^n$ and $\Phi_{\mathbf{r}} : \mathcal{X} \to \{0,1\}^n$, one can naturally define a double-block function

[3] When $\mathcal{X} = \{0,1\}^n$, we write Func to denote $\mathsf{Func}(\{0,1\}^n)$.

$\Phi : \mathcal{X} \to \{0,1\}^{2n}$ such that $\Phi(x) = (\Phi_\ell(x), \Phi_\mathbf{r}(x))$, which we write as $\Phi = (\Phi_\ell, \Phi_\mathbf{r})$.

$\{0,1\}^n$ denotes the set of all binary strings of length n and $\{0,1\}^*$ denotes the set of all binary strings of arbitrary finite length. This text may sometimes denote $|\{0,1\}^n|$ by N. We denote 0^n (i.e., the n-bit string of zeroes) by $\mathbf{0}$. For any element $X \in \{0,1\}^*$, $|X|$ denotes the number of bits in X. For any two elements $X, Y \in \{0,1\}^*$, $X\|Y$ denotes the concatenation of X to Y. For $X, Y \in \{0,1\}^n$, $X \oplus Y$ denotes the addition of X and Y, modulo 2. For any $X \in \{0,1\}^*$, parse X as $X = X_1\|X_2\|\dots\|X_l$, where for each $i = 1, \dots, l-1$, X_i is an element of $\{0,1\}^n$ and $1 \le |X_l| \le n$. We call each X_i a *block*. For a pair of blocks $x = (x_\ell, x_\mathbf{r}) \in \{0,1\}^{2n}$, we write $\mathsf{left}(x)$ to denote x_ℓ and $\mathsf{right}(x)$ to denote $x_\mathbf{r}$. For $x \in \{0,1\}^n$, where $x = x_{n-1}\|\dots\|x_0$, $\mathsf{lsb}(x)$ denotes the least significant bit x_0 of x. A function $\mathsf{chop}_{\mathsf{LSB}} : \{0,1\}^n \to \{0,1\}^{n-1}$ removes the least significant bit of a string $X \in \{0,1\}^n$. For a given bit b, fix_b is a function from $\{0,1\}^n$ to $\{0,1\}^n$ that takes an n-bit binary string $x = x_{n-1}\|\dots\|x_0$ and returns an another binary string $x' = (x_{n-1}\|\dots\|b)$, where $\mathsf{lsb}(x)$ is fixed to bit b. For a tuple $X := (X_1, \dots, X_q)$ of length q, an element X_i of X is called *fresh* if for all $j \neq i$, $X_i \neq X_j$. Otherwise, we say X_i is *not fresh* or *repeated* in X. Sometimes we denote tuple X as $(X_i)_{i\in[q]}$. X is said to be *distinct* if each of its elements is fresh. Otherwise, we say it is *not a fresh tuple*.

Lazy Sampling of Random Permutations. Consider a distinguisher A interacting with an n-bit random permutation $\pi \xleftarrow{\$} \{0,1\}^n$. We simulate this interaction by a simulator $\mathcal{S}$ that maintains a partial function Ψ. Ψ is initially defined to be an empty function (a function with empty domain), i.e., $\mathsf{Dom}(\Psi) \leftarrow \phi$. We consider two dynamically growing sets $\mathsf{Dom}(\Psi)$ and $\mathsf{Ran}(\Psi)$ associated to Ψ, such that the points at which Ψ has already been defined gets included in $\mathsf{Dom}(\Psi)$ and their respective defined values get included in $\mathsf{Ran}(\Psi)$. Initially, $\mathsf{Dom}(\Psi), \mathsf{Ran}(\Psi) \leftarrow \phi$. On the i^{th} query x_i, the simulator checks whether $x_i \in \mathsf{Dom}(\Psi)$. If so, the corresponding response is $y_i \leftarrow \Psi(x_i)$. Else, the response is sampled uniformly from $\{0,1\}^n \setminus \mathsf{Ran}(\Psi)$ and x_i, y_i are added to $\mathsf{Dom}(\Psi)$ and $\mathsf{Ran}(\Psi)$ respectively, i.e.

$$\mathsf{Dom}(\Psi) \leftarrow \mathsf{Dom}(\Psi) \cup \{x_i\},\ \mathsf{Ran}(\Psi) \leftarrow \mathsf{Ran}(\Psi) \cup \{y_i\}.$$

Note that at any point, $\mathsf{Dom}(\Psi), \mathsf{Ran}(\Psi) \subseteq \{0,1\}^n$.

1.2.1. Cryptographic Security Models

The Standard Model

In the standard model, no special Mathematical objects such as infinite random strings or random oracles are used, and communication systems

are usually abstracted into a reliable but insecure channel. Even the most common encryption goals require some complexity-theoretic hardness assumptions in the standard model [34].

The Random Oracle Model

The random oracle model formalized by Bellare and Rogaway [16] assumes that a hash function is replaced by a publicly accessible random function (the random oracle). This means that the adversary cannot compute the result of the hash function by itself, and must query the random oracle [57]. This model often allows one to design very simple, intuitive and efficient protocols for many tasks, while simultaneously providing a seemingly convincing security guarantee for such practical constructions [66]. Thus, a proof in the random oracle model does not imply that the scheme will remain secure when the random oracle is replaced by a concrete hash function.

Upper Bound on $\mathsf{Adv}_f^{\mathsf{MAC}}$ (Page 5, [68])

To get an upper bound for $\mathsf{Adv}_f^{\mathsf{MAC}}$, we consider a random oracle $\psi \xleftarrow{\$} \mathsf{Func}(\mathcal{K} \times \mathcal{N} \times \mathcal{M}, \mathcal{T})$ and reject oracle $\mathsf{Rej} : \mathcal{N} \times \mathcal{M} \times \mathcal{T} \to \{0\}$. The advantage $\mathsf{Adv}_f^{\mathsf{MAC}}$ is bounded above by

$$\max_{\mathcal{D}} \left| \mathsf{Pr}[\mathcal{D}^{(f_K^\pi, \mathsf{Ver}_K^\pi, \pi, \pi^{-1})} = 1] - \mathsf{Pr}[\mathcal{D}^{(\psi, \mathsf{Rej}, \pi, \pi^{-1})} = 1] \right|.$$

The Ideal Permutation and Ideal Cipher Models

Instead of a publicly accessible random function, the ideal permutation model assumes the adversary's access to a random permutation in addition to the construction (or random) oracle(s). On the other hand, the ideal cipher model provides a publicly accessible random block cipher (or ideal cipher) with a k-bit key and an n-bit input. All parties including the adversary can make both encryption and decryption queries to the ideal block cipher, for any given key [57, 66]. [66] and [57] are works that relate the random oracle and ideal cipher models.

1.2.2. Security Definitions

Distinguishing Advantage An adversary A is modeled as a randomized algorithm with access to an external oracle $\mathcal{O}$. Such an adversary is called an *oracle adversary*. An oracle $\mathcal{O}$ is an algorithm that may be a cryptographic scheme being analyzed. The interaction between A and $\mathcal{O}$,

denoted by $\mathsf{A}^{\mathcal{O}}$, generates a transcript $\tau = \{(x_1, y_1), (x_2, y_2), \ldots, (x_q, y_q)\}$, where $x_1, x_2, \ldots, x_q$ are q queries of A to oracle $\mathcal{O}$ and $y_1, y_2, \ldots, y_q$ be the corresponding responses, where $y_i = \mathcal{O}(x_i)$. We assume that A is **adaptive**, which means that x_i is dependent on the previous $i-1$ responses.

Distinguishing Game. Let F and G be two random systems and an adversary A is given oracle access to either of F or G. After interaction with an oracle $\mathcal{O} \in \{\mathsf{F}, \mathsf{G}\}$, A outputs 1, which is denoted as $\mathsf{A}^{\mathcal{O}} \Rightarrow 1$. Such an adversary is called a *distinguisher* and the game is called a *distinguishing game*. The task of the distinguisher in a distinguishing game is to tell with which of the two systems it has interacted. The advantage of the distinguisher A in distinguishing the random system F from G is defined as

$$\mathbf{Adv}^{\mathsf{F}}_{\mathsf{G}}(\mathsf{A}) \coloneqq \mid \Pr[\mathsf{A}^{\mathsf{F}} \Rightarrow 1] - \Pr[\mathsf{A}^{\mathsf{G}} \Rightarrow 1] \mid,$$

here the above probability is defined over the probability spaces of A and $\mathcal{O}$. The maximum advantage in distinguishing F from G is defined as

$$\max_{\mathsf{A} \in \mathcal{A}} \mathbf{Adv}^{\mathsf{F}}_{\mathsf{G}}(\mathsf{A}),$$

where $\mathcal{A}$ is the class of all possible distinguishers. One can easily generalize this setting when the distinguisher interacts with multiple oracles, which are separated by commas. For example, $\mathbf{Adv}^{\mathsf{F}_1,\ldots,\mathsf{F}_m}_{\mathsf{G}_1,\ldots,\mathsf{G}_m}(\mathsf{A})$ denotes the advantage of A in distinguishing $(\mathsf{F}_1, \ldots, \mathsf{F}_m)$ from $(\mathsf{G}_1, \ldots, \mathsf{G}_m)$.

Pseudo Random Function (PRF) and Psuedo Random Permutation (PRP). A keyed function $\mathsf{F} : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ with key space $\mathcal{K}$, domain $\mathcal{X}$ and range $\mathcal{Y}$ is a function for which $\mathsf{F}(K, X)$ shall be denoted by $\mathsf{F}_K(X)$. Given an algorithm A that has oracle access to a function from $\mathcal{X}$ to $\mathcal{Y}$, makes at most q queries in time at most t, and returns a single bit, the prf advantage of A against the family of keyed functions F, $\mathbf{Adv}^{\mathrm{PRF}}_{\mathsf{F}}(\mathsf{A})$, is defined as

$$\left| \Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathsf{A}^{\mathsf{F}_K(\cdot)} = 1\right] - \Pr\left[\mathsf{RF} \xleftarrow{\$} \mathrm{Func}(\mathcal{X}, \mathcal{Y}) : \mathsf{A}^{\mathsf{RF}(\cdot)} = 1\right] \right|.$$

F is said to be a $(q, \ell, \sigma, t, \epsilon)$-secure PRF if

$$\mathbf{Adv}^{\mathrm{PRF}}_{\mathsf{F}}(q, \ell, \sigma, t) \coloneqq \max_{\mathsf{A}} \mathbf{Adv}^{\mathrm{PRF}}_{\mathsf{F}}(\mathsf{A}) \leq \epsilon,$$

where the maximum is taken over all adversaries A that make q queries, with a maximum of ℓ data blocks in a single query and the total number of data blocks at most σ, with maximum running time t. Similarly, the prp-advantage of A against a family of keyed permutations E is defined as

$$\mathbf{Adv}^{\mathrm{PRP}}_{\mathsf{E}}(\mathsf{A}) \coloneqq \left| \Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathsf{A}^{\mathsf{E}_K(\cdot)} = 1\right] - \Pr\left[\Pi \xleftarrow{\$} \mathrm{Perm}(\mathcal{X}) : \mathsf{A}^{\Pi(\cdot)} = 1\right] \right|.$$

E is said to be a (q, t, ϵ)-secure PRP if $\mathbf{Adv}^{\text{PRP}}_{\mathsf{E}}(q, t) := \max_{\mathsf{A}} \mathbf{Adv}^{\text{PRP}}_{\mathsf{E}}(\mathsf{A}) \leq \epsilon$, where maximum is taken over all adversaries A that make q queries and have running time at most t.

PRF Security in the Random Permutation Model. Consider a function $f : \mathcal{K} \times \mathcal{M} \to \mathcal{T}$, where $\mathcal{K}$, $\mathcal{M}$ and $\mathcal{T}$ are the key space, message space and the tag space respectively. We discuss the pseudorandom security of f under the random permutation model. We assume that f makes internal public-random-permutation calls to π and π^{-1} (f can make calls to multiple random permutations when all of them are independent and uniform on the set of message blocks $\mathsf{Perm}(\mathcal{B})$). For simplicity, we use f^{π}_{K} to denote f with uniform K and uniform π. The distinguisher $\mathcal{D}$ is given access to either $(f^{\pi}_{K}, \pi, \pi^{-1})$ for $K \xleftarrow{\$} \{0,1\}^k$ or (ψ, π, π^{-1}) where $\psi \xleftarrow{\$} \mathsf{Func}(\mathcal{K} \times \mathcal{M}, \mathcal{T})$ is a random oracle. The distinguishing probability of $\mathcal{D}$ is represented by $\mathsf{Adv}^{\mathsf{prf}}_{f}(\mathcal{D})$, such that

$$\mathsf{Adv}^{\mathsf{prf}}_{f}(\mathcal{D}) = |\mathsf{Pr}[\mathcal{D}^{(f^{\pi}_{K}, \pi, \pi^{-1})} = 1] - \mathsf{Pr}[\mathcal{D}^{(\psi, \pi, \pi^{-1})} = 1]|.$$

To be precise, we call f an ϵ-PRF against (q_m, p)-adversaries if $\mathrm{Adv}^{PRF}_{F}(D) \leq \epsilon$ for all distinguishers $\mathcal{D}$ making q_m queries to f^{π}_{K} and p offline queries to π.

MAC Security in the Random Permutation Model. Consider f and another function $\mathsf{Ver} : \mathcal{K} \times \mathcal{M} \times \mathcal{T} \to \{0,1\}$ (similar to f^{π}_{K}, we use the notation Ver^{π}_{K}) such that for (M, T), if $f^{\pi}_{K}(M) = T$ then $\mathsf{Ver}^{\pi}_{K}(M, T) = 1$ (otherwise $\mathsf{Ver}^{\pi}_{K}(M, T) = 0$). Consider a (q_m, p, q_v) adversary $\mathcal{A}$ making q_m queries to f^{π}_{K}, p queries to π and q_v queries to Ver^{π}_{K}. We say that $\mathcal{A}$ *forges* if any of its queries (M, T) to Ver^{π}_{K} returns 1, such that M has not been queried to f^{π}_{K} before. The advantage of $\mathcal{A}$ against the MAC security of f is defined as

$$\mathsf{Adv}^{\mathsf{MAC}}_{f}(\mathcal{A}) = \mathsf{Pr}[K \xleftarrow{\$} \mathcal{K}, \pi \xleftarrow{\$} \mathsf{Perm}(\mathcal{B}) : \ \mathcal{A} \textit{ forges}].$$

To be precise, we call f an ϵ-MAC against (q_m, p, q_v)-adversaries if $\mathsf{Adv}^{\mathsf{MAC}}_{f}(\mathcal{A}) \leq \epsilon$ for all adversaries $\mathcal{A}$ making q_m queries to f^{π}_{K}, p queries to π and q_v queries to Ver^{π}_{K}.

Nonce-Based MAC Security in the Random Permutation Model. Consider nonce based versions of f and Ver (takes an additional input $N \in \mathcal{N}$.) such that for an input (N, M, T), $\mathsf{Ver}^{\pi}_{K}(N, M, T) = 1$ if $f^{\pi}_{K}(N, M) = T$ and 0 otherwise. Consider a (q_m, p, q_v) adversary $\mathcal{A}$ making q_m queries to f^{π}_{K} without repeating the nonce, p queries to π and q_v queries to Ver^{π}_{K}. We say that $\mathcal{A}$ *forges* if any of its queries (N, M, T) to Ver^{π}_{K}, such that (N, M) has not been queried to f^{π}_{K}, returns 1. The advantage of $\mathcal{A}$ against the MAC security of f is defined as

$$\mathsf{Adv}^{\mathsf{MAC}}_{f}(\mathcal{A}) = \mathsf{Pr}[K \xleftarrow{\$} \mathcal{K}, \pi \xleftarrow{\$} \mathsf{Perm}(\mathcal{B}) : \ \mathcal{A} \textit{ forges}].$$

We call f an ϵ-MAC against (q_m, p, q_v)-adversaries if $\mathsf{Adv}_f^{\mathsf{MAC}}(\mathcal{A}) \leq \epsilon$ for all (q_m, p, q_v)-adversaries $\mathcal{A}$.

PRF Security in the Ideal-Cipher Model A *keyed function* with the key space $\mathcal{K}$, domain $\mathcal{X}$ and range $\mathcal{Y}$ is a function $\mathsf{F} : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$. We denote $\mathsf{F}(k, x)$ by $\mathsf{F}_k(x)$. A random function RF from $\mathcal{X}$ to $\mathcal{Y}$ is a uniform random variable over the set $\mathsf{Func}(\mathcal{X}, \mathcal{Y})$, i.e., $\mathsf{RF} \xleftarrow{\$} \mathsf{Func}(\mathcal{X}, \mathcal{Y})$. We define the pseudorandom security of F under the ideal-cipher model. We assume that F makes internal calls to a publicly evaluated block cipher E with a randomly sampled block cipher key $K \xleftarrow{\$} \mathcal{K}$ (F can make calls to multiple block ciphers when all of them are independent and uniform over the set $\mathsf{BC}(\mathcal{K}, \{0,1\}^n)$). For simplicity, we write $\mathsf{F}_K^{\mathsf{E}}$ to denote F with a uniformly sampled block cipher $\mathsf{E} \xleftarrow{\$} \mathsf{BC}(\mathcal{K}, \{0,1\}^n)$, which is keyed by a randomly sampled $K \xleftarrow{\$} \mathcal{K}$. The distinguisher A is given access to either $(\mathsf{F}_K^{\mathsf{E}}, \mathsf{E}^{\pm})$ for $K \xleftarrow{\$} \mathcal{K}$ or $(\mathsf{RF}, \mathsf{E}^{\pm})$, where $\mathsf{E} \xleftarrow{\$} \mathsf{BC}(\mathcal{K}, \{0,1\}^n)$ is a uniformly sampled n-bit block cipher such that A can make forward or inverse queries to E, which is denoted as $\mathsf{E}^{\pm}$. We define the PRF advantage of A against a keyed function F in the ideal cipher model as

$$\mathbf{Adv}_{\mathsf{F}}^{\mathrm{PRF}}(\mathsf{A}) := \mathbf{Adv}_{(\mathsf{RF},\mathsf{E}^{\pm})}^{(\mathsf{F}_K^{\mathsf{E}},\mathsf{E}^{\pm})}(\mathsf{A}).$$

We say F is a (q, p, ϵ, t)-PRF if $\mathbf{Adv}_{\mathsf{F}}^{\mathrm{PRF}}(\mathsf{A}) \leq \epsilon$ for all adversaries A that make q queries to F, p forward and inverse offline queries to E and run for time at most t.

Multi-User PRF Security in Ideal Cipher Model We assume there are u users in the multi-user setting, such that the i^{th} user executes $\mathsf{F}_{K_i}^{\mathsf{E}}$. Furthermore, the i^{th} user key K_i is independent of the keys of all other users. An adversary A has access to all the u users as oracles. A make queries to the oracles in the form of (i, M) to the i^{th} user and obtains $T \leftarrow \mathsf{F}_{K_i}^{\mathsf{E}}(M)$. We call these **construction queries**. For $i \in [u]$, we assume A makes q_i queries to the i^{th} oracle. We also assume that A make queries to the underlying block cipher E and its inverse with some chosen keys k^j. We call these **primitive queries**. Suppose A chooses s distinct block cipher keys $(k^1, \ldots, k^s)$ and makes p_j primitive queries to the block cipher E with chosen keys k^j for $1 \leq j \leq s$. Let A be a (u, q, p, t)-adversary against the PRF security of F for all u users such that $q = q_1 + \ldots + q_u$ is the total number of construction queries and $p = p_1 + \ldots + p_s$ is the total number of primitive queries to the block cipher E with the total running time A being at most t. We assume that for any $i \in [u]$, A does not repeat any construction query to the i^{th} user. Similarly, A does not repeat any primitive query for any chosen block cipher key k^j to the block cipher E. The advantage of A in distinguishing $(\mathsf{F}^{\mathsf{E}}, \mathsf{E}^{\pm})$ from $(\mathsf{RF}, \mathsf{E}^{\pm})$ in the multi-user setting, where $\mathsf{RF} \xleftarrow{\$} \mathsf{Func}(\mathcal{X}, \mathcal{Y})$, is defined

as

$$\mathbf{Adv}_{\mathsf{F}}^{\text{mu-PRF}}(\mathsf{A}) := \left|\Pr\left[\mathsf{A}^{((\mathsf{F}_{K_1}^{\mathsf{E}},\ldots,\mathsf{F}_{K_u}^{\mathsf{E}}),\mathsf{E}^{\pm})} \Rightarrow 1\right] - \Pr\left[\mathsf{A}^{((\mathsf{RF},\ldots,\mathsf{RF}),\mathsf{E}^{\pm})} \Rightarrow 1\right]\right|,$$

where the randomness is defined over $K_1,\ldots,K_u \xleftarrow{\$} \mathcal{K}$, $\mathsf{E} \xleftarrow{\$} \mathsf{BC}(\mathcal{K},\{0,1\}^n)$ and the randomness of the adversary (if any). We write

$$\mathbf{Adv}_{\mathsf{F}}^{\text{mu-PRF}}(u,q,p,\mathsf{t}) := \max_{\mathsf{A}} \mathbf{Adv}_{\mathsf{F}}^{\text{mu-PRF}}(\mathsf{A}),$$

where the maximum is over all (u,q,p,t)-adversaries A. In this chapter, we skip the time parameter of the adversary as we shall assume that the adversary is computationally unbounded. This also leads to the assumption that the adversary is deterministic. When $u = 1$, it makes $\mathbf{Adv}_{\mathsf{F}}^{\text{mu-PRF}}(u,q,p,\mathsf{t})$ the single-user distinguishing advantage.

1.2.3. Keyed Hash

Security of a Keyed Hash Function Let $\mathcal{K}_h$ and $\mathcal{X}$ be two non-empty finite sets. A keyed function $\mathsf{H} : \mathcal{K}_h \times \mathcal{X} \to \{0,1\}^n$ is ϵ-almost-XOR universal (AXU) if for any distinct $x, x' \in \mathcal{X}$ and for any $\Delta \in \{0,1\}^n$,

$$\Pr[K_h \xleftarrow{\$} \mathcal{K}_h : \mathsf{H}_{K_h}(x) \oplus \mathsf{H}_{K_h}(x') = \Delta] \leq \epsilon_{\text{axu}}.$$

Moreover, H is an ϵ-universal hash function if for any distinct $x, x' \in \mathcal{X}$,

$$\Pr[K_h \xleftarrow{\$} \mathcal{K}_h : \mathsf{H}_{K_h}(x) = \mathsf{H}_{K_h}(x')] \leq \epsilon_{\text{univ}}.$$

A keyed hash function is said to be ϵ-regular if for any $x \in \mathcal{X}$ and for any $\Delta \in \{0,1\}^n$,

$$\Pr[K_h \xleftarrow{\$} \mathcal{K}_h : \mathsf{H}_{K_h}(x) = \Delta] \leq \epsilon_{\text{reg}}.$$

Regular Hash. A function $\mathcal{H} : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is said to be an ϵ-regular hash function if $\forall\ d \in \mathcal{D}$ and $r \in \mathcal{R}$,

$$\Pr_{K_h \xleftarrow{\$} \mathcal{K}}[\mathcal{H}(K_h, d) = r] \leq \epsilon.$$

AXU Hash. A function $\mathcal{H} : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is said to be an ϵ-AXU hash function if for two distinct d and d' from $\mathcal{D}$ and $r \in \mathcal{R}$,

$$\Pr_{K_h \xleftarrow{\$} \mathcal{K}}[\mathcal{H}(K_h, d) \oplus \mathcal{H}(K_h, d') = r] \leq \epsilon.$$

(d, d') is called a *colliding pair* for a function $\mathcal{H}(K_h, *)$ (or $\mathcal{H}_{K_h}(*)$) if $\mathcal{H}(K_h, d) = \mathcal{H}(K_h, d')$.

3-Way Regular Hash. A function $\mathcal{H} : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is said to be an ϵ-3-way regular hash function if for three distinct d, d' and d'' from $\mathcal{D}$ and for any non-zero r from $\mathcal{R}$,

$$\Pr_{K_h \xleftarrow{\$} \mathcal{K}} [\mathcal{H}(K_h, d) \oplus \mathcal{H}(K_h, d') \oplus \mathcal{H}(K_h, d'') = r] \leq \epsilon.$$

An example of 3-way regular hash is Poly hash (with the secret key K_h) where the padded message $x^* = x_1 \| \cdots \| x_\ell$ is processed as

$$\mathsf{Poly}_{K_h}(x^*) = x_\ell \cdot K_h \oplus x_{\ell-1} \cdot K_h^2 \oplus \cdots \oplus x_1 \cdot K_h^\ell.$$

1.3. Cryptographic Encryption and Authentication

1.3.1. Message Encryption

The process of message encryption converts the original plaintext into an incomprehensible ciphertext. Ideally, only authorized parties should be able to decrypt the ciphertext back into plaintext and gain access to the original data. Thus, encryption serves as a mechanism to ensure confidentiality. A pseudorandom encryption key produced by a *key-generation algorithm* is typically used in an encryption system. The encryption and decryption keys are identical in symmetric-key schemes. This secret key must be held by the sender as well as the receiver in order to achieve secure communication. The sender uses this key in an *encryption algorithm* to generate the ciphertext from the plaintext and send it to the receiver. Then the receiver uses it in a *decryption algorithm* to again obtain the plaintext. Some typical examples of symmetric-key encryption schemes include AES, 3-DES, and SNOW.

1.3.2. Message Authenticated Codes

Let $\mathcal{K}, \mathcal{N}, \mathcal{M}$ and $\mathcal{T}$ be four non-empty finite sets and $\mathsf{F} : \mathcal{K} \times \mathcal{N} \times \mathcal{M} \to \mathcal{T}$ a nonce-based MAC. For $K \in \mathcal{K}$, let Auth_K be the authentication oracle (which takes as input $(N, M) \in \mathcal{N} \times \mathcal{M}$ and outputs $T = \mathsf{F}(K, N, M)$) and let Ver_K be the verification oracle (which takes as input $(N, M, T) \in \mathcal{N} \times \mathcal{M} \times \mathcal{T}$ and outputs 1 if $\mathsf{F}(K, N, M) = T$ and 0 otherwise). An authentication query (N, M) by an adversary A is called a *faulty query* if A has already queried to the first oracle with the same nonce and a different message.

A (μ, q_m, q_v, t)-adversary against the unforgeability of F is an adversary A with oracle access to Auth_K and Ver_K such that it makes at most μ faulty authentication queries out of at most q_m authentication queries, and q_v

verification queries, with running time at most t. The adversary is said to be *nonce respecting* if $\mu = 0$ and *nonce misusing* if $\mu \geq 1$; any number of nonce repetitions in verification queries is allowed. A is said to *forge* F if for any of its verification queries (not obtained through a previous authentication query), the verification oracle returns 1. The forging advantage of A against F is defined as

$$\mathbf{Adv}^{\mathrm{MAC}}_{\mathsf{F}}(\mathsf{A}) := \Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathsf{A}^{\mathsf{Auth}_K(\cdot,\cdot),\mathsf{Ver}_K(\cdot,\cdot,\cdot)} \text{ forges}\right].$$

We write $\mathbf{Adv}^{\mathrm{MAC}}_{\mathsf{F}}(\mu, q_m, q_v, t) := \max_{\mathsf{A}} \mathbf{Adv}^{\mathrm{MAC}}_{\mathsf{F}}(\mathsf{A})$, where the maximum is taken over all (μ, q_m, q_v, t)-adversaries. In all of these definitions, we skip the parameter t, whenever we maximize over all unbounded adversaries.

1.3.3. Authenticated Encryption

An authenticated encryption (AE) mode is a cryptographic scheme that guarantees the privacy and authenticity of a message concurrently. Authenticated encryption has received much attention from the cryptographic community mostly due to its application to TLS and many other protocols. The recently concluded CAESAR competition [43], which aimed to identify a portfolio of authenticated encryption schemes, had drafted three use cases namely *lightweight*, *high-performance*, and *defense-in-depth*. The competition considered GCM [101] as the baseline algorithm as it is widely adopted (e.g. in TLS 1.2 and in its variant RGCM [18], which shall soon be considered in TLS 1.3 [5]) and standardized. ChaCha20+Poly1305 [19] is a popular alternative for settings where AES-NI is not implemented.

Encrypt-then-MAC. Both ChaCha20+Poly1305 and GCM follow the Encrypt-then-MAC (EtM) paradigm [15]. Some other popular AE designs following the same paradigm are CWC [93], GCM/2^+ [3], CHM [88], CIP [87], GCM-RUP [4], OGCM1 [137], OGCM2 [137], etc. The authenticated encryption of this paradigm is described as follows: Let $\mathcal{E}$ be a nonce-based encryption scheme and $\mathcal{I}$ be a message authentication code. Given a nonce N, a message M and an associated data A, the ciphertext $C = \mathcal{E}^N(M)$ is first computed, which is then used to compute the tag $T = \mathcal{I}(N, A, C)$. All the aforementioned algorithms can be described by an encryption $\mathcal{E}^N$ (involving stream cipher encryptions) and a MAC $\mathcal{I}$ (all constructions are algebraic hash function-based and most of them use the Wegman-Carter MAC (WC) [133]). EtM is a popular design paradigm due to its generic security guarantee. Informally stating a result in [44], if $\mathcal{E}$ is a secure symmetric encryption scheme and $\mathcal{I}$ is a secure MAC family then this method of implementing EtM results in secure channels. This was later also analyzed by [15, 116].

1.4. Cryptographic Primitives

1.4.1. PRPs and PRFs

In their seminal work, Luby and Rackoff [97] showed how to construct a keyed pseudorandom permutation (PRP) – i.e. a block cipher – from secret keyed pseudorandom functions (PRFs). Their work was a theoretical model for formally arguing the security of the DES block cipher, which consists of r rounds of Feistel constructions invoking independent instances of keyed functions.

PRP-Based PRFs. The most obvious way a PRF can be constructed is to consider a PRP π (a popular choice is an n-bit block cipher with a uniformly sampled key e_K for some integer n) itself as a PRF. However, this leads to an $n/2$-bit secure PRF. This result comes from the fact that $2^{n/2}$ evaluations of the PRF will lead to a collision with significant probability while the collision probability in case of a PRP will be zero. This is also termed the PRP-PRF switching lemma [13, 17, 49, 84]. In light of the recent research in lightweight cryptography, this bound may not be acceptable to designers. The value of n is generally chosen to be small because the state size of the PRF directly depends on n and lightweight designs aim to optimize it. For example, several lightweight block ciphers [8, 41, 9, 38, 7] that are proposed with a 64-bit state (i.e, $n = 64$) achieve only 32-bit security and can be broken with practical query complexity. This idea has resulted in several attempts to design a PRF from a PRP with more than $n/2$-bit security. They are popularly known as *Beyond-the-Birthday-bound* (BBB)-secure PRFs.

A first attempt to construct such a BBB-secure PRF denoted by $\mathsf{Trunc}_m(e_K(x))$ was proposed by Hall et al. [84], where $m < n$ (note that a block cipher is a popular candidate for a PRP), and its security was bounded by $2^{n-m/2}$ queries [10, 77]. Later in [14], Bellare et al. proposed n-bit security [10, 60, 98] of $e_{K_1}(x) \oplus e_{K_2}(x)$ where K_1 and K_2 are independently sampled. Seurin et al. proposed a $2^{2n/3}$ query-secure PRF, which they called EDM [55],

$$e_{K_2}(e_{K_1}(x) \oplus x).$$

The security of this construction has been improved by Mennink [103] using Patarin's mirror theory [113, 122, 120, 121]. Note that all constructions are deterministic (no use of nonce) and are instantiated with block ciphers with inputs considered to be of fixed length. However, there are a number of BBB-secure constructions that deal with arbitrary length inputs.

The usual technique is to incorporate a nonce and a keyed hash. The nonce is processed with a deterministic PRF and the output is properly integrated with the hashed value of the arbitrary length message. Excepting a few, most PRFs do not allow nonce misuse. The WC-MAC [45, 133] (Wegman-Carter MAC) is one such construction, where the nonce is processed with

a PRP-based PRF and a universal hash processes the message. Next, both the outputs are added and passed through another instance of a PRP to generate the output. This design is vulnerable to nonce misuse but secure only up to the birthday bound when respecting the nonce. Later, Cogliati and Seurin updated the WC MAC and designed the EWC-MAC [55] (Encrypted Wegman-Carter): $e_{K_2}(f_{K_1}(x) \oplus \mathcal{H}_{K_h}(x))$ (f is a deterministic PRF, $\mathcal{H}$ is a key universal hash and K_1, K_2 and K_h are uniform and independent), which is birthday bound secure under both nonce misuse and respect scenario (can be proved using the PRP-PRF switching lemma). The most important question that arises is *"How can a BBB secure PRF be designed?"* The first prominent design in this area is the EWCDM construction:

$$e_{K_2}((e_{K_1}(N) \oplus N) \oplus \mathcal{H}_{K_h}(x))$$

by Cogliati et al. [55], where the PRF is instantiated by Davis-Meyer and is used in the EWC mode. This design achieves BBB security of $2n/3$-bits (though n-bit security was conjectured and proved by Mennink et al. [103] using mirror theory) under nonce respect and birthday bound under nonce misuse. However, this construction is not minimal in structure as it uses two independent instances of keys K_1 and K_2. Datta et al. in [62] proposed DWCDM, which is a BBB secure construction (under nonce respect), and uses only one instance of the PRP (where e_{K_2} is replaced by $e^{-1}_{K_1}$). In the security proof, the authors extended mirror theory and provided a concrete proof of security up to $2^{2n/3}$ queries under nonce respecting conditions and birthday bound complexity under nonce misuse. Nevertheless, the bound is not tight as there does not exist any attack below 2^n queries. In fact the design is conjectured to have n-bit security.

1.4.2. Block Ciphers

A block cipher $\mathsf{E} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ is a function that takes a key $k \in \mathcal{K}$ and an n-bit input data $x \in \{0,1\}^n$ and produces an n-bit output y such that for each key $k \in \mathcal{K}$, $\mathsf{E}(k,\cdot)$ is a permutation over $\{0,1\}^n$. $\mathcal{K}$ is called the key space of the block cipher and $\{0,1\}^n$ is its input-output space. In shorthand notation, we write $\mathsf{E}_k(x)$ to represent $\mathsf{E}(k,x)$. Let $\mathsf{BC}(\mathcal{K},\{0,1\}^n)$ denotes the set of all n-bit block ciphers with key space $\mathcal{K}$. We say that a block cipher E is an (q,ϵ,t)-secure strong pseudorandom permutation, if for all distinguishers A that make a total of q forward and inverse queries with run time at most t, the following holds:

$$\mathbf{Adv}^{\mathsf{E}}_{\Pi}(\mathsf{A}) := \mid \Pr[K \xleftarrow{\$} \mathcal{K} : \mathsf{A}^{\mathsf{E}_K} \Rightarrow 1] - \Pr[\Pi \xleftarrow{\$} \mathsf{Perm} : \mathsf{A}^{\Pi} \Rightarrow 1] \mid \leq \epsilon.$$

1.4.3. Block cipher-based PRFs

EDM: Encrypted Davis-Meyer. It encrypts the output of Davis-Meyer:

$$\mathsf{EDM}_{K_1,K_2}[e] := e_{K_2}(\mathsf{DM}_{K_1}[e](x)).$$

EDM is a $2n/3$-bit BBB secure PRF. The query complexity of the attack against EDM is $\mathcal{O}(2^n)$ query complexity. Hence, the security bound is not tight. Later, Mennink proposed the dual of EDM defined as $\mathsf{DM}_{K_2}[e](e_{K_1}(x))$. This design achieves the same security bound as EDM but the bound is not tight. It has even been proven to be n-bit secure using mirror theory. The proof is not verified and the attack complexity is again up to $\mathcal{O}(2^n)$ queries.
DDM: Decrypted Davis Meyer. DDM optimizes EDM in the number of block cipher instances. In other words DDM replaces the outer e_{K_2} by $e_{K_1}^{-1}$. Formally,

$$\mathsf{DDM}_K[e] := e_K^{-1}(\mathsf{DM}_K(x)).$$

The proven security bound of DDM is exactly the same as EDM. However, this bound is not known to be tight and is accompanied by an attack with $\mathcal{O}(2^n)$ queries.
EWCDM: All the constructions above can handle fixed length inputs.
EWCDM [56] extends the input domain of EDM to handle multi-block inputs. It takes a nonce $N \in \mathcal{N}$ and an input $x \in \mathcal{M}$ (where $\mathcal{M}$ is the set of all multi-block inputs) to generate a tag $T \in \mathcal{T}$. $\mathsf{EWCDM}_{K_1,K_2,K_h}[e,\mathcal{H}]$ with N and x as the inputs is defined as

$$T = e_{K_2}(e_{K_1}(N) \oplus N \oplus \mathcal{H}_{K_h}(x)).$$

Here, $\mathcal{H}$ is ϵ_1-regular hash, ϵ_2-AXU hash and ϵ_3-3-way regular hash. For Poly hash, we have $\epsilon_1 = \epsilon_2 = \epsilon_3 = \frac{\ell}{2^n}$.
DWCDM: In CRYPTO 2018 [62], Datta et al. proposed DWCDM which optimizes EWCDM the number of block cipher instances to one without any compromise in the security level.

It takes a nonce $N \in \mathcal{N}$ and an input $x \in \mathcal{M}$ ($\mathcal{M}$ is the set of all multi-block inputs) to generate a tag $T \in \mathcal{T}$. $\mathsf{DWCDM}_{K,K_h}[e,e^{-1},\mathcal{H}]$ with N and x is defined as

$$T = e_K^{-1}(e_K(N) \oplus N \oplus \mathcal{H}_{K_h}(x)).$$

Here, the last $n/3$-bits of N are 0 and $\mathcal{H}$ is ϵ_1-regular hash, ϵ_2-AXU hash and ϵ_3-3-way regular hash. For Poly hash, we have $\epsilon_1 = \epsilon_2 = \epsilon_3 = \frac{\ell}{2^n}$.

1.4.4. Permutation-Based Cryptography

All the PRFs discussed so far are built using block ciphers as their underlying primitive. As block ciphers are designed to be efficient in the forward as

well as the inverse direction, they are over-engineered as primitives for such purposes [53]. At the other extreme, cryptographic public permutations are particularly designed to be fast in the forward direction, but not necessarily in the inverse. Examples of such permutations include Keccak [23], Gimli [20], SPONGENT [40], etc. In most cases, evaluating an unkeyed public permutation is faster than evaluating a keyed block cipher, as the latter involves evaluating the underlying key scheduling algorithm each time the block cipher is invoked in the design. [4] Moreover, we do not need to store the round keys in permutation-based designs, and designing a permutation is usually simpler than designing a block cipher. In this regard, we quote Bertoni et al. [22]:

" . . . the inverse mapping of block ciphers imposes a separation of the processing of the $n + k$ bits of the input. The key is processed in a key schedule and the data in the data path, and there can be no diffusion from the data path to the key schedule, which strongly limits the potential diffusion . . . Such a restriction is not present in the design of cryptographic permutations as they do not make a distinction between the processing of key and data input as there is no specific key input."

Numerous public permutation-based inverse-free hash and authenticated encryption designs were proposed [127, 46, 115, 64, 20, 65, 48, 59] with the advent of public permutation-based designs and the advantage of their efficient evaluation in the forward direction. The use of cryptographic permutations gained momentum during the SHA-3 competition [127]. The selection of the permutation-based Keccak sponge function as the SHA-3 standard further gave the community a high level of confidence in this primitive. Permutation-based sponge constructions have presently become a successful and full-fledged alternative to the block cipher-based modes. In fact, in the first round of the ongoing NIST lightweight competition [119], 24 out of 57 submissions are based on cryptographic permutations. Of these 24, 16 permutation-based proposals have qualified for the second round. These statistics depict the wide adoption of permutation-based designs [46, 20, 26, 48, 59, 64] in the community.

The necessity of designing PRFs using PRPs as primitives of cryptographic designs [14] was therefore realized because we usually seek PRF security from a mode of operation and it is generally easier to design PRPs than PRFs. Designing a secure non-invertible round function that can be iterated multiple times to produce a secure PRF is also a big challenge. As collision probabilities are amplified with each iteration [30, 105], it is hard to correctly iterate a non-invertible round function multiple times. Although Mennink and Neves [105] designed a dedicated PRF called FastPRF from scratch, their

[4] While caching the round keys of the block cipher may seem to eliminate the problem, this requires more storage space than storing the master key of the block cipher, e.g., storing the round keys of AES-128 requires ten times more space than storing its master key.

design is based on grouping the round functions of a PRP. Moreover, there are plenty of cryptographic modes that do not require the invertibility of its underlying primitives [101, 35, 88, 133, 46, 26, 64, 61, 62, 56, 100]. As PRFs are designed to be efficient in both forward and inverse directions, the choice to use them over PRPs as the underlying primitive for realizing the PRF security of a mode of operation is a more economical one in such cases. The fact that the counter mode of encryption generally offers a better security guarantee when instantiated with a PRF over a PRP (one can distinguish the PRP-based counter mode from the random encryption with $2^{n/2}$ queries, where n is the block size of the PRP, while the PRF-based the counter mode behaves identically with the random encryption scheme modulo the PRF advantage of the keyed function) is substantial evidence of our argument.

Due to the classical PRF-PRP switching lemma [49, 12, 17], a PRP E_k can be replaced with a PRF F_k until the number of invocations to the primitive exceeds $2^{n/2}$, where n is the block size of the permutation. Such a solution is adequate when the block size of the permutation is large (e.g. AES 128). This may however, not be a good solution when the block size is small (e.g. 64 bits). This is particularly relevant when instantiating cryptographic schemes using lightweight block ciphers like PRESENT [38], GIFT [7], etc., whose block size is typically 64 bits. Consequently, using them as PRFs in cryptographic designs can ensure only 32 bits of security, which is not practical in terms of the present computational power. As a remedy, exploration of cryptographic designs that retain security even after invoking the primitive more than $2^{n/2}$ times ensued. Such designs are popularly known as *beyond the birthday bound* (BBB) secure designs. Wherefore, Hall et al. [84] proposed a BBB secure PRF called Truncation that takes an n-bit block cipher E_k and truncates the result to a bits. This construction was later proven secure up to $2^{n-a/2}$ queries [10, 77]. Bellare et al. [14] proposed the Sum of Permutations (SoP) construction that returns the XOR of the outputs of two n-bit independent permutations:

$$\mathsf{SoP}^{\pi_1,\pi_2}(x) := \pi_1(x) \oplus \pi_2(x).$$

This construction was proven secure up to $2^{2n/3}$ queries [98], and was recently shown as secure up to 2^n queries [60]. Cogliati and Seurin [56] proposed another candidate for a beyond the birthday bound secure PRF, which they call *Encrypted Davis Meyer* (EDM) and have shown to achieve $2n/3$-bit security:

$$\mathsf{EDM}^{\pi_1,\pi_2}(x) := \pi_2(\pi_1(x) \oplus x).$$

Later in [103], Mennink and Neves showed an optimal security of the construction. In the same paper, they also proposed a dual variant of EDM, which they called EDMD:

$$\mathsf{EDMD}^{\pi_1,\pi_2}(x) := \pi_2(\pi_1(x)) \oplus \pi_1(x),$$

and proved its optimal PRF security. However, both proofs of security are inherently based on a debated result of Mirror theory for a general $\xi_{\max}$ [71]. Guo et al. [79] proposed SUMPIP, a contender of SoP:

$$\mathsf{SUMPIP}^{\pi}(x) := \pi(x) \oplus \pi^{-1}(x).$$

Unlike the single permutation variant of SoP that takes an $n-1$-bit input, SUMPIP is the first single permutation-based PRF that takes an n-bit input and returns an n-bit output. Authors also show in this paper that the single permutation variants of EDM and EDMD achieve $2n/3$-bit security. Concurrently, Cogliati and Seurin [55] too showed $2n/3$-bit security for the single-keyed EDM construction. Very recently, Gunsing and Mennink [78] proposed a new approach to designing a block cipher-based PRF, called the *Summation-Truncation Hybrid* (STH) technique. STH takes an $(n-1)$-bit input x, truncates the leftmost a bits of $\mathsf{E}(x\|0), \mathsf{E}(x\|1)$, and sums the discarded $n-a$ bits of $\mathsf{E}(x\|0)$ and $\mathsf{E}(x\|1)$ to produce an $(n+a)$-bit output. They showed $2^{n-a/2}$ bits of security for the construction, where $n-a$ is the number of discarded bits.

1.4.5. Public Permutation-Based Pseudorandom Functions

Let $\mathsf{F} : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ be a keyed function with $\mathcal{K}, \mathcal{X}$ and $\mathcal{Y}$ respectively the key space, input space and output space. We assume that F makes internal calls to the public random permutations $\mathbf{P} = \{\pi_1, \ldots, \pi_d\}$ for $d \geq 1$, where all of the d permutations are independent and uniformly sampled from $\mathsf{Perm}(n)$ for some $n \in \mathbb{N}$. We write $\mathbf{P}^{-1} = \left\{\pi_1^{-1}, \ldots, \pi_d^{-1}\right\}$ to denote the d-tuple of inverse permutations. For simplicity, $\mathsf{F}_k^{\mathbf{P}}$ denotes F with uniform k and uniform $\mathbf{P}$.

A distinguisher D is given access to either the oracle $\mathsf{F}_k^{\mathbf{P}}$ in the real world or a random function RF that maps elements from $\mathcal{X}$ to $\mathcal{Y}$ in the ideal world. Apart from querying to either of these two oracles, D can also make queries to the permutations $\mathbf{P}$ and $\mathbf{P}^{-1}$ in both of these worlds. Queries of the former type, where the distinguisher is interacting with either $\mathsf{F}_k^{\mathbf{P}}$ or RF, are called *construction queries*, and queries of the latter type are called *primitive queries*. A primitive query to a permutation is called a *forward primitive query* and to the inverse of a permutation is called an *inverse primitive query*. The prf advantage of D against F in the public permutation model is defined as follows:

$$\mathbf{Adv}_{\mathsf{F}}^{\mathrm{prf}}(\mathsf{D}) := \left| \Pr\left[\mathsf{D}^{(\mathsf{F}_k^{\mathbf{P}}, \mathbf{P}, \mathbf{P}^{-1})} \Rightarrow 1\right] - \Pr\left[\mathsf{D}^{(\mathsf{RF}, \mathbf{P}, \mathbf{P}^{-1})} \Rightarrow 1\right] \right|.$$

Here, $\mathsf{D}^{\mathbf{O}} \Rightarrow 1$ represents the distinguisher D being given access to the oracle $\mathbf{O}$, with which it interacts and then outputs 1. This probability is over the

randomness of $k \xleftarrow{\$} \mathcal{K}$, $\pi_1, \ldots, \pi_d \xleftarrow{\$} \mathsf{Perm}(n)$ and that of the distinguisher (if any). We say that D is a (q, p, t)-distinguisher if D makes a total of q construction queries and p primitive queries, and runs in at most t steps.

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F}}(q, p, \mathsf{t}) := \max_{\mathsf{D}} \mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F}}(\mathsf{D}),$$

where the maximum is taken over all (q, p, t)-distinguishers D. This text skips the time parameter of the distinguisher as the assumption throughout shall be that the distinguisher is computationally unbounded, and hence deterministic.

1.4.6. Some More Examples of Permutation-based PRFs

SoEM: Sum of Even-Mansour. It is a permutation based PRF that uses two instances of EM to simply add them up to output the sum. Precisely,

$$\mathsf{SoEM}_{K_1,K_2}[\pi_1, \pi_2](x) := \mathsf{EM}_{K_1}[\pi_1](x) \oplus \mathsf{EM}_{K_2}[\pi_2](x).$$

SOEM has three instances denoted by

- SoEM1 with $\pi_1 = \pi_2$ and K_1, K_2 are independent,
- SoEM21 with π_1, π_2 are independent with $K_1 = K_2$ and
- SoEM22 with π_1, π_2 are independent and K_1, K_2 are independent.

Security of SoEM: Both SoEM1 and SoEM21 achieves the birthday bound security and associated with matching birthday attacks in query complexity. SoEM22 achieves BBB security of $2n/3$-bits with a matching attack in query complexity. Below, we will briefly discuss about the birthday bound attack on SoEM with a single random permutation (i.e, SoEM1). Note that, we use $\mathcal{O}(f(n))$ to denote $c \cdot f(n)$ computations, where c is a small constant. From now on, we use this notation throughout the chapter when needed.

Attack Idea: The attack exploits the parallel structure of SoEM as well the usage of the same permutation in both the branches. In other words, if the inputs to the two branches swap then the final outputs will collide. Such a structure of inputs (M, M') can be obtained using $\mathcal{O}(2^{n/2})$ queries by adjusting the left and the right half of the inputs. The condition on the choice of (M, M') is $M \oplus M' = K_1 \oplus K_2$. This condition can be easily detected as the output of the messages M and M' would be same (see Fig.1.1).

SoKAC: These are mainly Even-Mansour followed by Davis-Meyer type of constructions. More precisely,

$$\mathsf{SoKAC1}_{K_1,K_2}[\pi_1](x) := \mathsf{DM}_{K_1}[e](\mathsf{EM}_{K_1,K_2}[\pi_1](x))$$
$$\mathsf{SoKAC21}(x) := \mathsf{DM}_{K}[e'](\mathsf{EM}_{K}[\pi_1](x))$$

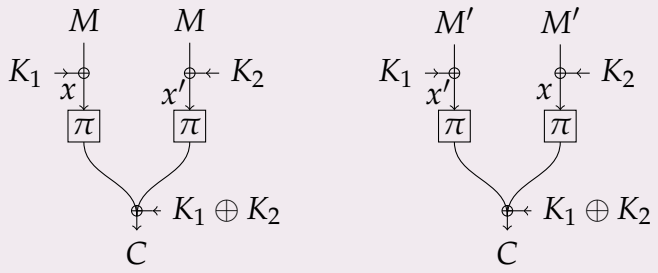


Figure 1.1.: SoEM1 - Swapping x and x'.

where $e_K(x) = \pi_1(x) \oplus K$ and $e'_K(x) = \pi_2(x) \oplus K$.

Proposition 5 in [53] claims that the same birthday bound attack as on SoEM1 can be applied to SoKAC1. Also, Proposition 6 of the same paper claims that the same beyond birthday bound attack as on SoEM21 can be applied to SoKAC21. We observe that the attacks possibly do not work with the claimed complexities. The main reason is the serial structure of SoKAC, wherein a fresh input to the first permutation π_1 makes the internal state random. Hence, an extended attack on SoKAC is unknown to us. Recently, Nandi proposed a birthday bound attack on SoKAC21 in [118], giving SoKAC21 a birthday bound security; a $2n/3$-bit security was claimed in Theorem 2 of [53]. Additionally, Fig. 1.3 presents an independent attack against SoKAC1 with $\mathcal{O}(2^{2n/3})$ query complexity.

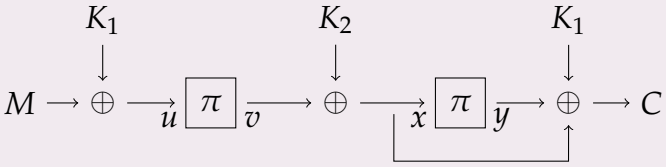


Figure 1.2.: SoKAC1 - One permutation instance π ($= \pi_1 = \pi_2$), two key instances K_1 and K_2.

Analysis of the attack: Observe that for the values $q = p_1 = p_2 = 2 \cdot 2^{2n/3}$, the set Ext_K has size $\mathcal{O}(1)$ with high probability, for each value $K \in \mathcal{K}$. Furthermore, if $K^\star$ denotes the true key of the construction, then $\Pr[K^\star \in \hat{\mathcal{K}}] = \Pr[|\mathrm{Ext}_{K^\star}| \geq 2] \geq \frac{1}{4}$, and thus, the expected size, $\mathsf{E}\left[\left|\hat{\mathcal{K}}\right|\right]$, of the guess-key set $\hat{\mathcal{K}}$ is $\mathcal{O}(1)$.

$5 \cdot 2^{2n/3}$-Query Attack on SoKAC1

1 : Make queries $M_1, \ldots, M_q \leftarrow \{0,1\}^n$ to the authentication oracle $\mathbf{O}$ with $q = 2 \cdot 2^{2n/3}$ $\Big($say $M_i = \langle i \rangle_{2n/3} \| 0^{n/3}$ for $i < 2^{2n/3}$,

$M_i = \langle i - 2^{2n/3} + 1 \rangle_{2n/3} \| 1 \| 0^{n/3-1}$ for $2^{2n/3} \leq i < 2 \cdot 2^{2n/3}\Big)$.

2 : Make $\tilde{u}_1, \ldots, \tilde{u}_{p_1}$ with $p_1 = 2 \cdot 2^{2n/3}$ forward queries to the primitive π $\Big($say $\tilde{u}_a = 0^{n/3} \| \langle a \rangle_{2n/3}$ for $a < 2^{2n/3}$,

$\tilde{u}_a = 0^{n/3-1} \| 1 \| \langle a - 2^{2n/3} + 1 \rangle_{2n/3}$ for $2^{2n/3} \leq a < 2 \cdot 2^{2n/3}\Big)$;

receive responses $\tilde{v}_a = \pi(\tilde{u}_a), a \in [p_1]$.

3 : Make $\tilde{y}_1, \ldots, \tilde{y}_{p_2} \xleftarrow[\text{wor}]{\$} \{0,1\}^n$ with $p_2 = 2 \cdot 2^{2n/3}$ backward primitive queries to the primitive π; receive responses $\tilde{x}_b, b \in [p_2]$.

4 : Set $\text{Ext}_K := \{(i,a,b) \in [q] \times [p_1] \times [p_2] :$ $(M_i \oplus \tilde{u}_a = K_1) \wedge (C_i \oplus \tilde{x}_b \oplus \tilde{y}_b = K_1)\}$ and set $\hat{\mathcal{K}} = \phi$.

5 : For all $K \in \mathcal{K}$ with $|\text{Ext}_K| \geq 2$, check whether :
For all pairs of tuples $(i,a,b) \neq (i',a',b')$ in Ext_K,
if $(\tilde{v}_a \oplus \tilde{x}_b \oplus \tilde{v}_{a'} \oplus \tilde{x}_{b'} = 0)$, then add K to $\hat{\mathcal{K}}$.

Figure 1.3.: Interaction of the adversary with $(\mathbf{O}, \pi)$, where $\mathbf{O}$ is either the random oracle or the real construction oracle SoKAC1_K^{π} and the primitive π.

1.5. The Coeffients-H Technique

System and Distinguisher. Consider a computationally unbounded distinguisher A (hence assumed deterministic) that interacts with either of the possibly randomized stateful systems $\mathbf{S}_{\text{re}}$ or $\mathbf{S}_{\text{id}}$, after which it returns a single bit 0 or 1. For any such system $\mathbf{S}_{\text{re}}$ or $\mathbf{S}_{\text{id}}$, the interaction between A and the system defines an ordered sequence of queries and responses, $\tau = ((X_1, Y_1), (X_2, Y_2), \ldots, (X_q, Y_q))$ called a *transcript*, where X_i is the i^{th} query of A and Y_i is the corresponding response from the system. Let X_{re} (resp. X_{id}) be the random variable that takes a transcript resulting from the interaction between A and $\mathbf{S}_{\text{re}}$ (resp. A and $\mathbf{S}_{\text{id}}$). Then the advantage of A in distinguishing $\mathbf{S}_{\text{re}}$ from $\mathbf{S}_{\text{id}}$ is bounded from above by the statistical distance

between the two random variables X_{re} and X_{id}, which is

$$\Delta(X_{re}, X_{id}) := \sum_{\tau} \max\{0, \Pr[X_{id} = \tau] - \Pr[X_{re} = \tau]\}.$$

1.5.1. Revisiting the Expectation Method

In the following, we briefly state the main result of the *Expectation Method* and show that the *coefficients-H technique* [124] is a special case of the expectation method. Both these techniques are used for bounding the information theoretic distinguishing advantage of two random systems as defined above.
EXPECTATION METHOD. The expectation method was introduced by Hoang and Tessaro to derive a tight multi-user security bound of the key-alternating cipher [85]. Subsequently, this technique has been used for proving the multi-user security of the double encryption method in [86] and recently by Bose et al. to bound the multi-user security of AES-GCM-SIV [42]. This method is a generalization of coefficients-H technique. Let $\phi : \Theta \to [0, \infty)$ be a non-negative function which maps any attainable transcript to a non-negative real value. Suppose there is a set of good transcripts such that for any good transcript τ,

$$\frac{\Pr[X_{re} = \tau]}{\Pr[X_{id} = \tau]} \geq 1 - \phi(\tau). \tag{1.1}$$

The statistical distance between the two random variables X_{re} and X_{id} can then be bounded as

$$\Delta(X_{re}, X_{id}) \leq \mathbf{E}[\phi(X_{id})] + \Pr[X_{id} \in \Theta_{bad}], \tag{1.2}$$

where Θ_{bad} is the set of all bad transcripts. In other words, the advantage of A in distinguishing $\mathbf{S}_{re}$ from $\mathbf{S}_{id}$ is bounded by $\mathbf{E}[\phi(X_{id})] + \Pr[X_{id} \in \Theta_{bad}]$. coefficients-H technique can be seen as a simple corollary of the expectation method when ϕ is taken to be a constant function.

COEFFICIENTS-H TECHNIQUE. Consider two oracles $\mathbf{O}_0 = (\$, \perp)$ (the ideal oracle for the relaxed[5] game) and $\mathbf{O}_1$ (real, i.e. our construction in the same relaxed game). Let $\mathcal{T}$ denote the set of all possible transcripts an adversary can obtain (i.e. the set of all *attainable* transcripts in the ideal world). We let X_{re} be the random variable that takes values $\tau \in \mathcal{T}$ when the adversary interacts with the real world and X_{id} to be the random variable that takes values $\tau \in \mathcal{T}$ when it interacts with the ideal world. Without loss of generality, we assume that the adversary is deterministic and fixed. Then the sample space for X_{re} and X_{id} is uniquely determined by the underlying oracle. As we deal with stateless oracles, these probabilities are independent

[5] "*relaxed*" denotes that in addition to the query input-output tuples, additional state values may be supplied to the adversary (after all the queries are made) as a part of the transcripts

of the order of query responses in the transcript. Suppose we have a set of transcripts, $\mathcal{T}_{\text{good}} \subseteq \mathcal{T}$, which we call *good* transcripts, and the following conditions hold:

1. In the game involving the ideal oracle $\mathbf{O}_0$ (and the fixed adversary), the probability of getting a transcript in $\mathcal{T}_{\text{good}}$ is at least $1 - \epsilon_1$.
2. For any transcript $\tau \in \mathcal{T}_{\text{good}}$, $\Pr[\mathsf{X}_{\text{re}} = \tau] \geq (1 - \epsilon_2) \cdot \Pr[\mathsf{X}_{\text{id}} = \tau]$.

Then $|\Pr[\mathcal{D}^{\mathbf{O}_0} = 1] - \Pr[\mathcal{D}^{\mathbf{O}_1} = 1]| \leq \epsilon_1 + \epsilon_2$. The proof can be found in (say) [132].

1.5.2. Coefficients-H Technique in the Multi-user Setting

Even in a multi-user setting in the ideal cipher model (see Chapter 6 for a detailed description), one may consider X_{re} to denote the random variable that takes a transcript τ realized in the real world and X_{id} to denote the random variable that takes a transcript τ realized in the ideal world. The probability of realizing a transcript τ in the ideal (resp. real) world is called the *ideal* (*resp. real*) *interpolation probability*. A transcript τ is said to be attainable with respect to an adversary A if its ideal interpolation probability is non-zero, and Θ shall denote the set of all such attainable transcripts. Following these notations, we now state the main theorem of the Coefficients-H technique [124] for the multi-user setting:

Theorem 1 (Coefficients-H Technique). *Let* $\Theta = \mathsf{GoodT} \sqcup \mathit{Bad\text{-}Tag}$ *be a partition of the set of attainable transcripts. Suppose there exists* $\epsilon_{\text{ratio}} \geq 0$ *such that for any* $\tau = (\tau_c, \tau_p) \in \mathsf{GoodT}$,

$$\frac{\mathsf{p}_{\text{re}}(\tau)}{\mathsf{p}_{\text{id}}(\tau)} := \frac{\Pr[\mathsf{X}_{\text{re}} = \tau]}{\Pr[\mathsf{X}_{\text{id}} = \tau]} \geq 1 - \epsilon_{\text{ratio}},$$

and there exists $\epsilon_{\text{bad}} \geq 0$ *such that* $\Pr[\mathsf{X}_{\text{id}} \in \mathit{Bad\text{-}Tag}] \leq \epsilon_{\text{bad}}$. *Then*

$$\mathbf{Adv}^{\text{mPRF}}_{\Pi}(\mathsf{A}) \leq \epsilon_{\text{ratio}} + \epsilon_{\text{bad}}. \tag{1.3}$$

2. A Beyond the Birthday Bound Secure MAC in the Faulty Nonce Model

Abstract

Although Encrypt-then-MAC (EtM) is a popular mode for authenticated encryption (AE), almost all designs following this paradigm (including the AE suites for TLS) are unfortunately vulnerable against nonce misuse: a single repetition of the nonce value reveals the hash key, leading to a universal forgery attack. There are only two authenticated encryption schemes following the EtM paradigm that can resist nonce misuse attacks – the GCM-RUP (CRYPTO-17) and the GCM/2^+ (INSCRYPT-12). However, they are secure only up to the birthday bound in the nonce respecting setting, resulting in a restriction on the data limit for a single key. This chapter introduces nEHtM, which is a nonce-based variant of EHtM (FSE-10) constructed using a block cipher, and which has a beyond the birthday bound (BBB) unforgeable security that gracefully degrades under nonce misuse. It also combines nEHtM with the CENC (FSE-06) mode of encryption using the EtM paradigm to realize a nonce-based AE, CWC+. CWC+ is very close to the CWC AE scheme (FSE-04) (requiring only a few more XOR operations); not only does it provide BBB security, but also gracefully degrading security on nonce misuse.

Keywords – graceful degradation of security, faulty nonce, Mirror Theory, Extended Mirror Theory, Expectation Method, GCM, EHtM, nEHtM, CWC.

2.1. Introduction

2.1.1. Nonce Misuse Resistance Security

A *nonces* is a random or pseudorandom string that may be attached to a Cryptographic protocol. The term stands for "number once" and is commonly referred to as a cryptographic nonce. It can be in the form of a timestamp, a visit counter on a webpage or a special marker intended to limit or prevent the unauthorized replay or reproduction of a file. Typically, a nonce is a value that varies with time to verify that specific values are not reused.

Attacks similar to Joux's *forbidden attack* [1] (which demonstrates how GCM leaks the hash key on execution of an encryption query with a repeated nonce) can be applied against most popular AEs such as ChaCha20+Poly1305, GCM, CWC, CHM, CIP, OGCM1, OGCM2, etc. that follow the EtM paradigm. GCM-RUP and GCM/2^+ are two exceptions, as they use variants of the WC MAC:

- GCM-RUP resists this attack as it uses the XEX [126] construction to define the tag, which is computed for a data D as $\mathsf{E}_{\mathbf{k}}(\mathsf{H}_{\mathbf{k}_h}(D) \oplus N) \oplus \mathsf{H}_{\mathbf{k}_h}(D)$. However, the following attack ensures only a nonce-respecting birthday bound security for XEX.

1. Make $2^{n/2}$ nonce-respecting queries (D_i, N_i) and receive tags T_i for each $i \in [2^{n/2}]$, where n is the block size of the underlying block cipher.
2. A collision amongst the values of $\mathsf{H}_{\mathbf{k}_h}(D_i) \oplus N_i$ is expected and can be detected through a collision amongst the values of $N \oplus T$.
3. Whenever $N_i \oplus T_i = N_j \oplus T_j$ for $i \neq j$, the difference between the hash outputs may be computed as $N_i \oplus N_j$, which eventually leaks the hash key.

Figure 2.1.: Birthday bound attack on XEX

- GCM/2^+ resists the attack as it uses the Encrypted Wegman-Carter-Shoup (EWCS) [130] construction to define the tag. The tag of EWCS for a data D is computed as $\mathsf{E}_{\mathbf{k}_2}(\mathsf{E}_{\mathbf{k}_1}(N) \oplus \mathsf{H}_{\mathbf{k}_h}(D))$. However, a nonce-respecting adversary can make $2^{n/2}$ queries with the same message and observe no collision in the tag, thus reducing the PRF security to the birthday bound.

Challenges in maintaining the uniqueness of the nonce may arise on implementations in a stateless device or in cases where the nonce is chosen

randomly from a small set. Faults in implementation of the cipher, accidental nonce-resets, etc. may also cause repetition of the nonce. In fact, the internet-wide scan by Böck et al. [36] finding 184 devices that used a duplicate nonce makes a convincing case for security against nonce misuse in MAC and AE constructions.

2.1.2. Beyond the Birthday Bound Security with Graceful Degradation on Nonce Misuse

GCM-RUP possesses a forging advantage bounded by $\ell q_m^2/2^n$ in the nonce-respecting model (where a message and its associated data can consist of at most ℓ data blocks and an adversary can make at most q_m encryption queries). This only allows the AES-based GCM-RUP (say) to process no more than $q_m \leq 2^{32}$ queries of size at most $\ell = 2^{32}$ blocks with a maximum advantage of 2^{-32}, a tolerance level much smaller than BBB security.

Achieving BBB security primarily equips a construction with a larger data limit for a single key. Furthermore, the PRF security of a MAC contributes to the privacy of the EtM encryption. *Encrypted Wegman-Carter with Davies-Meyer* [56] (or EWCDM) and *Decrypted Wegman-Carter with Davies-Meyer* [62] (or DWCDM) are two constructions that were proposed with an objective of achieving security beyond the birthday bound in a nonce respecting setting. However, these constructions only provided a birthday bound security with even a single misuse of the nonce. There are other known constructions such as *Dual Encrypted Wegman-Carter with Davies-Meyer* (or EWCDMD) [103, 117], *Encrypted Wegman-Carter-Shoup* [56] (or EWCS) and single hash-key variants of CLRW2 [94] possessing beyond the birthday bound nonce-respecting security but these too immediately degrade to a birthday bound PRF security whenever the nonce is not respected.

Goal of the chapter. The main goal of this chapter is to find *an efficient MAC which is BBB secure both as a PRF and a MAC*. It must provide *graceful degradation of security in the nonce-misuse setting*. Deterministic MACs (not requiring any nonce) that provide BBB security and mainly follow a double-block hash-then-sum approach [61, 63] and thus require the computation of two blocks of algebraic hashes (or one pass of block cipher or tweakable block cipher executions) notwithstanding, a single-block hash (which is certainly faster than two blocks of hash and requires a smaller hash key) is undeniably a better option. This motivates the chapter to focus on constructing a single-block algebraic hash-based design (e.g., a single call of the polynomial hash [108]).

Graceful Degradation of Security on Nonce Misuse. The most popular metric to measure nonce misuse is the maximum number of multicollisions

in nonce values amongst all queries [125]. To the best of our knowledge, none of the existing block cipher-based nonce-based MACs adhere to this notion with BBB security guarantee. We have also explored many other variants of MAC constructions using at most two block cipher calls and a single hash function call. Unfortunately, we found that none of them give beyond the birthday bound security in terms of multicollision nonce misuse, even with multicollisions of size 2.

This chapter instead considers another natural definition of nonce misuse, called the number of faulty nonces. An authentication query is said to be a *faulty query* if there exists a previous MAC query such that their corresponding nonces match. The nonce in a faulty query is called a *faulty nonce*. The notion of a faulty nonce is weaker than multicollision. When a counter is implemented in an aperiodic manner (e.g., timely nonce [36] used in TLS 1.2), a simple reset does not give a large number of faulty nonces; there are easy countermeasures to prevent a large number of faulty nonce encryptions.

2.1.3. Our Contribution

Our contribution in this chapter is threefold:

1. Multicollision on the Universal Hash. We study the probability of occurrence of multicollisions in a universal hash function. In particular, we show the probability of obtaining a $(\xi+1)$-multicollision tuple amongst q inputs to be at most $q^2\epsilon/\xi$ (see Sect. 2.4). This is clearly an improved bound as compared to a straightforward application of the union bound. We believe this problem can generate an independent interest and can also be used to get improved bounds for other constructions.

2. BBB Secure MAC with Graceful Security. [107] analyzes a probabilistic MAC called EHtM and shows a roughly $3n/4$-bit tight MAC security (tightness shown in [68]). This chapter analyzes a construction denoted as nEHtM, where

1. the random salt is replaced by the nonce and
2. the two independent pseudorandom functions are replaced by a single-keyed block cipher.

Given a data D and a nonce N, the tag is computed as follows (see Fig. 2.2(b)):

$$\mathsf{nEHtM}_{\mathbf{k},\mathbf{k}_h}(N,D) := \mathsf{E}_{\mathbf{k}}(0\|N) \oplus \mathsf{E}_{\mathbf{k}}(1\|\mathsf{H}_{\mathbf{k}_h}(D) \oplus N).$$

We show that nEHtM is secure roughly up to $2^{2n/3}$ authentication queries and 2^n verification queries in the nonce-respecting setting. Moreover, this security

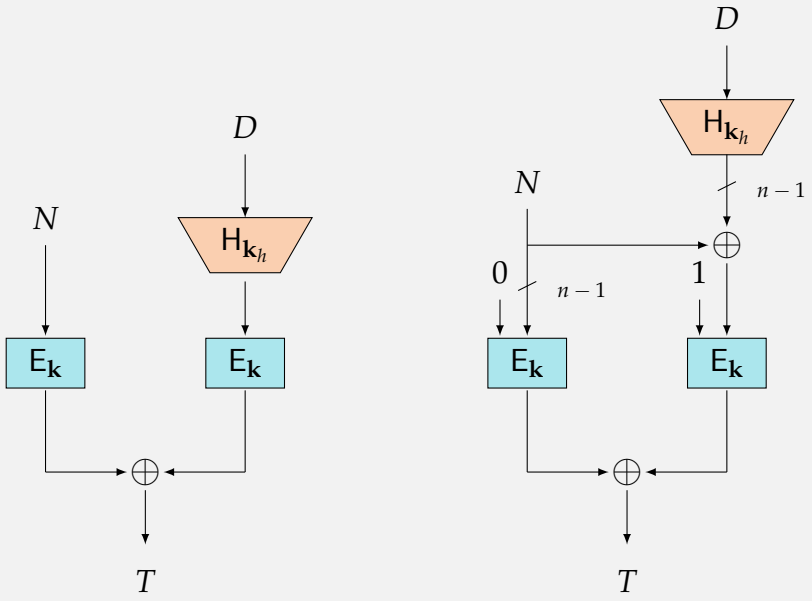


Figure 2.2.: (a) On the left is the CWC MAC (MAC algorithm used in CWC); (b) on the right is the domain separation variant of *nonce-based Enhanced Hash-then-Mask*.

degrades gracefully on the introduction of faults in the nonce. An extended distinguishing game shows the unforgeability of this construction. We apply the expectation method (as it shall later be shown to give a better bound than the coefficients-H technique) to bound the distinguishing advantage of two worlds. In the ideal world, once we realize the random tags T_i, we need to sample the hash key so as to determine all inputs of the underlying block cipher. The equality patterns amongst the nonce values are deterministic and we bound the number of faulty nonces by a parameter μ. However, the equality patterns amongst other inputs of the form $X := \mathsf{H}_{\mathbf{k}_h}(D) \oplus N$ are probabilistic due to the randomness of the hash key. As there may not be sufficient entropy in the hash key (which could be n-bit for say, the polynomial hash), the number of multicollisions amongst the values of X may not be easy to compute. We tackle this problem using the multicollision result (as stated in the first contribution) of the underlying hash function.

Limiting multicollisions in the values of both X and N allows us to apply mirror theory to show a beyond the birthday bound security on the distinguishing advantage of nEHtM. Note that mirror theory cannot give a beyond the birthday bound security without restricting the number of multicollisions.

It must be noted here that nEHtM (like all other candidates) is not secure beyond the birthday bound under the notion of multicollision nonce misuse

security and the corresponding attack is discussed in Sect. 2.2.6.

3. Application to a CWC-like AE Construction. We propose CWC+, which is an instance of the EtM composition based on the CENC type encryption with a maximum width parameter and the nEHtM MAC. Moreover, we apply an appropriate domain separation to make it a single-keyed construction (even the hash key is generated from the the block cipher). The construction is a very close variant of CWC as it requires only a few additional XOR computations and no extra calls to the block cipher. Furthermore, CWC+ gives both (1) BBB security and (2) graceful security degradation in the faulty nonce misuse model. In particular, we have the following forging advantage of CWC+:

$$\mathbf{Adv}^{\mathsf{auth}}_{\mathsf{CWC+}} = \frac{105\sigma^3\ell}{2^{2n}} + \frac{6\sigma\ell}{2^n} + \frac{2q_d}{2^\rho} + \frac{2q_d\ell}{2^n} + \frac{(2q_e+q_d)2\ell\mu}{2^n} + \left(\frac{5\sigma\ell\mu}{2^n}\right)^2,$$

where q_e and q_d is the number of encryption and decryption queries, ρ is the tag size, ℓ is the maximum number of message blocks queried including the associated data blocks, σ is the total number of message blocks queried and μ is the total number of faulty queries. Moreover, the security of CWC+ gracefully drops to the birthday bound when $\ell\mu$ is about $2^{n/2}$. However, when $\ell \leq 2^{n/4}$, then the security bound of CWC+ caps at roughly $2^{7n/12}$, which is strictly greater than the birthday bound. A better bound can be obtained if we assume some restrictions over all the message lengths.

(3) Another notable feature of CWC+ is that the scheme remains secure even with short tag lengths. In GCM, if the tag length is only 32 bits, then an adversary forges the construction with just 1024 verification attempts by querying with a single message consisting of 2^{22} blocks. However, for the same tag size, the authenticity advantage of CWC+ is 2^{-21} when the adversary forges the construction with 1024 verification attempts.

2.2. Design and Security of nEHtM and CWC+

This section discusses the design and security results of our proposed nonce-based message authentication code, which we call nEHtM, and a nonce-based authenticated encryption scheme called CWC+. We begin our discussion with a result on EtM composition that combines a standard encryption and a MAC scheme to achieve authenticated encryption.

2.2.1. Encrypt-then-MAC: Generic Composition Result

Bellare and Namprempre in [15] and Canetti and Krawczyk in [44] explored ways to combine standard encryption schemes with MACs to achieve authenticated encryption schemes. Their results yield three different types of

combinations: (a) Encrypt-and-MAC (E&M), (b) MAC-then-Encrypt (MtE) and (c) Encrypt-then-MAC (EtM). This chapter focuses only on EtM.

Let $\mathcal{E} = (\mathcal{E}.\mathsf{KGen}, \mathcal{E}.\mathsf{Enc}, \mathcal{E}.\mathsf{Dec})$ be a nonce-based symmetric key encryption scheme and $\mathcal{I} = (\mathcal{I}.\mathsf{KGen}, \mathcal{I}.\mathsf{Tag}, \mathcal{I}.\mathsf{Ver})$ be a nonce-based message authentication code. The function $\mathcal{E}.\mathsf{Enc} : \mathbf{k}_e \times \mathcal{N} \times \mathcal{M} \to \mathcal{C}$ maps a tuple $(\mathbf{k}_e, N, M)$ to a ciphertext C and the decryption function $\mathcal{E}.\mathsf{Dec} : \mathcal{K}_e \times \mathcal{N} \times \mathcal{C} \to \mathcal{M} \cup \{\perp\}$ either maps a tuple $(\mathbf{k}_e, N, C)$ — if legitimate — to the corresponding message M or otherwise returns the error symbol $\perp$. For the message authentication code $\mathcal{I}$, the function $\mathcal{I}.\mathsf{Tag} : \mathcal{K}_m \times \mathcal{N} \times \mathcal{D} \to \mathcal{T}$ maps a tuple $(\mathbf{k}_m, N, D)$ to a tag T, and the verification function $\mathcal{I}.\mathsf{Ver} : \mathcal{K}_m \times \mathcal{N} \times \mathcal{M} \times \mathcal{T} \to \{\top, \perp\}$ maps a quadruple $(\mathbf{k}_e, N, D, T)$ to one of the two symbols $\top$ (i.e. accept), $\perp$ (i.e. reject) according to whether T is a valid tag for the tuple $(\mathbf{k}_n, N, D)$ or not, respectively.

Based on these two schemes, we define the EtM authenticated encryption scheme $\mathsf{AE}_{\mathcal{E},\mathcal{I}} = (\mathsf{AE.KGen}, \mathsf{AE.Enc}, \mathsf{AE.Dec})$, where the key-generation algorithm AE.KGen generates a random pair of keys $(\mathbf{k}_e, \mathbf{k}_m) \in \mathcal{K}_e \times \mathcal{K}_m$. The encryption and decryption algorithms are defined as follows:

$$\mathsf{AE.Enc}(\mathbf{k}_e \| \mathbf{k}_m, N, A, M) = \begin{cases} C \leftarrow \mathcal{E}.\mathsf{Enc}(\mathbf{k}_e, N, M) \\ T \leftarrow \mathcal{I}.\mathsf{Tag}(\mathbf{k}_m, N, A \| C) \end{cases}$$

$$\mathsf{AE.Dec}(\mathbf{k}_e \| \mathbf{k}_m, N, A, C, T) = \begin{cases} M \leftarrow \mathcal{E}.\mathsf{Dec}(\mathbf{k}_e, N, C), \text{ if } Z = \top \\ \perp, \text{ if } Z = \perp, \end{cases}$$

for $Z \leftarrow \mathcal{I}.\mathsf{Ver}(\mathbf{k}_m, N, A \| C, T)$. We consider two security notions for the AE scheme: privacy and authenticity. The privacy advantage of the AE is defined as follows:

$$\mathbf{Adv}^{\mathrm{priv}}_{\mathsf{AE}}(\mathsf{A}) := \Pr[(\mathbf{k}_e, \mathbf{k}_m) \xleftarrow{\$} (\mathcal{K}_e \times \mathcal{K}_m) : \mathsf{A}^{\mathsf{AE.Enc}_{(\mathbf{k}_e, \mathbf{k}_m)}} = 1] - \Pr[\mathsf{A}^{\$} = 1],$$

where the random oracle $\$$ takes (N, A, M) as input and returns $(C, T) \xleftarrow{\$} \{0,1\}^{|M|+\rho}$. We assume that the adversary A is nonce respecting i.e. it does not make two queries with the same nonce.

We say that the adversary A *forges* if the oracle AE.Dec returns a bit string (which is not $\perp$) for a query (N, A, C, T) such that for no message M does the encryption query (N, A, M) to the oracle AE.Enc return (C, T). If an adversary A interacts with the encryption and decryption oracles of the AE, then the authenticity advantage of the AE is defined as follows:

$$\mathbf{Adv}^{\mathrm{auth}}_{\mathsf{AE}}(\mathsf{A}) := \Pr[(\mathbf{k}_e, \mathbf{k}_m) \xleftarrow{\$} (\mathcal{K}_e \times \mathcal{K}_m) : \mathsf{A}^{\mathsf{AE.Enc}_{(\mathbf{k}_e, \mathbf{k}_m)}, \mathsf{AE.Dec}_{(\mathbf{k}_e, \mathbf{k}_m)}} \text{ forges}],$$

where we assume that A can repeat nonces in decryption queries and can also use the nonces used in encryption queries.

The security of an AE scheme refers to the sum of its privacy and authenticity advantages. The privacy advantage of a nonce-based encryption scheme $\mathcal{E}$ that forms an AE with a MAC $\mathcal{I}$ is bound by the PRF advantages of $\mathcal{E}$ and $\mathcal{I}$, while its authenticity advantage is bound by the forging advantage of $\mathcal{I}$. The achievement of a beyond the birthday bound secure nonce-based AE scheme following the EtM paradigm thus requires a nonce respecting BBB secure nonce-based encryption scheme and a MAC mode that gives beyond the birthday bound security for PRF-distinguishability and unforgeability (possibly in the nonce misuse model).

2.2.2. Encryption Modes used in Encrypt-then-MAC-Based AEs

A symmetric encryption scheme is generally defined through a pseudo-random number generator (PRNG) that takes a short master key $\mathbf{k}$ and an initial value or nonce N that generates a keystream $(S_1, S_2, \ldots)$. Then the ciphertext is generated from the plaintext and the keystream by applying the one-time padding technique.

The counter mode of encryption (CTR) is a popular symmetric key encryption scheme, which gives birthday bound security in terms of the number of blocks and is used as the underlying encryption scheme in AE constructions such as CWC [93], GCM [101], GCM/2+ [3] and GCM-RUP [4]. On the other hand Multi-EDM [137] and Multi-EDMD [137], which give an almost n-bit security, are used as underlying encryption schemes in OGCM1 [137] and OGMC2 [137] respectively.

Cipher-Based Encryption. The cipher-based encryption [88] (CENC) is parametrized by a fixed non-negative integer w and so can be denoted as CENC_w. The PRNG of CENC_w takes a key $\mathbf{k}$, a nonce ctr and a length l as its inputs and gives a sequence of fixed length keystream blocks as output, where the i^{th} keystream block is defined as

$$S_i := \mathsf{E}_{\mathbf{k}}(\mathsf{ctr} + j(w+1)) \oplus \mathsf{E}_{\mathbf{k}}(\mathsf{ctr} + j(w+1) + i),\ j \in \left[0, \frac{l}{w} - 1\right], i \in [1, w].$$

The optimal security of CENC_w is shown in [27]. It is used as the underlying encryption scheme of the CHM and CIP AEs. An optimally secure nonce-based encryption mode $\mathsf{CENC}_{\max}$ [27], in which w is set to the maximum number of message blocks, is applied as the underlying encryption scheme of mGCM [27].

2.2.3. MACs used in Encrypt-then-MAC-Based AEs

Wegman-Carter MAC. The Wegman-Carter (WC) MAC [133] is an early and popular nonce-based MAC that authenticates a message by masking

its hash value with a random number generated through a pseudorandom function applied on a nonce i.e.

$$\mathsf{WC}[\mathsf{F},\mathsf{H}](N,M) := \mathsf{F}_{\mathbf{k}}(N) \oplus \mathsf{H}_{\mathbf{k}_h}(M).$$

If ϵ is the hash differential probability and q_v is the number of verification attempts, then the WC MAC provides $O(\epsilon q_v)$ security when nonces are never reused. However, the construction has no security when the nonce repeats even once. For some constructions, the hash key is revealed and for others, a simple forgery is possible. Different instantiations of the pseudorandom function and hash function give different instances of the WC MAC. The Wegman-Carter-Shoup (WCS) MAC [130] is a popular instantiation of the WC MAC in which the pseudorandom function is replaced by a block cipher. WCS is used as the underlying MAC in GCM, CHM and CIP. EDM and EDMD are used as instantiations of the PRF in the WC MAC and the resultant MACs are used as underlying MAC algorithms in OGCM1 and OGCM2, respectively. CWC MAC [93] (used as the MAC function in the CWC AE construction) is another variant of the WC MAC, in which the pseudorandom function is replaced by a block cipher and the hash function is defined as $E_{\mathbf{k}_2}(\mathsf{H}_{\mathbf{k}_h}(M))$.

Encrypted Wegman-Carter-Shoup. The Encrypted Wegman-Carter-Shoup (EWCS) MAC [56] was proposed as a remedy to the problem of nonce misuse security over the WC MAC. The EWCS MAC encrypts the output of the WCS MAC to generate the tag. This tag is then used as the underlying MAC of the GCM/2+ construction. EWCS gives a security of around $2^{n/2}$ when nonces do not repeat. An attacker can make approximately $2^{n/2}$ queries with distinct nonces but the same message and observe no collisions in the tag.

XOR-Encrypt-XOR. XOR-Encrypt-XOR (XEX) was originally proposed as a mode of designing a tweakable block cipher [126]. Luykx et al. [4] used it as the underlying MAC in GCM-RUP. For a nonce N and a message M, XEX works as follows:

$$\mathsf{XEX}[\mathsf{E},\mathsf{H}](N,M) := \mathsf{E}_{\mathbf{k}}(N \oplus \mathsf{H}_{\mathbf{k}_h}(M)) \oplus \mathsf{H}_{\mathbf{k}_h}(M).$$

XEX is secure up to the birthday bound when nonces do not repeat. It is easy to see that a collision amongst the values of $N \oplus \mathsf{H}_{\mathbf{k}_h}(M)$ leads to a forgery, which can be readily detected by finding collisions in the values of $N \oplus T$.

EWCDM [56] and a single-keyed hash variant of CLRW2 [94] are some possible alternatives of nonce-based MACs that can be potentially applied as the MAC function of any EtM-based AE mode. EWCDM is proven secure up to approximately $2^{2n/3}$ queries when nonces do not repeat [56], and the

single-keyed hash variant of CLRW2 is also birthday bound secure in the nonce respecting setting.

All these constructions have a birthday bound PRF security as an attacker can make $2^{n/2}$ queries with the same message but distinct nonces and observe no collision in the tag.

2.2.4. Security of nEHtM: A Nonce-Based Version of EHtM

The previous section demonstrates how the MACs used in the existing AE modes are not secure beyond the birthday bound when nonces repeat just once, making them unsuitable for use in designing an AE that is resilient in the faulty nonce model. This section introduces the *nonce-based Enhanced Hash-then-Mask* (nEHtM) and gives upto $2n/3$-bit unforgeability in the faulty nonce model. The Enhanced Hash-then-Mask (EHtM) proposed by Minematsu [107] is the first BBB secure PRF-based probabilistic MAC that uses only an n-bit random salt and an n-bit PRF. nEHtM is structurally similar to EHtM, except that the random salt is replaced by a nonce and the PRF by a block cipher. For the purpose of domain separation, we consider an $(n-1)$-bit nonce and an $(n-1)$-bit keyed hash function. For any message M and nonce N, nEHtM is defined as follows:

$$\mathsf{nEHtM}[\mathsf{E}, \mathsf{H}_{\mathbf{k}_h}](N, M) := E_{\mathbf{k}}(0\|N) \oplus \mathsf{E}_{\mathbf{k}}(1\|(N \oplus \mathsf{H}_{\mathbf{k}_h}(M))).$$

We now state Theorem 2, which bounds the unforgeability of nEHtM in the faulty nonce model. We also demonstrate a birthday bound forging attack on nEHtM when the number of faulty nonces reaches an order of $2^{n/2}$. The underlying idea of the attack is to form an alternating cycle of length 4 in the input of the block cipher; details may be found in Sect. 2.2.6.

Theorem 2. *Let $\mathcal{M}, \mathbf{k}$ and $\mathbf{k}_h$ be finite and non-empty sets. Let $\mathsf{E} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and $\mathsf{H} : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^{n-1}$ be an $(n-1)$-bit ϵ-AXU hash function. Let μ be a fixed parameter. Then the forging advantage for any (μ, q_m, q_v, t)-adversary against $\mathsf{nEHtM}[\mathsf{E}, \mathsf{H}]$ that makes q_m authentication queries with at most μ faulty nonces and q_v verification queries in time t is given by*

$$\mathbf{Adv}^{\mathrm{MAC}}_{\mathsf{nEHtM}[\mathsf{E},\mathsf{H}]}(\mu, q_m, q_v, t) \leq \mathbf{Adv}^{\mathrm{prp}}_{\mathsf{E}}(\mu, q_m + q_v, t') + \frac{48q_m^3}{2^{2n}} + \frac{12q_m^4\epsilon}{2^{2n}} + \frac{12\mu^2 q_m^2}{2^{2n}} + \frac{q_m + 2q_v}{2^n} + \frac{4q_m^3\epsilon}{2^n} + (2q_m + q_v)\mu\epsilon + q_v\epsilon,$$

where the time parameter t' is of the order of $t + (q_m + q_v)t_{\mathsf{H}}$ and t_{H} is the time required for computing the hash function. Assuming $\epsilon \approx 2^{-(n-1)}$ and $q_m \leq \epsilon^{-1}$

simplifies this bound to

$$\mathbf{Adv}^{\mathrm{MAC}}_{\mathsf{nEHtM}[\mathsf{Perm},\mathsf{H}]}(\mu, q_m, q_v, t) \leq \frac{72q_m^3}{2^{2n}} + \left(\frac{12\mu^2 q_m^2}{2^{2n}} + \frac{(4q_m + 2q_v)\mu}{2^n}\right) + \left(\frac{q_m + 4q_v}{2^n}\right).$$

The proof of this theorem is deferred to Sect. 2.5. The forging advantage of nEHtM for $\mu \leq 2^{n/3}$ and $q_m \leq 2^{2n/3}$ is thus

$$\mathbf{Adv}^{\mathrm{MAC}}_{\mathsf{nEHtM}[\mathsf{Perm},\mathsf{H}]}(q_m, q_v, t) \leq \frac{13q_m}{2^{2n/3}} + \frac{4q_v}{2^{2n/3}}.$$

Remark 1. *EHtM offers $3n/4$-bit security [68], whereas its nonce-based variant offers $2n/3$-bit security. This is because while EHtM also involves the random salts as an additional source of entropy, the number of multicollisions in the underlying hash function of nEHtM must be bound, for which the only source of randomness is the hash key.*

2.2.5. Security of CWC+: A Beyond the Birthday Bound Variant of CWC

We have already seen that $\mathsf{CENC}_{\max}$ is a highly efficient optimally secure nonce respecting encryption scheme and that nEHtM is a nonce-based MAC that is secure beyond the birthday bound in the faulty nonce model. Gluing them together using the EtM paradigm, we realize CWC+, a beyond the birthday bound secure AE in the faulty nonce model. The encryption and decryption functions of CWC+ are shown in Fig. 2.3. The privacy and the authenticity advantages of CWC+ are stated in the following theorem, the proof of which is deferred to Sect. 2.6.

Theorem 3. *Let $\mathsf{E} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and $\mathsf{Poly} : \{0,1\}^n \times \{0,1\}^* \to \{0,1\}^{n-1}$ be the $(n-1)$-bit truncated PolyHash function (which truncates the first bit of the PolyHash output). Let ρ and μ be two fixed parameters. Then the privacy advantage for any $(q_e, q_d, \ell, \sigma, t)$-nonce respecting adversary against $\mathsf{CWC}+[\mathsf{E}, \rho]$ is given by*

$$\mathbf{Adv}^{\mathrm{priv}}_{\mathsf{CWC}+[\mathsf{E},\rho]}(q_e, q_d, \ell, \sigma, t) \leq \mathbf{Adv}^{\mathrm{PRP}}_{\mathsf{E}}(\sigma + 2q, t') + \frac{105\sigma^3\ell}{2^{2n}} + \frac{6\sigma\ell}{2^n} + \frac{2q_d}{2^\rho} + \frac{2q_d\ell}{2^n}.$$

The authenticity advantage for any $(\mu, q_e, q_d, \ell, \sigma, t)$-adversary against $\mathsf{CWC}+[\mathsf{E}, \rho]$

is given by

$$\mathbf{Adv}^{\text{auth}}_{\text{CWC+}[\mathsf{E},\rho]}(\mu, q_e, q_d, \ell, \sigma, t) \leq \mathbf{Adv}^{\text{PRP}}_{\mathsf{E}}(\sigma + 2q, t') + \frac{105\sigma^3\ell}{2^{2n}} + \frac{6\sigma\ell}{2^n} + \frac{2q_d}{2^\rho} + \frac{2q_d\ell}{2^n} + \frac{(2q_e + q_d)2\ell\mu}{2^n} + \left(\frac{5\sigma\ell\mu}{2^n}\right)^2.$$

We denote $q_e + q_d$ by q — the total number of encryption and decryption queries, and $O(t + qt_{\mathsf{H}} + \sigma + 2q)$ by t', where t_{H} denotes the time required for computing the hash function and μ denotes the total number of encryption queries with faulty nonces.

$\text{CWC+}.\mathsf{Enc}_{\mathbf{k}}(N, A, M)$

1 : $L \leftarrow \mathsf{E}_{\mathbf{k}}(\mathbf{0}),\ N' \leftarrow N \| 0^{n/4-1}$.
2 : $l \leftarrow \lceil |M|/n \rceil$.
3 : $S \leftarrow \mathsf{CENC}_{\max}(\mathbf{k}, 0\|N', l)$.
4 : $C \leftarrow M \oplus \mathsf{first}(S, |M|)$.
5 : $\widetilde{T} \leftarrow \mathsf{nEHtM}[\mathsf{E}, \mathsf{Poly}_{\mathsf{E}_{\mathbf{k}}(\mathbf{0})}](N', C\|A)$.
6 : $T \leftarrow \mathsf{chop}_\rho(\widetilde{T})$.
7 : `return` (C, T).

$\text{CWC+}.\mathsf{Dec}_{\mathbf{k}}(N, A, C, T)$

1 : $L = \mathsf{E}_{\mathbf{k}}(\mathbf{0}),\ N' \leftarrow N \| 0^{n/4-1}$.
2 : $l \leftarrow \lceil |C|/n \rceil$.
3 : $\widetilde{T}' \leftarrow \mathsf{nEHtM}[\mathsf{E}, \mathsf{Poly}_{\mathsf{E}_{\mathbf{k}}(\mathbf{0})}](N', C\|A)$.
4 : `if` $\mathsf{chop}_\rho(\widetilde{T}') \neq T$ `then return` $\perp$.
5 : $S \leftarrow \mathsf{CENC}_{\max}(\mathbf{k}, N', l)$.
6 : $M \leftarrow C \oplus \mathsf{first}(S, |C|)$.
7 : `return` M.

Figure 2.3.: Encryption and Decryption functions of CWC+. $\mathsf{Poly}_{\mathsf{E}_{\mathbf{k}}(\mathbf{0})}$ denotes the PolyHash function with its n-bit hash key set to the encrypted value of $\mathbf{0}$. $\mathsf{first}(S, |M|)$ denotes the first $|M|$ bits in the sequence S. chop_ρ is a function that truncates the last $n - \rho$ bits of its input.

2.2.6. Nonce Misuse Attack on nEHtM

In the following, we discuss a birthday bound forging attack on nEHtM when the number of faulty queries is roughly $2^{n/2}$. The underlying idea

of the attack is to form an alternating cycle of length 4 in the input of the block cipher. For this, an adversary A makes two sets of $2^{n/2}$ MAC queries – one with message M and another with message $M'(\neq M)$ – and finds four queries such that the sum of their tag becomes zero. A mounts the attack in two phases: (a) In the first phase, it finds a quadruple that makes the tag-sum zero. (b) In the second phase, it forges the MAC. The attack is described algorithmically in part (b) of Fig. 2.4.

First Phase of the Attack.

1. A makes $q^* = 2^{n/2}$ MAC queries (N_i, M), $i \in [q^*]$ and receives responses $T_i \leftarrow \mathsf{nEHtM}(N_i, M)\ \forall\ i \in [q^*]$.
2. A makes $q^* = 2^{n/2}$ MAC queries (N_{q^*+i}, M'), $i \in [q^*]$ and receives responses $T_{q^*+i} \leftarrow \mathsf{nEHtM}(N_{q^*+i}, M)\ \forall\ i \in [q^*]$.
3. A finds two distinct query indices $i, j \in [q^*]$ such that $T_i \oplus T_j \oplus T_{q^*+i} \oplus T_{q^*+j} = \mathbf{0}$.

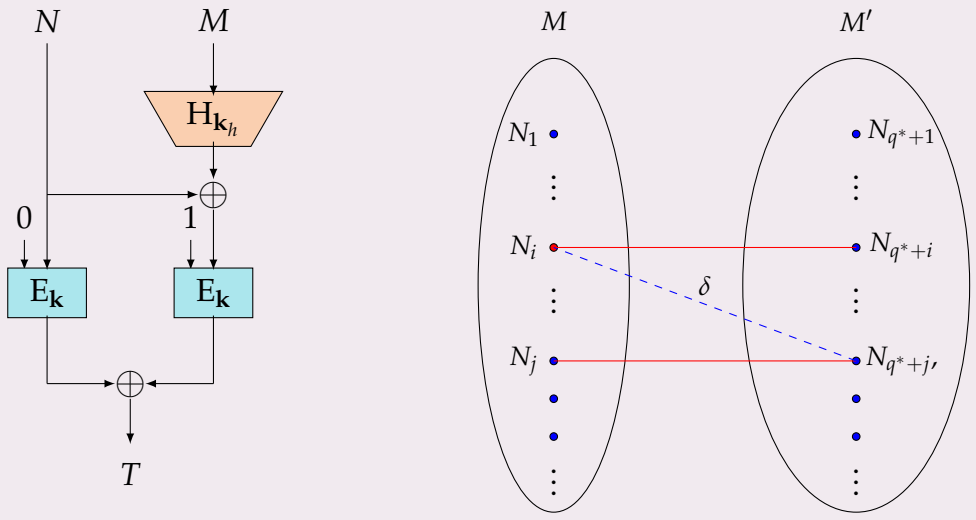


Figure 2.4.: (a) On the left is the domain-separated variant of *Nonce-based Enhanced Hash-then-Mask* with an n-bit keyed hash function H_{k_h} and an n-bit block cipher E_k; (b) On the right is the forging attack on the construction.

Note that the event CollT : $\exists\, i, j \in [q^*]$: $T_i \oplus T_j \oplus T_{q^*+i} \oplus T_{q^*+j} = \mathbf{0}$ can take place either because of (i) the collision of the hash i.e., $H_k(M) \oplus H_k(M') = N_i \oplus N_j$ or (ii) the random output of the underlying permutation Π. The probability of occurrence of the second case is extremely low (we call it a *false positive*) and therefore, when CollT takes place, we can assume with high probability that the hash value collides. As a result, A obtains the hash difference $N_i \oplus N_j$.

Second Phase of the Attack. A chooses two distinct nonces $N_{2q^*+1}, N_{2q^*+2} \notin \{N_1, \ldots, N_{q^*}\}$ such that $N_{2q^*+1} \oplus N_{2q^*+2} = N_i \oplus N_j$ and makes queries $T_{2q^*+1} \leftarrow (N_{2q^*+1}, M)$, $T_{2q^*+2} \leftarrow (N_{2q^*+1}, M')$, $T_{2q^*+3} \leftarrow (N_{2q^*+2}, M)$. This allows A to forge with $(N_{2q^*+2}, M', T_{2q^*+1} \oplus T_{2q^*+2} \oplus T_{2q^*+3})$.

As step 3 of the first phase holds with probability $(q^*)^2/2^n$, this attack holds for $q \approx 2^{n/2+1}$ and $\mu = 2^{n/2}$, when $\nu = 2$.

2.3. Mirror Theory

Introduced by Patarin in [120], mirror theory is a technique that provides a lower bound for the number of solutions to a given system of linear (more precisely, affine) bivariate equations and non-equations in a finite field (e.g., GF(2^n)). Solving a system of linear or affine equations is a straightforward and common problem in linear algebra. However, complications arise when non-equations are involved. A special form of problems involving non-equations is to find distinct values for all the variables present in the system. If $Y_1, \ldots, Y_s$ are the variables, the system of non-equations $Y_i \oplus Y_j \neq \mathbf{0}$ for all $i \neq j$ essentially restricts the solutions to those in which all variables take distinct values. We call such a solution an *injective solution*. Patarin did not consider any other forms of non-equations [120, 123, 121]. Datta et al. [62] considered other forms and termed the results *extended mirror theory* — the authors provided a lower bound on the number of injective solutions when the maximum component size $w_{\max}$ (a parameter that shall be defined soon) is three or less. This chapter extends their analysis for an arbitrary $w_{\max}$.

Injective Solution of Equations. Let $G = (\mathcal{V} := \{Y_1, \ldots, Y_\alpha\}, \mathcal{S})$ be a simple acyclic graph with an *edge-labelling function* $\mathcal{L} : \mathcal{S} \to \{0,1\}^n$. For an edge $\{Y_i, Y_j\} \in \mathcal{S}$, we write $\mathcal{L}(\{Y_i, Y_j\}) = \lambda_{ij}$ (and so $\lambda_{ij} = \lambda_{ji}$). *The system of equations induced by* G, denoted $\mathcal{E}_G$, is then defined as:

$$\mathcal{E}_G := \{Y_i \oplus Y_j = \lambda_{ij};\ \{Y_i, Y_j\} \in \mathcal{S}\}. \tag{2.1}$$

Thus, each vertex of G denotes a variable in the system of equations and each edge of G denotes an equation in $\mathcal{E}_G$. We denote the set of components in G by $\mathsf{comp}(G) = \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$, k being the total number of components. w_i denotes the size of (i.e. the number of vertices in) the component $\mathcal{C}_i$, $w_{\max}$ denotes the quantity $\max\{w_1, \ldots, w_k\}$ (also commonly denoted as ξ in Patarin's papers) and σ_i the sum $(w_1 + \cdots + w_i)$, with the convention $\sigma_0 = 0$.

Definition 1. *With respect to the system of equations $\mathcal{E}_G$ (as defined above), an injective function $\Phi : \mathcal{V} \to \{0,1\}^n$ is said to be an* injective solution *if $\Phi(Y_i) + \Phi(Y_j) = \lambda_{ij}$ for all $\{Y_i, Y_j\} \in \mathcal{S}$.*

As the graph G is acyclic, there exists a unique path between any two vertices Y_s and Y_t in the same component, which shall be denoted by P_{st}. Adding all equations induced by the edges of any such path P_{st} gives

$$\mathcal{L}(P_{st}) := \sum_{e \in P_{st}} \mathcal{L}(e).$$

So, for an injective solution to exist, the graph G (along with the label function $\mathcal{L}$) must satisfy the following property:

NPL (non-zero path label): *For all paths P in graph G, $\mathcal{L}(P) \neq \mathbf{0}$.*

It may be noted here that the NPL condition formalizes the notion of non-degeneracy mentioned in [120, 104]. The restriction on the graph to be acyclic implies that the equations are linearly independent (since otherwise, there is a possibility that the system becomes inconsistent).

Having identified the necessary condition for the existence of an injective solution to $\mathcal{E}_G$ corresponding to any simple edge-labeled undirected acyclic graph G, we now state the following claim due to Patarin [120], which gives a lower bound on the number of injective solutions to $\mathcal{E}_G$: Suppose G has α vertices and q edges. Then the number of injective solutions to $\mathcal{E}_G$ is at least $\frac{(2^n)_\alpha}{2^{nk}}$, provided $\sigma_k(w_{\max} - 1) \leq 2^n/64$. Unfortunately, the proof of this claim is unverifiable. [62] gives a detailed proof for the following lower bound on the number of injective solutions: $\frac{(2^n)_\alpha}{2^{nk}} \cdot (1 - \epsilon)$, with $\epsilon \approx 0$ and $\sigma_k^3 w_{\max}^2 \ll 2^{2n}$.

Injective Solution to a System of Equations and Non-Equations. We shall now examine an extended system involving a system of non-equations along with a system of equations. Let $G = (\mathcal{V} := \{Y_1, \ldots, Y_\alpha\}, \mathcal{S} \sqcup \mathcal{S}', \mathcal{L})$ be a simple undirected edge-labelled graph ($\mathcal{L}$ is a label function), whose edge set is partitioned into two disjoint sets $\mathcal{S}$ and $\mathcal{S}'$. As before, we simply write $\mathcal{L}(\{Y_i, Y_j\}) = \lambda_{ij}$ for all $\{Y_i, Y_j\} \in \mathcal{S}$ and $\mathcal{L}(\{Y_i, Y_j\}) = \lambda'_{ij}$ for all $\{Y_i, Y_j\} \in \mathcal{S}'$. Let such a graph G induce a system of equations and non-equations $\mathcal{E}_G$ as follows:

$$Y_i \oplus Y_j = \lambda_{ij} \; \forall \; \{Y_i, Y_j\} \in \mathcal{S}, \tag{2.2}$$

$$Y_i \oplus Y_j \neq \lambda'_{ij} \; \forall \; \{Y_i, Y_j\} \in \mathcal{S}', \tag{2.3}$$

For a system of equations and non-equations $\mathcal{E}_G$, an injective function $\Phi : \mathcal{V} \to \{0,1\}^n$ is said to be an *injective solution function* if $\Phi(Y_i) \oplus \Phi(Y_j) = \lambda_{ij}$ for all $\{Y_i, Y_j\} \in \mathcal{S}$ and $\Phi(Y_i) \oplus \Phi(Y_j) \neq \lambda'_{ij}$ for all $\{Y_i, Y_j\} \in \mathcal{S}'$.

Good Graphs. We shall first investigate the case when $\mathcal{E}_G$ has at least one solution. To ensure this, the subgraph $G^= := (\mathcal{V}, \mathcal{S}, \mathcal{L}_{|\mathcal{S}})$, where $\mathcal{L}_{|\mathcal{S}}$ is the function $\mathcal{L}$ restricted over the set $\mathcal{S}$, must

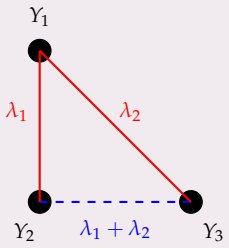


Figure 2.5.: $\mathcal{E}_G := \{Y_1 \oplus Y_2 = \lambda_1, Y_1 \oplus Y_3 = \lambda_2, Y_2 \oplus Y_3 \neq \lambda_1 \oplus \lambda_2\}$. The continuous red edges represent equations and the dashed blue edge represents a non-equation. Clearly, the system of equations and non-equations is inconsistent.

(i) be acyclic (i.e. **No Cycle** or **NC**)
(ii) satisfy the **NPL** condition and
(iii) satisfy the **NCL (non-zero cycle label)** property: *For all cycles C in G such that the edge set of C contains exactly one non-equation edge $e' \in \mathcal{S}'$, $\mathcal{L}(C) \neq \mathbf{0}$* (see Fig.4.1 for an example).

If a graph G satisfies the above three conditions (i)-(iii), it is said to be a **good graph**. In [62], authors have proved the following lower bound for $w_{\max} = 3$. Let $G = (\mathcal{V}, \mathcal{S} \sqcup \mathcal{S}', \mathcal{L})$ be a good graph with $|\mathcal{V}| = \alpha, |\mathcal{S}| = q_m, |\mathcal{S}'| = q_v$. Let $\mathsf{comp}(G^{=}) = \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$ with $|\mathcal{C}_i| = w_i$ (≤ 3) and $\sigma_i = (w_1 + \cdots + w_i)$. Let $\mathcal{Z} \subseteq \{0,1\}^n$ such that $|\{0,1\}^n \setminus \mathcal{Z}| = c$. The total number of injective solutions (each solution chosen from the set $\mathcal{Z}$) for the induced system of equations and non-equations $\mathcal{E}_G$ is at least:

$$\frac{(2^n)_\alpha}{2^{nk}} \left(1 - \frac{5k^3}{2^{2n}} - \frac{q_v + c\alpha}{2^{n-1}}\right). \tag{2.4}$$

Observe that $q_v + c\alpha$ is the number of non-equations, considering univariate non-equations arising from the constraint of each solution being from the set of size $2^n - c$. Now we state our theorem, which generalizes this result for any $w_{\max}$.

Theorem 4. *Let $G = (\mathcal{V}, \mathcal{S} \sqcup \mathcal{S}', \mathcal{L})$ be a good graph with α vertices such that $|\mathcal{S}| = q_m$ and $|\mathcal{S}'| = q_v$. Let $\mathsf{comp}(G^{=}) = \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$ and $|\mathcal{C}_i| = w_i$, $\sigma_i = (w_1 + \cdots + w_i)$. Then the total number of injective solutions chosen from a set $\mathcal{Z}$ of size $2^n - c$, for some $c \geq 0$, for the induced system of equations and non-equations $\mathcal{E}_G$ is at least:*

$$\frac{(2^n)_\alpha}{2^{nq_m}} \left(1 - \sum_{i=1}^{k} \frac{6\sigma_{i-1}^2 \binom{w_i}{2}}{2^{2n}} - \frac{2(q_v + c\alpha)}{2^n}\right), \tag{2.5}$$

provided $\sigma_k w_{\max} \leq 2^n/4$.

Idea of the Proof. The proof begins by counting the number of solutions in each of the k components. Let $\tilde{w}_{ij}$ be the number of edges from $\mathcal{S}'$ connecting vertices between the i^{th} and j^{th} components of $G^{=}$ and w'_i the number of edges in $\mathcal{S}'$ incident on $v_i \in \mathcal{V} \setminus G^{=}(\mathcal{V})$. It is easy to see that the number of solutions for the first component is exactly $(2^n - cw_1)$. We fix a solution and count the number of solutions for the second component, which is $(2^n - w_1w_2 - \tilde{w}_{1,2} - cw_2)$ as it must discard (i) w_1 values $(y_{i_1}, \ldots, y_{i_{w_1}})$ from the first component, (ii) $w_1(w_2 - 1)$ values $(y_{i_1} \oplus \mathcal{L}(P_j), \ldots, y_{i_{w_1}} \oplus \mathcal{L}(P_j))$ for all possible paths P_j from a fixed vertex to any other vertex in the second component and (iii) $cw_2 + \tilde{w}_{12}$ values to compensate for the fact that the set of admissible values $\mathcal{Z}$ is no longer a group. In general, the total number of solutions for the i^{th} component is at least $\prod_{i=1}^{k} \left(2^n - \sigma_{i-1}w_i - \sum_{j=1}^{i-1} \tilde{w}_{ij} - cw_i\right)$. Suppose there are k' vertices that do not belong to the set of vertices of the subgraph $G^{=}$. Fix such a vertex Y_{σ_k+i} and let us assume that w'_{σ_k+i} blue dashed edges are incident on it. If y_{σ_k+i} is a valid solution to the variable Y_{σ_k+i}, then (i) y_{σ_k+i} should be distinct from the previous σ_k assigned values, (ii) y_{σ_k+i} should be distinct from the $(i-1)$ values assigned to the variables that do not belong to the set of vertices of the subgraph $G^{=}(\mathcal{V})$ and (iii) y_{σ_k+i} should not take those w'_{σ_k+i} values.

Therefore, the total number of solutions is at least

$$\prod_{i=1}^{k} \left(2^n - \sigma_{i-1}w_i - \sum_{j=1}^{i-1} \tilde{w}_{ij} - cw_i\right) \cdot \prod_{i=1}^{k'} \left(2^n - \sigma_k - i + 1 - w'_{\sigma_k+i}\right). \tag{2.6}$$

The result follows after a few simple computations. □

Let us first consider the same problem as Thm. 4, but with only a system of affine equations, and on the entire field $\{0,1\}^n$:

Lemma 1. *Let $G = (\mathcal{V}, \mathcal{S}, \mathcal{L})$ be a simple edge-labelled undirected acyclic graph that satisfies the NPL condition. Let $\mathsf{comp}(G) = \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$ be the set of components of G such that $|\mathcal{C}_i| = w_i$ for each $i = 1, \ldots, k$, and let the number of edges in G be q. We denote by $\sigma_i = (w_1 + \ldots, w_i)$, the number of vertices in the first i components of G (and $\sigma_0 = 0$). Then the total number of injective solutions for the induced system of equations $\mathcal{E}_G$, denoted by h_α, is at least*

$$\frac{(2^n)_\alpha}{2^{nq}} \left(1 - \sum_{i=1}^{k} \frac{6\sigma_{i-1}^2 \binom{w_i}{2}}{2^{2n}}\right),$$

provided $\sigma_k w_{\max} \leq 2^n/4$, where $w_{\max} = \max\{w_1, \ldots, w_k\}$.

Proof. Consider the first component $\mathcal{C}_1$ of the graph G. Let $Y_{i_1} \in \mathcal{V}$ be an arbitrary vertex of $\mathcal{C}_1$. There are 2^n choices for assigning values to the variable Y_{i_1}. Let the value assigned to Y_{i_1} be y_{i_1}. For any other variable Y_{i_2} of $\mathcal{C}_1$, consider the path P from Y_{i_1} to Y_{i_2} and assign the value $y_{i_1} \oplus \mathcal{L}(P)$ to the variable Y_{i_2}. Let this value be y_{i_2}. Since the acyclic graph ensures that the path P is unique and $\mathcal{L}(P) \neq \mathbf{0}$ due to the NPL property, $y_{i_1} \neq y_{i_2}$. Therefore, assigned values to all other variables in $\mathcal{C}_1$ are different from y_{i_1}. Furthermore, if P_j and P_k are the paths from the vertex Y_{i_1} to Y_j and Y_k respectively and P is the (possibly empty) common prefix of P_j and P_k, then we can write $P_j = P\|P'_j$, $P_k = P\|P'_k$, so that

$$y_j \oplus y_k = \mathcal{L}(P_j) \oplus \mathcal{L}(P_k) = \mathcal{L}(P'_j) \oplus \mathcal{L}(P'_k) = \mathcal{L}(P'_j\|P'_k) \neq \mathbf{0},$$

where the last equality holds due to the NPL condition. That for all edges $\{j,k\} \in \mathcal{S}$, $y_j \oplus y_k = \lambda_{jk}$ is also a straightforward verification. Therefore, y_{i_1} sets the solution uniquely for all the variables in $\mathcal{C}_1$. Let $(y_{i_1}, \ldots, y_{i_{w_1}})$ denote one such possible (hence injective) solution for the variables in the first component. Once such a value is fixed for Y_{i_1}, we consider the second component.

We proceed with a similar computation for the second component $\mathcal{C}_2$. Let $Y_{i_{w_1+1}} \in \mathcal{V}$ be a variable in $\mathcal{C}_2$. For any *valid solution* $y_{i_{w_1+1}}$ for $Y_{i_{w_1+1}}$, we set $y_{i_{w_1+1}} \oplus \mathcal{L}(P)$ as a solution to the variable $Y_j \in \mathcal{V}$, where Y_j is an arbitrary vertex in $\mathcal{C}_2$ and P is the unique path from $Y_{i_{w_1+1}}$ to Y_j. Therefore, $y_{i_{w_1+1}}$ uniquely determines the values of the remaining $w_2 - 1$ variables. If $y_{i_{w_1+1}}$ is a valid assignment for $Y_{i_{w_1+1}}$, then $y_{i_{w_1+1}}$ must be –

- distinct from the values $y_{i_1}, \ldots, y_{i_{w_1}}$ already assigned to the variables in $\mathcal{C}_1$ and
- distinct from any value in $\{y_{i_1} \oplus \mathcal{L}(P_j), \ldots, y_{i_{w_1}} \oplus \mathcal{L}(P_j)\}$, for all possible paths P_j from the vertex $Y_{i_{w_1+1}}$ to any other vertex Y_j in $\mathcal{C}_2$.

Thus, at most $w_1 w_2$ values get discarded for assignment to the vertex $Y_{i_{w_1+1}}$, leaving at least $(2^n - w_1 w_2)$ choices of values for this vertex, hence also for injective solutions to all vertices of the second component.

In general, for the i^{th} component, once an injective solution is fixed for the previous $i-1$ components, there are at least $(2^n - w_1 w_i - \cdots - w_{i-1} w_i) = (2^n - \sigma_{i-1} w_i)$ ways for an injective solution to exist for the i^{th} component (as vertices of the first $i-1$ components are already assigned values). Hence, the total number of possible injective solutions for the induced system of equations is at least

$$h_\alpha \geq \prod_{i=1}^{k} (2^n - \sigma_{i-1} w_i)\,. \tag{2.7}$$

Recall that q and α are respectively the number of edges and vertices in G. Therefore,

$$h_\alpha \frac{2^{nq}}{(2^n)_\alpha} \geq \frac{2^{nq}}{(2^n)_\alpha} \prod_{i=1}^{k} (2^n - \sigma_{i-1} w_i) = \prod_{i=1}^{k} \frac{(2^n - \sigma_{i-1} w_i) 2^{n(w_i-1)}}{(2^n - \sigma_{i-1})_{w_i}}, \text{ where } (2.8)$$

$$(2^n - \sigma_{i-1})_{w_i} \leq 2^{nw_i} - 2^{n(w_i-1)} \left(\sigma_{i-1} w_i + \binom{w_i}{2} \right) + 2^{n(w_i-2)} \left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i - 1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right).$$

Plugging this inequality into Eqn. (2.8) gives

$$\begin{aligned}
& h_\alpha \frac{2^{nq}}{(2^n)_\alpha} \\
\geq & \prod_{i=1}^{k} \left(1 + \frac{2^{n(w_i-1)} \cdot \binom{w_i}{2} - 2^{n(w_i-2)} \cdot \left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i-1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right)}{2^{nw_i} - 2^{n(w_i-1)} (\sigma_{i-1} w_i + \binom{w_i}{2}) + 2^{n(w_i-2)} \left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i-1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right)} \right) \\
\geq & \prod_{i=1}^{k} \left(1 - \frac{\left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i-1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right)}{2^{2n} - 2^n (\sigma_{i-1} w_i + \binom{w_i}{2}) + \left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i-1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right)} \right) \\
\geq & \prod_{i=1}^{k} \left(1 - \frac{2 \left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i-1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right)}{2^{2n}} \right), \\
& \text{since } 2^n \left(\sigma_{i-1} w_i + \binom{w_i}{2} \right) - \left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i-1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right) \leq 2^{2n}/2 \\
& \text{when } \sigma_k w_{\max} \leq 2^n/4 \\
\geq & \left(1 - \sum_{i=1}^{k} \frac{6 \sigma_{i-1}^2 \binom{w_i}{2}}{2^{2n}} \right),
\end{aligned}$$

$$\text{since } \left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i-1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right) \leq 3 \sigma_{i-1}^2 \binom{w_i}{2} \left(\binom{w_i}{2} \sigma_{i-1}^2 + \binom{w_i}{2} (w_i-1) \sigma_{i-1} + \binom{w_i}{2} \frac{(w_i-2)(3w_i-1)}{12} \right) \leq 3 \sigma_{i-1}^2 \binom{w_i}{2}. \quad \square$$

Lemma 2 computes a bound on the total number of injective solutions when non-equation edges are incorporated:

Lemma 2. *Let $G = (\mathcal{V}, \mathcal{S} \sqcup \mathcal{S}', \mathcal{L})$ be a good graph such that $\mathcal{V} = \{Y_1, \ldots, Y_\alpha\}$, $|\mathcal{S}| = q_m$ and $|\mathcal{S}'| = q_v$. Let $\mathsf{comp}(G^=) = (\mathcal{C}_1, \ldots, \mathcal{C}_k)$ be the set of components of $G^=$ such that $|\mathcal{C}_i| = w_i$ for each $i = 1, \ldots, k$ and $\sigma_i = (w_1 + \ldots + w_i)$ the number of vertices in the first i components of $G^=$ ($\sigma_0 := 0$). For every $i \neq j \in [k]$, suppose there are $\tilde{w}_{ij}$ edges from $\mathcal{S}'$ connecting vertices of the i^{th} and j^{th} components of $G^=$. Let $|\mathcal{V} \setminus G^=(\mathcal{V})| = k'$ and for any vertex $v_i \in \mathcal{V} \setminus G^=(\mathcal{V})$, let w'_i be the*

number of blue dashed edges incident on v_i. Then the total number of injective solutions for the induced system of equations and non-equations $\mathcal{E}_G$, chosen from a set $\mathcal{Z}$ of size $2^n - c$ ($c \geq 0$), denoted h_α, is at least

$$\prod_{i=1}^{k} \left(2^n - \sigma_{i-1} w_i - \sum_{j=1}^{i-1} \tilde{w}_{ij} - c w_i \right) \cdot \prod_{i \in [k']} \left(2^n - \sigma_k - i - w'_i \right). \tag{2.9}$$

Proof. There are clearly $(2^n - cw_1)$ ways to assign values to any one of the vertices of the first component $\mathcal{C}_1$ of $G^=$, thus uniquely determining the values for the rest of the variables in $\mathcal{C}_1$. Thus, there are $(2^n - cw_1)$ ways for an injective solution to exist for the first component. Once such a solution is fixed for the first component, we consider the second component.

For any arbitrary vertex $Y_{i_{w_1+1}} \in \mathcal{V}$ in the second component $\mathcal{C}_2$ of $G^=$ a valid solution $y_{i_{w_1+1}}$ should not take the $w_1 w_2$ values constrained by the vertices of the first component. Additionally, as there are $\tilde{w}_{12}$ blue dashed edges connecting the components $\mathcal{C}_1$ and $\mathcal{C}_2$, there are $\tilde{w}_{12}$ paths from the vertex $Y_{i_{w_1+1}}$ to the vertices of the component $\mathcal{C}_1$. The size of the domain set $\mathcal{Z}$ restricts another cw_2 values to the assignment of $y_{i_{w_1+1}}$. Thus, at most $w_1 w_2 + \tilde{w}_{12} + cw_2$ values get discarded for assignment to $Y_{i_{w_1+1}}$, and as a result there are at least $(2^n - w_1 w_2 - \tilde{w}_{1,2} - cw_2)$ valid choices for $Y_{i_{w_1+1}}$. Once this value is assigned to $Y_{i_{w_1+1}}$, the remaining variables in the second component are assigned uniquely. Thus, there are $(2^n - w_1 w_2 - \tilde{w}_{12} - cw_2)$ ways for an injective solution to exist for the second component.

In general, once the injective solution is fixed for the previous $i-1$ components, there are at least $(2^n - \sigma_{i-1} w_i - \tilde{w}_{i1} - \ldots - \tilde{w}_{i,i-1} - cw_i)$ ways for an injective solution to exist for the i^{th} component. Hence, the total number of possible injective solutions for the induced system of equations and non-equations is at least

$$\prod_{i=1}^{k} \left(2^n - \sigma_{i-1} w_i - \sum_{j=1}^{i-1} \tilde{w}_{ij} - c w_i \right).$$

There may also exist vertices that do not belong to the set $G^=(\mathcal{V})$. Let there be k' such vertices. Fix such a vertex Y_{σ_k+i} and assume that w'_{σ_k+i} blue dashed edges are incident on Y_{σ_k+i}. If y_{σ_k+i} is a valid solution to the variable Y_{σ_k+i}, then it must –

- be distinct from the previous σ_k assigned values,
- be distinct from the $(i-1)$ values assigned to variables of the set $\mathcal{V} \setminus G^=(\mathcal{V})$, and
- not take the w'_{σ_k+i} values that violates the non-equality conditions of the w'_{σ_k+i} blue dashed edges.

Therefore, the number of valid choices for y_{σ_k+i} is at least $(2^n - \sigma_k - i + 1 - w'_{\sigma_k+i})$. Summarizing the above, the total number of possible injective solutions for the induced system of equations and non-equations is at least

$$\prod_{i=1}^{k} \left(2^n - \sigma_{i-1} w_i - \sum_{j=1}^{i-1} \tilde{w}_{ij} - c w_i \right) \cdot \prod_{i=1}^{k'} (2^n - \sigma_k - i + 1 - w'_{\sigma_k+i}),$$

which proves the result. □

Proof of Theorem 4

Lemma 2 bounds the number of injective solutions to $\mathcal{E}_G$ to at least $\prod_{i=1}^{k} (2^n - \sigma_{i-1} w_i - \tilde{w}_{i1} - \ldots - \tilde{w}_{i,i-1} - c w_i) \cdot \prod_{i \in [k']} \left(2^n - \sigma_k - i + 1 - w'_{\sigma_k+i}\right)$, where w_i is the size of the i^{th} component $\mathcal{C}_i$, $\sigma_{i-1} = (w_1 + \ldots + w_{i-1})$, k' is the number of vertices in $G \setminus G^{=}(\mathcal{V})$ and w'_{σ_k+i} is the number of blue dashed edges incident on the vertex Y_{σ_k+i}. Denoting $(\tilde{w}_{i1} + \ldots + \tilde{w}_{i,i-1})$ by p_i for notational ease, a similar computation as in Lemma 1 gives

$$\begin{aligned} h_\alpha \frac{2^{nq_m}}{(2^n)_\alpha} &\geq \frac{2^{nq_m}}{(2^n)_\alpha} \prod_{i=1}^{k} (2^n - \sigma_{i-1} w_i - p_i - c w_i) \prod_{i=1}^{k'} \left(2^n - \sigma_k - i + 1 - w'_{\sigma_k+i}\right) \\ &= \prod_{i=1}^{k} \frac{(2^n - \sigma_{i-1} w_i - p_i - c w_i)\, 2^{n(w_i-1)}}{(2^n - \sigma_{i-1})_{w_i}} \prod_{i=1}^{k'} \frac{\left(2^n - \sigma_k - i + 1 - w'_{\sigma_k+i}\right)}{(2^n - \sigma_k - i + 1)}. \end{aligned} \tag{2.10}$$

Now observe that

$$\begin{aligned}
&\prod_{i=1}^{k} \frac{(2^n - \sigma_{i-1}w_i - p_i - cw_i)\, 2^{n(w_i-1)}}{(2^n - \sigma_{i-1})_{w_i}} \\
\geq\ &\prod_{i=1}^{k} \left[1 - \frac{\binom{w_i}{2}\sigma_{i-1}^2 + \binom{w_i}{2}(w_i-1)\sigma_{i-1} + \binom{w_i}{2}\frac{(w_i-2)(3w_i-1)}{12}}{2^{2n} - 2^n(\sigma_{i-1}w_i + \binom{w_i}{2}) + \left(\binom{w_i}{2}\sigma_{i-1}^2 + \binom{w_i}{2}(w_i-1)\sigma_{i-1} + \binom{w_i}{2}\frac{(w_i-2)(3w_i-1)}{12}\right)} \right. \\
&\left. - \frac{2^n(p_i + cw_i)}{2^{2n} - 2^n(\sigma_{i-1}w_i + \binom{w_i}{2}) + \left(\binom{w_i}{2}\sigma_{i-1}^2 + \binom{w_i}{2}(w_i-1)\sigma_{i-1} + \binom{w_i}{2}\frac{(w_i-2)(3w_i-1)}{12}\right)} \right] \\
\geq\ &\prod_{i=1}^{k} \left[1 - \frac{2\left(\binom{w_i}{2}\sigma_{i-1}^2 + \binom{w_i}{2}(w_i-1)\sigma_{i-1} + \binom{w_i}{2}\frac{(w_i-2)(3w_i-1)}{12}\right)}{2^{2n}} - \frac{2(p_i + cw_i)}{2^n} \right] \\
&\text{since } 2^n \left(\sigma_{i-1}w_i + \binom{w_i}{2}\right) - \left(\binom{w_i}{2}\sigma_{i-1}^2 + \binom{w_i}{2}(w_i-1)\sigma_{i-1} \right. \\
&\left. + \binom{w_i}{2}\frac{(w_i-2)(3w_i-1)}{12}\right) \leq 2^{2n}/2 \text{ when } \sigma_k w_{\max} \leq 2^n/4 \\
\geq\ &\left(1 - \sum_{i=1}^{k} \frac{6\sigma_{i-1}^2\binom{w_i}{2}}{2^{2n}} - \sum_{i=1}^{k} \frac{2(p_i + cw_i)}{2^n}\right) \qquad (2.11)
\end{aligned}$$

because $\binom{w_i}{2}\sigma_{i-1}^2 + \binom{w_i}{2}(w_i-1)\sigma_{i-1} + \binom{w_i}{2}\frac{(w_i-2)(3w_i-1)}{12} \leq 3\sigma_{i-1}^2\binom{w_i}{2}$. Moreover, as the total number of blue dashed edges across the components of $G^{=}$ is denoted by $q'_v = p_1 + \cdots + p_k$ and $w_1 + \ldots + w_k \leq \alpha(p_1 + \ldots + p_k) = q'_v$, the expression 2.11 is

$$\geq \left(1 - \sum_{i=1}^{k} \frac{6\sigma_{i-1}^2\binom{w_i}{2}}{2^{2n}} - \frac{2q'_v}{2^n} - \frac{2c\alpha}{2^n}\right),$$

and $\prod_{i=1}^{k'} \frac{\left(2^n-\sigma_k-i+1-w'_{\sigma_k+i}\right)}{(2^n-\sigma_k-i+1)}$

$$= \prod_{i=1}^{k'} \frac{\left(2^n-\sigma_k-i+1-w'_{\sigma_k+i}\right)}{(2^n-\sigma_k-i+1)} \geq \prod_{i=1}^{k'} \left(1-\frac{w'_{\sigma_k+i}}{(2^n-\sigma_k-i+1)}\right)$$

$$\geq \left(1-\sum_{i=1}^{k'} \frac{2w'_{\sigma_k+i}}{2^n}\right),$$

which follows from the fact that $(\sigma_k+i-1) \leq 2^n/2$,

$$\geq \left(1-\frac{2q''_v}{2^n}\right)$$ since we denote the total number of blue dashed edges incident on the vertices outside of the set $G^{=}(\mathcal{V})$ by $(w'_{\sigma_k+1}+\ldots+w'_{\sigma_k+k'}) = q''_v$.

Thus, $h_\alpha \frac{2^{nq_m}}{(2^n)_\alpha} \geq \left(1-\sum_{i=1}^{k} \frac{6\sigma_{i-1}^2 \binom{w_i}{2}}{2^{2n}} - \frac{2(q_v+c\alpha)}{2^n}\right)$ from Eqn. 2.10,

where $q_v = q'_v + q''_v$, the total number of non-equation edges in G. □

2.4. Mutlicollision in a Universal Hash Function

This section explores some results on the number of multicollisions in the outputs of a universal hash function. Suppose H_{K_h} is an ϵ-universal hash function with the hash key K_h chosen uniformly at random from the hash key space. For any q distinct messages $M_1,\ldots,M_q$, the probability that there exists $i \neq j$, such that M_i and M_j collide under the hash function H_{K_h} is at most $\epsilon\binom{q}{2}$ (by the union bound). We say that $(M_1,\ldots,M_\xi)$ is a *ξ-multicollision tuple* for H_{K_h} if $\mathsf{H}_{K_h}(M_1) = \mathsf{H}_{K_h}(M_2) = \cdots = \mathsf{H}_{K_h}(M_\xi)$. Then extending the 2-collision bound for multicollisions, the probability that a ξ-tuple $(M_1,\ldots,M_\xi)$ is a ξ-multicollision tuple for a ξ-wise independent hash function [133] H_{K_h} is $1/2^{n(\xi-1)}$. Clearly, this cannot be concluded for a universal hash function. In fact, one can easily construct a ξ-tuple of messages such that the multicollision probability under the PolyHash function is $\ell/2^n$.

In the following, is a better bound on the existence of a multicollision tuple for any collection of q messages; the proof can be found in 2.4.

Theorem 5 (Multicollision Theorem). *Let $X_1,\ldots,X_q$ be q distinct messages and H_{K_h} an ϵ-universal hash function. Then for $\xi \in \mathbb{N}$, the probability that a $(\xi+1)$-multicollision tuple exists in this set of messages is no more than $q^2\epsilon/2\xi$.*

Proof. Consider the graph $G = (\mathcal{V}, \mathcal{S})$ with vertex set $\mathcal{V}$ containing each of the q messages $Z_i := \mathsf{H}_{K_h}(X_i)$, $i \in [q]$. Let $\mathbf{X}$ denote a $(\xi+1)$-tuple $(X_1, \ldots, X_{\xi+1}) \in \mathcal{V}^{\xi+1}$. An edge between two nodes exists in $\mathcal{S}$ if and only if the hash values of the corresponding messages collide. Therefore, the event $\mathsf{H}_{K_h}(X_1) = \ldots = \mathsf{H}_{K_h}(X_{\xi+1})$ boils down to the existence of a clique of size $\xi+1$ in G. Due to Lemma 3, if G has $\left\lceil \frac{q^2}{2\xi} \right\rceil$ edges, then any collection of $\xi+1$ (out of the total q) vertices in $\mathcal{V}$ must contain at least one pair which is in $\mathcal{S}$. Also, for $s = q^2/\xi$, there must exist a multiset $\{v_1, \ldots, v_s\} \subseteq [q]$ (with $v_{2i-1} \neq v_{2i}$, $v_{2i-1} = v_{2j-1} \implies v_{2i} \neq v_{2j}$ and $v_{2i} = v_{2j} \implies v_{2i-1} \neq v_{2j-1}$ for all $i, j \in [s/2]$) such that

$$\begin{aligned} &Z_1 = Z_2 = \ldots = Z_{\xi+1} \Rightarrow \\ &\quad Z_{v_1} = Z_{v_2} \vee Z_{v_3} = Z_{v_4} \vee \ldots \vee Z_{v_s-1} = Z_{v_s} \text{ and } |\{v_1, \ldots, v_s\}| \geq \xi, \end{aligned} \tag{2.12}$$

hence bounding the required probability as follows:

$$\begin{aligned} &\max_{\mathbf{X}} \Pr\left[K_h \xleftarrow{\$} \{0,1\}^n : \exists i_1, \cdots, i_\xi \in [q], \mathsf{H}_{K_h}(X_{i_1}) = \cdots = \mathsf{H}_{K_h}(X_{i_\xi})\right] \\ \leq\ & \Pr[Z_{v_1} = Z_{v_2} \vee \ldots \vee Z_{v_s-1} = Z_{v_s}] \leq \sum_{i=1}^{s/2} \Pr[Z_{v_i} = Z_{v_{i+1}}] \leq \frac{s\epsilon}{2} = \frac{q^2\epsilon}{2\xi}. \end{aligned}$$

Lemma 3. *Let $q, \xi \in \mathbb{N}$. Then for any set $\mathcal{V}$ with $|\mathcal{V}| = q$, there exists a graph $G = (\mathcal{V}, \mathcal{S})$ with $|\mathcal{S}| = \left\lceil \frac{q^2}{2\xi} \right\rceil$ such that any collection C of $\xi+1$ vertices has at least one edge in $\mathcal{S}$ joining two vertices in C.*

Proof. Divide the q vertices into ξ subcollections of size $\left\lceil \frac{q}{\xi} \right\rceil$ each, the last subcollection possibly containing a lesser number of vertices. Suppose $\mathcal{S}$ contains all the edges required to form cliques C_i ($i \in [\xi]$). As there are at most $\xi \cdot \binom{\left\lceil \frac{q}{\xi} \right\rceil}{2}$ edges in all the ξ cliques,

$$\xi \cdot \binom{\frac{q}{\xi}}{2} < \frac{q^2}{2\xi} \leq \left\lceil \frac{q^2}{2\xi} \right\rceil = |\mathcal{S}|.$$

Hence, $\mathcal{S}$ must contain more edges, distinct from those involved in the ξ cliques, which must exist between at least one pair of vertices in different cliques C_i and C_j ($i \neq j$). Since there are $\xi+1$ vertices in C and a total of ξ cliques C_i formed so far, it can thus be inferred from the pigeonhole principle that at least one clique C_i contains more than one edge from $\mathcal{S}$, making clear the existence of an edge from $\mathcal{S}$ in C. □

2.5. Proof of Theorem 2

In this section, we prove Theorem 2. We shall also refer to the construction $\mathsf{nEHtM}[\mathsf{E},\mathsf{H}]$ as simply nEHtM when the underlying primitives are understood.

The first step of the proof is the standard switch from the computational setting to an information theoretic one by replacing the block cipher $\mathsf{E}_{\mathbf{k}}$ with an n-bit uniform random permutation Π at a cost of $\mathbf{Adv}^{\mathsf{prp}}_{\mathsf{E}}(q_m+q_v,t')$, where $t'=O(t+(q_m+q_v)t_H)$ and t_H is the time required for computing the hash function. Let us denote this modified construction as $\mathsf{nEHtM}^*[\Pi,\mathsf{H}]$. Hence,

$$\mathbf{Adv}^{\mathsf{MAC}}_{\mathsf{nEHtM}}(q_m,q_v,t)\leq \mathbf{Adv}^{\mathsf{prp}}_{\mathsf{E}}(q_m+q_v,t')+\mathbf{Adv}^{\mathsf{MAC}}_{\mathsf{nEHtM}^*}(q_m,q_v,t). \quad (2.13)$$

To get an upper bound for $\mathbf{Adv}^{\mathsf{MAC}}_{\mathsf{nEHtM}^*}(q_m,q_v,t)$, we consider a perfect random oracle Rand, which on input (N,M) returns T, sampled uniformly at random from $\{0,1\}^n$, and an oracle Rej which always returns $\perp$ (i.e., rejects) for all inputs (N,M,T). Now, due to [56, 68, 62] we have

$$\mathbf{Adv}^{\mathsf{MAC}}_{\mathsf{nEHtM}^*}(q_m,q_v,t)\leq \max_{\mathsf{D}} \Pr[\mathsf{D}^{\mathsf{TG}[\Pi,\mathsf{H}_{\mathbf{k}_h}],\mathsf{VF}[\Pi,\mathsf{H}_{\mathbf{k}_h}]}=1]-\Pr[\mathsf{D}^{\mathsf{Rand},\mathsf{Rej}}=1],$$

where the maximum is taken over all non-trivial distinguishers D. This formulation allows us to apply the expectation method [85, 42] to prove that

$$\mathbf{Adv}^{\mathsf{MAC}}_{\mathsf{nEHtM}^*}(q_m,q_v,t)\leq \frac{48q_m^3}{2^{2n}}+\frac{12q_m^4\epsilon}{2^{2n}}+\frac{12\mu^2q_m^2}{2^{2n}}+\frac{q_m+2q_v}{2^n}+\frac{4q_m^3\epsilon}{2^n} + (2q_m+q_v)\mu\epsilon+q_v\epsilon. \quad (2.14)$$

Attack Transcript. Henceforth, we fix a deterministic non-trivial (i.e., one that makes no repeated queries) distinguisher D that interacts with

1. either the real oracle $(\mathsf{TG}[\Pi,\mathsf{H}_{\mathbf{k}_h}],\mathsf{VF}[\Pi,\mathsf{H}_{\mathbf{k}_h}])$ for a uniform random permutation Π and a random hashing key $\mathbf{k}_h$,
2. or the ideal oracle $(\mathsf{Rand},\mathsf{Rej})$,

making at most q_m queries to its left (authentication) oracle with at most μ faulty nonces, and at most q_v queries to its right (verification) oracle, and returning a single bit. Then,

$$\mathbf{Adv}(\mathsf{D})=\left|\Pr\left[\mathsf{D}^{\mathsf{TG}[\Pi,\mathsf{H}_{\mathbf{k}_h}],\mathsf{VF}[\Pi,\mathsf{H}_{\mathbf{k}_h}]}=1\right]-\Pr\left[\mathsf{D}^{\mathsf{Rand},\mathsf{Rej}}=1\right]\right|.$$

$$\text{Let } \tau_m:=\{(N_1,M_1,T_1),(N_2,M_2,T_2),\ldots,(N_{q_m},M_{q_m},T_{q_m})\}$$

be the list of authentication queries made by D and the corresponding responses it receives. Also let

$$\tau_v:=\{(N'_1,M'_1,T'_1,b'_1),(N'_2,M'_2,T'_2,b'_2),\ldots,(N'_{q_v},M'_{q_v},T'_{q_v},b'_{q_v})\}$$

be the list of verification queries made by D and the corresponding responses it receives, where for all j, $b'_j \in \{\top, \bot\}$ denotes the set of accept $(b'_j = \top)$ and reject $(b'_j = \bot)$ responses. The pair $\tau = (\tau_m, \tau_v)$ constitutes the query transcript of the attack. For convenience, we slightly modify the experiment to reveal to the distinguisher (after obtaining all responses corresponding to its queries, but before outputting its decision), the hashing key $\mathbf{k}_h$ if D interacts with the real world, or a uniformly random dummy key $\mathbf{k}_h$ if D interacts with the ideal world. Hence, the *extended transcript* of the attack is $\tau' = (\tau, \mathbf{k}_h)$. We shall often simply call a tuple $(N, M, T) \in \tau_m$ an *authentication query*, and a tuple $(N', M', T', b') \in \tau_v$ a *verification query*.

A transcript τ' is said to be an *attainable transcript* (with respect to D) if the probability of realizing it in the ideal world is non-zero. It must be noted that since attainability is with respect to the ideal world, any verification query (N'_i, M'_i, T'_i, b'_i) even in an attainable transcript $\tau' = (\tau, \mathbf{k}_h)$ is such that $b'_i = \bot$. We denote Θ to be the set of all attainable transcripts and X_{re} and X_{id} to be the random variables that take an extended transcript τ' induced by the real world and the ideal world respectively.

2.5.1. Definition and Probability of Bad Transcripts

In this section, we define and bound the probability of bad transcripts in the ideal world. For notational simplicity, we denote $N_i \oplus \mathsf{H}_{\mathbf{k}_h}(M_i)$ as X_i. Note that X_i is an $(n-1)$-bit string.

Definition 2 (Bad Transcript). *Given a parameter $\xi \in \mathbb{N}$, where $\xi \geq \mu$ (optimal value of ξ determined later in the proof), an attainable transcript $\tau' = (\tau_m, \tau_v, \mathbf{k}_h)$ is called a* bad *transcript if any one of the following holds:*

- B1 : $\exists\, i_1 \in [q_m]$ *such that* $T_{i_1} = \mathbf{0}$.
- B2 : $\exists\, i_1 \neq i_2 \neq i_3$ *such that* $N_{i_1} = N_{i_2}$ *and* $X_{i_2} = X_{i_3}$.
- B3 : $\{i_1, \ldots, i_{\xi+1}\} \subseteq [q_m]$ *such that* $X_{i_1} = X_{i_2} = \ldots = X_{i_{\xi+1}}$.
- B4 $\exists\, a \in [q_v]$, $\exists\, i \in [q_m]$ *such that* $N_i = N'_a$, $X_i = X'_a$ *and* $T_i = T'_a$.

We denote by Θ_{bad} (resp. Θ_{good}) the set of bad (resp. good) transcripts. We bound the probability of bad transcripts in the ideal world as follows.

Lemma 4. *Let X_{id} and Θ_{bad} be defined as above. Then*

$$\Pr[X_{\text{id}} \in \Theta_{\text{bad}}] \leq \epsilon_{\text{bad}} = \frac{q_m}{2^n} + \frac{q_m^2 \epsilon}{2\xi} + (2q_m + q_v)\mu\epsilon + q_v\epsilon.$$

Proof. By the union bound,

$$\Pr[X_{\text{id}} \in \Theta_{\text{bad}}] \leq \Pr[\text{B1}] + \Pr[\text{B2}] + \Pr[\text{B3}] + \Pr[\text{B4}]. \tag{2.15}$$

We now bound the probabilities of all bad events individually. The lemma will follow by adding the individual bounds. Clearly,

$$\Pr[\mathsf{B1}] \leq \frac{q_m}{2^n}. \tag{2.16}$$

Bounding B2. Let $\mathcal{F}$ be the set of all query indices i_1 for which there is a $i_2 \neq i_1$ such that $N_{i_1} = N_{i_2}$. It is easy to see that $|\mathcal{F}| \leq 2\mu$. Event B2 occurs if for some $i_2 \in \mathcal{F}$, $\mathsf{H}_{\mathbf{k}_h}(M_{i_2}) = N_{i_3} \oplus \mathsf{H}_{\mathbf{k}_h}(M_{i_3})$ for some $i_3 \neq i_2$. For any such fixed i_1, i_2, i_3, the probability of the event is at most ϵ. The number of such choices of (i_2, i_3) is at most $2\mu q_m$. Hence,

$$\Pr[\mathsf{B2}] \leq 2\mu q_m \epsilon. \tag{2.17}$$

Bounding B3. Event B3 occurs if there exist $\xi + 1$ distinct authentication query indices $\{i_1, \ldots, i_{\xi+1}\} \subseteq [q_m]$ such that $X_{i_1} = \ldots = X_{i_{\xi+1}}$. This event is thus a $(\xi+1)$-multicollision on the ϵ-universal hash function mapping (N, M) to $\mathsf{H}_{\mathbf{k}_h}(M) \oplus N$ (as $\mathsf{H}_{\mathbf{k}_h}$ is an ϵ-almost xor universal). Therefore by Theorem 5,

$$\Pr[\mathsf{B3}] \leq q_m^2 \epsilon / 2\xi. \tag{2.18}$$

Bounding B4. For some $a \in [q_v]$ and $i \in [q_m]$, if $N_i = N'_a$, $X_i = X'_a$ and $T_i = T'_a$, then $M_i \neq M'_a$ (as the adversary does not make any trivial query). Hence the probability that $X_i = X'_a$ holds is at most ϵ. Now, for any a, there can be at most $(\mu + 1)$ indices i such that $N_i = N'_a$. Hence, the required probability is bounded as

$$\Pr[\mathsf{B4}] \leq (\mu + 1) q_v \epsilon. \tag{2.19}$$

The proof follows from Eqn.s (2.15)-(2.19). □

2.5.2. Analysis of Good Transcripts

In this section, we show that for a good transcript $\tau' = (\tau, \mathbf{k}_h)$, realizing τ' is almost as likely in the real world as in the ideal world. Consider a good transcript $\tau' = (\tau_m, \tau_v, \mathbf{k}_h)$. Since the authentication oracle is perfectly random and the verification oracle always rejects in the ideal world,

$$\Pr[X_{\mathrm{id}} = \tau'] = \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{2^{nq_m}} \tag{2.20}$$

We must now lower bound $\Pr[X_{\mathrm{re}} = \tau']$, i.e. the probability of getting τ' in the real world. We say that a permutation Π is *compatible with* τ_m (respectively with τ_v) if (A) (respectively (B)) holds:

$$\text{(A) } \forall i \in [q_m], \Pi(\widehat{N}_i) \oplus \Pi(\widehat{X}_i) = T_i, \text{ (B) } \forall a \in [q_v], \Pi(\widehat{N'}_a) \oplus \Pi(\widehat{X'}_a) \neq T'_a,$$

where $\widehat{N}_i = 0\|N_i$, $\widehat{X}_i = 1\|X_i$, $\widehat{N'}_a = 0\|N'_a$ and $\widehat{X'}_a = 1\|X'_a$. We simply say that Π is compatible with τ if it is compatible with both τ_m and τ_v. We denote by $\mathsf{Comp}(\tau)$ the set of permutations Π that are compatible with τ. Therefore,

$$\begin{aligned}\Pr[X_{\mathrm{id}} = \tau'] &= \frac{1}{|\mathcal{K}_h|} \cdot \Pr[\Pi \xleftarrow{\$} \mathsf{Perm} : \Pi \in \mathsf{Comp}(\tau)] \\ &= \frac{1}{|\mathcal{K}_h|} \cdot \underbrace{\Pr[\Pi(\widehat{N}_i) \oplus \Pi(\widehat{X}_i) = T_i, \Pi(\widehat{N'}_a) \oplus \Pi(\widehat{X'}_a) \neq T'_a]}_{\mathsf{P}_{mv}}. \quad (2.21)\end{aligned}$$

We refer to the system of equations as "*authentication equations*" as they involve only the authentication queries and to the system of non-equations as "*verification non-equations*" as they involve only the verification queries. We denote the system of authentication equations by $\mathcal{E}_m$ and the system of verification non-equations by $\mathcal{E}_v$:

$$\mathcal{E}_m = \begin{cases} \Pi(\widehat{N}_1) \oplus \Pi(\widehat{X}_1) = T_1 \\ \Pi(\widehat{N}_2) \oplus \Pi(\widehat{X}_2) = T_2 \\ \mathcal{V}dots \\ \Pi(\widehat{N}_{q_m}) \oplus \Pi(\widehat{X}_{q_m}) = T_{q_m} \end{cases} \qquad \mathcal{E}_v = \begin{cases} \Pi(\widehat{N'}_1) \oplus \Pi(\widehat{X'}_1) \neq T'_1 \\ \Pi(\widehat{N'}_2) \oplus \Pi(\widehat{X'}_2) \neq T'_2 \\ \mathcal{V}dots \\ \Pi(\widehat{N'}_{q_v}) \oplus \Pi(\widehat{X'}_{q_v}) \neq T'_{q_v} \end{cases}$$

Equation and Non-Equation Inducing Graph. From the above system of bivariate affine equations and non-equations, we induce the edge-labelled undirected graph $G_{\tau'} = (\mathcal{V}, \mathcal{S} \sqcup \mathcal{S}')$, where the set of nodes $\mathcal{V}$ is the set of variables $\{Y_1, \ldots, Y_\alpha\}$, $\mathcal{S}$ is the set of edges corresponding to each authentication equation and $\mathcal{S}'$ is the set of edges corresponding to each verification non-equation. Moreover, if there is an authentication equation $Y_s \oplus Y_t = T_i$, then the corresponding edge $\{Y_s, Y_t\} \in \mathcal{S}$ is labelled T_i. Similarly, if there is a verification non-equation $Y_s \oplus Y_t \neq T'_i$, then the corresponding edge $\{Y_s, Y_t\} \in \mathcal{S}'$ is labelled T'_i. Moreover, $G^{=}_{\tau'} = (\mathcal{V}, \mathcal{S})$ is a subgraph of $G_{\tau'}$.

Claim: *If the transcript τ' is good, then the induced graph $G_{\tau'}$ is good.*
Proof of the Claim: To prove that $G_{\tau'}$ is good, we need to show that

1. $G^{=}_{\tau'}$ is acyclic,
2. $G_{\tau'}$ satisfies the NPL condition, and
3. $G_{\tau'}$ satisfies the NCL condition.

For this, we inherit the notations introduced while analyzing the probability of good transcripts in the proof of Theorem 2: $\widehat{N}_i$ denotes $0\|N_i$, $\widehat{X}_i$ denotes $1\|X_i$, $\widehat{N'}_a$ denotes $0\|N'_a$ and $\widehat{X'}_a$ denotes $1\|X'_a$ where $X_i = N_i \oplus \mathsf{H}_{\mathbf{k}_h}(M_i)$.

$G^{=}_{\tau'}$ is acylic. For the sake of contradiction, let us assume there is a cycle C in the graph $G^{=}_{\tau}$. If $|C| = 2$, then there must exist two authentication equations

$$\Pi(\widehat{N}_{i_1}) \oplus \Pi(\widehat{X}_{i_1}) = T_{i_1}, \qquad \Pi(\widehat{N}_{i_2}) \oplus \Pi(\widehat{X}_{i_2}) = T_{i_2}$$

in $\mathcal{E}_m$ with $N_{i_1} = N_{i_2}$ and $X_{i_1} = X_{i_2}$. But this event is exactly the bad event (B2) in Definition 2. As the transcript τ' is good, this event cannot hold and therefore, there cannot be any cycle of length 2 in $G^{=}_{\tau'}$. A careful observation reveals that there cannot be any cycle of length 3 in the graph. Moreover, if there is any cycle of length at least 4 in $G^{=}_{\tau'}$, there must exist three authentication equations

$$\Pi(\widehat{N}_{i_1}) \oplus \Pi(\widehat{X}_{i_1}) = T_{i_1}, \quad \Pi(\widehat{N}_{i_2}) \oplus \Pi(\widehat{X}_{i_2}) = T_{i_2}, \quad \Pi(\widehat{N}_{i_3}) \oplus \Pi(\widehat{X}_{i_3}) = T_{i_3}$$

in $\mathcal{E}_m$ with $N_{i_1} = N_{i_2}$ and $X_{i_2} = X_{i_3}$. But this event is again the bad event (B2) in Definition 2. As the transcript τ' is good, this event cannot occur and therefore, there cannot be any cycle in $G^{=}_{\tau'}$ with length at least 4. This shows that $G^{=}_{\tau'}$ is acyclic.

$G_{\tau'}$ satisfies NPL. Each edge-label in the graph is non-zero as τ' is good. Consider any path P of length 2 in $G^{=}_{\tau}$. Let the edge-labels of the edges in the path be T_{i_1} and T_{i_2}. This implies that there must be two authentication equations

$$\Pi(\widehat{N}_{i_1}) \oplus \Pi(\widehat{X}_{i_1}) = T_{i_1} \text{ and } \Pi(\widehat{N}_{i_2}) \oplus \Pi(\widehat{X}_{i_2}) = T_{i_2}$$

in $\mathcal{E}_m$ with $N_{i_1} = N_{i_2}$ or $X_{i_1} = X_{i_2}$. If $T_{i_1} = T_{i_2}$, then this would create a cycle of length 2 in $G^{=}_{\tau'}$, which is impossible as $G^{=}_{\tau'}$ is acyclic. Therefore, there cannot be any path of length 2 in $G^{=}_{\tau'}$ such that the path-label becomes zero. Moreover, one cannot have any path of length at least 3 in $G^{=}_{\tau'}$ as otherwise, the bad condition (B2) would be satisfied. Therefore, $G_{\tau'}$ satisfies **Non-zero path label** condition.

$G_{\tau'}$ satisfies NCL. Consider first a cycle of length 2, where one edge is a blue dotted edge. Then there must be one authentication equation and one verification non-equation

$$\Pi(\widehat{N}_i) \oplus \Pi(\widehat{X}_i) = T_i, \qquad \Pi(\widehat{N'}_a) \oplus \Pi(\widehat{X'}_a) \neq T'_a,$$

respectively, such that $N_i = N'_a, X_i = X'_a$ and $T_i = T'_a$. However, this implies that the event satisfies the bad condition (B4) in Definition 2. As the transcript τ' is good, this event cannot occur and therefore, there cannot be any cycle of length 2 with one blue dotted edge. Moreover as argued before, there cannot be any cycle of length 3 with exactly one non-equation edge. Now, for the existence of a cycle with length at least 4 that contains exactly one non-equation edge, there must exist a path with minimum length 3 in $G^{=}_{\tau'}$. This is clearly impossible, ensuring that $G_{\tau'}$ satisfies the **Non-zero cycle label** condition. □

Suppose there are k components in the subgraph $G^{=}_{\tau'}$ and the size of the i^{th} component is W_i. Thus, W_i is a random variable, and so is $W_{\max}$, which denotes the size of the largest component. It is easy to see that $W_{\max} \leq \xi$. As the graph $G_{\tau'}$ is good (follows from the claim above), we assume $\xi \leq 2^n/8q_m$, which allows us to apply Theorem 4 with $c = 0$ to obtain

$$\mathsf{P}_{mv} \geq \frac{1}{2^{nq_m}} \cdot \left(1 - \sum_{i=1}^{k} \frac{6\sigma_{i-1}^2 \binom{W_i}{2}}{2^{2n}} - \frac{2q_v}{2^n}\right). \tag{2.22}$$

Therefore, Eqn.s (2.20)-(2.22) imply that the ratio $\frac{\Pr[X_{\text{re}}=\tau']}{\Pr[X_{\text{id}}=\tau']}$ is no less than

$$\left(1 - \sum_{i=1}^{k} \frac{6\sigma_{i-1}^2 \binom{W_i}{2}}{2^{2n}} - \frac{2q_v}{2^n}\right) \geq 1 - \left(\sum_{i=1}^{k} \frac{24q_m^2 \binom{W_i}{2}}{2^{2n}} + \frac{2q_v}{2^n}\right), \tag{2.23}$$

since $\sigma_{i-1} \leq 2q_m$. Define $\phi(\tau') := \sum_{i=1}^{k} \frac{24q_m^2 \binom{W_i}{2}}{2^{2n}} + \frac{2q_v}{2^n}$. We now compute the expectation of $\phi(X_{\text{id}})$ as follows:

$$\mathbf{E}\left[\left(\sum_{i=1}^{k} \frac{24q_m^2 \binom{W_i}{2}}{2^{2n}} + \frac{2q_v}{2^n}\right)\right] = \left(\frac{2q_v}{2^n} + \frac{24q_m^2}{2^{2n}} \mathbf{E}\left[\sum_{i=1}^{k} \binom{W_i}{2}\right]\right). \tag{2.24}$$

Let $\widetilde{W}_i = W_i - 1$ and therefore,

$$\mathbf{E}\left[\sum_{i=1}^{k} \binom{W_i}{2}\right] = \mathbf{E}\left[\sum_{i=1}^{k} \binom{\widetilde{W}_i}{2}\right] + \mathbf{E}\left[\sum_{i=1}^{k} \widetilde{W}_i\right] \overset{(2)}{\leq} \mathbf{E}\left[\sum_{i=1}^{k} \binom{\widetilde{W}_i}{2}\right] + 2q_m \tag{2.25}$$

due to the fact that $(\widetilde{W}_1 + \ldots \widetilde{W}_k) = \sigma_k - k \leq 2q_m$.
Next, consider the following two indicator random variables:

$$\mathbb{1}_{i_1 i_2} = \begin{cases} 1, & \text{if } X_{i_1} = X_{i_2} \\ 0, & \text{otherwise} \end{cases} \qquad \tilde{\mathbb{1}}_{i_1 i_2} = \begin{cases} 1, & \text{if } N_{i_1} = N_{i_2} \\ 0, & \text{otherwise.} \end{cases}$$

Therefore, $\mathbf{E}\left[\sum_{i=1}^{k} \binom{\widetilde{W}_i}{2}\right]$

$$\begin{aligned}
&= \sum_{i_1 \neq i_2}^{q_m} \mathbf{E}[\mathbb{1}_{i_1 i_2}] + \sum_{i_1 \neq i_2}^{\mu} \mathbf{E}[\tilde{\mathbb{1}}_{i_1 i_2}] \text{ (by linearity of expectation)} \\
&= \sum_{i_1 \neq i_2}^{q_m} \Pr[\mathsf{H}_{\mathbf{k}_h}(M_{i_1}) \oplus \mathsf{H}_{\mathbf{k}_h}(M_{i_2}) = N_{i_1} \oplus N_{i_2}] + \mu^2/2 \text{ (by definition)} \\
&\leq \binom{q_m}{2}\epsilon + \mu^2/2 \text{ (by the } \epsilon\text{-almost xor universal probability} \\
&\qquad\qquad\qquad\qquad \text{of the underlying hash function)} \\
&\leq q_m^2\epsilon/2 + \mu^2/2.
\end{aligned} \tag{2.26}$$

Therefore from Eqn.s (2.24)-(2.26),

$$\mathbf{E}[\phi(X_{\mathrm{id}})] \leq \left(\frac{12q_m^4\epsilon}{2^{2n}} + \frac{12\mu^2 q_m^2}{2^{2n}} + \frac{48q_m^3}{2^{2n}} + \frac{2q_v}{2^n}\right). \tag{2.27}$$

Finally, we have assumed $\xi \geq \mu$ and $\xi \leq 2^n/8q_m$; if $\mu \leq 2^n/8q_m$, we also choose $\xi = 2^n/8q_m$ (if not, the bound becomes vacuously true). The result then follows from Eqn. (1.2), Lemma 4 and Eqn. (2.27). □

2.5.3. Security Bound Using the Coefficients-H Technique

We instantiate the underlying hash function of nEHtM by a truncated n-bit $2\ell/2^n$-axu PolyHash function that truncates the first bit of the output [61], where ℓ is the maximum number of message blocks. Therefore, from Lemma 4, Eqn. (2.23) and the inequality $\sum_{i=1}^{k} \binom{W_i}{2} \leq \xi q_m$, we obtain the following bound using the coefficients-H technique:

$$\delta_{\mathrm{hc}} \leq \frac{q_m + 2q_v}{2^n} + \frac{q_m^2\ell}{2^n\xi} + \frac{(2q_m + q_v)2\ell\mu}{2^n} + \frac{2q_v\ell}{2^n} + \frac{24q_m^3\xi}{2^{2n}}. \tag{2.28}$$

We choose the optimal value of ξ such that the right hand side of Eqn. (2.28) gets maximized. This happens when $\frac{q_m^2\ell}{2^n\xi} = \frac{24q_m^3\xi}{2^{2n}}$. Solving this equality for ξ gives $\xi_{\mathrm{opt}} = \left(\frac{\ell 2^n}{24q_m}\right)^{\frac{1}{2}}$. Plugging in this value into Eqn. (2.28) then gives

$$\delta_{\mathrm{hc}} \leq \frac{q_m + 2q_v}{2^n} + \frac{(2q_m + q_v)2\ell\mu}{2^n} + \frac{2q_v\ell}{2^n} + 10\left(\frac{q_m^5\ell}{2^{3n}}\right)^{\frac{1}{2}}.$$

The above bound holds as long as $q \leq 2^{3n/5}/\ell^{1/5} \approx O(2^{3n/5})$, which is weaker than the bound $O(2^{2n/3})$ obtained using the expectation method.

2.6. Proof of Theorem 3

Instead of separately proving the privacy and authenticity of the construction, this section bounds the distinguishing advantage of the following random systems: (i) the pair of oracles $(\mathsf{CWC+.Enc}, \mathsf{CWC+.Dec})$ for a random permutation Π, which is called the real system or the real world and (ii) the pair of oracles $(\mathsf{Rand}, \mathsf{Rej})$, which is called the ideal system or the ideal world. The privacy and authenticity bounds of CWC+ then follow as a simple corollary of this result. We prove the following information theoretic

bound of CWC+:

$$\delta^* \leq \frac{97\sigma^3\ell}{2^{2n}} + \frac{5\sigma}{2^n} + \frac{\sigma\ell}{2^n} + \frac{8\sigma^3}{2^{2n}} + \frac{2q_d}{2^\rho}\left(1 + \frac{\ell}{2^{n-\rho}}\right) + \frac{(2q_e + q_d)2\ell\mu}{2^n} + \left(\frac{5\sigma\ell\mu}{2^n}\right)^2, \quad (2.29)$$

assuming $q_e\ell \approx \sigma$, $\sigma \leq 2^n/48$. Here, δ^* is the maximum advantage of distinguishing the real world from the ideal world.

Description of the ideal world. We begin with the assumption that all messages queried by an adversary have lengths in multiples of n and that the i^{th} message has l_i blocks. Consider a deterministic distinguisher A that interacts either with the real world or with the ideal world. Rej simply rejects all the verification attempts of A whereas Rand works on the i^{th} encryption query (N_i, M_i, A_i) as shown in Fig. 2.6.

Rand(N_i, A_i, M_i)

1 : if $N_i \in \mathcal{D}$, let $N_i = N$.
2 : if $l_i = l_N$, then $S_i \leftarrow \mathcal{L}(N)$.
3 : if $l_i < l_N$, then $S_i \leftarrow \mathcal{L}(N)[1, nl_i]$.
4 : if $l_i > l_N$, then
5 : $R \xleftarrow{\$} (\{0,1\}^n)^{l_i - l_N}, S_i \leftarrow \mathcal{L}(N) \| R$.
6 : $l_N = l_i$.
7 : else
8 : $S_i \xleftarrow{\$} (\{0,1\}^n)^{l_i}$, $\mathcal{L}(N_i) \leftarrow S_i, l_{N_i} = l_i$.
9 : $\mathcal{D} \leftarrow \mathcal{D} \cup \{N_i\}$.
10 : $\widetilde{T}_i \xleftarrow{\$} \{0,1\}^n$, $T_i \leftarrow \mathsf{chop}_\rho(\widetilde{T}_i)$.
11 : return (S_i, T_i).

Figure 2.6.: Random oracle for the ideal world. l_N denotes the updated number of keystream blocks for nonce N and $\mathcal{L}(N)$ denotes the updated keystream blocks for nonce N of length l_N. $\mathcal{D}$ denotes the domain of the nonce. chop_ρ is a function that truncates the last $n - \rho$ bits of its input.

Attack Transcript. Let D be a fixed non-trivial computationally unbounded deterministic distinguisher that interacts with either the real world or the ideal world, making at most q_e queries to the left (encryption) oracle with at

most μ faulty nonces, and at most q_d queries to its right (decryption) oracle, returning a single bit.
Let $\tau_e := \{(N_1, M_1, A_1, S_1, T_1), \ldots, (N_{q_e}, M_{q_e}, A_{q_e}, S_{q_e}, T_{q_e})\}$ be the list of encryption queries and $\tau_d := \{(N'_1, A'_1, C'_1, T'_1, Z_1), \ldots, (N'_{q_d}, A'_{q_d}, C'_{q_d}, T'_{q_d}, Z_{q_d})\}$ be the list of decryption queries, where $Z_i = M_i \cup \{\perp\}$. Note that the encryption oracle in both worlds releases the keystream as it determines the ciphertext uniquely. For convenience, we reveal the hash key $\mathbf{k}_h$ (which is $\mathsf{E}_{\mathbf{k}}(\mathbf{0})$ if D interacts with the real world, and a uniform random element from $\{0,1\}^n$ if D interacts with the ideal world), and also the (un-truncated) n-bit tag $\mathbf{T} := (\widetilde{T}_1, \ldots, \widetilde{T}_{q_e})$ to the distinguisher after it has made all its queries and obtained the corresponding responses, but before it outputs its decision. Thus, the *extended transcript* of the attack is $\tau' = (\tau, \mathbf{k}_h, \widetilde{\mathbf{T}})$.

Bad Transcripts. Recall that N_i is a $3n/4$-bit string. We denote $0\|N_i\|0^{n/4-1}$ by $\widehat{N}_i$ and $1\|X_i$ by $\widehat{X}_i$, where $X_i := N_i\|0^{n/4-1} \oplus \mathsf{Poly}_{\mathbf{k}_h}(M_i)$. We also denote by $S_i[j]$, the j^{th} keystream block for the i^{th} message. With these notations, we define the bad transcript as follows: a transcript $\tau' = (\tau_e, \tau_d, \mathbf{k}_h, \widetilde{\mathbf{T}})$ is called **bad** if any one of the following holds:

- B.1 : $\exists\ i \in [q_e]$ and $j \in [l_i]$ such that $S_i[j] = \mathbf{k}_h$.
- B.2 : $\exists\ i \in [q_e]$ and $j \in [l_i]$ such that $S_i[j] = \mathbf{0}$.
- B.3 : $\exists\ i \in [q_e]$ and $j, j' \in [l_i]$ such that $S_i[j] = S_i[j']$.
- B.4 : $\exists\ i \in [q_e]$ such that $\widetilde{T}_i = \mathbf{0}$.
- B.5 : $\exists i_1 \neq i_2 \neq i_3$ such that $\widehat{N}_{i_1} = \widehat{N}_{i_2}$ and $\widehat{X}_{i_2} = \widehat{X}_{i_3}$.
- B.6 : $\{i_1, \ldots, i_{\xi+1}\} \subseteq [q_e]$ such that $\widehat{X}_{i_1} = \widehat{X}_{i_2} = \ldots = \widehat{X}_{i_{\xi+1}}$ for some parameter $\xi \geq \mu$.
- B.7 $\exists\ a \in [q_d]$, $i \in [q_e]$ such that $\widehat{N}_i = \widehat{N'}_a$, $\widehat{X}_i = \widehat{X'}_a$ and $\widetilde{T}_i = T'_a$.

Θ_{bad} (resp. Θ_{good}) denotes the set of bad (resp. good) transcripts, and X_{re} and X_{id} denote random variables that realize an extended transcript τ' in the real and the ideal world, respectively. We bound the probability of bad transcripts in the ideal world as follows.

Lemma 5. *Let X_{id} and Θ_{bad} be defined as above. Then*

$$\Pr[X_{\text{id}} \in \Theta_{\text{bad}}] \leq \epsilon_{\text{bad}} = \frac{2\sigma}{2^n} + \frac{q_e\ell^2}{2^n} + \frac{q_e}{2^n} + \frac{q_e^2\ell}{\xi 2^n} + \frac{(2q_e + q_d)2\ell\mu}{2^n} + \frac{2q_d\ell}{2^n}.$$

Proof. By the union bound,

$$\Pr[X_{\text{id}} \in \Theta_{\text{bad}}] \leq \sum_{i=1}^{7} \Pr[\text{B.i}]. \tag{2.30}$$

In the following, we only bound $\Pr[\text{B.1}], \Pr[\text{B.2}]$ and $\Pr[\text{B.3}]$ as the bound for the remaining events can be found in the proof of Lemma 4. Clearly,

$$\Pr[\text{B.1}] \leq \frac{\sigma}{2^n}. \tag{2.31}$$

Bounding B.2. Event B.2 occurs if there exists a zero keystream block in any query. For a fixed query and block, the probability of this event is exactly 2^{-n}. When the i^{th} query is not faulty, then the probability of any block taking the output $\mathbf{0}$ is exactly 2^{-n}. If the i^{th} query is faulty, then we have the following two cases:

- **Case (i):** When the j^{th} block is sampled while executing the i^{th} query, then the probability is 2^{-n}.
- **Case (ii):** When the j^{th} block is not sampled while executing the i^{th} query, then there must be some previous encryption query for which the j^{th} block is freshly sampled and hence, the probability is 2^{-n}.

Summing over all choices of i and j,

$$\Pr[\text{B.2}] \leq \frac{\sigma}{2^n}. \tag{2.32}$$

Bounding B3. Event B3 occurs if there is a collision between two different keystream blocks in an encryption query. For a fixed query and two distinct fixed blocks, the probability of this event is exactly 2^{-n}. When the i^{th} query is not faulty, then the probability of such a collision is exactly 2^{-n}. If the i^{th} query is faulty, then we have the following two cases:

- **Case (i):** When either of the blocks is sampled while executing the i^{th} query, then the probability is 2^{-n}.
- **Case (ii):** When none of the two blocks are sampled while executing the i^{th} query, it means that there must be some previous encryption query for which either of the blocks was freshly sampled, and hence the probability is 2^{-n}.

Summing over all choices of i, j and j',

$$\Pr[\text{B.3}] \leq \frac{q_e \ell^2}{2^n}. \tag{2.33}$$

$\Pr[\text{B.4}] + \Pr[\text{B.5}] + \Pr[\text{B.6}] + \Pr[\text{B.7}]$ can be bound similarly as in Lemma 4. Therefore, the result follows from Eqn.s (2.30), (2.31), (2.32) and (2.33) and Lemma 4, with $\epsilon \leq 2\ell/2^n$ (ϵ being the almost xor universal probability of the truncated PolyHash). □

Good Transcripts. We now show that for a good transcript $\tau' = (\tau, \mathbf{k}_h, \widetilde{\mathbf{T}})$, realizing τ' is almost as likely in the real world as in the ideal world.

Lemma 6. *Let $\tau' = (\tau_e, \tau_d, \mathbf{k}_h, \widetilde{\boldsymbol{T}})$ be a good transcript. Then*

$$\frac{\Pr[X_{\text{re}} = \tau']}{\Pr[X_{\text{id}} = \tau']} \geq \left(1 - \sum_{i=1}^{k} \frac{24\sigma^2 \binom{W_i}{2}}{2^{2n}} - \frac{2q_d}{2^\rho} - \frac{2\sigma}{2^n}\right),$$

where σ is the number of message blocks queried and ρ is the size of the tag.

Proof. Let $\tau' = (\tau_e, \tau_d, \mathbf{k}_h, \widetilde{\mathbf{T}})$ be a good transcript. Since in the ideal world, the encryption oracle is perfectly random and the decryption oracle always rejects,

$$\Pr[X_{\text{id}} = \tau'] = \frac{1}{2^n} \cdot \prod_{t=1}^{r} \frac{1}{2^{nl_t}} \cdot \frac{1}{2^{nq_e}}, \tag{2.34}$$

where r is the number of groups of nonces and l_t the updated number of generated keystream blocks for group t.

We say that a permutation Π is **compatible with** τ_e if

$$(\text{A}) : \begin{cases} \forall i \in [q_e], j \in [l_i], \Pi(\widehat{N}_i) \oplus \Pi(0 \| N_i \| \langle j \rangle) = S_i[j] \\ \forall i \in [q_e] \Pi(\widehat{N}_i) \oplus \Pi(\widehat{X}_i) = T_i, \end{cases}$$

and is **compatible with** τ_d if

$$(\text{B}) : \ \forall a \in [q_d], \Pi(\widehat{N'}_a) \oplus \Pi(\widehat{X'}_a) \neq T'_a \| \beta,$$

where $\langle j \rangle$ denotes the $(n/4 - 1)$-bit binary representation of the non-zero integer j. Π is compatible with τ' if it is compatible with both τ_e and τ_d. Let $\mathsf{Comp}(\tau)$ denote the set of all permutations that are compatible with τ. Then

$$\begin{aligned} \mathsf{p}_{\text{re}}(\tau) \coloneqq \Pr[X_{\text{re}} = \tau'] &= \frac{1}{|\mathcal{K}_h|} \cdot \Pr[\Pi \xleftarrow{\$} \mathsf{Perm} : \Pi \in \mathsf{Comp}(\tau)] \\ &\quad \text{(due to randomness of the hash key } \mathsf{E}_{\mathbf{k}}(\mathbf{0})) \\ &= 2^{-n} \cdot \Pr[\text{(A), (B) holds}]. \end{aligned} \tag{2.35}$$

On modelling the system of equations and non-equations into a graph theoretic setting by translating the system of $\sigma + q_e$ equations and $2^{n-\rho} q_d$ non-equations into a graph $G_{\tau'}$. As τ' is a good transcript, it is induced by the good graph $G_{\tau'}$ (i.e. it satisfies the NC, NPL and NCL conditions). Thus by Thm. 4, assuming $\xi \leq 2^n/8\sigma\ell$ with $c = 1$, $\sigma_{i-1} \leq \sigma_k \leq 2\sigma$ and $\alpha \leq \sigma$ gives

$$\Pr[\text{(A), (B) holds}] \geq \frac{1}{2^{nq_e}} \prod_{t=1}^{r} \frac{1}{2^{nl_t}} \cdot \left(1 - \sum_{i=1}^{k} \frac{6\sigma'^2_{i-1} \binom{W'_i}{2}}{2^{2n}} - \frac{2q_d}{2^\rho} - \frac{2\sigma}{2^n}\right), \tag{2.36}$$

where k is the number of components of $G_{\tau'}$, W'_i denotes the size of the i^{th} component and $\sigma'_i = W'_1 + \ldots W'_i$.

The result follows from Eqn.s (2.34), (2.35) and (2.36), and the inequality $\sigma'_{i-1} \leq \sigma'_k = (W'_1 + \ldots + W'_k) \leq 2\sigma$. □

Next, dividing Eqn. (2.36) by Eqn. (2.34) gives

$$\frac{\Pr[X_{\text{re}} = \tau']}{\Pr[X_{\text{id}} = \tau']} \geq 1 - \left(\sum_{i=1}^{k} \frac{24\sigma^2 \binom{W'_i}{2}}{2^{2n}} + \frac{2q_d}{2^\rho} + \frac{2\sigma}{2^n}\right); \tag{2.37}$$

observe that $\left(\sum_{i=1}^{k} \frac{24\sigma^2\binom{W_i'}{2}}{2^{2n}} + \frac{2q_d}{2^\rho} + \frac{2\sigma}{2^n}\right)$ depends upon the transcript τ', so that we can write it as the function $\phi(\tau')$ and calculate the expectation of $\phi(X_{\mathrm{id}})$ as follows:

$$\mathbf{E}\left[\phi\left(X_{\mathrm{id}}\right)\right] = \left(\frac{2q_d}{2^\rho} + \frac{2\sigma}{2^n} + \frac{24\sigma^2}{2^{2n}}\mathbf{E}\left[\sum_{i=1}^{k}\binom{W_i'}{2}\right]\right). \tag{2.38}$$

It is easy to see that $\binom{W_i'}{2} \leq \binom{W_i}{2}\binom{2\ell}{2}$ (where ℓ is the maximum number of message blocks and W_i is as in the proof of Theorem 2). Therefore,

$$\mathbf{E}\left[\sum_{i=1}^{k}\binom{W_i'}{2}\right] \leq 2\ell^2\mathbf{E}\left[\sum_{i=1}^{k}\binom{W_i}{2}\right] \leq 2\ell^2\mathbf{E}\left[\sum_{i=1}^{k}\binom{\tilde{W}_i}{2}\right] + 4q_e\ell^2. \tag{2.39}$$

Moreover from Eqn. (2.26),

$$\mathbf{E}\left[\sum_{i=1}^{k}\binom{\tilde{W}_i}{2}\right] \leq \sum_{i_1\neq i_2}^{q_e}\mathbf{E}[I_{i_1 i_2}] + \sum_{i_1\neq i_2}^{\mu}\mathbf{E}[\tilde{I}_{i_1 i_2}] \leq q_e^2\ell/2^n + \mu^2/2, \tag{2.40}$$

where the almost xor universal probability of the truncated PolyHash is at most $2\ell/2^n$. Finally, from Eqn.s (2.38), (2.39) and (2.40), we have

$$\mathbf{E}[\phi(X_{\mathrm{id}})] \leq \left(\frac{2q_d}{2^\rho} + \frac{2\sigma}{2^n} + \frac{48\sigma^4\ell}{2^{3n}} + \left(\frac{5\sigma\ell\mu}{2^n}\right)^2 + \frac{96\sigma^3\ell}{2^{2n}}\right), \tag{2.41}$$

where we assume $\ell q_e \approx \sigma$, the total number of message blocks queried.

Since $\xi \geq \mu$ and $\xi \leq 2^n/8\sigma\ell$, assuming $\mu \leq 2^n/8\sigma\ell$ (otherwise the bound holds trivially) lets us choose $\xi = 2^n/8\sigma\ell$. Hence, the bound stated in Eqn. (2.29) follows from Eqn. (1.2), Lemma 5, Eqn. (2.41), and $\sigma \leq 2^n/48$. □

The privacy bound of CWC+ is thus derived from Eqn. (2.29) by setting $\mu = 0$ and the bound stated in Eqn. (2.29) is itself the authenticity bound of CWC+.

3. On the Composition of Single-Keyed Tweakable Even-Mansour for Achieving BBB Security

Abstract

Observing the growing popularity of random permutation (RP)-based designs (e.g, Sponge), Bart Mennink in CRYPTO 2019 has initiated an interesting research in the direction of RP-based pseudorandom functions (PRFs). Both are claimed to achieve beyond-the-birthday-bound (BBB) security of $2n/3$ bits (n being the input block size in bits) but require two instances of RPs and can handle only one-block inputs. In this work, we extend research in this direction by providing two new BBB-secure constructions by composing the tweakable Even-Mansour appropriately. Our first construction requires only one instance of an RP and requires only one key. Our second construction extends the first to a nonce-based Message Authentication Code (MAC) using a universal hash to deal with multi-block inputs. We show that the hash key can be derived from the original key when the underlying hash is the Poly hash. We provide matching attacks for both constructions to demonstrate the tightness of the proven security bounds.

Keywords – PDMMAC, Davis-Meyer, PRF, MAC, permutation, beyond the birthday bound security.

3.1. Introduction

There is significant research on the design of PRFs from PRPs and vice versa. The most relevant work based on PRP-from-PRF has been the Luby-Rackoff construction [97]. However, constructions in this direction are not very popular as PRPs are easier to build than PRFs and several cryptographic designs desire instantiation with PRFs. In fact, the research community has found it a better proposition to go the other way around - constructing PRFs from PRPs, as a PRP can be more easily designed from a PRF than a PRF from a PRP.

Permutation-Based Designs

With the advent of public permutation-based designs and the efficiencies of permutations in the forward direction, several inverse-free hash and authenticated encryption schemes have been proposed. The most prominent of such designs are the Sponge designs introduced in SHA3 through the Keccak hash [24], this research direction later being extended by popular designs like PHOTON [80]. Several AEAD designs like Keyed Sponge [2, 25, 21, 106], SPONGENT [39], ASCON [64], Beetle [46] have later been proposed. Permutation-based designs generally provide lower security bounds and it can be highly interesting to design RP-based PRFs with BBB security on the permutation size. In CRYPTO 2019 [53], Mennink et al. studied permutation-based PRFs and proposed two BBB secure constructions denoted as SOEM and SOKAC. However, both designs are not minimal in structure and cannot handle arbitrary-length data. Both use two independent instances of random permutations and at least one randomly sampled key. They are deterministic and do not handle nonce. In this chapter, we explore this direction of research and address the following relevant questions: *Can we design minimally structured PRF? (i.e, with one instance of random permutation) Does there exist a nonce-based MAC constructed using an RP which is again minimal in structure and can handle arbitrary-length data?* We found the answer to be "yes", and we mainly propose two BBB secure deterministic and nonce-based designs using only one instance of a random permutation and one uniformly sampled construction key. We list our contributions below.

3.1.1. Motivation

The initial motivation for our construction arises from the fact that there are no single key, single permutation-based MACs with BBB security. No similar BBB secure permutation-based (or even nonce-based) MAC construction currently exists other than SoEM22 [53], which is also based on two permutations. In fact, [53] also provides birthday bound attacks for

the 2-permutations-1-keyed (π_1, π_2, K) and 1-permutation-2-keyed (π, K_1, K_2) constructions, thus leaving no scope for improvement in SoEM. It is therefore clear that a sequential construction is required for a minimization; SoKAC [53] is the only existing sequential construction, a birthday bound attack to which is present in [118].

Two variants of SoKAC, namely SoKAC1 and SoKAC21 seem to have the following inconsistencies:

1. The authors claim a birthday bound security of SoKAC1 in Proposition 5 of [53], whose proof claims a distinguishing attack that does not seem to work. Hence, a corrected attack is required for SoKAC1.
2. SoKAC21 is claimed to achieve a tight $2n/3$-bit security in Proposition 6 of [53], accompanied by an attack with a query complexity of $\mathcal{O}\left(2^{2n/3}\right)$. This security is proved flawed in [118], which shows a birthday bound attack on SoKAC21.

The main reason behind the above inconsistencies is the fixing of the input to the second permutation π_2 (or π) by the output of the first permutation π_1 (or π). Thus, although the final tag is a sum of the outputs of π_1, π_2 and a secret key, the fixing of the permutation input prevents construction of a transcript-inducing graph and subsequent use of Mirror theory.
This implies that the current form of SoKAC may not be a convincing construction to build upon. Our construction takes a different direction from SoKAC, and is inspired by DWCDM [62] - the output of only one permutation is involved in the tag generation and the sum of permutations occurs between the two permutation instances, allowing a query fixing the input and output of the construction (not the permutations) to be clearly described by an inducing graph, which was not the case in SoKAC. Thus, Mirror theory in its present form can be directly applied to our construction.

3.1.2. Our Contributions

We address the problem of designing a generic BBB secure MAC based on a RP with the minimal structure. The term "minimal" refers to the number of instances of the internal mathematical components (similar to DWCDM - Decrypted Wegman-Carter with Davies-Meyer, which minimizes the number of block cipher instances). Our proposal only uses one key and two calls of the same permutation (one forward and one inverse). The key is used to generate three sub-keys that are injected in between the two permutation calls. Precisely:

- We propose a deterministic MAC denoted by PDMMAC (***P**ermutation-based **D**avis-**M**eyer*) using one permutation and one key instance. We prove its PRF (which also upper bounds the MAC security) security

Table 3.1.: Comparison of existing PRFs. #Keys and #Primitives denote number of key and primitive instances.

Construction	#Key Instances	#Primitive Instances	MAC Security in n-bits (tightness)	Nonce Based	Multi-Block Inputs
Based on permutations					
PDMMAC **[This work]**	1	1	$2n/3$ (tight)		
PDM*MAC **[This work]**	$1+1$ (hash key)	1	$2n/3$ (tight)	✓	✓
1K-PDM*MAC **[This work]**	1	1	$2n/3$ (tight)	✓	✓
SoEM1 [53]	2	1	- (birthday attack)		
SoEM21 [53]	1	2	- (birthday attack)		
SoEM22 [53]	2	2	$2n/3$ (tight)		
SoKAC1 [53]	2	1	- (birthday attack)		
SoKAC21 [53]	1	2	- (birthday attack) [118]		
Based on Block Ciphers					
EDM [55]	2	2	$2n/3$ (not tight)		
EWCDM [55]	$2+1$ (hash key)	2	$2n/3$ (not tight)	✓	✓
DWCDM [62]	$1+1$ (hash key)	1	$2n/3$ (not tight)	✓	✓
1K-DWCDM [62]	1	1	$2n/3$ (not tight)	✓	✓

up to $2^{2n/3}$ queries under the random permutation model. We provide a proof using the coefficients-H technique. The bound has been proven to be tight with a matching attack with query complexity $2^{2n/3}$.

- The previous result sparks curiosity about the achievability of $2n/3$-bit security by a minimal construction that can process arbitrary length inputs. We propose a nonce-based MAC denoted by PDM*MAC using an additional keyed hash. We provide a BBB secure nonce-based MAC security proof of $2^{2n/3}$ query complexity under the nonce-respect scenario. We show the tightness of the proven security bound by demonstrating a matching attack.
- We propose a one-keyed instance of PDM*MAC denoted by 1K-PDM*MAC by instantiating the hash key K_h as $K_h = \pi(K)$, where π is the underlying RP. In addition, the underlying nonce is chosen to be non zero and the hash function is chosen as Poly hash. This instance achieves the same security bound as PDM*MAC.

Table 3.1 describes the structures of several well-known constructions in terms of primitives and other design properties.

3.2. Mirror Theory

Mirror Theory: Mirror theory is a tool for finding the number of solutions to affine systems of equalities and non-equalities. Mirror theory by Patarin [113, 122, 120, 121] provides a lower bound on such a number for a finite set of affine bi-variate equations, which is such that its variables are sampled without replacement. The proof is verifiable up to a bound of $2n/3$ bits.

Equation-Inducing Graph: Consider an undirected graph $\mathcal{G}_{\text{eq}} = (\mathsf{V}_{\text{eq}}, \mathsf{E}_{\text{eq}}, \mathcal{L})$, where $\mathsf{V}_{\text{eq}} = \{X_1, \ldots, X_m\}$ and the edge-label function $\mathcal{L} : \mathsf{E}_{\text{eq}} \to \mathbb{F}_{2^n}$ assigns a label λ to each edge $e \in \mathsf{E}_{\text{eq}}$.
If each vertex X_i is assumed to represent a unique variable (also denoted X_i, for the sake of convenience), then such a graph $\mathcal{G}_{\text{eq}}$ can be considered to induce a system of equations defined by-

$$X_i \oplus X_j = \lambda_{i,j}, \text{ whenever } e_{i,j} := \{X_i, X_j\} \in \mathsf{E}_{\text{eq}} \text{ and } \mathcal{L}(e_{i,j}) = \lambda_{i,j}.$$

Observe that should any of the following cases occur, the graph $\mathcal{G}_{\text{eq}}$ might induce a system of equations which is either inconsistent or has redundant equations:

- **Existence of a cycle:** A cycle arises in $\mathcal{G}_{\text{eq}}$ if there exists a sequence of edges $\{X_{i_1}, X_{j_1}\}, \ldots, \{X_{i_r}, X_{j_r}\} \in \mathsf{E}_{\text{eq}}$ such that $X_{j_a} = X_{i_{a+1}}$ $\forall\ a \in [r-1]$ and $X_{j_r} = X_{i_1}$. A loop i.e. $X_i = X_j$ for some edge $\{X_i, X_j\} \in \mathsf{E}$ is also considered a cycle.
- **Zero Path Label:** The path label of a path P of edges in E_{eq} is defined as $\mathcal{L}(P) = \sum_{e \in P} \mathcal{L}(e)$. Thus, a zero path-label arises when there exists a path P in $\mathcal{G}$ such that $\mathcal{L}(P) = 0$.

Extended Mirror Theory: Extended mirror theory gives a lower bound for the number of solutions to a combination of a system of bi-variate affine equations (as in Mirror Theory) and a system of bi-variate affine non-equations of the form $X_i \oplus Y_i \neq c$. [72] contains a detailed treatment of such a combination of systems.

Equations-and-Non-Equations-Inducing Graph: Consider an undirected graph $\mathcal{G}_{\text{eq}} = (\mathsf{V}_{\text{eq}}, \mathsf{E}_{\text{eq}}, \mathcal{L}_{\text{eq}})$, where $\mathsf{V}_{\text{eq}} = \{X_1, \ldots, X_m\}$ and the edge-label function $\mathcal{L}_{\text{eq}} : \mathsf{E}_{\text{eq}} \to \mathbb{F}_{2^n}$ assigns a label λ to each edge $e \in \mathsf{E}_{\text{eq}}$.

If each vertex X_i is assumed to represent a unique variable (also denoted X_i, for the sake of convenience), then such a graph $\mathcal{G}_{\text{eq}}$ can be considered to induce a system of equations defined by-

$$X_i \oplus X_j = \lambda_{i,j}, \text{ whenever } \mathsf{E}_{i,j} := \{X_i, X_j\} \in \mathsf{E}_{\text{eq}} \text{ and } \mathcal{L}_{\text{eq}}(\mathsf{E}_{i,j}) = \lambda_{i,j}.$$

Now consider an undirected graph $\mathcal{G}_{\text{eq,neq}} = (\mathsf{V}, \mathsf{E}_{\text{eq}} \sqcup \mathsf{E}_{\text{neq}}, \mathcal{L})$, where $\mathsf{V}_{\text{eq}} = \{X_1, \ldots, X_m\} \subseteq V = \{X_1, \ldots, X_v\}$ and the edge-label function $\mathcal{L} : \mathsf{E}_{\text{eq}} \sqcup \mathsf{E}_{\text{neq}} \to \mathbb{F}_{2^n}$ assigns a label λ to each edge $e \in \mathsf{E}_{\text{eq}} \sqcup \mathsf{E}_{\text{neq}}$.

Again assuming each vertex X_i, $\mathcal{G}_{\text{eq,neq}}$ can be considered to induce a system of equations and a system of non-equations defined by-

$$\begin{aligned} X_i \oplus X_j = \lambda_{i,j}, \quad \text{whenever} \quad & e_{i,j} := \{X_i, X_j\} \in \mathsf{E}_{\text{eq}} \text{ and} \\ & \mathcal{L}(e_{i,j}) = \lambda_{i,j}, \forall\, X_i, X_j \in \mathsf{V}_{\text{eq}} \\ X'_i \oplus X'_j \neq \lambda'_{i,j}, \quad \text{whenever} \quad & e'_{i,j} := \{X'_i, X'_j\} \in \mathsf{E}_{\text{neq}} \text{ and} \\ & \mathcal{L}(e'_{i,j}) = \lambda'_{i,j}, \forall\, X'_i, X'_j \in V. \end{aligned}$$

Let $\mathcal{G}_{\text{eq,neq}} = (\mathsf{V}, \mathsf{E}_{\text{eq}} \sqcup \mathsf{E}_{\text{neq}}, \mathcal{L})$ be a graph that induces a system of affine bivariate equations and non-equations over α distinct variables. Suppose $\mathcal{G}_{\text{eq,neq}}$ has α vertices and $q'_m + q_v$ edges with $|\mathsf{E}_{\text{eq}}| = q'_m, |\mathsf{E}_{\text{neq}}| = q_v$. Let $\mathcal{C}_1, \ldots, \mathcal{C}_k$ be all the components (i.e. maximal subgraphs where any two vertices are connected to each other by a path) of $\mathcal{G}_{\text{eq}} = (\mathsf{V}, \mathsf{E}_{\text{eq}}, \mathcal{L}|_{\mathsf{E}_{\text{eq}}})$, $\mathcal{C}_i$ of size w_i, and let $\sigma_i = (w_1 + \cdots + w_i)$. Denote by $\xi_{\max}$, the size of the component of $\mathcal{G}_{\text{eq}}$ with the maximum number of vertices. Using an *extended version* of mirror theory, we can provide a lower bound on the number of injective solutions when the maximum component size is $\xi_{\max}$. We now state the following lemma, which summarizes the result of Theorem 3 in [72]

Lemma 7. *The total number of injective solutions chosen from a set $\mathcal{Z}$ of size $2^n - c$, for some $c \geq 0$, for the induced system of equations and non-equations $\mathcal{G}_{\text{eq,neq}}$ is at least:*

$$(2^n)_\alpha \left(1 - \sum_{i=1}^{k} \frac{6\sigma_{i-1}^2 \binom{\xi_i}{2}}{2^{2n}} - \frac{2(q_v + c\alpha)}{2^n}\right),$$

provided $\sigma_k \xi_{\max} \leq 2^n/4$, and assuming $\sigma_0 = 0$.

This lemma thus provides a bound for a solution from a subset of $\{0,1\}^n$. However, applying this lemma to our results (Thm. 6, 7 and 8) generates the term $\frac{p(p+q)}{2^n}$ for the non-equations, as c takes the value p, which is not a constant. We wish for a beyond-the-birthday bound on this number, which could possibly have been achieved by the results in [62, 62] . In spite of this providing a stronger bound, there are two problems. First, non-equations are unaccounted, which could be easily included (by the same method as in the proof of Cor. 2). Second, a maximum component size of only 2 is allowed for the equations-inducing subgraph. A modification of this result is presented here (Cor. 1 and Cor. 2), which not only takes non-equations into account and allows for a maximum size of 3 for equation-components, but also provides an improved bound.

3.2.1. Extended Mirror Theory

Some Probability Results

Recall the following result from [62]: *Let* $S' \subseteq \{0,1\}^n$ *be a subset of size* $(2^n - s')$ *and* $U_n \leftarrow \{0,1\}^n$. *Let* $(V, W) \xleftarrow[wor]{\$} S'^{(2)}$ *be a WOR sample of size* 2 *drawn from* S'. *Then,*

$$V \oplus W \succ_{\epsilon_1(s')} U_n \text{ over } \mathbb{F}_{2^n}^\star := \mathbb{F}_{2^n} \setminus \{0^n\}, \tag{3.1}$$

where $\epsilon_1(s')$ is a quantity with value at most $\frac{s'^2}{(2^n - s')^2}$. This result can be extended for three random variables as follows:

Lemma 8. *Let* $S' \subseteq \{0,1\}^n$ *be a subset of size* $(2^n - p')$ *and* $U_n, V_n \leftarrow \{0,1\}^n$. *Let* $(P, Q, R) \xleftarrow[wor]{\$} S'$ *be a WOR sample of size* 3 *drawn from* $S'^{(3)}$. *Then,*

$$(P \oplus Q, Q \oplus R) \succ_{\epsilon_2(p')} (U_n, V_n), \tag{3.2}$$

where $\epsilon_2(p')$ *is a quantity with value at most* $\frac{3 \cdot 2^n \cdot p'^2 - p'^3}{(2^n - p')^3}$.

The proof is similar to that provided by [62] for Eqn. (3.1):

Proof. Consider a set S' of size $2^n - p'$, and three random variables $P, Q, R \xleftarrow[\text{wor}]{\$} S'$. Fix $\lambda_1, \lambda_2 \in \mathbb{F}_{2^n}$. For $i \in \{1,2,3\}$, let

$$A_i = \{(a_1, a_2, a_3) | a_1 \oplus a_2 = \lambda_1, a_2 \oplus a_3 = \lambda_2, a_i \notin S'\},$$

so that $|A_i| \leq p'$. Thus,

$$\{(p,q,r) \in S'^{(3)} | p \oplus q = \lambda_1, q \oplus r = \lambda_2\} = \\ \{(p, p \oplus \lambda_1, p \oplus \lambda_1 \oplus \lambda_2) | p \in \{0,1\}^n\} \setminus (A_1 \cup A_2 \cup A_3),$$

which is a set of size no less than $2^n - 3p'$. Hence,

$$\begin{aligned} \Pr\left[\begin{smallmatrix} P \oplus Q = \lambda_1, \\ Q \oplus R = \lambda_2 \end{smallmatrix}\right] &= \frac{2^n - |A_1 \cup A_2 \cup A_3|}{(2^n - p')(2^n - p' - 1)(2^n - p' - 2)} \\ &\geq \frac{2^n - 3p'}{(2^n - p')(2^n - p')(2^n - p')} \\ &= \frac{1}{2^{2n}} \left(1 - \frac{3 \cdot 2^n \cdot p'^2 - p'^3}{(2^n - p')^3}\right). \end{aligned}$$

□

Results on Mirror Theory

Eqn.s (3.1) and (3.2) can be easily extended for systems of equations as follows-

Corollary 1. *Let $S' \subseteq \{0,1\}^n$ be a subset of size $(2^n - p')$ and*

$$(X_1, X_2, \ldots, X_t, Y_1, Y_2, \ldots, Y_t, Z_1, Z_2, \ldots Z_t) \xleftarrow[wor]{\$} S'$$

be a WOR sample of size $3t$ drawn from $S'^{(3)}$. Then for constants $\lambda_1, \lambda_2, \ldots, \lambda_{2t}$ in $\{0,1\}^n$,

$$\Pr\left[(X_1 \oplus Y_1 = \lambda_1) \wedge (X_2 \oplus Y_2 = \lambda_2) \wedge \ldots \wedge (X_t \oplus Y_t = \lambda_t)\right] \geq \frac{1}{2^n}\left(1 - \frac{t \cdot p'^2}{(2^n - p')^2}\right), \tag{3.3}$$

by Eqn. (3.1), and

$$\Pr\left[\begin{pmatrix} X_1 \oplus Y_1 = \lambda_1, \\ Z_1 \oplus Y_1 = \lambda_2 \end{pmatrix} \wedge \begin{pmatrix} X_2 \oplus Y_2 = \lambda_3, \\ Z_2 \oplus Y_2 = \lambda_4 \end{pmatrix} \wedge \ldots \wedge \begin{pmatrix} X_t \oplus Y_t = \lambda_{2t-1}, \\ Z_t \oplus Y_t = \lambda_{2t} \end{pmatrix}\right] \geq \frac{1}{2^{2nt}}\left(1 - \frac{3t \cdot 2^n \cdot p'^2}{(2^n - p')^3}\right), \tag{3.4}$$

by Eqn. (3.2) of Lemma 8.

Proof. Observe that by Eqn. (3.1),

$$\Pr\left[(X_1 \oplus Y_1 = \lambda_1) \wedge (X_2 \oplus Y_2 = \lambda_2) \wedge \ldots \wedge (X_t \oplus Y_t = \lambda_t)\right]$$

$$\begin{aligned}
&= \Pr\left[X_1 \oplus Y_1 = \lambda_1 \;\middle|\; \begin{matrix} X_1, Y_1 \in S' \\ \text{are distinct} \end{matrix}\right] \times \cdots \times \Pr\left[X_t \oplus Y_t = \lambda_t \;\middle|\; \begin{matrix} X_t, Y_t \in S' \\ \setminus \{X_1, \ldots, X_{t-1}, Y_1, \ldots, Y_{t-1}\} \\ \text{are distinct} \end{matrix}\right] \\
&\geq \frac{1}{2^n}\left(1 - \epsilon_1(p')\right) \times \frac{1}{2^n}\left(1 - \epsilon_1(p' - 2)\right) \cdots \times \frac{1}{2^n}\left(1 - \epsilon_1(p' - (2t-2))\right) \\
&\geq \prod_{i=1}^{t} \frac{1}{2^n}\left(1 - \epsilon_1(p')\right) \;\geq \frac{1}{2^{nt}} \sum_{i=1}^{t} \frac{1}{2^n}\left(1 - \epsilon_1(p')\right) \;\geq \frac{1}{2^{nt}}\left(1 - \frac{tp'^2}{(2^n - p')^2}\right).
\end{aligned}$$

Similarly, by Eqn. (3.2),

$$\Pr\left[\begin{pmatrix} X_1 \oplus Y_1 = \lambda_1, \\ Z_1 \oplus Y_1 = \lambda_2 \end{pmatrix} \wedge \begin{pmatrix} X_2 \oplus Y_2 = \lambda_3, \\ Z_2 \oplus Y_2 = \lambda_4 \end{pmatrix} \wedge \ldots \wedge \begin{pmatrix} X_t \oplus Y_t = \lambda_{2t-1}, \\ Z_t \oplus Y_t = \lambda_{2t} \end{pmatrix}\right]$$

$$
\begin{aligned}
&= \Pr\left[\begin{matrix}X_1\oplus Y_1=\lambda_1,\\ Z_1\oplus Y_1=\lambda_2\end{matrix} \;\middle|\; \begin{matrix}X_1,Y_1,Z_1\in S'\\ \text{are distinct}\end{matrix}\right]\\
&\times \Pr\left[\begin{matrix}X_2\oplus Y_2=\lambda_3,\\ Z_2\oplus Y_2=\lambda_4\end{matrix} \;\middle|\; \begin{matrix}X_2,Y_2,Z_2\in S'\setminus\{X_1,Y_1,Z_1\}\\ \text{are distinct}\end{matrix}\right]\\
&\vdots\\
&\times \Pr\left[\begin{matrix}X_t\oplus Y_t=\lambda_{2t-1},\\ Z_t\oplus Y_t=\lambda_{2t}\end{matrix} \;\middle|\; \begin{matrix}X_t,Y_t,Z_t\in S'\setminus\{X_1,\ldots,X_{t-1},Y_1,\ldots,Y_{t-1},Z_1,\ldots,Z_{t-1}\}\\ \text{are distinct}\end{matrix}\right]\\
&\geq \frac{1}{2^{2n}}\left(1-\epsilon_2(p')\right)\times\frac{1}{2^{2n}}\left(1-\epsilon_2(p'-3)\right)\cdots\times\frac{1}{2^{2n}}\left(1-\epsilon_2(p'-3(t-1))\right)\\
&\geq \prod_{i=1}^{t}\frac{1}{2^{2n}}\left(1-\epsilon_2(p')\right) \geq \frac{1}{2^{2nt}}\left(1-\sum_{i=1}^{t}\epsilon_2(p')\right) \geq \frac{1}{2^{2nt}}\left(1-\frac{3t\cdot 2^n\cdot p'^2}{(2^n-p')^3}\right).
\end{aligned}
$$

□

The following bound on probability of a valid solution for a combination of a system of equations and a system of non-equations can also be obtained from Eqn.s (3.1) and (3.2)-

Corollary 2. *Let $\mathcal{G}_{\text{eq,neq}} = (\mathsf{V}, \mathsf{E}_{\text{eq}} \sqcup \mathsf{E}_{\text{neq}}, \mathcal{L})$ be an equations-and-non-equations-inducing graph such that the subgraph $\mathcal{G}_{\text{eq}}$ only has components of size 2 or 3. If $|\mathsf{V} \setminus \mathsf{V}_{\text{eq}}| = q_v$ and λ_i $(i \in [q_m])$ are edge-labels of the edges in E_{eq} in the same order as the components, then the probability of the induced systems of equations and non-equations attaining any solution from a set $S' \subseteq \{0,1\}^n$ of size $(2^n - p')$ for all the variables represented only by the vertices in V_{eq} is bounded by-*

$$
\frac{1}{2^{nq_m}}\left(1-\frac{1200q_m^3+312(p'+3q_v)q_m^2+2(p'+3q_v)^2q_m}{2^{2n}}\right)\left(1-\frac{q_v}{2^n}\right). \quad (3.5)
$$

Proof. Suppose $G^{=}$ has exactly $q_m - t$ components with-

1. t components $(X_i, Y_i, Z_i)_{i=1}^{t}$ of size 3 and
2. $q_m - 2t$ components $(X_i, Y_i)_{i=t+1}^{q_m-2t}$ of size 2.

. Let $w_{i,j}$ be the number of edges in E_{neq} that connect one vertex of the i^{th} component of $G^{=}$ to one vertex of its j^{th} component. Also let $w(v)$ be the number of edges in E_{neq} from some vertex in $\mathsf{V} \setminus \mathsf{V}_{\text{eq}}$ incident on a vertex $v \in \mathsf{V}_{\text{eq}}$. The number of solutions for all the variables represented by vertices in V_{eq} can then be computed as-

$$
\Pr\left[\begin{pmatrix}X_1\oplus Y_1=\lambda_1,\\ Z_1\oplus Y_1=\lambda_2\end{pmatrix},\begin{pmatrix}X_1\oplus Y_1=\lambda_2,\\ Z_1\oplus Y_1=\lambda_4\end{pmatrix},\ldots,\begin{pmatrix}X_t\oplus Y_t=\lambda_{2t-1},\\ Z_t\oplus Y_t=\lambda_{2t}\end{pmatrix},\begin{matrix}(X_{t+1}\oplus Y_{t+1}=\lambda_{2t+1}),(X_{t+2}\oplus Y_{t+2}=\lambda_{t+2}),\\ \ldots,\left(X_{q_m-2t}\oplus Y_{q_m-2t}=\lambda_{q_m}\right)\end{matrix}\right]
$$

$$
\begin{aligned}
&= \Pr\left[\begin{matrix}X_1\oplus Y_1=\lambda_1,\\ Z_1\oplus Y_1=\lambda_2\end{matrix} \;\middle|\; \begin{matrix}X_1,Y_1,Z_1\in S'\\ \text{are distinct}\end{matrix}\right]\\
&\times \Pr\left[\begin{matrix}X_2\oplus Y_2=\lambda_3,\\ Z_2\oplus Y_2=\lambda_4\end{matrix} \;\middle|\; \begin{matrix}X_2,Y_2,Z_2\in S'\setminus\{X_1,Y_1,Z_1\}\\ \text{are distinct}\end{matrix}\right]\\
&\vdots
\end{aligned}
$$

$$\times \Pr\left[\begin{matrix}X_t\oplus Y_t=\lambda_{2t-1},\\ Z_t\oplus Y_t=\lambda_{2t_1}\end{matrix} \;\middle|\; \begin{matrix}X_t,Y_t,Z_t\in S'\setminus\{X_1,\ldots,X_{t-1},Y_1,\ldots,Y_{t-1},Z_1,\ldots,Z_{t-1}\}\\ \text{are distinct}\end{matrix}\right]$$
$$\times \Pr\left[X_{t+1}\oplus Y_{t+1}=\lambda_{2t+1} \;\middle|\; \begin{matrix}X_{t+1},Y_{t+1}\in S'\\ \setminus\{X_1,\ldots,X_t,Y_1,\ldots,Y_t,Z_1,\ldots,Z_t\}\\ \text{are distinct}\end{matrix}\right]$$
$$\vdots$$
$$\times \Pr\left[X_{q_m-t}\oplus Y_{q_m-t}=\lambda_{q_m} \;\middle|\; \begin{matrix}X_{q_m-t},Y_{q_m-t}\in S'\\ \setminus\{X_1,\ldots,X_{q_m-t},Y_1,\ldots,Y_{q_m-t},Z_1,\ldots,Z_{q_m-t}\}\\ \text{are distinct}\end{matrix}\right].$$

The vertices in $\mathcal{G}_{\text{eq,neq}}$ representing X_1, Y_1 and Z_1 can be chosen after removing one value from S' for each non-equation edge joining one of these vertices to some other vertex of $\mathcal{G}_{\text{eq,neq}}$. Thus, the choice for their values must be made from a set of size $p' + w(X_1) + w(Y_1) + w(Z_1)$. Next, X_2, Y_2 and Z_2 can be chosen only after all the previously assigned values, all values conflicting with any non-equation edges connecting X_2, Y_2 and Z_2 to some vertex in $\mathsf{V} \setminus \mathsf{V}_{\text{eq}}$ and all values conflicting with any non-equation edges joining some vertex of the first component (i.e. X_1, Y_1 or Z_1) with the second component (i.e. X_2, Y_2 or Z_2) are removed from the set S'. This leaves a set of size no less than $p' + 3 + w(X_2) + w(Y_2) + w(Z_2) + w_{1,2}$. Similar calculations for the remaining components give the following lower bound for $\Pr\left[\begin{pmatrix}X_1\oplus Y_1=\lambda_1, Z_1\oplus Y_1=\lambda_2,\ldots,\\ X_t\oplus Y_t=\lambda_{2t-1}, Z_t\oplus Y_t=\lambda_{2t}\end{pmatrix}, \begin{pmatrix}X_{t+1}\oplus Y_{t+1}=\lambda_{2t+1},\ldots,\\ X_{q_m-2t}\oplus Y_{q_m-2t}=\lambda_{q_m}\end{pmatrix}\right]$:

$$\frac{1}{2^{2n}}\left(1-\varepsilon_2\left(p'+w(X_1)+w(Y_1)+w(Z_1)\right)\right)$$
$$\times \quad \frac{1}{2^{2n}}\left(1-\varepsilon_2\left(p'+3+w(X_2)+w(Y_2)+w(Z_2)+w_{1,2}\right)\right)$$
$$\vdots$$
$$\times \quad \frac{1}{2^{2n}}\left(1-\varepsilon_2\left(p'+3(t-1)+w(X_t)+w(Y_t)+w(Z_t)+\sum_{j=1}^{t-1} w_{j,t}\right)\right)$$
$$\times \quad \frac{1}{2^{n}}\left(1-\varepsilon_1\left(p'+3t+w(X_{t+1})+w(Y_{t+1})+\sum_{j=1}^{t} w_{j,t+1}\right)\right)$$
$$\vdots$$
$$\times \quad \frac{1}{2^{n}}\left(1-\varepsilon_1\left(p'+3t+2(q_m-t-1)+w(X_{q_m-t})+w(Y_{q_m-t})+\sum_{j=1}^{q_m-t-1} w_{j,q_m-t}\right)\right)$$

$$
\begin{aligned}
&\geq \frac{1}{2^{2n}}\left(1-\frac{24(p'+3q_v)^2}{2^{2n}}\right)\\
&\times \frac{1}{2^{2n}}\left(1-\frac{24(p'+3+3q_v+9)^2}{2^{2n}}\right)\\
&\vdots\\
&\times \frac{1}{2^{2n}}\left(1-\frac{24(p'+3(t-1)+3q_v+9(t-1))^2}{2^{2n}}\right)\\
&\times \frac{1}{2^{n}}\left(1-\frac{4(p'+3t+2q_v+6t)^2}{2^{2n}}\right)\\
&\vdots\\
&\times \frac{1}{2^{n}}\left(1-\frac{4(p'+3t+2(q_m-t-1)+2q_v+6t+4(q_m-t-1))^2}{2^{2n}}\right)\\
&\geq \frac{1}{2^{2nt}}\left(1-\frac{24}{2^{2n}}\sum_{i=0}^{t-1}(p'+12i+3q_v)^2\right)\\
&\times \frac{1}{2^{n(q_m-2t)}}\left(1-\frac{4}{2^{2n}}\sum_{i=t}^{q_m-t}(p'+7t+6i+2q_v)^2\right)\\
&\geq \frac{1}{2^{2nt}}\left(1-\frac{24}{2^{2n}}\left(48q_m^3+12(p'+3q_v)q_m^2+(p'+3q_v)^2q_m\right)\right)\\
&\times \frac{1}{2^{n(q_m-2t)}}\left(1-\frac{4}{2^{2n}}\left(12q_m^3+6(p'+2q_v)q_m^2+(p'+2q_v)^2q_m\right)\right), \quad \text{since } t\leq q_m.
\end{aligned}
$$

Next, observe that the only vertices in V that remain after this computation are those connected by edges in E_{neq}. The number of valid solutions for these vertices is minimum when they form a single component. Since there can be at most $2q_v' \leq 2q_v$ vertices in $\mathsf{V}\setminus\mathsf{V}_{\text{eq}}$, the lower bound for the probability of any combination of values represented by these $2q_v'$ vertices is:

$$
\begin{aligned}
& \Pr\Big[\left(X'_1 \oplus X'_2 \neq \lambda'_1\right) \wedge \left(X'_2 \oplus X'_3 \neq \lambda'_2\right) \wedge \\
& \ldots \wedge \left(X'_{2q'_v-1} \oplus X'_{2q'_v} \neq \lambda'_{2q'_v-1}\right) \Big] \\
= \; & 1 - \Pr\Big[\left(X'_1 \oplus X'_2 = \lambda'_1\right) \vee \left(X'_2 \oplus X'_3 = \lambda'_2\right) \vee \\
& \ldots \vee \left(X'_{2q'_v-1} \oplus X'_{2q'_v} = \lambda'_{2q'_v-1}\right) \Big] \\
\geq \; & 1 - \Big(\Pr\left[X'_1 \oplus X'_2 = \lambda'_1\right] + \Pr\left[X'_2 \oplus X'_3 = \lambda'_2\right] + \\
& \ldots + \Pr\left[X'_{2q'_v-1} \oplus X'_{2q'_v} = \lambda'_{2q'_v-1}\right] \Big)
\end{aligned}
$$

$$
\begin{aligned}
& \geq \; 1 - \sum_{a=1}^{2q'_v-1} \frac{1}{2^n} \left(1 - \frac{p'^2}{(2^n - p')^2}\right) \\
& = \; 1 - \frac{(2q'_v - 1)\,(2^n - 2p')}{(2^n - p')^2} \\
& \geq \; 1 - \frac{q'_v}{2^n}, \text{ since } 2p' \leq 2^n/2.
\end{aligned}
$$

Since $q'_v \leq q_v$, any solution to the combined systems of equations and non-equations must therefore have a probability of at least-

$$\frac{1}{2^{nq_m}} \left(1 - \frac{1200q_m^3 + 312(p' + 3q_v)q_m^2 + 2(p' + 3q_v)^2 q_m}{2^{2n}}\right) \left(1 - \frac{q_v}{2^n}\right).$$

□

3.3. Related Work

We describe some constructions relevant for our proposals. We have also identified an issue in the cryptanalysis of SoKAC proposed in CRYPTO-19 [53].

Tweakable Even-Mansour: Even and Mansour pioneered the design and analysis of random permutation-based blockciphers [76]. Let π be an ideal n-bit (public) permutation and $K_1, K_2 \in \{0,1\}^n$ be the secret keys. The Even-Mansour construction is defined as follows:

$$\mathsf{EM}_{K_1,K_2}[\pi](x) := \pi(x \oplus K_1) \oplus K_2, \; \forall x \in \{0,1\}^n.$$

When $K_1 = K_2$, we simply write $\mathsf{EM}_{K_1}[\pi]$. In order to incorporate a tweak t in the Even-Mansour construction, Cogliati et al. replace the round keys by some functions $f_i(K_i, t)$ and called it Tweakable Even-Mansour (TEM) construction. This is exactly the spirit of the TWEAKEY framework introduced by Jean et al. [91]. In this chapter, we consider the following simple instantiation of TEM.

$$\mathsf{TEM}_K[\pi](x,t) := \pi(x \oplus (2^t \cdot K)) \oplus (2^t \cdot K),\ \forall x, t \in \{0,1\}^n.$$

Here, 2 denotes a primitive element in the binary field $\{0,1\}^n$. Other similar known approaches can be found in [96, 126, 54, 102] etc.

Davis-Meyer: For a permutation π (public or keyed), Davis Meyer construction is defined as $\mathsf{DM}[\pi](x) := \pi(x) \oplus x$. This method has been popularly adopted to design both hash and PRF from an ideal permutation or cipher. When the permutation π is a blockcipher e_K, we write $\mathsf{DM}_K[e](x) := e_K(x) \oplus x$.

3.4. PDMMAC and PDM*MAC Constructions

3.4.1. Specification and Security of PDMMAC

Specification of PDMMAC: Let $K \xleftarrow{\$} \{0,1\}^n$ and $\pi \xleftarrow{\$} \mathrm{Perm}(n)$. The PRF that we propose in this chapter is a construction that takes a message $M \in \{0,1\}^n$ as an input and return n-bit tag $T := \mathsf{PDMMAC}^{\pi}_K(M)$. The construction PDMMAC is defined as

$$T = \pi^{-1}\left(\pi(K \oplus M) \oplus 3K \oplus M\right) \oplus 2K. \tag{3.6}$$

Design Rationale: Our design PDMMAC is motivated by DDM. Let

$$\mathsf{TEM}_K(t, M) = \pi(M \oplus 2^t \cdot K) \oplus 2^t \cdot K$$

be a specific instantiation of the tweakable Even-Mansour construction. The construction PDMMAC can be equivalently described as (see Fig.3.1)

$$T = \mathsf{TEM}_K^{-1}\left(1, \mathsf{TEM}_K(0, M) \oplus M\right). \tag{3.7}$$

Security of PDMMAC: We prove that PDMMAC for one instance of uniform π and uniform key K is secure up to attack complexity $\mathcal{O}(2^{2n/3})$. We also propose an attack matching this bound.

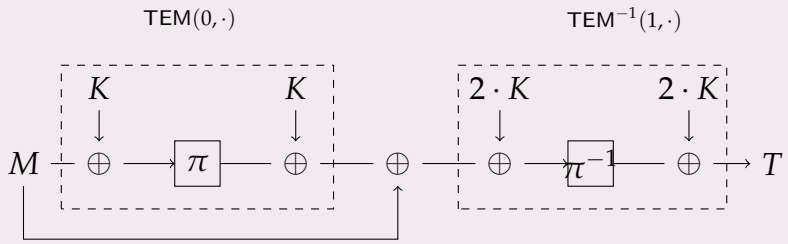


Figure 3.1.: The construction PDMMAC

Theorem 6. *Let $M \in \mathcal{M}$, and consider PDMMAC$^{\pi}_{K}$ based on one permutation $\pi \xleftarrow{\$} Perm(\{0,1\}^n)$ and one key $K \xleftarrow{\$} \{0,1\}^n$. For any distinguisher $\mathcal{D}$ making at most q construction queries at most p primitive queries to $\pi^{\pm}$, we have,*

$$Adv^{\mathsf{prf}}_{PDMMAC}(\mathcal{D}) \leq \frac{q^2 + 2q^3 + 3pq^2 + p^2q + 8q(p+q)^2}{2^{2n}} + \frac{6 + q + q\sqrt{3np} + \sqrt{6npq} + p\sqrt{3nq}}{2^n}.$$

The proof for this theorem can be found in Sect. 3.5. Note that the dominating term of advantage is $\sqrt{\frac{3n(pq^2+qp^2)}{2^{2n}}}$. So the construction is secure as long as $p, q \ll \frac{2^{2n/3}}{n^{1/3}}$.

A Matching Attack with $\mathcal{O}(2^{2n/3})$ Queries: We have a matching attack (up to the logarithmic factor). The attack is similar to that of PDM*MAC, and henceforth omitted. We include the attack for PDM*MAC instead of PDMMAC as it is more robust.

3.4.2. Specification and Security of PDM*MAC

Specification of PDM*MAC: The previous construction does not allow arbitrary-length messages. We now propose a construction similar to DWCDM, which uses a single ideal permutation $\pi \xleftarrow{\$} \mathrm{Perm}(n)$ and an n-bit key K. To process a message $M \in \{0,1\}^*$, a hash function $\mathcal{H}$ with a key K_h sampled independently of K is required, which is almost xor-universal, regular and 3-way regular. The construction PDM*MAC for an n-bit nonce N and a message $M \in \{0,1\}^*$, with $\mathcal{B} = \{0,1\}^n$ computes $T = \mathsf{PDM^*MAC}^{\pi}_{K,K_h}(N, M)$ as follows:

$$T = \pi^{-1}\left(\pi(K \oplus N) \oplus 3K \oplus N \oplus \mathcal{H}_{K_h}(M)\right) \oplus 2K. \tag{3.8}$$

INSTANCE OF $\mathcal{H}$: PolyHash [108] is an example of a keyed hash which is $\frac{\ell}{2^n}$-regular, AXU and 3-way regular, where ℓ is the maximum number of n-bit blocks. The hash first uses an injective 10^* (one followed by zeros) padding to pad an input $M \in \{0,1\}^*$ to multiple of n-bits. Precisely, $M\|10^j = M_1\|M_2\|\ldots\|M_\ell$ where $j = n - |M| \mod n-1$. The hash value is generated as

$$\mathsf{Poly}_{\mathcal{H}}(M) = M_\ell \cdot K_h \oplus M_{\ell-1} \cdot K_h^2 \oplus \cdots \oplus M_1 \cdot K_h^\ell.$$

DESIGN RATIONALE: This construction is motivated by DWCDM. Like PDM-MAC, the nonce and the hash of the message are XOR-ed between two permutation calls. Similar designs have been adapted for DWCDM from DDM. The construction PDM*MAC can be equivalently described as (see Fig.3.2)-

$$T = \mathsf{TEM}_K^{-1}\left(1, \mathsf{TEM}_K(0,N) \oplus N \oplus \mathcal{H}_{K_h}(M)\right). \tag{3.9}$$

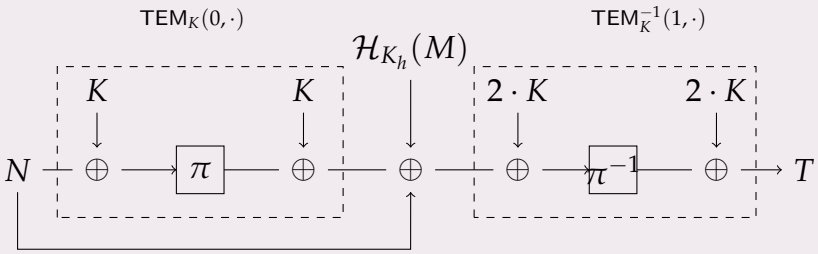


Figure 3.2.: The construction PDM*MAC

SECURITY OF PDM*MAC: We prove the security of PDM*MAC up to an attack complexity of $\mathcal{O}(2^{2n/3})$ for one instance of uniform π and uniform key K. We also propose an attack matching this bound in Fig. 3.3.

Analysis of the attack: Observe that since $I_K := \{(i,a) \mid N_i \oplus \tilde{u}_a = K\}$ has size $\mathcal{O}(2^{n/3})$ for each value $K \in \mathcal{K}$, and for the values $q = p_1 = p_2 = 2 \cdot 2^{2n/3}$, the set Ext_K has size $\mathcal{O}(1)$ with high probability. Furthermore, if $K^\star$ denotes the true key of the construction, then $\Pr[K^\star \in \hat{\mathcal{K}}] = \Pr\left[|\mathrm{Ext}_{K^\star}| \geq 2\right] \geq \frac{1}{4}$, and thus, the expected size, $\mathsf{E}\left[|\hat{\mathcal{K}}|\right]$, of the guess-key set $\hat{\mathcal{K}}$ is $\mathcal{O}(1)$.

Theorem 7. *Let $n \in \mathcal{N}$, and consider PDM*MAC$^{\pi}_{K,K_h}$ based on one permutation $\pi \xleftarrow{\$}$ Perm$(\{0,1\}^n)$, one key $K \xleftarrow{\$} \{0,1\}^n$ and one hash key $K_h \xleftarrow{\$} \{0,1\}^n$. For any distinguisher $\mathcal{D}$ making at most q_m construction queries, at most p primitive*

A Matching Attack on PDM*MAC with $\mathcal{O}(2^{2n/3})$ Queries

1 : Make queries $(N_1, M), \ldots, (N_q, M)$ with $q = 2 \cdot 2^{2n/3}$ to authentication oracle $\mathbf{O}\Big($say $N_i = \langle i \rangle_{2n/3} \| 0^{n/3}$ for $i < 2^{2n/3}$, $N_i = \langle i - 2^{2n/3} + 1 \rangle_{2n/3} \| 1 \| 0^{n/3-1}$ for $2^{2n/3} \leq i < 2 \cdot 2^{2n/3}\Big)$; receive responses $T_i = \mathbf{O}(M_i), i \in [q]$.

2 : Make $\tilde{u}_1, \ldots, \tilde{u}_{p_1}$ forward queries to the primitive π with $p_1 = 2 \cdot 2^{2n/3}$ $\Big($say $\tilde{u}_a = 0^{n/3} \| \langle a \rangle_{2n/3}$ for $a < 2^{2n/3}, \tilde{u}_a = 0^{n/3-1} \| 1 \| \langle a - 2^{2n/3} + 1 \rangle_{2n/3}$ for $2^{2n/3} \leq a < 2 \cdot 2^{2n/3}\Big)$; receive responses $\tilde{v}_a = \pi(\tilde{u}_a), a \in [p_1]$.

3 : Make $\tilde{y}_1, \ldots, \tilde{y}_{p_2} \xleftarrow[\text{wor}]{\$} \{0,1\}^n$ backward queries to the primitive π with $p_2 = 2 \cdot 2^{2n/3}$; receive responses $\tilde{x}_b, b \in [p_2]$.

4 : Set $\text{Ext}_K := \{(i, a, b) \in [q] \times [p_1] \times [p_2] : (N_i \oplus \tilde{u}_a = K) \wedge (T_i \oplus \tilde{x}_b = 2K)\}$ and set $\hat{\mathcal{K}} = \phi$.

5 : For all $K \in \mathcal{K}$ with $|\text{Ext}_K| \geq 2$, carry out the following check : For all pairs of tuples $(i, a, b) \neq (i', a', b')$ in Ext_K, if $(N_i \oplus \tilde{v}_a \oplus \tilde{y}_b \oplus N_{i'} \oplus \tilde{v}_{a'} \oplus \tilde{y}_{b'} = 0)$, then add K to $\hat{\mathcal{K}}$.

Figure 3.3.: Interaction of the adversary with $(\mathbf{O}, \pi)$, where $\mathbf{O}$ is either the random oracle or the real construction oracle $\text{PDM}^*\text{MAC}_K^{\pi}$ and the primitive π.

queries to $\pi^{\pm}$ *and at most* q_v *queries to the verification oracle, we have*

$$\begin{aligned} Adv^{\text{MAC}}_{\text{PDM}^*\text{MAC}}(\mathcal{D}) \leq\ & q_v\epsilon + \\ & \frac{q_m^2(1 + 1202q_m + 3p + 312(p + q_m + 3q_v)) + p^2(q_m + q_v)}{2^{2n}} + \\ & \frac{2(p + q_m + 3q_v)^2 q_m}{2^{2n}} + \\ & \frac{6 + 2q_m^2\epsilon + q_m + \sqrt{6npq_m} + q_m\sqrt{3np} + p\sqrt{3nq_m} + 3q_m^2 q_v\epsilon + q_v}{2^n}. \end{aligned}$$

The proof for this theorem can be found in Sect. 3.6. If we assume $\epsilon \approx 2^{-n}$, the dominating term of advantage is $\sqrt{\frac{3n(pq_m^2 + q_m p^2)}{2^{2n}}}$. So the construction is secure as long as $p, q \ll \frac{2^{2n/3}}{n^{1/3}}$.

3.4.3. Single Keyed Version of PDM*MAC: 1K-PDM*MAC

The PDM*MAC construction calls one permutation, one key K associated with the permutation and one independent hash key K_h. We extend the specification of PDM*MAC to a single keyed version denoted by 1K-PDM*MAC. We use the technique of instantiating the hash key K_h by $K_h = \pi(K)$. We also assume that $N \neq 0$ and $\mathcal{H}$ is Poly hash. However, this technique is similar to that used in DWCDM (where $K_h = E_K(0)$). We prove that 1K-PDM*MAC for one instance of uniform π and uniform key K is secure up to attack complexity $\mathcal{O}(2^{2n/3})$.

Theorem 8. *Let $n \in \mathcal{N}$, and consider 1K-PDM*MAC$^{\pi}_{K}$ based on one permutation $\pi \xleftarrow{\$} \mathsf{Perm}(\{0,1\}^n)$, one key $K \xleftarrow{\$} \{0,1\}^n$. For any distinguisher $\mathcal{D}$ making at most q_m construction queries, at most p primitive queries to $\pi^{\pm}$ and at most q_v queries to the verification oracle, we have*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{MAC}}_{\mathsf{1K-PDM^*MAC}}(\mathcal{D}) \leq\ & q_v\epsilon + \\ & \frac{q_m^2(1+1202q_m+3p+312(p+q_m+3q_v))+p^2(q_m+q_v)}{2^{2n}} + \\ & \frac{2(p+q_m+3q_v)^2 q_m}{2^{2n}} + \\ & \frac{6+q_m^2\epsilon(2+3q_v)+3q_m+2p+\sqrt{6npq_m}+q_m\sqrt{3np}+p\sqrt{3nq_m}+q_v}{2^n}.\end{aligned}$$

The proof for this theorem can be found in Sect. 3.7.

3.5. Proof of Theorem 6

We use Coefficient-H technique [75, 132] (described in Sect. 1.5.1) to prove the theorem. The details are given below.

Game Description

We denote by q, the number of queries that $\mathcal{D}$ makes to one of the construction oracles PDMMAC$^{\pi}_{K}$ or φ, the queries being summarized by the transcript $\tau_q = \{(M_1, T_1), \ldots, (M_q, T_q)\}$. $\mathcal{D}$ also makes p queries to the primitive π, which are summarized by $\tau_p = \{(\tilde{u}_1, \tilde{v}_1), \ldots, (\tilde{u}_p, \tilde{v}_p)\}$. It may be assumed without loss of generality that both τ_q and τ_p have distinct elements.

After $\mathcal{D}$ has interacted with the oracles but before it has output its decision, the key K is also revealed to it. In the real world, this is the key used in the construction, while in the ideal world, it is a dummy value drawn uniformly at random from $\{0,1\}^n$. The full transcript of the interaction is denoted by $\tau = (\tau_q, \tau_p, K)$. The set of all attainable transcripts is denoted by $\mathcal{T}$, and we

partition $\mathcal{T}$ as $\mathcal{T}_{\text{good}} \sqcup \mathcal{T}_{\text{bad}}$, as described shortly. We let X_{re} be the random variable that takes values $\tau \in \mathcal{T}$ when $\mathcal{D}$ interacts with the real world and X_{id} to be the random variable that takes values $\tau \in \mathcal{T}$ when $\mathcal{D}$ interacts with the ideal world.

Transcript Equations Induced by the Distinguishing Game

This distinguishing game results in a system of equations obtained through the queries to the construction and primitive oracles. These are of the form-

Construction equations:			**Queries to primitive π:**		
$\pi(M_1 \oplus K) \oplus \pi(T_1 \oplus 2K)$	$=$	$3K \oplus M_1$	$\pi(\tilde{u}_1)$	$=$	$\tilde{v}_1$
$\vdots$			$\vdots$		
$\pi(M_q \oplus K) \oplus \pi(T_q \oplus 2K)$	$=$	$3K \oplus M_q$	$\pi(\tilde{u}_p)$	$=$	$\tilde{v}_p$

Furthermore, these equations can be expressed graphically as described in the *supplementary material*.

3.5.1. Bad Events

A transcript $\tau = (\tau_q, \tau_p, K)$ is said to be in $\mathcal{T}_{\text{bad}}$ and is called a *bad transcript* if and only if at least one of the following is satisfied-

Collision amongst two construction queries-

- B1. There exist $i \neq j \in [q]$ such that $\big(T_i \oplus M_j = 3K\big) \wedge \big(T_j \oplus M_i = 3K\big)$.

Collision within one construction query-

- B2. There exists $i \in [q]$ such that $T_i \oplus M_i = 3K$.

Collision amongst three construction queries-

- B3. There exist $i, j, k \in [q]$ such that $T_i \oplus M_j = T_j \oplus M_k = 3K$.
- B4. There exist $i, j, k \in [q]$ such that $T_i = T_j = T_k$.
- B5. There exist $i, j, k \in [q]$ such that $T_i = T_j = M_k \oplus 3K$.

Collision amongst two construction queries and one primitive query-

- B6. There exist $i \neq j \in [q], k \in [p]$ such that $\big(M_i \oplus T_j = 3K\big) \wedge \big(2K \oplus T_i = \tilde{u}_k\big)$.

- B7. There exist $i \neq j \in [q], k \in [p]$ such that $\big(M_i \oplus T_j = 3K\big) \wedge \big(K \oplus M_j = \tilde{u}_k\big)$.

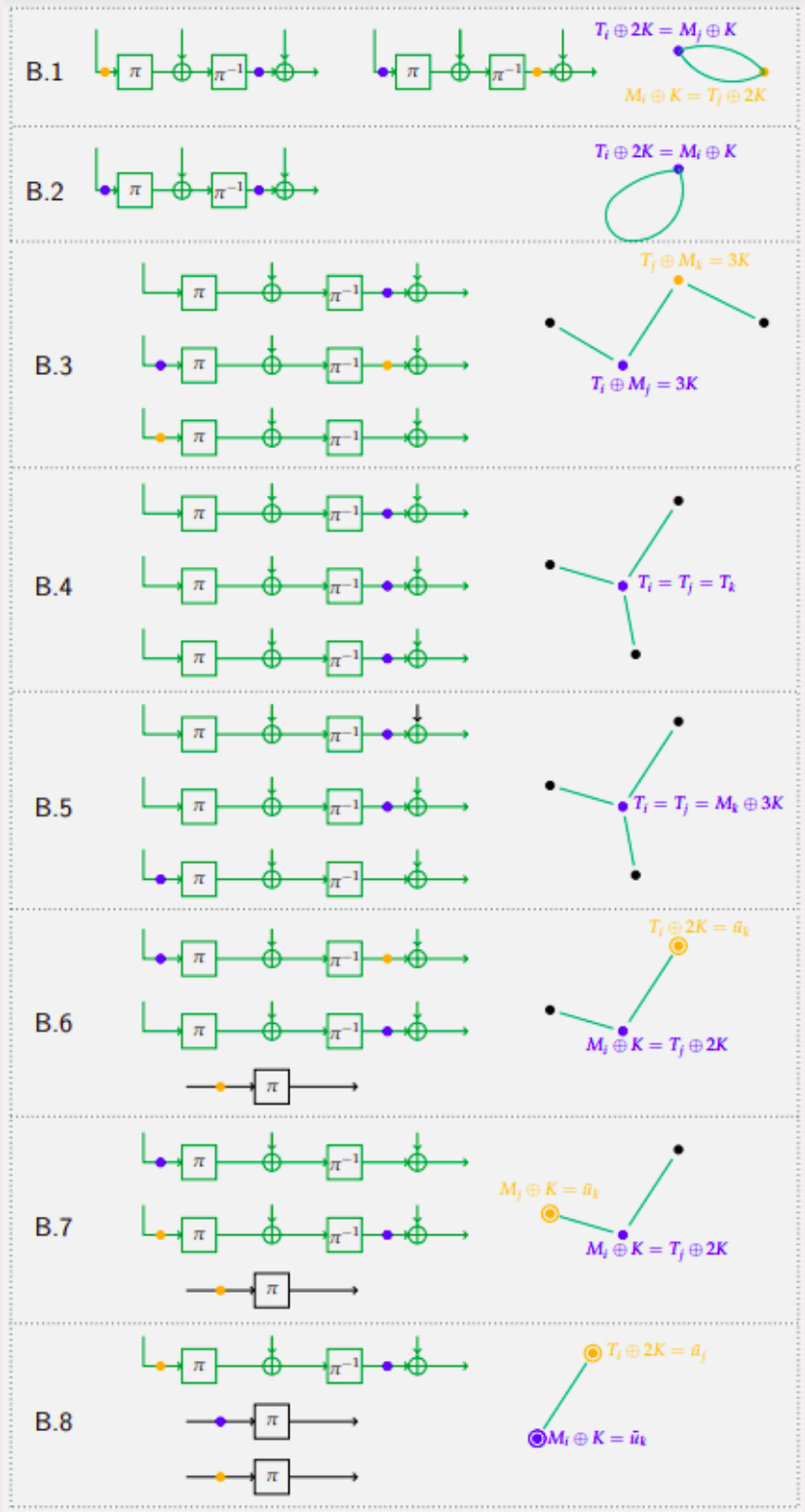


Figure 3.4.: Collisions amongst construction equations and/or primitive queries - figurative and graphical representations of the bad events.

Collision amongst one construction queries and two primitive queries-

- B8. There exist $i \in [q], j, k \in [p]$ such that $(K \oplus M_i = \tilde{u}_k) \wedge (2K \oplus T_i = \tilde{u}_j)$.

Any transcript $\tau \in \mathcal{T}_{\text{good}} = \mathcal{T} \setminus \mathcal{T}_{\text{bad}}$ is said to be a *good transcript*. A figurative and graphical description of the bad events is given in Fig.3.4. In addition, a circled vertex in any graph describing a bad event denotes a collision with a primitive query.

Probability of Bad Transcripts

$$\text{Now, } \Pr[\tau \in \mathcal{T}_{\text{bad}}] \leq \sum_{i=1}^{8} \Pr[\text{B}i].$$

Probability of events B1, B2, B4 and B5. Consider event B1. Since there are q construction queries (with randomness only in T_i and T_j, but not in M_i and M_j), $\Pr[\text{B1}] \leq \frac{q^2}{2^{2n}}$, $\Pr[\text{B2}] \leq \frac{q}{2^n}$, $\Pr[\text{B4}] \leq \frac{q^3}{2^{2n}}$ and $\Pr[\text{B5}] \leq \frac{q^3}{2^{2n}}$.

Probability of event B3. Let A_3 be any constant value.
Define $\Omega_3 = \{(j,i,k) | T_j \oplus M_j = T_i \oplus M_k\}$. Then-

$$\begin{aligned}
\Pr[\text{B3}] &= \Pr\left[(T_j \oplus M_j = T_i \oplus M_k) \wedge (3K = T_j \oplus M_k)\right] \\
&\leq \Pr\left[(3K = T_j \oplus M_k) \wedge (|\Omega_3| \geq A_3)\right] + \\
&\quad \Pr\left[(3K = T_j \oplus M_k) \wedge (|\Omega_3| \leq A_3)\right] \\
&\leq \Pr[|\Omega_3| \geq A_3] \cdot \frac{1}{2^n} + A_3 \cdot \frac{1}{2^n}.
\end{aligned}$$

If $A_3 = \frac{pq^2}{2^n} + \sqrt{\frac{6npq}{2^n}}$, then by Lemma 23 of appendix B,

$$\Pr[\text{B7}] \leq \frac{pq^2}{2^{2n}} + \frac{\sqrt{6npq}}{2^n} + \frac{2}{2^n}.$$

Probability of event B6. Since there are q construction queries and p queries to the primitive, $\Pr[\text{B6}] \leq \frac{pq^2}{2^{2n}}$.

Probability of event B7. Let A_7 be any constant value.
Define $\Omega_7 = \{(j,i,k) | T_j \oplus 3M_j = M_i \oplus 3\tilde{u}_k\}$. Then-

$$\begin{aligned}
\Pr[\text{B7}] &= \Pr\left[(T_j \oplus 3M_j = M_i \oplus 3\tilde{u}_k) \wedge (3K = T_j \oplus M_i)\right] \\
&\leq \Pr\left[(3K = T_j \oplus M_i) \wedge (|\Omega_7| \geq A_7)\right] + \\
&\quad \Pr\left[(3K = T_j \oplus M_i) \wedge (|\Omega_7| \leq A_7)\right] \\
&\leq \Pr[|\Omega_7| \geq A_7] \cdot \frac{1}{2^n} + A_7 \cdot \frac{1}{2^n}.
\end{aligned}$$

If $A_7 = \frac{pq^2}{2^n} + q\sqrt{3np}$, then by Lemma 22 of appendix B,

$$\Pr[\mathsf{B7}] \leq \frac{pq^2}{2^{2n}} + \frac{q\sqrt{3np}}{2^n} + \frac{2}{2^n}.$$

Probability of event B8. Let A_8 be any constant value.
Define $\Omega_8 = \{(i,k,j) | 2M_i \oplus T_i = 2\tilde{u}_k \oplus \tilde{u}_j\}$. Then-

$$\begin{aligned}
\Pr[\mathsf{B8}] &= \Pr\left[\left(2M_i \oplus T_i = 2\tilde{u}_k \oplus \tilde{u}_j\right) \wedge \left(2K = T_i \oplus \tilde{u}_j\right)\right] \\
&\leq \Pr\left[\left(2K = T_i \oplus \tilde{u}_j\right) \wedge \left(|\Omega_8| \geq A_8\right)\right] + \\
&\quad \Pr\left[\left(2K = T_i \oplus \tilde{u}_j\right) \wedge \left(|\Omega_8| \leq A_8\right)\right] \\
&\leq \Pr[|\Omega_8| \geq A_8] \cdot \frac{1}{2^n} + A_8 \cdot \frac{1}{2^n}.
\end{aligned}$$

If $A_8 = \frac{p^2q}{2^n} + p\sqrt{3nq}$, then by Lemma 22 of appendix B,

$$\Pr[\mathsf{B8}] \leq \frac{p^2q}{2^{2n}} + \frac{p\sqrt{3nq}}{2^n} + \frac{2}{2^n}.$$

Thus,

$$\Pr[\tau \in \mathcal{T}_{\text{bad}}] \leq \frac{q^2 + 2q^3 + 3pq^2 + p^2q}{2^{2n}} + \frac{6 + q + q\sqrt{3np} + \sqrt{6npq} + p\sqrt{3nq}}{2^n}.$$

3.5.2. Good Transcripts

Observe that any good transcript $\tau \in \mathcal{T}_{\text{good}}$ must necessarily be induced by a graph $\mathcal{G}^{\tau}_{\text{eq}}$, which satisfies the following conditions:

- There is no cycle in $\mathcal{G}^{\tau}_{\text{eq}} = (\mathsf{V}_{\text{eq}}, \mathsf{E}_{\text{eq}}, \mathcal{L}_{\text{eq}})$.
- There is no path P in $\mathcal{G}^{\tau}_{\text{eq}}$ such that $\mathcal{L}_{\text{eq}}(P) := \sum_{e \in P} \mathcal{L}(e) = 0$.

Also, it may perhaps contain some circled vertices (denoting collisions with some permutation queries). In fact, every component of $\mathcal{G}^{\tau}_{\text{eq}}$ has size at most 3, due to the restrictions of bad events B3, B4 and B5. Furthermore, no component of $\mathcal{G}^{\tau}_{\text{eq}}$ of size 3 has a circled vertex due to B6 and B7, and components of size 2 may have at most one circled vertex due to B8. We first modify the good transcripts so as to make certain that none of the vertices of $\mathcal{G}^{\tau}_{\text{eq}}$ are circled, as follows:

- If there exists $i \in [q]$ and $k \in [p]$ such that $K \oplus M_i = \tilde{u}_k$, then remove (M_i, T_i) from τ_q and add $(2K \oplus T_i, 3K \oplus M_i \oplus \tilde{v}_k)$ to τ_p.

- If there exists $i \in [q]$ and $j \in [p]$ such that $2K \oplus T_i = \tilde{u}_j$, then remove (M_i, T_i) from τ_q and add $(K \oplus M_i, 3K \oplus M_i \oplus \tilde{v}_j)$ to τ_p.

Denote the new transcript of primitive queries by F, so that $|F| = p' = p + s$ and $q' = q - s$. Let $S' = \{0,1\}^n \setminus \{\tilde{v}_k \,|\, (\tilde{u}_k, \tilde{v}_k) \in F\}$. Denoting $Q = T \oplus 2K$ and $P = M \oplus K$, assume that for a modified good transcript τ, there are t_1 construction equations of the form

$$\begin{aligned} \pi(P_1) \oplus \pi(Q) &= \lambda_1 \\ \pi(P_2) \oplus \pi(Q) &= \lambda_2, \end{aligned}$$

t_2 construction equations of the form

$$\begin{aligned} \pi(P) \oplus \pi(Q_1) &= \lambda_1 \\ \pi(Q_1) \oplus \pi(Q_2) &= \lambda_2, \end{aligned}$$

and $q' - t_1 - t_2$ construction equations of the form $\pi(P) \oplus \pi(Q) = \lambda$.
Let p_{re} be the probability of a modified transcript τ satisfying the system of equations $\pi(M_i \oplus K) \oplus \pi(T_i \oplus 2K) = 3K \oplus M_i$, $i \in [q']$.

Good Transcript Analysis

The probabilities that X_{re} and X_{id} attain a particular value τ can be computed as

$$\begin{aligned} \Pr[\mathsf{X}_{\text{id}} = \tau] &= \frac{1}{2^{nq}} \cdot \frac{1}{(2^n)_p} \cdot \frac{1}{2^n} \text{ and} \\ \Pr[\mathsf{X}_{\text{re}} = \tau] &= p_{\text{re}} \cdot \frac{1}{(2^n)_{p'}} \cdot \frac{1}{2^n}, \end{aligned}$$

where p_{re} can be computed using Eqn.s (3.1) and (3.2)) as follows.

Probability that construction equations are satisfied.

Cases I and II. $\left(\pi(\mathbf{P_{2i-1}}) \oplus \pi(\mathbf{Q_i}) = \pi\boldsymbol{\lambda}_{\mathbf{2i-1}}, \text{ß}(\mathbf{P_{2i}}) \oplus \pi(\mathbf{Q_i}) = \boldsymbol{\lambda}_{\mathbf{2i}} \text{ or } \pi(\mathbf{P}_{\mathbf{2t_1+j}}) \oplus \pi\mathbf{Q}_{\mathbf{t_1+2j}}) = \boldsymbol{\lambda}_{\mathbf{2t_1+2j-1}}, \pi\mathbf{Q}_{\mathbf{t_1+2j-1}}) \oplus \pi\mathbf{Q}_{\mathbf{t_1+2j}}) = \boldsymbol{\lambda}_{\mathbf{2t_1+2j}}\right)$.
By Eqn. (3.4),

$$\begin{aligned} &\Pr\left[\begin{smallmatrix} \pi(P_1)\oplus\pi(Q_1)=\lambda_1, \pi(P_2)\oplus\pi(Q_1)=\lambda_2, \ldots, \\ \pi(P_{2t_1+t_2})\oplus\pi(Q_{t_1+2t_2-1})=\lambda_{2t_1+2t_2-1}, \pi(Q_{t_1+2t_2-1})\oplus\pi(Q_{t_1+2t_2})=\lambda_{2t_1+2t_2} \end{smallmatrix}\right] \\ \geq\ & \frac{1}{2^{2n(t_1+t_2)}}\left(1 - \frac{3 \cdot q' \cdot 2^n \cdot p'^2}{(2^n - p')^3}\right), \text{ since } t_1 + t_2 \leq q'. \end{aligned}$$

Case III. $\left(\pi(\mathbf{P_{2t_1+t_2+1}})\oplus\pi\mathbf{Q_{t_1+2t_2+1}}) = \boldsymbol{\lambda_{2t_1+2t_2+1}}\right)$.
By Eqn. (3.3),

$$\Pr\begin{bmatrix}\pi(P_{2t_1+t_2+1})\oplus\pi(Q_{t_1+2t_2+1})=\lambda_{2t_1+2t_2+1},\ldots,\\ \pi(P_{q'-t_2})\oplus\pi(Q_{q'-t_1})=\lambda_{q'}\end{bmatrix}$$

$$\geq \frac{1}{2^{n(q'-2t_1-2t_2)}}\left(1-\frac{q'\cdot p'^2}{(2^n-p')^2}\right), \text{ since } q'-2t_1-2t_2\leq q'.$$

$$\begin{aligned}\text{Thus, } p_{\text{re}} &\geq \frac{1}{2^{nq'}}\left(1-\frac{3\cdot q'\cdot 2^n\cdot p'^2}{(2^n-p')^3}\right)\left(1-\frac{q'\cdot p'^2}{(2^n-p')^2}\right)\\ &\geq \frac{1}{2^{nq'}}\left(1-\frac{6\cdot 2^n\cdot q(p+q)^2}{2^{2n}}\right)\left(1-\frac{2\cdot 2^n\cdot q(p+q)^2}{2^{2n}}\right)\\ &\quad\left(\text{since } q\geq q', \frac{2}{2^n}\geq p'\geq p \text{ and } (p+q)\geq p'\right)\\ &\geq \frac{1}{2^{nq'}}\left(1-\frac{8\cdot 2^n\cdot q(p+q)^2}{2^{2n}}\right).\end{aligned}$$

$$\begin{aligned}\text{Thus, } \frac{\Pr[\mathsf{X}_{\text{re}}]}{\Pr[\mathsf{X}_{\text{id}}]} &\geq \frac{2^{nq}}{2^{nq'}}\cdot\frac{(2^n)_p}{(2^n)_{p'}}\cdot\left(1-\frac{8q(p+q)^2}{2^{2n}}\right)\geq\left(1-\frac{8q(p+q)^2}{2^{2n}}\right),\\ \text{i.e. } \frac{\Pr[\mathsf{X}_{\text{re}}]}{\Pr[\mathsf{X}_{\text{id}}]} &\geq \left(1-\epsilon_{\text{good}}\right), \text{ where } \epsilon_{\text{good}}=\frac{8q(p+q)^2}{2^{2n}}.\end{aligned}$$

3.6. Proof of Theorem 7

We use Coefficient-H technique [75, 132] (described in Sect. 1.5.1) to prove the theorem.

Forging Game

An upper bound for the nonce-based MAC advantage can be computed by adapting the distinguishing game in Sect. 1.2.2 (the game is described in Page 5, [68]) as follows. $\mathcal{D}$ makes q_m queries to one of the construction (authentication, or Auth) oracles $\mathsf{PDM}^*\mathsf{MAC}^{\pi}_{K,K_h}$ or φ, the queries being summarized by the authentication transcript

$$\tau_0^m = \{(N_1, M_1, T_1), \ldots, (N_{q_m}, M_{q_m}, T_{q_m})\},$$

and by q_v, the number of verification queries that $\mathcal{D}$ makes to one of the construction (verification, or Ver) oracles $\mathsf{Ver}^{\pi}_{K,K_h}$ or $\perp$, the queries being summarized by the verification transcript $\tau_0^v = \{(N_1', M_1', T_1', b_1), \ldots$

$, (N'_{q_v}, M'_{q_v}, T'_{q_v}, b_{q_v})\}$, where $\forall a, b_a \in \{0,1\}$ are the output values of the verification oracle (in the real world, the oracle checks if $\mathsf{Auth}(N'_a, M'_a) = T'_i$, and returns 1 or 0 according to whether the equality holds or not, respectively, while in the ideal world, $b_a = 0$ for all a). $\mathcal{D}$ also makes p queries to the primitive π, which are summarized by $\tau_p = \{(\tilde{u}_1, \tilde{v}_1), \ldots, (\tilde{u}_p, \tilde{v}_p)\}$. It may be assumed without loss of generality that each of τ_0^m, τ_0^v, τ has distinct elements.

After $\mathcal{D}$ has interacted with the oracles but before it has output its decision, the keys K and K_h are also revealed to it. In the real world, these are the keys used in the construction, while in the ideal world, they are dummy values drawn uniformly at random from $\{0,1\}^n$. The full transcript of the interaction is denoted by $\tau = (\tau_0^m, \tau_0^v, \tau_p, K, K_h)$. The set of all attainable transcripts is denoted by $\mathcal{T}$, and we partition $\mathcal{T}$ as $\mathcal{T}_{\text{good}} \sqcup \mathcal{T}_{\text{bad}}$, as described shortly.

Transcript Equations Induced by the Forging Game: The system of equations has a similar form, and is extended by a system of non-equations, as given below-

Authentication equations:

$$\begin{aligned} \pi(N_1 \oplus K) \oplus \pi(T_1 \oplus 2K) &= 3K \oplus N_1 \oplus H_1 \\ &\vdots \\ \pi(N_{q_m} \oplus K) \oplus \pi(T_{q_m} \oplus 2K) &= 3K \oplus N_{q_m} \oplus H_{q_m} \end{aligned}$$

Verification non-equations:

$$\begin{aligned} \pi(N'_1 \oplus K) \oplus \pi(T'_1 \oplus 2K) &\neq 3K \oplus N'_1 \oplus H'_1 \\ &\vdots \\ \pi(N'_{q_v} \oplus K) \oplus \pi(T'_{q_v} \oplus 2K) &\neq 3K \oplus N'_{q_v} \oplus H'_{q_v} \end{aligned}$$

Queries to primitive π:

$$\begin{aligned} \pi(\tilde{u}_1) &= \tilde{v}_1 \\ &\vdots \\ \pi(\tilde{u}_p) &= \tilde{v}_p, \end{aligned}$$

where $H_i = \mathcal{H}_{K_h}(M_i), \forall i \in [q_m]$ and $H'_j = \mathcal{H}'_{K_h}(M'_j), \forall j \in [q_v]$.

3.6.1. Bad Events

A transcript $\tau = (\tau_0^m, \tau_0^v, \tau_p, K, K_h)$ is said to be in $\mathcal{T}_{\text{bad}}$ and is called a *bad transcript* if and only if there exists a tuple $(N_i, M_i, T_i) \in \tau_0^m$, $(N'_a, M'_a, T'_a) \in$

τ_0^v and $(\tilde{u}_j, \tilde{v}_j), (\tilde{x}_k, \tilde{y}_k) \in \tau_p$ such that at least one of the following is satisfied-

Collision amongst two authentication queries-

- B1. There exist $i \neq j \in [q_m]$ such that

$$\left(T_i = T_j\right) \wedge \left(N_i \oplus H_i = N_j \oplus H_j\right).$$

- B2. There exist $i \neq j \in [q_m]$ such that

$$\left(T_i \oplus N_j = 3K\right) \wedge \left(N_i \oplus H_i = N_j \oplus H_j\right).$$

- B3. There exist $i \neq j \in [q_m]$ such that

$$\left(T_i \oplus N_j = 3K\right) \wedge \left(T_j \oplus N_i = 3K\right).$$

Collision within one authentication query-

- B4. There exists $i \in [q_m]$ such that $T_i \oplus N_i = 3K$.

Collision amongst three authentication queries-

- B5. There exist $i, j, k \in [q_m]$ such that $T_i \oplus N_j = T_j \oplus N_k = 3K$.
- B6. There exist $i, j, k \in [q_m]$ such that $T_i = T_j = T_k$.
- B7. There exist $i, j, k \in [q_m]$ such that $T_i = T_j = N_k \oplus 3K$.

Collision amongst two authentication queries and one primitive query-

- B8. There exist $i \neq j \in [q_m], k \in [p]$ such that

$$\left(N_i \oplus T_j = 3K\right) \wedge \left(2K \oplus T_i = \tilde{u}_k\right).$$

- B9. There exist $i \neq j \in [q_m], k \in [p]$ such that

$$\left(N_i \oplus T_j = 3K\right) \wedge \left(K \oplus N_j = \tilde{u}_k\right).$$

Collision amongst one authentication query and two primitive queries-

- B10. There exist $i \in [q_m], j, k \in [p]$ such that

$$\left(K \oplus N_i = \tilde{u}_k\right) \wedge \left(2K \oplus T_i = \tilde{u}_j\right).$$

Collision amongst one verification query and two primitive queries-

- B11. There exist $a \in [q_v], j, k \in [p]$ such that

$$\left(K \oplus N'_a = \tilde{u}_k\right) \wedge \left(2K \oplus T'_a = \tilde{u}_j\right).$$

Collision amongst one authentication and one verification query-

- B12. There exist $i \in [q_m], a \in [q_v]$ such that
$$\left(N_i = N'_a\right) \wedge \left(H_i = H'_a\right) \wedge \left(T_i = T'_a\right).$$

Collision amongst two authentication queries and one verification query, with an extra condition-

- B13. There exist $i, j \in [q_m], a \in [q_v]$ such that
$$\left(H_i \oplus H_j \oplus H'_a = N_i \oplus N_j \oplus N'_a \oplus 2K\right) \text{ and}$$
$$\left(N'_a = N_i\right) \wedge \left(T_i \oplus N_j = 3K\right) \wedge \left(T_j = T'_a\right).$$
- B14. There exist $i, j \in [q_m], a \in [q_v]$ such that
$$\left(H_i \oplus H_j \oplus H'_a = N_i \oplus N_j \oplus N'_a \oplus 2K\right) \text{ and}$$
$$\left(T'_a \oplus N_i = 3K\right) \wedge \left(T_i \oplus N_j = 3K\right) \wedge \left(T_j \oplus N'_a = 3K\right).$$
- B15. There exist $i, j \in [q_m], a \in [q_v]$ such that
$$\left(H_i \oplus H_j \oplus H'_a = N_i \oplus N_j \oplus N'_a \oplus 2K\right) \text{ and}$$
$$\left(N'_a = N_i\right) \wedge \left(T_i = T_j\right) \wedge \left(T'_a \oplus N_j = 3K\right).$$

Any transcript $\tau \in \mathcal{T}_{\text{good}} = \mathcal{T} \setminus \mathcal{T}_{\text{bad}}$ is said to be a *good transcript*. A figurative and graphical description of the bad events is provided in appendix C. In these figures, a circled vertex in any graph describing a bad event denotes a collision with a primitive query.

Probability of Bad Transcripts

$$\text{Now, } \Pr[\tau \in \mathcal{T}_{\text{bad}}] \leq \sum_{i=1}^{15} \Pr[\text{B}i].$$

Probability of events B1, B2, B3, B4, B6 and B7. Consider event B1. Since there are q_m authentication queries (with randomness only in T_i and T_j, but not in N_i and N_j) and since $\mathcal{H}$ is an ϵ-differential hash function, $\Pr[\text{B1}] \leq \frac{q_m^2 \epsilon}{2^n}$. Similarly, $\Pr[\text{B2}] \leq \frac{q_m^2 \epsilon}{2^n}$, $\Pr[\text{B3}] \leq \frac{q_m^2}{2^{2n}}$, $\Pr[\text{B4}] \leq \frac{q_m}{2^n}$, $\Pr[\text{B6}] \leq \frac{q_m^3}{2^{2n}}$ and $\Pr[\text{B7}] \leq \frac{q_m^3}{2^{2n}}$.

Probability of event B5. Let A_5 be any constant value.
Define $\Omega_5 = \{(j, i, k) | T_j \oplus N_j = T_i \oplus N_k\}$. Then-

$$\begin{aligned}
\Pr[\text{B5}] &= \Pr\left[\left(T_j \oplus N_j = T_i \oplus N_k\right) \wedge \left(3K = T_j \oplus N_i\right)\right] \\
&\leq \Pr\left[\left(3K = T_j \oplus N_i\right) \wedge \left(|\Omega_5| \geq A_5\right)\right] + \\
&\quad \Pr\left[\left(3K = T_j \oplus N_i\right) \wedge \left(|\Omega_5| \leq A_5\right)\right] \\
&\leq \Pr[|\Omega_5| \geq A_5] \cdot \frac{1}{2^n} + A_5 \cdot \frac{1}{2^n}.
\end{aligned}$$

If $A_5 = \frac{pq_m^2}{2^n} + \sqrt{\frac{6npq_m}{2^n}}$, then by Lemma 23 of appendix B,

$$\Pr[\text{B5}] \leq \frac{pq_m^2}{2^{2n}} + \frac{\sqrt{6npq_m}}{2^n} + \frac{2}{2^n}.$$

Probability of event B8. Since there are q_m authentication queries and p queries to the primitive, $\Pr[\text{B8}] \leq \frac{pq_m^2}{2^{2n}}$.

Probability of event B9. Let A_9 be any constant value.
Define $\Omega_9 = \{(j,i,k) | T_j \oplus N_j = T_i \oplus N_k\}$. Then-

$$\begin{aligned}
\Pr[\text{B9}] &= \Pr\left[\left(T_j \oplus 3N_j = N_i \oplus 3\tilde{u}_k\right) \wedge \left(3K = T_j \oplus N_i\right)\right] \\
&\leq \Pr\left[\left(3K = T_j \oplus N_i\right) \wedge \left(|\Omega_9| \geq A_9\right)\right] + \\
&\quad \Pr\left[\left(3K = T_j \oplus N_i\right) \wedge \left(|\Omega_9| \leq A_9\right)\right] \\
&\leq \Pr[|\Omega_9| \geq A_9] \cdot \frac{1}{2^n} + A_9 \cdot \frac{1}{2^n}.
\end{aligned}$$

If $A_9 = \frac{pq_m^2}{2^n} + q_m\sqrt{3np}$, then by Lemma 22 of appendix B,

$$\Pr[\text{B9}] \leq \frac{pq_m^2}{2^{2n}} + \frac{q_m\sqrt{3np}}{2^n} + \frac{2}{2^n}.$$

Probability of event B10. Let A_{10} be any constant value.
Define $\Omega_{10} = \{(i,k,j) | 2N_i \oplus T_i = 2\tilde{u}_k \oplus \tilde{u}_j\}$. Then-

$$\begin{aligned}
\Pr[\text{B10}] &= \Pr\left[\left(2N_i \oplus T_i = 2\tilde{u}_k \oplus \tilde{u}_j\right) \wedge \left(2K = T_i \oplus \tilde{u}_j\right)\right] \\
&\leq \Pr\left[\left(2K = T_i \oplus \tilde{u}_j\right) \wedge \left(|\Omega_{10}| \geq A_{10}\right)\right] + \\
&\quad \Pr\left[\left(2K = T_i \oplus \tilde{u}_j\right) \wedge \left(|\Omega_{10}| \leq A_{10}\right)\right] \\
&\leq \Pr[|\Omega_{10}| \geq A_{10}] \cdot \frac{1}{2^n} + A_{10} \cdot \frac{1}{2^n}.
\end{aligned}$$

If $A_{10} = \frac{p^2q_m}{2^n} + p\sqrt{3nq_m}$, then by Lemma 22 of appendix B,

$$\Pr[\text{B10}] \leq \frac{p^2q_m}{2^{2n}} + \frac{p\sqrt{3nq_m}}{2^n} + \frac{2}{2^n}.$$

Probability of event B11. Since there are q_v verification queries and p queries to the primitive, $\Pr[\text{B11}] \leq \frac{p^2q_v}{2^{2n}}$.

Probability of event B12. Since there are q_v verification queries and H is an ϵ-differential hash function, $\Pr[\text{B12}] \leq q_v\epsilon$.

Probability of events B13, B14 and B15. For all three events, $H_i \oplus H_j \oplus H'_a = (N_i \oplus N_j \oplus N'_a) \oplus 3K$. Since there are q_m authentication queries and q_v verification queries and assuming $\mathcal{H}$ is an ϵ-3-way-regular hash function, Pr[B13], Pr[B14] and Pr[B15] are all at most $\frac{q_m^2 q_v \epsilon}{2^n}$.

Thus,

$$\Pr[\tau \in \mathcal{T}_{\text{bad}}] \leq \frac{q_m^2 + 2q_m^3 + 3pq_m^2 + p^2 q_m + p^2 q_v}{2^{2n}} + \frac{2q_m^2\epsilon + q_m + q_m\sqrt{3np} + \sqrt{6npq_m} + p\sqrt{3nq_m} + 6 + 3q_m^2 q_v \epsilon}{2^n} + q_v\epsilon.$$

3.6.2. Good Transcripts

Observe that any good transcript $\tau \in \mathcal{T}_{\text{good}}$ must necessarily be induced by a graph $\mathcal{G}^{\tau}_{\text{eq,neq}}$, which satisfies the following conditions:

- There is no cycle of equation-inducing edges in $\mathcal{G}^{\tau}_{\text{eq}} = (\mathsf{V}_{\text{eq}}, \mathsf{E}_{\text{eq}}, \mathcal{L}|_{\mathsf{E}_{\text{eq}}})$.
- There is no path P in $\mathcal{G}^{\tau}_{\text{eq}}$ such that $\mathcal{L}(P) := \sum_{e \in P} \mathcal{L}(e) = 0$.
- For all the cycles C in $\mathcal{G}^{\tau}_{\text{eq,neq}}$ whose edge set consists of all but one equation edges $e \in \mathsf{E}_{\text{eq}}$ and exactly one non-equation edge $e' \in \mathsf{E}_{\text{neq}}$, $\mathcal{L}(C) \neq 0$.

It may perhaps contain some circled vertices (denoting collisions with some permutation queries). It shall be assumed that the edges in $\mathsf{E}^{\tau}_{\text{eq}}$ are continuous edges, colored green, and edges in $\mathsf{E}^{\tau}_{\text{neq}}$ are dotted edges, colored red. In fact, every component of $\mathcal{G}^{\tau}_{\text{eq}}$ has size at most 3, due to the restrictions of bad events B5, B6 and B7. Furthermore, no component of $\mathcal{G}^{\tau}_{\text{eq}}$ of size 3 has a circled vertex due to B8 and B9, and components of size 2 of $\mathcal{G}^{\tau}_{\text{eq}}$ as well as $\mathcal{G}^{\tau}_{\text{eq,neq}}$ may have at most one circled vertex due to B10 and B11. Finally, the restrictions by bad events B13, B14 and B15 ensure that $\mathcal{G}^{\tau}_{\text{eq,neq}}$ satisfies the condition $\mathcal{L}(C) \neq 0$ for a cycle containing exactly one non-equation edge.

We first modify the good transcripts in such a way that no vertices remain circled:

- If there exists $i \in [q_m]$ and $k \in [p]$ such that $K \oplus N_i = \tilde{u}_k$, then remove (N_i, M_i, T_i) from τ_0^m and add $(2K \oplus T_i, 3K \oplus N_i \oplus H_i \oplus \tilde{v}_k)$ to τ_p.
- If there exists $i \in [q_m]$ and $j \in [p]$ such that $2K \oplus T_i = \tilde{u}_j$, then remove (N_i, M_i, T_i) from τ_0^m and add $(K \oplus N_i, 3K \oplus N_i \oplus H_i \oplus \tilde{v}_j)$ to τ_p.

Denote the new set of primitive transcripts by F, so that $|F| = p' = p + s$ and $q'_m = q_m - s$. Let $S' \subseteq \{0,1\}^n$ such that $S' = \{0,1\}^n \setminus \{\tilde{v}_k \mid (\tilde{u}_k, \tilde{v}_k) \in F\}$. Let p_{re} be the probability of a modified transcript τ satisfying the system of equations $\pi(N_i \oplus K) \oplus \pi(T_i \oplus 2K) = 3K \oplus N_i \oplus \mathcal{H}_{K_h}(M_i)$, $i \in [q']$.

Good Transcript Analysis

The probabilities of X_{re} and X_{id} attaining a particular value τ can be computed as follows-

$$\begin{aligned}\Pr[\mathsf{X}_{\text{id}}=\tau] &= \frac{1}{2^{nq_m}}\cdot 1\cdot\frac{1}{(2^n)_p}\cdot\left(\frac{1}{2^n}\right)^2 \text{ and}\\ \Pr[\mathsf{X}_{\text{re}}=\tau] &= p_{\text{re}}\cdot\frac{1}{(2^n)_{p'}}\cdot\left(\frac{1}{2^n}\right)^2.\end{aligned}$$

Probability that authentication equations and verification non-equations are satisfied.

By Corollary 2,

$$\Pr\left[\begin{array}{c}\Big(\pi(P_1)\oplus\pi(Q_1)=\lambda_1,\pi(P_2)\oplus\pi(Q_1)=\lambda_2,\ldots,\\ \pi(P_{2t_1+t_2})\oplus\pi(Q_{t_1+2t_2-1})=\lambda_{2t_1+2t_2-1},\pi(Q_{t_1+2t_2-1})\oplus\pi(Q_{t_1+2t_2})=\lambda_{2t_1+2t_2}\\ \pi(P_{2t_1+t_2+1})\oplus\pi(Q_{t_1+2t_2+1})=\lambda_{2t_1+2t_2+1},\ldots,\\ \pi(P_{q'_m-t_2})\oplus\pi(Q_{q'_m-t_1})=\lambda_{q'_m}\Big)\wedge\\ \Big(\big(\pi(X'_1)\oplus\pi(X'_2)\neq\lambda'_1\big)\wedge\big(\pi(X'_2)\oplus\pi(X'_3)\neq\lambda'_2\big)\wedge\ldots\wedge\big(\pi(X'_{2q'_v-1})\oplus\pi(X'_{2q'_v})\neq\lambda'_{2q'_v-1}\big)\Big)\end{array}\right]$$

$$\leq\frac{1}{2^{nq'_m}}\left(1-\frac{1200{q'_m}^3+312(p'+3q_v){q'_m}^2+2(p'+3q_v)^2q'_m}{2^{2n}}\right)\left(1-\frac{q_v}{2^n}\right).$$

Therefore, p_{re} must be at least $\frac{1}{2^{nq_m}}\left(1-\frac{1200q_m^3+312(p'+3q_v)q_m^2+2(p'+3q_v)^2q_m}{2^{2n}}\right)\left(1-\frac{q_v}{2^n}\right)$, so that-

$$\begin{aligned}\frac{\Pr[\mathsf{X}_{\text{re}}]}{\Pr[\mathsf{X}_{\text{id}}]} &\geq \frac{2^{nq_m}}{2^{nq'_m}}\cdot\frac{(2^n)_p}{(2^n)_{p'}}\left(1-\frac{q_v}{2^n}\right)\left(1-\frac{1200{q'_m}^3+312(p'+3q_v){q'_m}^2+2(p'+3q_v)^2q'_m}{2^{2n}}\right)\\ &\geq \left(1-\frac{q_v}{2^n}\right)\cdot\left(1-\frac{1200q_m^3+312(p+q_m+3q_v)q_m^2+2(p+q_m+3q_v)^2q_m}{2^{2n}}\right),\\ &\quad\text{since } q'_m\leq q_m, p'\leq p+q_m\\ &\geq \left(1-\epsilon_{\text{good}}\right),\text{ where}\\ \epsilon_{\text{good}} &= \frac{q_v}{2^n}+\frac{1200q_m^3+312(p+q_m+3q_v)q_m^2+2(p+q_m+3q_v)^2q_m}{2^{2n}}.\end{aligned}$$

3.7. Proof of Theorem 8

The proof is similar to that of PDM*MAC, except for some extra bad cases. We add the following cases after B15. The cases are as follows.

- B16. There exists $i \in [q_m]$ such that $T_i = 3K$.
- B17. There exists $i \in [q_m]$ such that $\pi(N_i \oplus K) \oplus \mathcal{H}_{K_h}(M_i) \oplus N_i \oplus 3K = K_h$.
- B18. There exists $k \in [p]$ such that $\tilde{u}_k = K$.
- B19. There exists $k \in [p]$ such that $\tilde{y}_k = K_h$.

Probability of B16. There are q_m authentication queries. Hence, $\Pr[\text{B16}] \leq \frac{q_m}{2^n}$.

Probability of B17. In this case, N_i and M_i are fixed. Thus, $\Pr[\text{B17}] = \Pr[\pi(N_i \oplus K) \oplus 3K = \mathcal{H}_{K_h}(M_i) \oplus N_i \oplus K_h]$. As K and K_h are independently sampled in the ideal world, we obtain $\Pr[\text{B17}] \leq \frac{q_m}{2^n}$, by conditioning H.

Probability of B18 and B19. Since there are p queries to the primitive, $\Pr[\text{B18}], \Pr[\text{B19}] \leq \frac{p}{2^n}$.

Good Transcript Analysis

The good transcript analysis is exactly the same except in this case $\Pr[\mathsf{X}_{\text{re}} = \tau] = p_{\text{re}} \cdot \frac{1}{(2^n)_{p'}} \cdot \left(\frac{1}{2^n}\right)$ (as only the construction key K needs to be sampled, the last term in the expression is $\frac{1}{2^n}$ instead $(\frac{1}{2^n})^2$). However, this does not change the lower bound of $\frac{\Pr[\mathsf{X}_{\text{re}}]}{\Pr[\mathsf{X}_{\text{id}}]}$.

3.8. Summary

Our designs are minimal in structure in the number of permutation and key instances. However, PDMMAC makes two calls to one permutation π, one forward call to π and another inverse call to π^{-1}. We already know that PRFs with one permutation call can not provide more than birthday bound security and hence we need at least two calls to the permutation. Thus, the question

Can we design a BBB secure PRF with one permutation with two forward calls?

remains unanswered and the design of such a construction can be interesting to the community. A possible approach to proceed with this problem is to prove the $2n/3$-bit BBB security of SoKAC1. This design has been mentioned to be at most $n/2$-bit secure [53] accompanied by a birthday bound attack. However, the attack is possibly wrong and SoKAC1 may provide $2n/3$-bit BBB security.

4. Permutation-Based EDM: An Inverse-Free BBB Secure PRF

Abstract

In CRYPTO 2019, Chen et al. initiated interesting research in the direction of designing PRFs based on public permutations. They proposed SoEM22 and SoKAC21, two beyond the birthday bound secure n-bit to n-bit PRF constructions built on public permutations, where n is the size of the permutation. However, both constructions require two independent instances of public permutations. In FSE 2020, Chakraborti et al. proposed a single public permutation-based n-bit to n-bit beyond the birthday bound secure PRF, to which they referred as PDMMAC. Although the construction is minimal in the number of permutations, it requires an inverse call of its underlying permutation. Coming up with a beyond the birthday bound secure public permutation-based n-bit to n-bit PRF with a single permutation and two forward calls was left as an open problem in their paper. In this work, we propose pEDM, a single permutation-based n-bit to n-bit PRF with two calls that does not require invertibility of the permutation. We have shown that our construction is secure against all adaptive information-theoretic distinguishers that make roughly up to $2^{2n/3}$ construction and primitive queries. Moreover, we have also shown a matching attack with similar query complexity that establishes the tightness of our security bound.

Keywords – Public permutations, EDM, PDMMAC, Expectation Method.

4.1. Introduction

Most permutation-based cryptographic schemes generally provide a lower security bound with respect to the permutation state size. For example, most sponge-based modes provide $c/2$ bits of security (exceptions are [46, 58]), where $c < b$ is the capacity of the permutation, and b is its total state size. As the state size of a permutation is typically larger than the block size of a message (e.g. the state size of KECCAK is 1600 bits), the birthday bound is nevertheless adequate in practice. Birthday bound solutions are inadequate for, say lightweight permutations such as SPONGENT [40] and PHOTON [80], whose state sizes go as low as 88 and 100 bits, respectively. Thus, designing public permutation-based cryptographic schemes that provide beyond the birthday bound security with respect to the permutation state size can be highly interesting.

Chen et al. initiated research in this direction in [53], where they proposed two fixed input- and fixed output-length beyond the birthday bound secure PRFs based on public permutations – one in parallel mode and the other in sequential mode. They showed that the sum of two independent instances of the Even-Mansour [76] cipher in parallel mode, which they refer to as SoEM22 -

$$\mathsf{SoEM22}^{\pi_1,\pi_2}_{\mathbf{k}_1,\mathbf{k}_2}(x) := \pi_1(x \oplus \mathbf{k}_1) \oplus \pi_2(x \oplus \mathbf{k}_2) \oplus \mathbf{k}_1 \oplus \mathbf{k}_2,$$

provides a tight $2n/3$-bit security. This construction was extended by Bhattacharya et al. [28], where they showed beyond the birthday bound security of the domain-separated variant of SoEM22. They also proved that one cannot reduce the number of keys of SoEM22 without degrading the security bound to the birthday limit. Chen et al. also proposed a sequential-mode sum SoKAC21 -

$$\mathsf{SoKAC21}^{\pi_1,\pi_2}_{\mathbf{k}}(x) := \pi_2(\pi_1(x \oplus \mathbf{k}) \oplus \mathbf{k}) \oplus \pi_1(x \oplus \mathbf{k}) \oplus \mathbf{k},$$

which they proved to have a tight $2n/3$-bit security. Later in [118], Nandi exhibited a birthday bound attack on SoKAC21, hence falsifying the security claim of this construction. In [47], Chakraborti et al. proposed PDMMAC, a beyond the birthday bound secure single permutation-based fixed input- and fixed output-length PRF that operates in sequential mode. The design of PDMMAC gets its motivation from the *Decrypted Davis-Meyer* (DDM) construction,

$$\mathsf{DDM}(x) := \pi^{-1}(\pi(x) \oplus x).$$

PDMMAC requires an n-bit key $\mathbf{k}$ and an n-bit public permutation π to generate its output:

$$\mathsf{PDMMAC}^{\pi}_{\mathbf{k}}(x) := \pi^{-1}(\pi(x \oplus \mathbf{k}) \oplus (x \oplus 3\mathbf{k})) \oplus 2\mathbf{k}.$$

They extended the construction to a BBB secure single-permutation and single-keyed variant of the nonce-based MAC [1]. Although minimally structured, PDMMAC and its related MAC constructions (i.e. PDM*MAC [47] and 1K-PDM*MAC [47]) require invertibility of the permutation π (similar to the design of DWCDM [62]). However, the inverse call in PDMMAC somewhat brings down one of the advantages of using cryptographic permutations in a mode – efficiency of evaluating the permutation in the forward direction. In fact, designing a BBB secure single permutation-based PRF with two forward calls was stated as an open problem in [47]. Not only this, inverse-free designs have become an important design aspect of cryptography today as designs that rely solely on forward call(s) of permutation(s) create a very low footprint in a combined implementation of the mode [29]. Therefore, we do not as yet have any BBB secure single permutation-based fixed input- and fixed output-length PRF that operates in sequential mode with two forward calls [2].

4.1.1. Our Contribution

In this chapter, we propose pEDM, the first fixed input- and fixed output-length single permutation-based beyond the birthday bound secure PRF that operates in sequential mode without requiring an inverse call of the permutation. Our design is motivated by the EDM construction. In particular, pEDM with a $2n$-bit key and n-bit public permutation takes an n-bit input and returns an n-bit output as follows:

$$\mathsf{pEDM}^{\pi}_{\mathbf{k}_1,\mathbf{k}_2}(x) := \pi(\pi(x \oplus \mathbf{k}_1) \oplus (x \oplus \mathbf{k}_1) \oplus \mathbf{k}_2) \oplus \mathbf{k}_1.$$

We have shown that pEDM is secure against all adaptive information theoretic distinguishers that make roughly up to $2^{2n/3}$ construction and primitive queries. We also show a matching attack, thus establishing the tightness of this security bound. While we could directly realize a permutation-based PRF by instantiating the block cipher of the single-keyed variant of EDM with a 2-round Even-Mansour cipher, this would lead to four permutation calls with keys totalling $6n$ bits. Compared to such a straightforward solution, our construction altogether saves two permutation calls and $4n$ bits of the key. Although pEDM uses a single permutation call with no inverse functionality, the number of keys required is one more than the number of keys required in PDMMAC. Presently, we do not know whether our construction is prone

[1] A single permutation-based nonce-based MAC that does not require invertibility of the permutation was also proposed in [70]

[2] Chen et al. [53] showed an $n/2$-bit attack on SoKAC1; $\mathsf{SoKAC1}^{\pi}_{\mathbf{k}_1,\mathbf{k}_2}(x) = \pi(\pi(x \oplus \mathbf{k}_1) \oplus \mathbf{k}_2) \oplus \pi(x \oplus \mathbf{k}_1) \oplus \mathbf{k}_2 \oplus \mathbf{k}_1$. However, Chakraborti et al. [47] claimed that the attack was possibly wrong and showed a $2n/3$-bit attack on it. They also conjectured the tightness of this attack bound.

to the birthday attack with a single key. However, we believe it can be proven secure beyond the birthday bound with only an n-bit key. We show the PRF advantage of this construction through an extended distinguishing game and apply the expectation method to bound its distinguishing advantage. In table 4.1, we compare the structures of several public permutation-based PRFs with single-block input, single-block output- and multi-block input, multi-block output-designs.

Table 4.1.: Comparison table for permutation-based PRFs. n denotes the state size of the permutation and Inv indicates whether the construction requires an inverse call of the permutation. $s := n - \log(w+1)$, where $w \geq 1$ is the size of a chunk in a CENC-based construction. The last three constructions require a keyed hash function with at most ℓ blocks of input. The number of keys for these constructions includes the hash keys as well. All the constructions except CENCPP* and DS-CENCPP* require two permutation calls. Although SoKAC1 was shown to have a birthday bound attack and SoKAC21 was shown beyond the birthday bound secure in [53], Chakraborti et al. [47] believed that the birthday bound attack on SoKAC1 was possibly wrong and showed an attack on it with a $2^{2n/3}$-query complexity. Moreover, Nandi [118] has shown a birthday bound attack on SoKAC21.

Constructions	(perm, keys)	Inv	(i/p, o/p)	Sec
SoEM1 [53]	$(1,2)$	x	(n,n)	$\Theta(n/2)$
SoEM21 [53]	$(2,1)$	x	(n,n)	$\Theta(n/2)$
SoEM22 [53]	$(2,2)$	x	(n,n)	$\Theta(2n/3)$
SoKAC1 [53] (†)	$(1,2)$	x	(n,n)	$\Omega(2n/3)$
SoKAC21 [53]	$(2,1)$	x	(n,n)	$\Theta(n/2)$
PDMMAC [47]	$(1,1)$	✓	(n,n)	$\Theta(2n/3)$
DS-SoEM [28]	$(1,2)$	x	$(n-1,n)$	$\Theta(2n/3)$
pEDM [This Chapter]	$(1,2)$	x	(n,n)	$\Theta(2n/3)$
CENCPP* [28]	$(w+1,2)$	x	(n,wn)	$O(2n/3)$
DS-CENCPP* [28]	$(1,2)$	x	(s,wn)	$O(2n/3)$
nEHtM$_p$ [70]	$(1,2)$	x	$(n-1+\ell n,n)$	$\Theta(2n/3)$
PDM*MAC [47]	$(1,2)$	✓	$(n+\ell n,n)$	$\Theta(2n/3)$
1K-PDM*MAC [47]	$(1,1)$	✓	$(n+\ell n,n)$	$O(2n/3)$

We would like to mention here that DS-CENCPP* with $w = 1$ is a parallel construction with an $(n-1)$-bit input, and requires field multiplication with a primitive element to derive its $2n$-bit keys. However, our proposed construction is sequential with an n-bit input and does not require field multiplication to derive the keys. Although both of them have similar security bounds (i.e. $2^{2n/3}$), pEDM requires a lesser state size in hardware implementation as compared to the parallel construction DS-CENCPP* due to its sequential nature.

4.2. pEDM: Permutation-Based Encrypted Davis Meyer

In this section, we propose pEDM, the first permutation-based sequential beyond the birthday bound secure pseudorandom function with two forward permutation calls. Our construction is the permutation variant of the Encrypted Davis-Meyer (EDM) construction with two independent n-bit round keys $\mathbf{k}_1$ and $\mathbf{k}_2$. pEDM takes an n-bit input M masked with an n-bit round key $\mathbf{k}_1$ to generate the input of the first permutation call. The resulting permutation output is masked with $\mathbf{k}_2 \oplus M \oplus \mathbf{k}_1$ to generate the input for the second permutation call. The second permutation output is then masked with the round key $\mathbf{k}_1$ to generate the final output T. A schematic diagram of the construction is shown in Fig. 4.1.

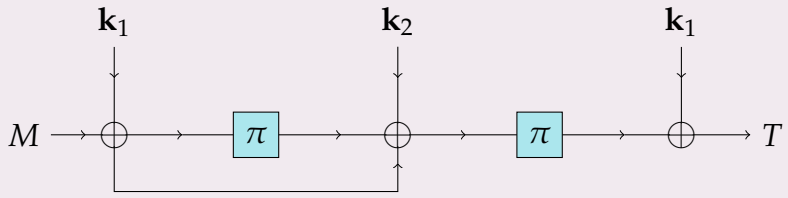


Figure 4.1.: The pEDM construction with independent keys $\mathbf{k}_1$ and $\mathbf{k}_2$, and an n-bit permutation π.

In the following, we prove that pEDM is $2n/3$-bit secure in the public permutation model, where n is the state size of the permutation.

4.2.1. Security of pEDM

We show that pEDM is secure against all adversaries that make roughly $2^{2n/3}$ construction and primitive queries in the random permutation model. The following result states the security of pEDM, the proof of which can be found in Sect. 4.3.

Theorem 9. *Let $\pi \xleftarrow{\$} \mathsf{Perm}$ be an n-bit public random permutation and let $\mathbf{k}_1, \mathbf{k}_2 \xleftarrow{\$} \{0,1\}^n$ be two independent n-bit keys. Then the PRF advantage for any (q,p)-distinguisher against the construction $\mathsf{pEDM}^{\pi}_{\mathbf{k}_1,\mathbf{k}_2}$ that makes at most q construction queries and p primitive queries is given by*

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{pEDM}}(q,p) \leq \frac{12q^2}{N^{4/3}} + \frac{2pq}{N^{4/3}} + \frac{15q}{N^{2/3}} + \frac{2\sqrt{q}}{N^{1/3}} + \frac{32pq^2}{N^2} + \frac{7qp^2}{N^2} + \frac{24q^3}{N^2} + \frac{2q^{3/2}}{N} + \frac{2p}{N^{4/3}} + \frac{3p\sqrt{nq}}{N} + \frac{2p\sqrt{q}}{N} + \frac{q\sqrt{p}}{N} + \frac{p^{3/2}}{N} + \frac{2}{N}.$$

Remark 2. *We would like to mention here that omitting the key $\mathbf{k}_1$ in the feed-forward connection of pEDM realizes the construction $\pi(\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_2) \oplus \mathbf{k}_1 = T$, which has a similar level of security. Thus, our proposed construction can be viewed as a 2-round key-alternating cipher based on the permutation-based Davis-Meyer construction along with an additional permutation, whereas the construction $\pi(\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_2) \oplus \mathbf{k}_1 = T$ can be viewed as the Even-Mansour cipher-based Davies-Meyer construction followed by a permutation. We believe both are similar in performance and security.*

4.2.2. Matching Attack on pEDM

In this section, we show a key-recovery attack on pEDM matching with Theorem 9, with a total of $q = 2^{2n/3+1}$ construction queries and $2p = 2^{2n/3+2}$ primitive queries (Fig. 4.2). The idea of the attack is to collect in a set $\mathcal{S}_{\mathbf{k}_1}$ for each key $\mathbf{k}_1$, a triplet of query indices $(i, a, b) \in [q] \times [p] \times [p]$ such that $(M_i \oplus \mathbf{k}_1 = \tilde{x}_a) \wedge (T_i \oplus \mathbf{k}_1 = \tilde{v}_b)$. $\mathbf{k}_1$ is considered a candidate guess-key if the number of triplets (i, a, b) in $\mathcal{S}_{\mathbf{k}_1}$ such that

$$\tilde{x}_a \oplus \tilde{y}_a \oplus \tilde{u}_b = \tilde{x}_{a'} \oplus \tilde{y}_{a'} \oplus \tilde{u}_{b'}$$

is at least 2. We show that the true key belongs to the set of candidate guess-keys with high probability and the size of the set of candidate keys is not very large.

Notation. For a tuple $(M_1, M_2, \ldots, M_s)$ of length s (each $M_i \in \{0,1\}^n$), $(M_1, M_2, \ldots, M_s) \xleftarrow{\text{wor}} \{0,1\}^n$ denotes $M_1 \xleftarrow{\$} \{0,1\}^n$ and $\forall\, i \geq 2$, $M_i \xleftarrow{\$} \{0,1\}^n \setminus \{M_1, \ldots, M_{i-1}\}$. Similarly, $(M_1, M_2, \ldots, M_s) \xleftarrow{\text{wr}} \{0,1\}^n$ denotes $M_1 \xleftarrow{\$} \{0,1\}^n$ and $\forall\, i \geq 2$, $M_i \xleftarrow{\$} \{0,1\}^n$ (independent of all $M_1, \ldots, M_{i-1}$).

Claim. *Let $(\mathbf{k}_1^*, \mathbf{k}_2^*)$ be the true key, i.e., the pair of keys used in the construction. Then*

$$\Pr[\mathbf{k}_1^* \in \mathcal{K}] \geq 0.687 \tag{4.1}$$

$$\Pr[|\mathcal{K} \setminus \{\mathbf{k}_1^*\}| \geq 128] \leq 0.5. \tag{4.2}$$

This claim shall be proved in the following section. Observe that the first equation states that the true key $\mathbf{k}_1^*$ belongs to the set of candidate keys with high probability, and the second equation states that the probability of the number of candidate keys being no less than 128 is at most $1/2$. Before proceeding with the analysis of the attack, we recall the Chernoff bound for the sum of independent Bernoulli trials:

Lemma 9. *Let $X_1, X_2, \ldots, X_n$ be independent random variables following the Bernoulli distribution such that X_i takes value 1 with probability p_i for each i. Let $X = X_1 + X_2 + \ldots + X_n$ and $\mu = \mathbf{E}[X]$. Then, for any $0 < \delta < 1$,*

$$\Pr[X \leq (1-\delta)\mu] \leq e^{-\mu\delta^2/2}.$$

Stage-I: Construction and Primitive Queries

1 : $M_i \in \{0,1\}^n \; \forall i \xleftarrow{\text{wor}} [2^{2n/3+1}],\; T_i \leftarrow \mathsf{pEDM}(M_i[1]) \forall i \in [2^{2n/3+1}].$

2 : $\tilde{x}_a \xleftarrow{\text{wor}} \{0,1\}^n \; \forall a \in [2^{2n/3+1}],\; \tilde{y}_a \leftarrow \pi_1(\tilde{x}_a) \; \forall a \in [2^{2n/3+1}].$

3 : $\mathcal{U}_1 \leftarrow \left\{ \tilde{x}_a \,|\, a \in [2^{2n/3+1}] \right\}.$

4 : $\tilde{u}_b \xleftarrow{\text{wor}} \{0,1\}^n \setminus \mathcal{U}_1 \; \forall b \in [2^{2n/3+1}],\; \tilde{v}_b \leftarrow \pi_1(\tilde{u}_b) \; \forall b \in [2^{2n/3+1}].$

5 : $\mathcal{U}_2 \leftarrow \left\{ \tilde{u}_b \,|\, b \in [2^{2n/3+1}] \right\}.$

Stage-II: Backward Equation Check

1 : $\forall \mathbf{k}_1 \in \{0,1\}^n,$
$\mathcal{S}_{\mathbf{k}_1} \leftarrow \left\{ (i,a,b) \in [2^{2n/3}+1]^3 \,:\, M_i \oplus \tilde{x}_a = \mathbf{k}_1 = T_i \oplus \tilde{v}_b \right\}.$

Stage-II: Constructing the Guess Key Set

1 : $\mathcal{K} \leftarrow \phi.$

2 : $\forall \mathbf{k}_1 \in \{0,1\}^n$ such that $|\mathcal{S}_{\mathbf{k}_1}| \geq 2,$
`if` $\tilde{x}_a \oplus v_a \oplus \tilde{u}_b \oplus \tilde{x}_{a'} \oplus \tilde{y}_{a'} \oplus \tilde{u}_{b'} = 0 \; \forall (i,a,b) \neq (i',a',b') \in \mathcal{S}_{\mathbf{k}_1},$
`then` $\mathcal{K} \leftarrow \mathcal{K} \cup \{\mathbf{k}_1\}.$

3 : `compute` $\mathbf{k}_2 \leftarrow M_i \oplus \mathbf{k}_1 \oplus \tilde{y}_a \oplus \tilde{u}_b, \; \forall \mathbf{k}_1 \in \mathcal{K}.$

Figure 4.2.: An attack on pEDM, where a computationally unbounded adversary makes $\mathcal{O}\left(2^{2n/3}\right)$ queries to the construction and primitives.

4.2.3. Analysis of the Key-Recovery Advantage

In this section, we prove the claim in Sect. 4.2.2.

Step I: The true key belongs to the set of candidate keys. According to step 2 of Stage-III of the algorithm, an element $\mathbf{k}_1$ belongs to the set $\mathcal{K}$ if the following two conditions hold:

(a) $|\mathcal{S}_{\mathbf{k}_1}| \geq 2$ and
(b) $\tilde{x}_a \oplus \tilde{y}_a \oplus \tilde{u}_b = \tilde{x}_{a'} \oplus \tilde{y}_{a'} \oplus \tilde{u}_{b'}, \; \forall \; (i,a,b),(i',a',b') \in \mathcal{S}_{\mathbf{k}_1},$

(where $\mathcal{S}_{\mathbf{k}_1}$ is the set of all triplets $(i,a,b) \in ([2^{2n/3+1}])^3$ as defined in Stage-II of the algorithm, i.e. Fig 4.2) such that

$$\begin{aligned} \mathbf{k}_1 &= M_i \oplus \tilde{x}_a \text{ and} \\ \mathbf{k}_1 &= T_i \oplus \tilde{v}_b. \end{aligned} \tag{4.3}$$

For the true key $(\mathbf{k}_1^*, \mathbf{k}_2^*)$, let $z_i := \pi(M_i \oplus \mathbf{k}_1^*) \oplus (M_i \oplus \mathbf{k}_1^* \oplus \mathbf{k}_2^*)$. Note that all the random variables M_i as well as $M_i \oplus \mathbf{k}_1^*$ are sampled without replacement. $\pi(M_i \oplus \mathbf{k}_1^*)$ are also sampled without replacement, independent of the variables $M_i \oplus \mathbf{k}_1^*$. Therefore, due to the result of [71, 60] on the *sum of two independent permutations*, all z_i follow the uniform distribution.

Observe next that for the first part $\mathbf{k}_1^*$ of the true key pair $(\mathbf{k}_1^*, \mathbf{k}_2^*)$, if Eqn. (4.3) holds when the value $\mathbf{k}_1$ is replaced by the true key $\mathbf{k}_1^*$ for some $(i, a, b) \in (\left[2^{2n/3+1}\right])^3$, then the second part $\mathbf{k}_2^*$ can be computed as $\mathbf{k}_2^* = \tilde{x}_a \oplus \tilde{y}_a \oplus \tilde{u}_b$. Hence, the relation

$$\tilde{x}_a \oplus \tilde{y}_a \oplus \tilde{u}_b = \tilde{x}_{a'} \oplus \tilde{y}_{a'} \oplus \tilde{u}_{b'}$$

is automatically satisfied for the true key $\mathbf{k}_1^*$ and $(i, a, b), (i', a', b') \in \mathcal{S}_{\mathbf{k}_1^*}$. Therefore, to bound Eqn. (4.1), it suffices to bound the probability of existence of at least two distinct tuples $(i, a, b), (i', a', b')$ such that

$$\begin{aligned} \mathbf{k}_1^* &= M_i \oplus \tilde{x}_a = T_i \oplus \tilde{v}_b \\ \mathbf{k}_1^* &= M_{i'} \oplus \tilde{x}_{a'} = T_{i'} \oplus \tilde{v}_{b'}. \end{aligned} \tag{4.4}$$

Again, for the first part $\mathbf{k}_1^*$ of the true key pair $(\mathbf{k}_1^*, \mathbf{k}_2^*)$, if the following equations are satisfied

$$\begin{aligned} \mathbf{k}_1^* &= M_i \oplus \tilde{x}_a \\ \mathbf{k}_1^* &= M_{i'} \oplus \tilde{x}_{a'} \\ z_i &= \tilde{u}_b \\ z_{i'} &= \tilde{u}_{b'} \end{aligned} \tag{4.5}$$

for some $(i, a, b), (i', a', b') \in (\left[2^{2n/3+1}\right])^3$, then it satisfies (4.4). As a result, it is enough to bound the probability that there exist at least two distinct tuples $(i, a, b), (i', a', b')$ such that Eqn. (4.5) is satisfied. We bound this probability in two stages. In the first stage, we bound the number of i such that $z_i \in \mathcal{U}_2$ and we store such i in list $\mathcal{L}$. Let $\mathcal{L}_M$ be the set of all $M_i \oplus \mathbf{k}_1^*$ such that $i \in \mathcal{L}$. In the second stage, we obtain a lower bound for the probability of the number of a such that $\tilde{x}_a \in \mathcal{L}_M$ being at least 2.

Stage I. Let $\mathbb{1}_i$ be the indicator random variable that takes value 1 if and only if $z_i \in \mathcal{U}_2$. It is easy to see that $\mathbb{1}_i$ are independent Bernoulli random variables with success probability $\frac{2}{2^{n/3}}$. Let $Z = (\mathbb{1}_1 + \ldots + \mathbb{1}_{2^{2n/3+1}})$. Then $Z \sim \mathsf{Bin}(2^{2n/3+1}, \frac{2}{2^{n/3}})$ and therefore, $\mathbf{E}[Z] = 4 \cdot 2^{n/3}$. Applying the Chernoff bound as stated in Lemma 9 with $\delta = 1/2$, we get

$$\Pr[Z > 2^{n/3+1}] \geq 1 - \frac{1}{e^{2^{n/3-1}}}. \tag{4.6}$$

It is therefore evident that the event bounding the size of $\mathcal{L}$ and in turn, the size of $\mathcal{L}_M$ by $2^{n/3} + 1$, holds with high probability.

Stage II. In order to bound the probability of existence of at least two distinct tuples $(i,a,b),(i',a',b)$ such that (4.4) holds, we bound the following:

$$\Pr[|a : \tilde{x}_a \in \mathcal{L}_M| \geq 2]. \tag{4.7}$$

Observe that Eqn. (4.7) is equivalent to

$$1 - \left(\Pr\left[\tilde{x}_a \notin \mathcal{L}_M, \forall a \in \left[2^{2n/3+1}\right]\right] + \sum_{a=1}^{2^{2n/3+1}} \Pr\left[\tilde{x}_a \in \mathcal{L}_M \wedge \tilde{x}_b \notin \mathcal{L}_M, \forall b \neq a\right] \right). \tag{4.8}$$

Since $\tilde{x}_1, \ldots, \tilde{x}_{2^{2n/3+1}}$ are sampled without replacement from $\{0,1\}^n$ and $|\mathcal{L}_M| = 2^{n/3+1}$,

$$\begin{aligned} \Pr\left[\tilde{x}_a \notin \mathcal{L}_M, \forall a \in \left[2^{2n/3+1}\right]\right] &= \Pr\left[\tilde{x}_1, \ldots, \tilde{x}_{2^{2n/3}+1} \notin \mathcal{L}_M\right] \\ &\leq \frac{(2^n - 2^{n/3+1})_{2^{2n/3+1}}}{(2^n)_{2^{2n/3}+1}} \\ &\leq \left(1 - \frac{2}{2^{2n/3}}\right)^{2 \cdot 2^{2n/3}} \leq \frac{1}{e^4}. \end{aligned} \tag{4.9}$$

Similarly,

$$\begin{aligned} & \sum_{a=1}^{2^{2n/3+1}} \Pr\left[\tilde{x}_a \in \mathcal{L}_M \wedge \tilde{x}_b \notin \mathcal{L}_M, \forall b \neq a\right] \\ = & \sum_{a=1}^{2^{2n/3+1}} \Pr\left[\tilde{x}_a \in \mathcal{L}_M\right] \cdot \Pr\left[\tilde{x}_b \notin \mathcal{L}_M, \forall b \neq a\right] \\ \leq & \sum_{a=1}^{2^{2n/3+1}} \frac{2^{n/3+1}}{(2^n - 2^{2n/3+1} + 1)} \cdot \left(1 - \frac{2}{2^{2n/3}}\right)^{2 \cdot 2^{2n/3} - 1} \\ \leq & \; 8 \left(1 - \frac{2}{2^{2n/3}}\right)^{-1} \cdot \left(1 - \frac{2}{2^{2n/3}}\right)^{2 \cdot 2^{2n/3}} \\ & \leq \frac{16}{e^4}, \end{aligned} \tag{4.10}$$

where the last inequality follows due to $\frac{1}{\left(1 - \frac{2}{2^{2n/3}}\right)} \leq 2$ as $n \geq 3$. Therefore, from Eqn.s (4.8), (4.9) and (4.10), plugging in $e \leq 3$ gives

$$\Pr\left[\mathbf{k}_1^* \in \mathcal{K}\right] \geq 1 - \frac{17}{e^4}. \tag{4.11}$$

Step II: Bounding the size of $\mathcal{K} \setminus \{\mathbf{k}_1^*\}$. We use Markov's inequality to find an upper bound for the probability that $|\mathcal{K} \setminus \{\mathbf{k}_1^*\}| \geq 128$ holds:

$$\Pr\left[|\mathcal{K} \setminus \{\mathbf{k}_1^*\}| \geq 128\right] \leq \frac{\mathbf{E}\left[|\mathcal{K} \setminus \{\mathbf{k}_1^*\}|\right]}{128}. \tag{4.12}$$

Therefore, it is sufficient to bound the expected size of the set of candidate keys $\mathcal{K} \setminus \{\mathbf{k}_1^*\}$. For each $\mathbf{k}_1 \in \{0,1\}^n$, let $\mathbb{1}_{\mathbf{k}_1}$ be the indicator random variable that takes value 1 if and only if there exist $(i,a,b),(i',a',b')$ such that

$$\begin{aligned} \tilde{x}_a \oplus \tilde{y}_a \oplus \tilde{u}_b &= \tilde{x}_{a'} \oplus \tilde{y}_{a'} \oplus \tilde{u}_{b'} \\ \mathbf{k}_1 &= M_i \oplus \tilde{x}_a \\ \mathbf{k}_1 &= T_i \oplus \tilde{v}_b \\ \mathbf{k}_1 &= M_{i'} \oplus \tilde{x}_{a'} \\ \mathbf{k}_1 &= T_{i'} \oplus \tilde{v}_{b'}. \end{aligned} \tag{4.13}$$

It is clear from the linearity of expectation that

$$\sum_{\mathbf{k}_1 \in \{0,1\}^n \setminus \{\mathbf{k}_1^*\}} \mathbb{1}_{\mathbf{k}_1} = |\mathcal{K} \setminus \{\mathbf{k}_1^*\}| \Rightarrow \mathbf{E}[|\mathcal{K} \setminus \{\mathbf{k}_1^*\}|] = \sum_{\mathbf{k}_1 \in \{0,1\}^n \setminus \{\mathbf{k}_1^*\}} \Pr\left[\mathbb{1}_{\mathbf{k}_1} = 1\right]. \tag{4.14}$$

This reduces the problem to bounding the probability of $\mathbb{1}_{\mathbf{k}_1}$ taking value 1. For a fixed choice of indices (i,a,b) and (i',a',b'), the above system of equations holds with probability at most 2^{-5n} as all the random variables are independent from each other. The number of choices of such indices is at most $(2^{n/3+1})^6$. Therefore,

$$\Pr\left[\mathbb{1}_{\mathbf{k}_1} = 1\right] \leq \frac{64}{2^n}. \tag{4.15}$$

Eqn.s (4.14) and (4.15) bound the expected size of the set of candidate keys by at most 8. Setting this value in Eqn. (4.12), thus gives

$$\Pr\left[|\mathcal{K} \setminus \{\mathbf{k}_1^*\}| \geq 128\right] \leq 1/2,$$

which concludes the proof of the claim. □

The distinguisher in the attack in Fig. 4.2 is information theoretically bounded and its run time [3] is more than 2^n. In particular, for each key $\mathbf{k}_1$, the number of steps required to populate the set $\mathcal{S}_{\mathbf{k}_1}$ is roughly 2^{2n}. Therefore, Stage-II of the algorithm takes at most 2^{4n} operations. In step 2 of Stage-III, the algorithm requires at least one checking operation for each set $\mathcal{S}_{\mathbf{k}_1}$, and therefore requires about 2^n operations. This adds up to give an overall time complexity of $O(2^{4n})$ for the algorithm; the number of construction queries is $2^{2n/3+1}$ and the total number of primitive queries is $2^{2n/3+2}$.

[3] The time complexity of this adversary does not account for the number of times the adversary makes offline primitive queries; it accounts solely for the time required to compute local operations.

4.3. Proof of Theorem 9

Let $\mathbf{k} = (\mathbf{k}_1, \mathbf{k}_2) \in \{0,1\}^{2n}$ be a pair of n-bit keys. We consider any information theoretic deterministic distinguisher D that interacts with the following oracles: $(\mathsf{pEDM}^{\pi}_{\mathbf{k}}, \pi)$ in the real world and (RF, π) in the ideal world, where RF is the random function over $\{0,1\}^n$ to $\{0,1\}^n$. We call the first oracle a *construction oracle* and the second a *primitive oracle*. Queries to the construction oracle are called *construction queries* and to the primitive oracle, *primitive queries*. A transcript $\tau_c = \{(M_1, T_1), \ldots, (M_q, T_q)\}$ summarizes the construction queries and a transcript $\tau_p = \{(\tilde{u}_1, \tilde{v}_1), \ldots, (\tilde{u}_p, \tilde{v}_p)\}$, the primitives queries, with the assumption that D makes a total of q construction and p primitive queries. Primitive queries can either be forward queries $\tilde{u}$ to the primitive π resulting in responses $\tilde{v}$, or inverse queries $\tilde{v}$ to π^{-1} resulting in responses $\tilde{u}$. Since we assume D to never make redundant queries, none of the transcripts contain any duplicate elements.

We modify the experiment by releasing internal values to D after it has finished its interaction with one of the oracles but has not yet output its decision bit. We also reveal the key $\mathbf{k}$ used in the construction in the real world, and a pair of n-bit dummy keys $\mathbf{k} = (\mathbf{k}_1, \mathbf{k}_2)$ sampled uniformly at random from the keyspace $\{0,1\}^n$ in the ideal world, to the distinguisher. Thus, the complete transcript is $\tau = (\tau_c, \tau_p, \mathbf{k})$. Since the modified experiment only makes the distinguisher more powerful, the distinguishing advantage of D in this experiment is no less than its distinguishing advantage in the former one. Let X_{re} denote the random variable that takes as its value, a transcript τ realized in the real world, and X_{id} the random variable that takes as its value, a transcript τ realized in the ideal world. The probability of realizing $\tau = (\tau_c, \tau_p, \mathbf{k})$ in the ideal (resp. real) world is called *ideal* (resp. *real*) *interpolation probability*. A transcript τ is *attainable* by D if its ideal interpolation probability is non-zero. Let Θ denote the set of all attainable transcripts and $\phi : \Theta \to [0, \infty)$ be a non-negative function that maps any attainable transcript to a non-negative real value. Following these notations and using the expectation method (1.2), we shall prove the following result:

Lemma 10. *Let* $\tau = (\tau_c, \tau_p, \mathbf{k}) \in \Theta$ *be an attainable transcript. Let* $p(\tau) := \Pr[\pi \xleftarrow{\$} \mathsf{Perm}(n) : \mathsf{pEDM}^{\pi}_{\mathbf{k}} \longrightarrow \tau_c \mid \pi \longrightarrow \tau_p]$. *Then*

$$\rho(\tau) := \frac{\pi_{\mathrm{re}}(\tau)}{\pi_{\mathrm{id}}(\tau)} = p(\tau) \cdot 2^{nq}.$$

Recall that $\mathsf{pEDM}^{\pi}_{\mathbf{k}} \longrightarrow \tau_c$ denotes $\mathsf{pEDM}^{\pi}_{\mathbf{k}}(M_i) = T_i$ for all $(M_i, T_i) \in \tau_c$, i.e. $\pi(\pi(M_i \oplus \mathbf{k}_1) \oplus M_i \oplus \mathbf{k}_1 \oplus \mathbf{k}_2) \oplus \mathbf{k}_1 = T_i$, where $\mathbf{k} = (\mathbf{k}_1, \mathbf{k}_2)$. The proof of this lemma is hence trivial, the ideal interpolation probability for a good transcript being $\frac{1}{(2^n)_p 2^{nq}}$ (since the random function RF always outputs a uniformly random n-bit string on each input query).

4.3.1. Definition and Probability of Bad Transcripts

In this section, we define and bound the probability of bad transcripts in the ideal world. For a transcript $\tau = (\tau_c, \tau_p, \mathbf{k}_1, \mathbf{k}_2)$,

$$
\begin{aligned}
\mathcal{U} &:= \left\{\tilde{u} \in \{0,1\}^n \,:\, (\tilde{u}, \tilde{v}) \in \tau_p\right\}, \\
\mathcal{V} &:= \left\{\tilde{v} \in \{0,1\}^n \,:\, (\tilde{u}, \tilde{v}) \in \tau_p\right\}, \\
\alpha &:= \left|\left\{(M,T) \in \tau_c \,:\, M \oplus \mathbf{k}_1 \in \mathcal{U}\right\}\right|, \\
\beta &:= \left|\left\{(M,T) \in \tau_c \,:\, T \oplus \mathbf{k}_1 \in \mathcal{V}\right\}\right|, \\
\mathsf{C} &:= \left|\left\{(M,T),(M',T') \in \tau_c \,:\, T = T'\right\}\right|, \\
\sigma &:= \left|\left\{(M,T),(M',T'),(M'',T'') \in \tau_c \,:\, T \oplus M' \oplus \mathbf{k}_2 = M'' \oplus \mathbf{k}_1\right\}\right|, \\
\theta &:= \left|\left\{(\tilde{u},\tilde{v}),(\tilde{u}',\tilde{v}') \in \tau_p \,:\, \tilde{u} \oplus \tilde{v} = \tilde{u}' \oplus \tilde{v}'\right\}\right|.
\end{aligned}
$$

We say that a construction query $(M,T) \in \tau_c$ is *non-colliding* if $\forall (M',T') \in \tau_c, T \neq T'$. The set of bad transcripts is characterized by identifying two-fold collisions, as depicted in Fig. 4.3:

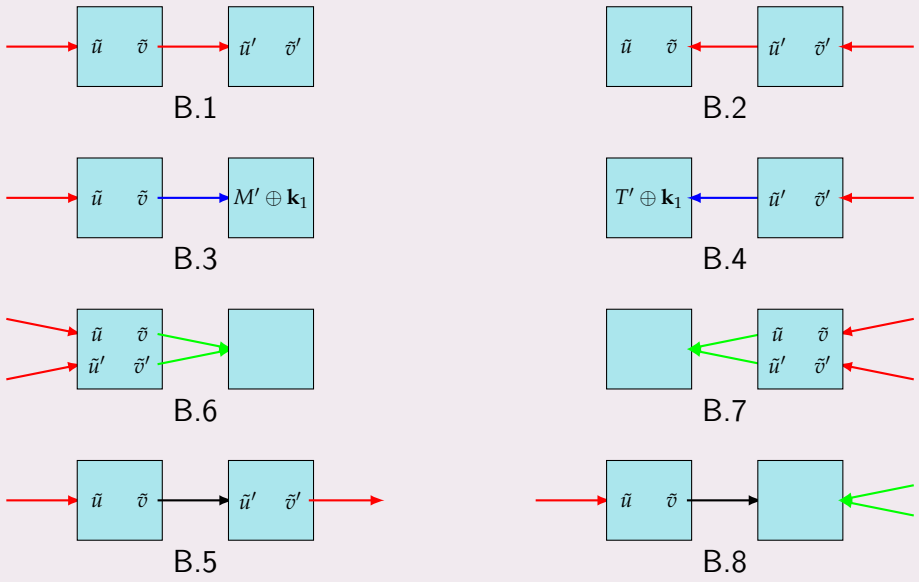


Figure 4.3.: Different cases of two-fold collisions. A red edge denotes a construction input/output collision with a primitive input/output, a blue edge denotes a primitive input colliding with the input or output of some construction query, and a green edge denotes a collision amongst similar types.

Definition 3. *An attainable transcript* $\tau = (\tau_c, \tau_p, \mathbf{k})$ *is called a* bad *transcript if any one of the following holds:*

1. Inputs to (resp. outputs of) the two consecutive permutation calls within one or two construction queries are not fresh:
   - B.1: $\exists\ (M,T) \in \tau_c, (\tilde{u},\tilde{v}), (\tilde{u}',\tilde{v}') \in \tau_p$ *such that* $M \oplus \mathbf{k}_1 = \tilde{u}, \tilde{v} \oplus \tilde{u} \oplus \mathbf{k}_2 = \tilde{u}'$.
   - B.2: $\exists\ (M,T) \in \tau_c, (\tilde{u},\tilde{v}), (\tilde{u}',\tilde{v}') \in \tau_p$ *such that* $T \oplus \mathbf{k}_1 = \tilde{v}, \tilde{u} \oplus (M \oplus \mathbf{k}_1) \oplus \mathbf{k}_2 = \tilde{v}'$.
   - B.3: $\exists\ (M,T), (M',T') \in \tau_c, (\tilde{u},\tilde{v}) \in \tau_p$ *such that* $M \oplus \mathbf{k}_1 = \tilde{u}, \tilde{v} \oplus \tilde{u} \oplus \mathbf{k}_2 = M' \oplus \mathbf{k}_1$.
   - B.4: $\exists\ (M,T), (M',T') \in \tau_c, (\tilde{u},\tilde{v}) \in \tau_p$ *such that* $T \oplus \mathbf{k}_1 = \tilde{v}, \tilde{u} \oplus (M \oplus \mathbf{k}_1) \oplus \mathbf{k}_2 = T' \oplus \mathbf{k}_1$.
2. The input and output of a construction query are both not fresh:
   - B.5: $\exists\ (M,T) \in \tau_c, (\tilde{u},\tilde{v}), (\tilde{u}',\tilde{v}') \in \tau_p$ *such that* $M \oplus \mathbf{k}_1 = \tilde{u}, T \oplus \mathbf{k}_1 = \tilde{v}'$.
3. Inputs to (resp. outputs of) the first (resp. second) permutation call of two construction queries collide with the inputs (resp. outputs) of two primitive queries, and the inputs to (resp. outputs of) the second (resp. from first) permutation call for these two construction queries also collide:
   - B.6: $\exists\ (M,T), (M',T') \in \tau_c, (\tilde{u},\tilde{v}), (\tilde{u}',\tilde{v}') \in \tau_p$ *such that* $M \oplus \mathbf{k}_1 = \tilde{u}, M' \oplus \mathbf{k}_1 = \tilde{u}', \tilde{u} \oplus \tilde{v} = \tilde{u}' \oplus \tilde{v}'$.
   - B.7: $\exists\ (M,T), (M',T') \in \tau_c, (\tilde{u},\tilde{v}), (\tilde{u}',\tilde{v}') \in \tau_p$ *such that* $T \oplus \mathbf{k}_1 = \tilde{v}, T' \oplus \mathbf{k}_1 = \tilde{v}', \tilde{u} \oplus M = \tilde{u}' \oplus M'$.
4. Additional bad events:
   - B.8: $\exists\ (M,T), (M',T') \in \tau_c, (\tilde{u},\tilde{v}) \in \tau_p$ *such that* $M \oplus \mathbf{k}_1 = \tilde{u}, T = T'$.
   - B.9: $\sigma \geq q^2/2^{n/3}$.
   - B.10: $\mathsf{C} \geq q/2^{n/3}$.
   - B.11: $\alpha \geq \sqrt{q}$.
   - B.12: $\beta \geq \sqrt{q}$.
   - B.13: $\theta \geq \sqrt{p}$.

Recall that $\mathsf{BadT} \subseteq \Theta$ is the set of all attainable bad transcripts and $\mathsf{GoodT} = \Theta \setminus \mathsf{BadT}$ is the set of all attainable good transcripts. We bound the probability of bad transcripts in the ideal world as follows:

Lemma 11. *Let* $\tau = (\tau_c, \tau_p, \mathbf{k})$ *be an attainable transcript. Then*

$$\Pr[\mathsf{X}_{\mathrm{id}} \in \mathsf{BadT}] \leq \frac{3qp^2}{2^{2n}} + \frac{4pq^2}{2^{2n}} + \frac{3p\sqrt{nq}}{2^n} + \frac{2p\sqrt{q}}{2^n} + \frac{q\sqrt{p}}{2^n} + \frac{p^{3/2}}{2^n} + \frac{2q}{2^{2n/3}} + \frac{2}{2^n}.$$

Proof. Let $\tau = (\tau_c, \tau_p, \mathbf{k}_1, \mathbf{k}_2)$ be any attainable transcript. As $\mathbf{k}_1$ and $\mathbf{k}_2$ are sampled uniformly and independently from the keyspace in the ideal world, the union bound gives

$$\Pr[\mathsf{X}_{\mathrm{id}} \in \mathsf{BadT}] \leq \Pr[\mathsf{B}.7 \vee \mathsf{B}.13] + \sum_{\substack{1 \leq i \leq 13 \\ i \neq 7,13}} \Pr[\mathsf{B}.i]. \tag{4.16}$$

We bound the probabilities of all bad events individually, adding which, we obtain the lemma.

Bounding B.1. For fixed values $(M, T) \in \tau_c$ and $(\tilde{u}, \tilde{v}), (\tilde{u}', \tilde{v}') \in \tau_p$,

$$\Pr\left[\mathbf{k}_1 = M \oplus \tilde{u}, \mathbf{k}_2 = \tilde{u}' \oplus \tilde{v} \oplus \tilde{u}\right] = \frac{1}{2^{2n}}$$

due to randomness of the keys $\mathbf{k}_1$ and $\mathbf{k}_2$. Summing over all possible choices of these values gives

$$\Pr[\mathsf{B}.1] \leq \frac{qp^2}{2^{2n}}. \tag{4.17}$$

Bounding B.2. For fixed values $(M, T) \in \tau_c$ and $(\tilde{u}, \tilde{v}), (\tilde{u}', \tilde{v}') \in \tau_p$,

$$\Pr\left[\mathbf{k}_1 = T \oplus \tilde{v}, \mathbf{k}_2 = \tilde{v}' \oplus \tilde{u} \oplus (M \oplus \mathbf{k}_1)\right] = \frac{1}{2^{2n}}$$

by randomness of $\mathbf{k}_1$ and $\mathbf{k}_2$. Summing over all possible choices of these values gives

$$\Pr[\mathsf{B}.2] \leq \frac{qp^2}{2^{2n}}. \tag{4.18}$$

Bounding B.3. Fixing $(M, T), (M', T') \in \tau_c$ and $(\tilde{u}, \tilde{v}) \in \tau_p$,

$$\Pr\left[\mathbf{k}_1 = \tilde{u} \oplus M, \mathbf{k}_2 = \tilde{v} \oplus (M' \oplus \mathbf{k}_1) \oplus \tilde{u}\right] = \frac{1}{2^{2n}}$$

by randomness of $\mathbf{k}_1$ and $\mathbf{k}_2$. Summing over all possible choices of these values gives

$$\Pr[\mathsf{B}.3] \leq \frac{pq^2}{2^{2n}}. \tag{4.19}$$

Bounding B.4. Similar to B.3,

$$\Pr[\mathsf{B}.4] \leq \frac{pq^2}{2^{2n}}. \tag{4.20}$$

Bounding B.5. Consider the set

$$\mathsf{BadK}_1 := \{\mathbf{k}_1 \in \{0,1\}^n : \exists (M, T) \in \tau_c, (\tilde{u}, \tilde{v}), (\tilde{u}', \tilde{v}') \in \tau_p \text{ such that } \mathbf{k}_1 = M \oplus \tilde{u} = T \oplus \tilde{v}'\}.$$

Therefore, for any $\Delta > 0$,

$$\begin{aligned}\Pr[\mathsf{B}.5] &= \Pr[\mathbf{k}_1 \in \mathsf{BadK}_1] \\ &= \Pr\left[(\mathbf{k}_1 \in \mathsf{BadK}_1) \wedge (|\mathsf{BadK}_1| \geq \Delta)\right] \\ &\quad + \Pr\left[(\mathbf{k}_1 \in \mathsf{BadK}_1) \wedge (|\mathsf{BadK}_1| < \Delta)\right] \\ &\leq \Pr[|\mathsf{BadK}_1| \geq \Delta]] + \frac{\Delta}{2^n}. \end{aligned} \tag{4.21}$$

Clearly, $|\mathsf{BadK}_1| \leq \mathcal{Z}$, where $\mathcal{Z} := |\{((M,T),\tilde{u},\tilde{v}') \in \tau_c \times \mathcal{U} \times \mathcal{V} : M \oplus \tilde{u} = T \oplus \tilde{v}'\}|$. By Lemma 21,

$$\Pr[|\mathcal{Z}| \geq qp^2/2^n + 3p\sqrt{nq}] \leq 2/2^n.$$

Therefore, setting $\Delta = qp^2/2^n + 3p\sqrt{nq}$ in Eqn. (4.21) gives

$$\Pr[\mathsf{B}.5] \leq \frac{qp^2}{2^{2n}} + \frac{3p\sqrt{nq}}{2^n} + \frac{2}{2^n}. \tag{4.22}$$

Bounding $\mathsf{B}.6 \vee \mathsf{B}.13$. Note that

$$\Pr[\mathsf{B}.6 \vee \mathsf{B}.13] \leq \Pr[\mathsf{B}.13] + \Pr[\mathsf{B}.6 \wedge \overline{\mathsf{B}.13}]. \tag{4.23}$$

To bound the probability of occurrence of B.13, we define an indicator random variable $\mathbb{1}_{ab}$ that takes value 1 if and only if $\exists\ (\tilde{u}_a,\tilde{v}_a),(\tilde{u}_b,\tilde{v}_b) \in \tau_p$ such that $\tilde{u}_a \oplus \tilde{v}_a = \tilde{u}_b \oplus \tilde{v}_b$. For fixed values of a, b, $\Pr[\mathbb{1}_{ab} = 1] = 1/2^n$ as either both $(\tilde{u}_a,\tilde{v}_a),(\tilde{u}_b,\tilde{v}_b)$ are backward queries with random values for $\tilde{u}_a,\tilde{u}_b$, or at least one of them (say $(\tilde{u}_b,\tilde{v}_b)$) is a forward query, with a random value for $\tilde{v}_b$. Hence, by the linearity of expectation,

$$\mathbf{E}[\theta] = \sum_{ab} \mathbf{E}[\mathbb{1}_{ab}] = \sum_{ab} \Pr[\mathbb{1}_{ab} = 1] \leq \frac{p^2}{2^n}. \tag{4.24}$$

Therefore, using Markov's inequality, we have

$$\Pr[\mathsf{B}.13] = \Pr[\theta \geq \sqrt{p}] \leq \frac{\mathbf{E}[\theta]}{\sqrt{p}} \leq \frac{p^{3/2}}{2^n}, \tag{4.25}$$

by Eqn. (4.24). Next, in order to bound the probability of occurrence of the event $\mathsf{B}.6 \wedge \overline{\mathsf{B}.13}$, we fix $(M,T),(M',T') \in \tau_c$ and $(\tilde{u},\tilde{v}),(\tilde{u}',\tilde{v}') \in \tau_p$. Then the probability of the event $(M \oplus \mathbf{k}_1 = \tilde{u}) \wedge (M' \oplus \mathbf{k}_1 = \tilde{u}') \wedge (\tilde{u} \oplus \tilde{v} = \tilde{u}' \oplus \tilde{v}')$ is $1/2^n$ due to the randomness of $\mathbf{k}_1$. This probability is well defined as $\tilde{u} \neq \tilde{u}'$.

Observe that the number of pairs $(\tilde{u},\tilde{v}),(\tilde{u}',\tilde{v}') \in \tau_p$ that satisfy this event is at most $\sqrt{p}$. Furthermore, the number of choices for $(M,T) \in \tau_c$ is q,

which restricts the number of choices for $(M', T') \in \tau_c$ to at most 1 (since choosing an (M, T) determines (M', T') as $M' = \tilde{u}' \oplus \tilde{u} \oplus M$). Hence,

$$\Pr[\mathsf{B.6} \wedge \overline{\mathsf{B.13}}] \leq \frac{q\sqrt{p}}{2^n}. \tag{4.26}$$

Combining Eqn.s (4.23), (4.25) and (4.26), we get

$$\Pr[\mathsf{B.6} \vee \mathsf{B.13}] \leq \frac{q\sqrt{p}}{2^n} + \frac{p^{3/2}}{2^n}. \tag{4.27}$$

Bounding B.7. For fixed pairs $(M, T), (M', T') \in \tau_c$ and $(\tilde{u}, \tilde{v}), (\tilde{u}', \tilde{v}') \in \tau_p$, this event occurs with probability $1/2^{2n}$ due to independence of T and T'. As the number of choices for $(M, T), (M', T')$ is at most q^2 and the number of choices for $(\tilde{u}, \tilde{v})$ is at most p, the number of choices for $(\tilde{u}', \tilde{v}')$ are restricted to at most 1. Varying over all possible choices of $(M, T), (M', T') \in \tau_c$ and $(\tilde{u}, \tilde{v}), (\tilde{u}', \tilde{v}') \in \tau_p$, we have

$$\Pr[\mathsf{B.7}] \leq \frac{pq^2}{2^{2n}}. \tag{4.28}$$

Bounding B.8. Fix $(M, T), (M', T') \in \tau_c$ and $(\tilde{u}, \tilde{v}) \in \tau_p$. Then

$$\Pr\left[\mathbf{k}_1 = \tilde{u} \oplus M, T = T'\right] = \frac{1}{2^{2n}}.$$

Summing over all possible choices of $(M, T), (M', T') \in \tau_c, (\tilde{u}, \tilde{v}) \in \tau_p$, we have

$$\Pr[\mathsf{B.8}] \leq \frac{pq^2}{2^{2n}}. \tag{4.29}$$

Bounding B.9. Consider an indicator random variable $\mathbb{1}_{i_1 i_2 i_3}$, which takes value 1 if and only if $(M_{i_1}, T_{i_1}), (M_{i_2}, T_{i_2}), (M_{i_3}, T_{i_3}) \in \tau_c$ such that $T_{i_1} \oplus M_{i_2} \oplus \mathbf{k}_2 = M_{i_3} \oplus \mathbf{k}_1$. Since $\Pr[\mathbb{1}_{i_1 i_2 i_3} = 1] = 1/2^n$ for fixed values of i_1, i_2 and i_3 (by the randomness of $\mathbf{k}_1$),

$$\mathbf{E}[\sigma] \;=\; \sum_{i_1 i_2 i_3} \mathbf{E}[\mathbb{1}_{i_1 i_2 i_3}] = \sum_{i_1 i_2 i_3} \Pr[\mathbb{1}_{i_1 i_2 i_3} = 1] \leq \frac{q^3}{2^n}. \tag{4.30}$$

Therefore, by Markov's inequality,

$$\Pr[\mathsf{B.9}] = \Pr[\sigma \geq q^2/2^{n/3}] \leq \frac{\mathbf{E}[\sigma]}{q^2/2^{n/3}} \leq \frac{q}{2^{2n/3}} \text{ (by Eqn. (4.30).)} \tag{4.31}$$

Bounding B.10. Consider an indicator random variable $\mathbb{1}_{i_1 i_2}$, which takes value 1 if and only if $\exists\, (M_{i_1}, T_{i_1}), (M_{i_2}, T_{i_2}) \in \tau_c$ such that $T_{i_1} = T_{i_2}$. Since $\Pr[\mathbb{1}_{i_1 i_2} = 1] = 1/2^n$ for fixed i_1, i_2 (by the independence of T_{i_1} and T_{i_2}),

$$\mathbf{E}[\mathsf{C}] \;=\; \sum_{i_1 i_2} \mathbf{E}[\mathbb{1}_{i_1 i_2}] = \sum_{i_1 i_2} \Pr[\mathbb{1}_{i_1 i_2} = 1] \leq \frac{q^2}{2^n}. \tag{4.32}$$

Therefore,

$$\Pr[\text{B.10}] = \Pr[\mathsf{C} \geq q/2^{n/3}] \leq \frac{\mathbf{E}[\mathsf{C}]}{q/2^{n/3}} \leq \frac{q}{2^{2n/3}} \text{ (by Eqn. (4.32))}. \quad (4.33)$$

Bounding B.11. Again consider an indicator random variable $\mathbb{1}_{ia}$, taking value 1 if and only if $\exists\ (M_i, T_i) \in \tau_c, (\tilde{u}_a, \tilde{v}_a) \in \tau_p$ such that $M_i \oplus \mathbf{k}_1 = \tilde{u}_a$. Fixing i, a implies $\Pr[\mathbb{1}_{ia} = 1] = 1/2^n$ by the randomness of $\mathbf{k}_1$. Thus,

$$\mathbf{E}[\alpha] \quad = \quad \sum_{ia} \mathbf{E}[\mathbb{1}_{ia}] = \sum_{ia} \Pr[\mathbb{1}_{ia} = 1] \leq \frac{qp}{2^n}. \quad (4.34)$$

Then by Markov's inequality,

$$\Pr[\text{B.11}] = \Pr[\alpha \geq \sqrt{q}] \leq \frac{\mathbf{E}[\alpha]}{\sqrt{q}} \leq \frac{p\sqrt{q}}{2^n} \text{ (by Eqn. (4.34))}. \quad (4.35)$$

Bounding B.12. Consider $\mathbb{1}_{ia}$, taking value 1 if and only if $\exists\ (M_i, T_i) \in \tau_c, (\tilde{u}_a, \tilde{v}_a) \in \tau_p$ such that $T_i \oplus \mathbf{k}_1 = \tilde{v}_a$. For fixed values i, a, since $\Pr[\mathbb{1}_{ia} = 1] = 1/2^n$ by the randomness of $\mathbf{k}_1$,

$$\mathbf{E}[\beta] \quad = \quad \sum_{ia} \mathbf{E}[\mathbb{1}_{ia}] = \sum_{ia} \Pr[\mathbb{1}_{ia} = 1] \leq \frac{qp}{2^n}. \quad (4.36)$$

Therefore, using Markov's inequality gives

$$\Pr[\text{B.12}] = \Pr[\beta \geq \sqrt{q}] \leq \frac{\mathbf{E}[\beta]}{\sqrt{q}} \leq \frac{p\sqrt{q}}{2^n} \text{ (by Eqn. (4.36))}. \quad (4.37)$$

The result follows from Eqn.s (4.16)- (4.37). □

4.3.2. Analysis of Good Transcripts

This section shows that for a good transcript $\tau = (\tau_c, \tau_p, \mathbf{k})$, realizing τ is almost as likely in the real world as in the ideal world:

Lemma 12 (Good Lemma). *Let $\tau = (\tau_c, \tau_p, \mathbf{k}) \in \mathsf{GoodT}$ be a good transcript, and X_{re}, X_{id} be as defined previously. Then there exists a positive integer t such that for $0 \leq t \leq q/2^{n/3}$,*

$$\frac{\Pr[\mathsf{X}_{\text{re}} = \tau]}{\Pr[\mathsf{X}_{\text{id}} = \tau]} \geq 1 - \left(\frac{12q^2}{2^{4n/3}} + \frac{2pq}{2^{4n/3}} + \frac{13q}{2^{2n/3}} + \frac{2pt}{q^2} + \frac{2\sqrt{q}}{2^{n/3}} + \frac{28pq^2}{2^{2n}} + \frac{4p^2q}{2^{2n}} + \frac{24q^3}{2^{2n}} + \frac{2q^{3/2}}{2^n}\right).$$

4.3.3. Proof of Good Lemma

In this section, we prove Lemma 12 by establishing a lower bound for $\mathsf{p}(\tau)$ – since it is clear from Lemma 10 that to compute the ratio of real to ideal interpolation probabilities for a good transcript τ, one must compare this probability with 2^{nq}.

Establishing a Lower Bound for $\mathsf{p}(\tau)$

$\tau = (\tau_c, \tau_p, \mathbf{k}_1, \mathbf{k}_2)$ being a good transcript, we now prove that the set of construction queries τ_c can be partitioned into the following subsets:

$$\begin{aligned}\mathcal{Q}_{\mathcal{U}} &:= \{(M,T) \in \tau_c : M \oplus \mathbf{k}_1 \in \mathcal{U}\}\\ \mathcal{Q}_{\mathcal{V}} &:= \{(M,T) \in \tau_c : T \oplus \mathbf{k}_1 \in \mathcal{V}\}\\ \mathcal{Q}_0 &:= \{(M,T) \in \tau_c : M \oplus \mathbf{k}_1 \notin \mathcal{U}, T \oplus \mathbf{k}_1 \notin \mathcal{V}\}\end{aligned}$$

Observe first that $\mathcal{Q}_{\mathcal{U}} \cup \mathcal{Q}_{\mathcal{V}} \cup \mathcal{Q}_0$. By definition of bad transcripts, $\mathcal{Q}_{\mathcal{U}} \cap \mathcal{Q}_{\mathcal{V}} = \phi$ and by definition of the subsets, $\mathcal{Q}_{\mathcal{U}} \cap \mathcal{Q}_0 = \phi$, $\mathcal{Q}_{\mathcal{V}} \cap \mathcal{Q}_0 = \phi$. Thus:

Proposition 1. *Let $\tau = (\tau_c, \tau_p, \mathbf{k}_1, \mathbf{k}_2) \in \mathsf{GoodT}$ be a good transcript. Then the sets $(\mathcal{Q}_{\mathcal{U}}, \mathcal{Q}_{\mathcal{V}}, \mathcal{Q}_0)$ are pairwise disjoint.*

Since τ is a good transcript, $\alpha = |\mathcal{Q}_{\mathcal{U}}| \leq \sqrt{q}$ and $\beta = |\mathcal{Q}_{\mathcal{V}}| \leq \sqrt{q}$. Let $\mathsf{E}_{\mathcal{U}}$ denote the event $\mathsf{pEDM}^{\pi}_{\mathbf{k}} \longrightarrow \mathcal{Q}_{\mathcal{U}}$. Similarly, let $\mathsf{E}_{\mathcal{V}}$ denote the event $\mathsf{pEDM}^{\pi}_{\mathbf{k}} \longrightarrow \mathcal{Q}_{\mathcal{V}}$ and E_0 denote the event $\mathsf{pEDM}^{\pi}_{\mathbf{k}} \longrightarrow \mathcal{Q}_0$. It is easy to see that

$$\begin{aligned}\mathsf{p}(\tau) &= \Pr[\mathsf{E}_{\mathcal{U}} \wedge \mathsf{E}_{\mathcal{V}} \wedge \mathsf{E}_0 \,|\, \pi \longrightarrow \tau_p]\\ &= \Pr[\mathsf{E}_{\mathcal{U}} \wedge \mathsf{E}_{\mathcal{V}} \,|\, \pi \longrightarrow \tau_p] \cdot \Pr[\mathsf{E}_0 \,|\, \mathsf{E}_{\mathcal{U}} \wedge \mathsf{E}_{\mathcal{V}} \wedge \pi \longrightarrow \tau_p]. \quad (4.38)\end{aligned}$$

Therefore, it is enough to establish a good lower bound on the final two probabilities to obtain a bound for τ.

To bound $\Pr[\mathsf{E}_{\mathcal{U}} \wedge \mathsf{E}_{\mathcal{V}} \,|\, \pi \longrightarrow \tau_p]$, we define the following sets:

$$\begin{aligned}\mathcal{S}_1 &:= \{M \oplus \mathbf{k}_1 : (M,T) \in \mathcal{Q}_{\mathcal{U}}\},\ \ \mathcal{S}_2 := \{M \oplus \mathbf{k}_1 : (M,T) \in \mathcal{Q}_{\mathcal{V}}\}\\ \mathcal{D}_1 &:= \{T \oplus \mathbf{k}_1 : (M,T) \in \mathcal{Q}_{\mathcal{U}}\},\ \ \mathcal{D}_2 := \{T \oplus \mathbf{k}_1 : (M,T) \in \mathcal{Q}_{\mathcal{V}}\}\end{aligned}$$

Note that $\mathcal{S}_1 \subseteq \mathcal{U}$, $|\mathcal{S}_1| = \alpha$, and $\mathcal{D}_2 \subseteq \mathcal{V}$, $|\mathcal{D}_2| = \beta$. Since $\pi \longrightarrow \tau_p$, π is fixed on exactly p input-output pairs. Therefore, for each $(M,T) \in \mathcal{Q}_{\mathcal{U}}$, $\exists!$ $(\tilde{u}, \tilde{v}) \in \tau_p$ such that $M \oplus \mathbf{k}_1 = \tilde{u}$, so that $\pi(M \oplus k)$ is well defined and equal to $\tilde{v}$. Similarly, for each $(M,T) \in \mathcal{Q}_{\mathcal{V}}$, $\exists!$ $(\tilde{u}, \tilde{v}) \in \tau_p$ such that $T \oplus \mathbf{k}_1 = \tilde{v}$, so that $\pi^{-1}(T \oplus \mathbf{k}_1)$ is well defined and equal to $\tilde{u}$. We can now define the following two sets:

$$\begin{aligned}\mathcal{X}_1 &:= \{\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 \,:\, (M,T) \in \mathcal{Q}_{\mathcal{U}}\}\\ \mathcal{X}_2 &:= \{\pi^{-1}(T \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 \,:\, (M,T) \in \mathcal{Q}_{\mathcal{V}}\}.\end{aligned}$$

Proposition 2. *Every element of $\mathcal{D}_1$ is distinct and does not collide with any primitive query output. Similarly, every element of $\mathcal{S}_2$ is distinct and does not collide with any primitive query input.*

Proof. The distinct property of $\mathcal{D}_1$ follows from ¬B.6. Moreover, if any element of $\mathcal{D}_1$ collides with a primitive query output then it would satisfy condition B.2. Thus, $\mathcal{D}_1 \cap \mathcal{V} = \phi \Rightarrow \mathcal{D}_1 \cap \mathcal{D}_2 = \phi$ and hence $|\mathcal{D}_1| = \alpha$. Similarly by definition, every element of $\mathcal{S}_2$ is unique and does not collide with any primitive query input (otherwise satisfies condition B.2). Hence, $\mathcal{S}_2 \cap \mathcal{U} = \phi \Rightarrow \mathcal{S}_2 \cap \mathcal{S}_1 = \phi$ and hence $|\mathcal{S}_2| = \beta$. □

Proposition 3. *Every element of $\mathcal{X}_1$ is distinct and $\mathcal{X}_1 \cap \mathcal{S}_1 = \phi, \mathcal{X}_1 \cap \mathcal{S}_2 = \phi$. Every element of $\mathcal{X}_2$ is distinct and $\mathcal{X}_2 \cap \mathcal{D}_1 = \phi, \mathcal{X}_2 \cap \mathcal{D}_2 = \phi$.*

Proof. For the sake of contradiction, assume that $\pi(M_{i_1} \oplus \mathbf{k}_1) \oplus M_{i_1} \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 = \pi(M_{i_2} \oplus \mathbf{k}_1) \oplus M_{i_2} \oplus \mathbf{k}_1 \oplus \mathbf{k}_2$ for some $(M_{i_1}, T_{i_1}),)(M_{i_2}, T_{i_2}) \in \mathcal{Q}_\mathcal{U}$. However, this requires validity of the condition B.7, necessitating τ to be a bad transcript. Thus, every element of $\mathcal{X}_1$ is distinct. Furthermore, no element of $\mathcal{X}_1$ collides with any primitive query input, as otherwise condition B.1 would be satisfied. This implies $\mathcal{X}_1 \cap \mathcal{S}_1 = \phi$. Also, $\mathcal{X}_1 \cap \mathcal{S}_2 = \phi$ follows due to ¬B.4. Thus, $\mathcal{X}_1 \cap \mathcal{U} = \phi \Rightarrow \mathcal{X}_1 \cap \mathcal{S}_1 = \phi$ and $\mathcal{X}_1 \cap \mathcal{S}_2 = \phi$. Hence, $|\mathcal{X}_1| = \alpha$.

Again assuming $\pi^{-1}(T_{i_1} \oplus \mathbf{k}_1) \oplus (M_{i_1} \oplus \mathbf{k}_1) \oplus \mathbf{k}_2 = \pi^{-1}(T_{i_2} \oplus \mathbf{k}_1) \oplus (M_{i_2} \oplus \mathbf{k}_1) \oplus \mathbf{k}_2$ for some $(M_{i_1}, T_{i_1}), (M_{i_2}, T_{i_2}) \in \mathcal{Q}_\mathcal{V}$ forces the condition B.8 to hold, necessitating τ to be a bad transcript. Thus, every element of $\mathcal{X}_2$ is distinct. Moreover, no element of $\mathcal{X}_2$ collides with any primitive query output, as otherwise condition B.3 would be satisfied. This implies $\mathcal{X}_2 \cap \mathcal{D}_2 = \phi$. Since $\mathcal{X}_2 \cap \mathcal{D}_1 = \phi$ follows due to ¬B.5, $\mathcal{X}_2 \cap \mathcal{V} = \phi \Rightarrow \mathcal{X}_2 \cap \mathcal{D}_2 = \phi$ and $\mathcal{X}_2 \cap \mathcal{D}_1 = \phi$. Hence, $|\mathcal{X}_2| = \beta$. □

Consider the following two sequences:

$$\begin{aligned}\mathcal{Q}_1 &:= (\pi(M_i \oplus \mathbf{k}_1) \oplus M_i \oplus \mathbf{k}_1 \oplus \mathbf{k}_2, T_i \oplus \mathbf{k}_1)_{(M_i,T_i)\in\mathcal{Q}_\mathcal{U}}\,.\\ \mathcal{Q}_2 &:= \left(M_i \oplus \mathbf{k}_1, \pi^{-1}(T_i \oplus \mathbf{k}_1) \oplus M_i \oplus \mathbf{k}_1 \oplus \mathbf{k}_2\right)_{(M_i,T_i)\in\mathcal{Q}_\mathcal{V}}.\end{aligned}$$

From propositions 2 and 3, it follows that the domain of $\mathcal{Q}_1$ is disjoint from the domain of $\mathcal{Q}_2$. Moreover, they are pairwise disjoint from $\mathcal{U}$. Similarly, the range of $\mathcal{Q}_1$ is disjoint from the range of $\mathcal{Q}_2$ and they are pairwise disjoint from $\mathcal{V}$. Therefore, $\mathfrak{X} = (\mathcal{U}, \mathcal{X}_1, \mathcal{S}_2)$ and $\mathfrak{Y} = (\mathcal{V}, \mathcal{D}_1, \mathcal{X}_2)$ are disjoint collections. Thus, from Proposition 1,

$$\begin{aligned}&\Pr[\mathsf{E}_\mathcal{U} \wedge \mathsf{E}_\mathcal{V} \mid \pi \longrightarrow \tau_p]\\ :=\ &\Pr[\pi \xleftarrow{\$} \mathsf{Perm}(n) : \mathfrak{X} \setminus \mathcal{U} \xrightarrow{\pi} \mathfrak{Y} \setminus \mathcal{V} \mid \pi \longrightarrow \tau_p]\\ =\ &\frac{1}{(2^n - p)_{\alpha+\beta}}.\end{aligned} \tag{4.39}$$

We now bound $\Pr[\mathsf{E}_0 \mid \mathsf{E}_{\mathcal{U}} \wedge \mathsf{E}_{\mathcal{V}} \wedge \pi \longrightarrow \tau_p]$. For the sake of simplicity, we rename the elements of $\mathcal{Q}_0$ as $\mathcal{Q}_0 = \{(M_1, T_1), (M_2, T_2), \ldots, (M_{q'}, T_{q'})\}$, and note that $|\mathcal{Q}_0| = q' = q - (\alpha + \beta)$. Let us define sets

$$\begin{aligned}\mathcal{X} &= \{M \in \{0,1\}^n : (M, T) \in (\tau_c \setminus \mathcal{Q}_{\mathcal{U}})\}\\ \mathcal{Y} &= \{T \in \{0,1\}^n : (M, T) \in \mathcal{Q}_0\}\\ \mathcal{S} &= \{(M, T) \in \mathcal{Q}_0 : \forall (M', T') \neq (M, T) \in \mathcal{Q}_0, T \neq T'\},\end{aligned}$$

where $r = |\mathcal{Y}|$, $\mathcal{S}$ is the set of non-colliding queries of $\mathcal{Q}_0$ and $s' = |\mathcal{S}|$. Since τ is a good transcript, $s' \geq q - \frac{q}{2^{n/3}}$, so as not to satisfy B.10. Thus, we must bound the probability that a permutation π realizes $\mathcal{Q}_0$, i.e. we must bound the number of permutations π that are already fixed on $\alpha + \beta$ input-output pairs such that

$$\forall (M, T) \in \mathcal{Q}_0, \pi(\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2) \oplus \mathbf{k}_1 = T. \tag{4.40}$$

Note that the equations in Eqn. (4.40) are not independent as the two permutations are identical: for two queries (M, T) and (M', T') in $\mathcal{Q}_0$, $\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 = M \oplus \mathbf{k}_1$ implies $\pi(M' \oplus \mathbf{k}_1) = T \oplus \mathbf{k}_1$. Similarly, $\pi(M \oplus \mathbf{k}_1) = T' \oplus \mathbf{k}_1$ implies $\pi(M' \oplus \mathbf{k}_1) \oplus M' \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 = M \oplus \mathbf{k}_1$. One could count only the permutations π that are already fixed on $\alpha + \beta$ input-output pairs such that for any query $(M, T) \in \mathcal{Q}_0$, $\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 \notin \mathcal{X} \oplus \mathbf{k}_1$. However this only leads to a birthday bound. To obtain a beyond the birthday bound, we need to allow for collisions and a more precise counting. We shall thus consider permutations π that are already fixed on $\alpha + \beta$ input-output pairs such that $\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 = M' \oplus \mathbf{k}_1$ for t pairs $((M, T), (M', T'))$ of distinct non-colliding queries, where t is a sufficiently large value. However, care must be taken in choosing the t-pairs of distinct non-colliding queries so as to not create incompatibility with other queries.

Counting Collisions

To this end, we define an index set $\mathcal{I} = \{i \in [q'] : (M_i, T_i) \in \mathcal{S}\}$, and the set $\mathcal{I}^{(\underline{2})} = \mathcal{I}^{(2)} = \{(i, j) : i, j \in \mathcal{I}, i \neq j\}$ i.e. the set of all ordered pairs of distinct elements of $\mathcal{I}$.

Definition 4. *For a fixed positive integer t, an unordered set of t ordered pairs of indices*

$$\mathcal{I}_t = \{(i_1, j_1), (i_2, j_2), \ldots, (i_t, j_t)\} \subseteq \mathcal{I}^{(\underline{2})},$$

is good *if it satisfies the following conditions:*

1. *for $l \in [t]$, $M_{j_l} \oplus M_{i_l}$ are distinct.*

2. *for* $l \in [t]$, $T_{i_l} \oplus M_{j_l}$ *are distinct.*
3. *for* $l \in [t]$, $M_{j_l} \oplus M_{i_l} \oplus \mathbf{k}_2 \notin \mathcal{V}$.
4. *for* $l \in [t]$, $T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 \notin \mathcal{U}$.
5. *for* $l \in [t]$, $T_{i_l} \oplus \mathbf{k}_2 \oplus M_{j_l} \notin \mathcal{X} \oplus \mathbf{k}_1$.
6. *for* $l \in [t]$, $M_{i_l} \oplus \mathbf{k}_2 \oplus M_{j_l} \notin \mathcal{Y} \oplus \mathbf{k}_1$.
7. *for* $l \in [t]$, $M_{j_l} \oplus M_{i_l} \oplus \mathbf{k}_2 \notin \mathcal{X}_2$.
8. *for* $l \in [t]$, $T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 \notin \mathcal{X}_1$.

We call an element in the set $\mathcal{I}_t$ *a* dependency pair.

Note that $\mathcal{I}_t$ is the set of t ordered pairs of indices of non-colliding queries. We justify below why the conditions listed above, are not incompatible with the other queries.

Rationale for the Conditions. A dependency pair $(i_l, j_l) \in \mathcal{I}_t$ is dependent in one of the following two ways:

(a) $\pi(M_{i_l} \oplus \mathbf{k}_1) \oplus (M_{i_l} \oplus \mathbf{k}_1) \oplus \mathbf{k}_2 = M_{j_l} \oplus \mathbf{k}_1$ or
(b) $\pi(M_{i_l} \oplus \mathbf{k}_1) = T_{j_l} \oplus \mathbf{k}_1$

for permutations π that are already fixed on $\alpha + \beta$ input-output pairs. Such a dependency pair is said to be of *length 1*. Thus, we have the following two equalities:

$(\widehat{a})$ $\pi(M_{i_l} \oplus \mathbf{k}_1) = M_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2$,
$(\widehat{b})$ $\pi(M_{j_l} \oplus \mathbf{k}_1) = M_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2$.

Both these equalities impose distinctness of the permutation outputs, which justifies condition (1) of Defn. 4. They also require the permutation outputs to not collide with any primitive output (i.e. the elements of $\mathcal{V}$), as their corresponding input does not collide with any primitive input (i.e. elements of $\mathcal{U}$), validating condition (3). Similarly, $M_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2$ should not collide with any element of $\mathcal{Y} \oplus \mathbf{k}_1$, establishing condition (6). Furthermore, $M_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2$ should not collide with any element of $\mathcal{X}_2$, as $(M_{i_l}, T_{i_l}), (M_{j_l, T_{j_l}}) \notin \mathcal{Q}_{\mathcal{V}}$. This justifies conditions (7). Finally, since Eqn. (a) also imposes the following equality:

$$\widehat{(c)}\ \pi(M_{j_l} \oplus \mathbf{k}_1) = T_{i_l} \oplus \mathbf{k}_1,$$

$\pi(M_{j_l} \oplus \mathbf{k}_1) \oplus (M_{j_l} \oplus \mathbf{k}_1 \oplus \mathbf{k}_2)$ (or equivalently $T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2$) should be distinct, which accounts for condition (2). Moreover, it should not collide with any other elements of $\mathcal{X} \oplus \mathbf{k}_1$, as it would otherwise extend the length of the dependency pair by 1. This validates condition (5) of the definition. Similarly $T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2$ should not collide with any primitive inputs, as $T_{j_l} \oplus \mathbf{k}_1$ does not collide with any primitive output. This establishes condition (4) of the definition. As $T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2$ should not collide with any element of $\mathcal{X}_2$ (otherwise $T_{j_l} \oplus \mathbf{k}_1 \in \mathcal{D}_1$, which is not possible as $(M_{j_l}, T_{j_l}) \notin \mathcal{Q}_{\mathcal{U}}$), this justifies condition (8).

Lemma 13. *Fix a positive integer t such that $0 \leq t \leq M$. Then the number of good sets $\mathcal{I}_t$ of t pairs of non-colliding queries is at least*

$$|\mathcal{I}_t| \geq \frac{(s')_{2t}}{t!}\left(1 - \frac{4q}{2^{2n/3}} - \frac{2pt}{q^2} - \frac{2\sqrt{q}}{2^{n/3}}\right).$$

Proof. First observe that amongst the $s'(s'-1)$ possible pairs of non-colliding query indices (i_1, j_1), at most $(2\sigma + 2p + \alpha + \beta)$ of them do not satisfy conditions (3)-(8). Indeed, by definition of a good transcript (more precisely, bad event B.9), there cannot be more than σ pairs $((M_{i_l}, T_{i_l}), (M_{j_l}, T_{j_l}))$ such that $T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 \in \mathcal{X} \oplus \mathbf{k}_1$, and there cannot be more than σ pairs $((M_{i_l}, T_{i_l}), (M_{j_l}, T_{j_l}))$ such that $M_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 \in \mathcal{Y} \oplus \mathbf{k}_1$. Similarly, due to B.11, there cannot be more than α pairs such that $T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 \in \mathcal{X}_1$ and due to B.12, there cannot be more than β pairs such that $M_{j_l} \oplus M_{i_l} \oplus \mathbf{k}_2 \in \mathcal{X}_2$. Hence, we obtain a lower bound $\mathcal{I}_t$ as follows:

- (i_1, j_1) can be chosen from at least $s'(s'-1) - 2\sigma - 2p - \alpha - \beta$ possibilities.
- Once (i_1, j_1) is fixed, i_2 can be chosen freely from the remaining $(s'-2)$ possibilities. Since j_2 must be different from i_1, j_1 and i_2, $M_{j_2} \oplus M_{i_2} \neq M_{j_1} \oplus M_{i_1}$ and $T_{i_2} \oplus M_{j_2} \neq T_{i_1} \oplus M_{j_1}$, the number of choices for j_2 is $(s'-5)$; after removing the at most $2\sigma + 2p + \alpha + \beta$ pairs of queries not satisfying (3)-(8), there remain at least $(s'-2)(s'-5) - 2\sigma - 2p - \alpha - \beta$ possibilities for the pair (i_2, j_2).
- Assuming $(i_1, j_1), (i_2, j_2), (i_{l-1}, j_{l-1})$ are already selected, i_l can be chosen freely from the $(s'-2l+2)$ remaining possibilities. As j_l must be different from $i_1, j_1, \ldots, i_{l-1}, j_{l-1}, i_l$ and it must be such that $M_{j_l} \oplus M_{i_l} \neq M_{j_d} \oplus M_{i_d}$ for $d \in [l-1]$, $T_{i_l} \oplus M_{j_l} \neq T_{i_d} \oplus M_{j_d}$ for $d \in [l-1]$, there are at least $(s'-4l+3)$ possibilities for j_l. After removing at most $2\sigma + 2p + \alpha + \beta$ pairs not satisfying (3)-(8), there remain at least $(s'-2l+2)(s'-4l+3) - 2\sigma - 2p - \alpha - \beta$ possibilities for the pair (i_l, j_l).

Since $\mathcal{I}_t$ is an unordered set of t pairs, the number $|\mathcal{I}_t|$ of good sets is at least

$$|\mathcal{I}_t| \geq \frac{1}{t!}\prod_{l=0}^{t-1}\left((s'-2l)(s'-4l-1) - 2\sigma - 2p - \alpha - \beta\right).$$

$$
\begin{aligned}
\text{Therefore, } |\mathcal{I}_t| &\geq \frac{(s')_{2t}}{t!}\prod_{l=0}^{t-1}\frac{(s'-2l)(s'-4l-1)-2p-2\sigma-\alpha-\beta}{(s'-2l)(s'-2l-1)} \\
&\geq \frac{(s')_{2t}}{t!}\prod_{l=0}^{t-1}\left(1-\frac{2ls'-4l^2+2p+2\sigma+\alpha+\beta}{(s'-2l)(s'-2l-1)}\right) \\
&\geq \frac{(s')_{2t}}{t!}\prod_{l=0}^{t-1}\left(1-\frac{2ls'+2p+2\sigma+\alpha+\beta}{(s'-2M)^2}\right) (\because l \leq t \leq M) \\
&\geq \frac{(s')_{2t}}{t!}\left(1-\sum_{l=0}^{t-1}\frac{2ls'+2p+2\sigma+\alpha+\beta}{(s'-2M)^2}\right) \\
&\geq \frac{(s')_{2t}}{t!}\left(1-\frac{2s'M^2+2pt+2\sigma M+\alpha M+\beta M}{q^2}\right) \\
&\qquad (\because t \leq M \text{ and } s'-2M \leq q) \\
&\geq \frac{(s')_{2t}}{t!}\left(1-\frac{2M^2}{q}-\frac{2pt}{q^2}-\frac{2\sigma M}{q^2}-\frac{\beta M}{q^2}-\frac{\alpha M}{q^2}\right) \\
&\qquad (\text{as } s' \leq q) \\
&\geq \frac{(s')_{2t}}{t!}\left(1-\frac{2q}{2^{2n/3}}-\frac{2pt}{q^2}-\frac{2q}{2^{2n/3}}-\frac{2}{\sqrt{q}2^{n/3}}\right) \\
&\qquad \left(\text{as } M \leq q/2^{n/3}, \sigma \leq q^2/2^{n/3} \text{ and } \alpha,\beta \leq \sqrt{q}\right) \\
&\geq \frac{(s')_{2t}}{t!}\left(1-\frac{2q}{2^{2n/3}}-\frac{2pt}{q^2}-\frac{2q}{2^{2n/3}}-\frac{2\sqrt{q}}{2^{n/3}}\right) (q \geq 1) \\
&\geq \frac{(s')_{2t}}{t!}\left(1-\frac{4q}{2^{2n/3}}-\frac{2pt}{q^2}-\frac{2\sqrt{q}}{2^{n/3}}\right). \qquad \square
\end{aligned}
$$

We shall henceforth work with a fixed positive integer t such that $0 \leq t \leq M$ and a good set $\mathcal{I}_t = \{(i_1, j_1), \ldots, (i_t, j_t)\}$. For a good set $\mathcal{I}_t$, let $\mathcal{Q}_{\mathcal{I}_t} \coloneqq \{(M_i, T_i) \in \mathcal{Q}_0 : (i, \star) \in \mathcal{I}_t \vee (\star, i) \in \mathcal{I}_t\}$. We shall now compute a lower bound on the number of permutations π that are already fixed on $\alpha + \beta$ input-output pairs and satisfy

$$\pi(\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2) \oplus \mathbf{k}_1 = T, \forall (M, T) \in \mathcal{Q}_0 \tag{4.41}$$

such that for any $l \in [t]$, $\pi(M_{i_l} \oplus \mathbf{k}_1) \oplus M_{i_l} \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 = M_{j_l} \oplus \mathbf{k}_1$. Such a permutation π for the $2t$ queries appearing in $\mathcal{I}_t$ exists if and only if $\forall l \in [t]$,

1. $\pi(M_{i_l} \oplus \mathbf{k}_1) = M_{j_l} \oplus M_{i_l} \oplus \mathbf{k}_2$,
2. $\pi(M_{j_l} \oplus \mathbf{k}_1) = T_{i_l} \oplus \mathbf{k}_1$ and
3. $\pi(T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2) = T_{j_l} \oplus \mathbf{k}_1$.

This set of $3t$ equalities is input-output compatible as $\mathcal{I}_t$ is a good set. Now, as the sets in the collection $\mathfrak{X}^+ = \{\mathcal{X} \oplus \mathbf{k}_1, \mathcal{U}, \mathcal{X}_1, \{T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 : l \in [t]\}\}$

are pairwise disjoint, and so are the sets in the collection $\mathfrak{Y}^+ = \{\mathcal{Y} \oplus \mathbf{k}_1, \mathcal{V}, \mathcal{X}_2, \{M_{j_l} \oplus M_{i_l} \oplus \mathbf{k}_2 : l \in [t]\}, \{M_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 : l \in [t]\}\}$, we can define

$$\begin{aligned}\mathcal{X}' &:= \mathcal{X} \oplus \mathbf{k}_1 \cup \mathcal{U} \cup \mathcal{X}_1 \cup \{T_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 : l \in [t]\} \text{ and} \\ \mathcal{Y}' &:= \mathcal{Y} \oplus \mathbf{k}_1 \cup \mathcal{V} \cup \mathcal{X}_2 \cup \{M_{j_l} \oplus M_{i_l} \oplus \mathbf{k}_2 : l \in [t]\} \\ &\qquad \cup \{M_{i_l} \oplus M_{j_l} \oplus \mathbf{k}_2 : l \in [t]\}.\end{aligned}$$

It is easy to see that $|\mathcal{X}'| = q' + p + \alpha + t$ and $|\mathcal{Y}'| = r + 2t + p + \beta$, where one may recall that $r = |\mathcal{Y}|$. For the remaining queries in $\mathcal{Q}_0$ such that $(M, T) \notin \mathcal{Q}_{\mathcal{I}_t}$, let $q'' = q' - 2t = q - \alpha - \beta - 2t$, $s'' = s' - 2t$ the number of non-colliding queries in $\mathcal{Q}_0 \setminus \mathcal{Q}_{\mathcal{I}_t}$ and $r' = r - 2t$ the number of distinct oracle responses appearing in these queries.
An approach similar to [55] allows us to regroup the elements of $\mathcal{Q}_0 \setminus \mathcal{Q}_{\mathcal{I}_t}$ such that all queries with the same output become consecutive. We write the queries as follows:

$$\tau' = \begin{cases} \Big((M_{11}, T_1), \ldots, (M_{1q_1}, T_1), \\ (M_{21}, T_2), \ldots, (M_{2q_2}, T_2), \\ \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\ (M_{r',1}, T_{r'}), \ldots, (M_{r',q_{r'}}, T_{r'}) \Big), \end{cases}$$

where $T_1, \ldots, T_{r'}$ are distinct. Furthermore, $(q_1 + \ldots + q_{r'}) = q''$. For the ease of later computations, we assume that all non-colliding queries appear first followed by colliding queries, i.e. $q_i = 1$ for $i \in [s'']$ and $q_i > 1$ for $i \in \{s'' + 1, \ldots, r'\}$. Our goal is to obtain a lower bound on the number of permutations $\pi \in \mathsf{Perm}(n)$ that are already fixed on $\alpha + \beta$ input-output pairs, and in addition to satisfying above $3t$ equalities, also satisfy the following:

$$\forall (M, T) \in \tau', \pi(\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2) \oplus \mathbf{k}_1 = T. \tag{4.42}$$

We thus sample all intermediate values $z = \pi^{-1}(T \oplus \mathbf{k}_1)$, which leads us to the second step of the proof.

Sampling Intermediate Values

Consider a sequence $\mathbf{z} = (z_1, z_2, \ldots, z_{r'})$ of r n-bit values. We say $\mathbf{z}$ is *good* if

1. each z_i is distinct
2. for all $i \in [r'], z_i \notin \mathcal{X}'$
3. for all $i \in [r']$ and $j \in [q_i]$, $z_i \oplus M_{i,j}$ are distinct
4. for all $i \in [r']$ and $j \in [q_i]$, $z_i \oplus M_{i,j} \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 \notin \mathcal{Y}'$.

For any good tuple $\mathbf{z} = (z_1, \ldots, z_{r'})$, the equalities

$$\forall i \in [r'], \forall j \in [q_i], \pi(M_{i,j} \oplus \mathbf{k}_1) = z_i \oplus \mathbf{k}_2 \oplus M_{i,j} \oplus \mathbf{k}_1$$

$$\forall i \in [r'], \pi(z_i) = T_i \oplus \mathbf{k}_1$$

are compatible with all previously defined input-output pairs. Likewise, a permutation π satisfying these equations is such that $\pi(\pi(M \oplus \mathbf{k}_1) \oplus M \oplus \mathbf{k}_1 \oplus \mathbf{k}_2) = T \oplus \mathbf{k}_1$ for all $(M, T) \in \tau'$.

Lemma 14. *Fix a positive integer t such that $0 \leq t \leq M$ and a good set $\mathcal{I}_t$. Then the number of good tuples $\mathbf{z} = (z_1, \ldots, z_{r'})$ is at least*

$$\mathcal{N}_{\mathbf{z}}(t) \geq (2^n - q - p - \alpha - \beta - 3t - r)_{s''} \prod_{i=0}^{s''-1} \left(1 - \frac{p+i}{2^n - (5q+p) - i}\right) (2^n)^{r'-s''} \left(1 - \frac{6q^2}{2^{4n/3}} - \frac{2pq}{2^{4n/3}}\right).$$

Proof. The number of valid choices for z_1 is at least $2^n - (q + p + \alpha + t) - q_1(r + 2t + p + \beta)$ as $z_1 \notin \mathcal{X}'$ and $z_1 \oplus M_{1,j} \oplus \mathbf{k}_1 \oplus \mathbf{k}_2 \notin \mathcal{Y}'$ for $j \in [q_1]$, where $|\mathcal{X}'| = q + p + \alpha + t$ and $|\mathcal{Y}'| = r + 2t + p + \beta$. Once the value of z_1 is fixed, z_2 can be chosen in the following way:

- $z_2 \neq z_1$
- $z_2 \notin \mathcal{X}'$
- $z_2 \oplus \mathbf{k}_2 \oplus M_{2,j} \oplus \mathbf{k}_1 \notin \mathcal{Y}'$ for $j \in [q_2]$
- $z_2 \neq z_1 \oplus M_{1,j} \oplus M_{2,j'}$ for $j \in [q_1], j' \in [q_2]$.

Thus, the number of valid choices for z_2 is at least $2^n - 1 - (q + p + \alpha + t) - q_2(r + 2t + p + \beta + q_1)$. In general, after choosing values for $z_1, \ldots, z_{i-1}$, the number of valid choices for z_i is at least

$$2^n - (i-1) - (q + p + \alpha + t) - q_i \left(r + 2t + p + \beta + \sum_{j=1}^{i-1} q_j\right).$$

This is because z_i cannot be equal to $z_1, \ldots, z_{i-1}$, which accounts for $i - 1$ terms in the above equation. The term $(q + p + \alpha + t)$ is present in the equation as $z_i \notin \mathcal{X}'$, and $j \in [q_i], z_i \oplus \mathbf{k}_2 \oplus M_{i,j} \oplus \mathbf{k}_1 \notin \mathcal{Y}'$, which establishes the term $q_i(r + 2t + p + \beta)$. Lastly, $z_i \neq z_l \oplus M_{l,j} \oplus M_{i,j'}$ for $l \in [i-1], j \in [q_l], j' \in [q_i]$ explains the term $q_i(q_1 + \ldots + q_{i-1})$ in the equation. Therefore, overall the number of good tuples $\mathbf{z}$ is at least

$$\mathcal{N}_{\mathbf{z}}(t) \geq \prod_{i=0}^{r'-1} \left(2^n - (q + p + \alpha + t) - i - q_{i+1} \left(r + 2t + p + \beta + \sum_{j=1}^{i} q_j\right)\right). \tag{4.43}$$

Next, we split Eqn. (4.43) into two parts – the first comprising the s'' non-colliding queries and the second, the colliding queries:

$$\begin{aligned}
\mathcal{N}_{\mathbf{z}}(t) \;&\geq\; \prod_{i=0}^{s''-1} (2^n - (q+p+\alpha+t) - i - r - 2t - p - \beta - i) \cdot \\
&\prod_{i=s''}^{r'-1} \left(2^n - (q+p+\alpha+t) - i - q_{i+1}\left(r + 2t + p + \beta + \sum_{j=1}^{i} q_j\right)\right) \\
&\geq \prod_{i=0}^{s''-1} (2^n - (q+2p+\alpha+\beta+3t) - 2i - r) \cdot \\
&\prod_{i=s''}^{r'-1} \left(2^n - (q+p+\alpha+t) - i - q_{i+1}\left(r + 2t + p + \beta + \sum_{j=1}^{i} q_j\right)\right).
\end{aligned} \tag{4.44}$$

$$\begin{aligned}
&\text{Now, } \prod_{i=0}^{s''-1} (2^n - (q+2p+\alpha+\beta+3t) - 2i - r) \\
&= (2^n - q - p - \alpha - \beta - 3t - r)_{s''} \cdot \prod_{i=0}^{s''-1} \frac{2^n - q - 2p - \alpha - \beta - 3t - 2i - r}{2^n - q - p - \alpha - \beta - 3t - r - i} \\
&= (2^n - q - p - \alpha - \beta - 3t - r)_{s''} \cdot \prod_{i=0}^{s''-1} \left(1 - \frac{p+i}{2^n - q - p - \alpha - \beta - 3t - r - i}\right) \\
&\geq (2^n - q - p - \alpha - \beta - 3t - r)_{s''} \cdot \prod_{i=0}^{s''-1} \left(1 - \frac{p+i}{2^n - (5q+p) - i}\right),
\end{aligned} \tag{4.45}$$

since $\alpha, \beta \leq q$, $r \leq q$ and $3t \leq q$. Also, for $i \in \{s'', s''+1, \ldots, r'-2, r'-1\}$,

$$\begin{aligned}
& q + p + \alpha + t + i \\
\leq\; & q + p + \alpha + t + r' - 1 \quad (\text{ as } i \leq r'_1) \\
=\; & q + p + \alpha + r - t - 1 \quad (\text{ as } r' = r - 2t) \\
\leq\; & 3q + p \quad (\text{ as } \alpha \leq q, r - t - 1 \leq q), \text{ and}
\end{aligned} \tag{4.46}$$

$$\begin{aligned}
& r + 2t + p + \beta + \sum_{j=1}^{i} q_j \\
\leq\; & r + 2t + p + \beta + q'' \quad (\text{ as } (q_1 + \ldots q_{r'}) = q'') \\
=\; & r + p + \beta + q' \quad (\text{ as } q'' = q' - 2t) \\
\leq\; & 3q + p \quad (\text{ as } q' \leq q, \beta \leq q \text{ and } r \leq q), \text{ so that}
\end{aligned} \tag{4.47}$$

$$
\begin{aligned}
&\prod_{i=s''}^{r'-1}\left(2^n-(q+p+\alpha+t)-i-q_{i+1}\left(r+2t+p+\beta+\sum_{j=1}^{i}q_j\right)\right)\\
=\;&\prod_{i=s''}^{r'-1}\left(2^n-(q+p+\alpha+t)-i-q_{i+1}\left(r+2t+p+\beta+\sum_{j=1}^{i}q_j\right)\right)\\
\geq\;&\prod_{i=s''}^{r'-1}(2^n-(3q+p)-q_{i+1}(3q+p)) \quad \text{(from Eqn.s (4.46) and (4.47))}\\
\geq\;&(2^n)^{r'-s''}\prod_{i=s''}^{r'-1}\left(\frac{2^n-(3q+p)-q_{i+1}(3q+p)}{2^n}\right)\\
\geq\;&(2^n)^{r'-s''}\prod_{i=s''}^{r'-1}\left(1-\frac{q_{i+1}(6q+2p)}{2^n}\right)\\
\geq\;&(2^n)^{r'-s''}\left(1-\sum_{i=s''}^{r'}\frac{q_{i+1}(6q+2p)}{2^n}\right)\\
\geq\;&(2^n)^{r'-s''}\left(1-\frac{(6q+2p)\sum_{i=s''}^{r'}q_{i+1}}{2^n}\right)\\
\geq\;&(2^n)^{r'-s''}\left(1-\frac{(6q+2p)M}{2^n}\right) \qquad \left(\text{since } \sum_{i=s''}^{r'-1}q_{i+1}\leq M\right)\\
\geq\;&(2^n)^{r'-s''}\left(1-\frac{6q^2}{2^{4n/3}}-\frac{2pq}{2^{4n/3}}\right) \qquad \left(\text{since } M\leq q/2^{n/3}\right). \qquad (4.48)
\end{aligned}
$$

Therefore, the result follows from Eqn.s (4.44), (4.45) and (4.48). □

Final Calculation

As Sect. 4.3.3 fixes π on $3t$ input-output pairs and Sect. 4.3.3 fixes π on $q''+r'$ input-output pairs, i.e. a total of $3t+q''+r' = q'+t+r'$ pairs (as $q''=q'-2t$), and also on $p+\alpha+\beta$ input-output pairs by the attack transcript, Lemmas 13 and 14 imply

$$
\begin{aligned}
&\Pr[\mathsf{E}_0 \mid \mathsf{E}_{\mathcal{U}} \wedge \mathsf{E}_{\mathcal{V}} \wedge \pi \longrightarrow \tau_p] \geq \sum_{0\leq t\leq M}\frac{|\mathcal{I}_t|\cdot\mathcal{N}_{\mathbf{z}}(t)}{(2^n-p-\alpha-\beta)_{q'+t+r'}}\\
\geq\;&(2^n)^{r'-s''}\cdot\left(1-\frac{6q^2}{2^{4n/3}}-\frac{2pq}{2^{4n/3}}\right)\cdot\sum_{0\leq t\leq M}\left[\frac{(s')_{2t}}{t!}\cdot\left(1-\frac{4q}{2^{2n/3}}-\frac{2pt}{q^2}-\frac{2\sqrt{q}}{2^{n/3}}\right)\right.\\
&\left.\cdot\prod_{i=0}^{s''-1}\left(1-\frac{p+i}{2^n-(5q+p)-i}\right)\cdot\cdot\frac{(2^n-q-p-\alpha-\beta-3t-r)_{s''}}{(2^n-p-\alpha-\beta)_{q'+t+r'}}\right]. \qquad (4.49)
\end{aligned}
$$

Therefore, from Lemma 10 and Eqn.s (4.38), (4.39) and (4.49),

$$\begin{aligned}
\rho(\tau) \;\geq\; & \frac{2^{n(q+r'-s'')}}{(2^n-p)_{\alpha+\beta}} \cdot \left(1-\frac{6q^2}{2^{4n/3}}-\frac{2pq}{2^{4n/3}}\right) \cdot \\
& \sum_{0\leq t\leq M} \left[\frac{(s')_{2t}}{t!} \cdot \left(1-\frac{4q}{2^{2n/3}}-\frac{2pt}{q^2}-\frac{2\sqrt{q}}{2^{n/3}}\right) \cdot \right. \\
& \left. \prod_{i=0}^{s''-1}\left(1-\frac{p+i}{2^n-(5q+p)-i}\right) \cdot \frac{(2^n-q-p-\alpha-\beta-3t-r)_{s''}}{(2^n-p-\alpha-\beta)_{q'+t+r'}}\right] \\
\geq\; & \underbrace{\frac{2^{n(\alpha+\beta)}}{(2^n-p)_{\alpha+\beta}}}_{\geq 1} \cdot \underbrace{\frac{(2^n)^{r'-s''}}{(2^n-p-\alpha-\beta-q')_{r'-s''}}}_{\geq 1} \cdot \\
& \underbrace{\frac{2^{nq'}}{(2^n-p-\alpha-\beta)_{q'}} \cdot \prod_{i=0}^{s''-1}\left(1-\frac{p+i}{2^n-(5q+p)-i}\right)}_{\text{D.1}} \\
& \cdot \left(1-\frac{6q^2}{2^{4n/3}}-\frac{2pq}{2^{4n/3}}\right) \cdot \left(1-\frac{4q}{2^{2n/3}}-\frac{2pt}{q^2}-\frac{2\sqrt{q}}{2^{n/3}}\right) \\
& \cdot \underbrace{\left[\sum_{0\leq t\leq M} \frac{(s')_{2t}\cdot(2^n-q-p-\alpha-\beta-3t-r)_{s''}}{t!\cdot(n-p-\alpha-\beta-q'-r'+s'')_{s''+t}}\right]}_{\text{D.2}}.
\end{aligned} \tag{4.50}$$

Computing D.1.

$$\begin{aligned}
\text{D.1} \;=\; & \frac{2^{nq'}}{(2^n-p-\alpha-\beta)_{q'}} \cdot \prod_{i=0}^{s''-1}\left(1-\frac{p+i}{2^n-(5q+p)-i}\right) \\
\overset{(1)}{\geq}\; & \prod_{i=0}^{q'-1}\left(1+\frac{p+i}{2^n-p-i}\right) \cdot \prod_{i=0}^{s''-1}\left(1-\frac{p+i}{2^n-(5q+p)-i}\right) \\
\overset{(2)}{\geq}\; & \prod_{i=0}^{q'-1}\left[\left(1+\frac{p+i}{2^n-p-i}\right) \cdot \left(1-\frac{p+i}{2^n-i-5q-p}\right)\right] \\
\geq\; & \prod_{i=0}^{q'-1}\left[1-\left(\frac{5q(p+i)+p^2+2pi+i^2}{(2^n-p-i)(2^n-p-i-5q)}\right)\right] \\
\overset{(3)}{\geq}\; & \prod_{i=0}^{q'-1}\left[1-\frac{20q(p+i)}{2^{2n}}-\frac{4p^2}{2^{2n}}-\frac{8pi}{2^{2n}}-\frac{4i^2}{2^{2n}}\right] \\
\geq\; & \left(1-\frac{20pq^2}{2^{2n}}-\frac{20q^3}{2^{2n}}-\frac{4p^2q}{2^{2n}}-\frac{8pq^2}{2^{2n}}-\frac{4q^3}{2^{2n}}\right) \\
\geq\; & \left(1-\frac{28pq^2}{2^{2n}}-\frac{4p^2q}{2^{2n}}-\frac{24q^3}{2^{2n}}\right),
\end{aligned} \tag{4.51}$$

where (1) holds as $2^n - p - \alpha - \beta \leq 2^n - p$. (2) holds as $s'' \leq q'$ and (3) holds as $p + i \leq 2^n/2, p + i + 5q \leq 2^n/2$.

Computing D.2.

$$\begin{aligned}
\mathsf{D.2} &= \sum_{0\leq t\leq M} \frac{(s')_{2t} \cdot (2^n - q - p - \alpha - \beta - 3t - r)_{s''}}{t! \cdot (2^n - p - \alpha - \beta - q' - r' + s'')_{s''+t}} \\
&\overset{(1)}{\geq} \sum_{0\leq t\leq M} \underbrace{\frac{(s')_{2t}}{(s')_t (s')_t}}_{\mathsf{E.1}} \cdot \underbrace{\frac{(s')_t (s')_t}{t!} \cdot \frac{(2^n - q - p - \alpha - \beta - s')_{s'-t}}{(2^n - q - p - \alpha - \beta)_{s'}}}_{\mathsf{E.2}} \\
&\quad \cdot \underbrace{\frac{(2^n - q - p - \alpha - \beta)_{s'}}{(2^n - q - p - \alpha - \beta - s')_{s'-t}} \cdot \frac{(2^n - q - p - \alpha - \beta - 3t - r)_{s''}}{(2^n - p - \alpha - \beta - q' - r + s')_{s''+t}}}_{\mathsf{E.3}},
\end{aligned} \tag{4.52}$$

where (1) holds as $r' = r - 2t$ and $s'' = s' - 2t$. Now, we individually bound E.1 and E.3 as follows:

Computing E.1.

$$\mathsf{E.1} = \frac{(s')_{2t}}{(s')_t (s')_t} \geq \frac{(s' - 2M)^{2t}}{(s')^{2t}} \geq 1 - \frac{4tM}{s'} \tag{4.53}$$

$$\overset{(2)}{\geq} 1 - \frac{4M^2}{q - M} \overset{(3)}{\geq} 1 - \frac{8M^2}{q} \overset{(4)}{\geq} \left(1 - \frac{8q}{2^{2n/3}}\right), \tag{4.54}$$

where (2) follows as $s' \geq q - M$, (3) follows as $q - 3M \leq q/2$ and finally (4) follows as $M = q/2^{n/3}$.

Computing E.3.

$$\begin{aligned}
\mathsf{E.3} &= \frac{(2^n - q - p - \alpha - \beta)_{s'}}{(2^n - q - p - \alpha - \beta - s')_{s'-t}} \cdot \frac{(2^n - q - p - \alpha - \beta - 3t - r)_{s''}}{(2^n - p - \alpha - \beta - q' - r + s')_{s''+t}} \\
&\overset{(5)}{=} \frac{(2^n - q - p - \alpha - \beta)_{s''+2t}}{(2^n - q - p - \alpha - \beta - s')_{s''+t}} \cdot \frac{(2^n - q - p - \alpha - \beta - 3t - r)_{s''}}{(2^n - p - \alpha - \beta - q' - r + s')_{s''+t}} \\
&= \frac{(2^n - q - p - \alpha - \beta)_{s''+t}}{(2^n - q - p - r + s')_{s''+t}} \cdot \frac{(2^n - q - p - \alpha - \beta - s'' - t)_t}{(2^n - q - p - \alpha - \beta - s')_t} \cdot \frac{(2^n - q - p - \alpha - \beta - 3t - r)_{s''}}{(2^n - q - p - \alpha - \beta - s' - t)_{s''}} \\
&\overset{(6)}{=} \underbrace{\frac{(2^n - q - p - \alpha - \beta)_{s''+t}}{(2^n - q - p - r + s')_{s''+t}}}_{\mathsf{E.3.1}} \cdot \underbrace{\frac{(2^n - q - p - \alpha - \beta - s' + t)_t}{(2^n - q - p - \alpha - \beta - s')_t}}_{\geq 1} \cdot \underbrace{\frac{(2^n - q - p - \alpha - \beta - 3t - r)_{s''}}{(2^n - q - p - \alpha - \beta - s' - t)_{s''}}}_{\mathsf{E.3.2}},
\end{aligned}$$

where (5) and (6) follows as $s'' = s' - 2t$. Now, we individually bound E.3.1 and E.3.2 as follows:

Computing E.3.1.

$$\begin{aligned}
\mathsf{E.3.1} &= \frac{(2^n-q-p-\alpha-\beta)_{s''+t}}{(2^n-q-p-r+s')_{s''+t}} = \prod_{i=0}^{s''+t-1} \frac{2^n-q-p-\alpha-\beta-i}{2^n-q-p-r+s'-i} \\
&= \prod_{i=0}^{s''+t-1} \left(1-\frac{\alpha+\beta-r+s'}{2^n-q-p-i-r+s'}\right) \\
&= \prod_{i=0}^{s''+t-1} \left(1-\frac{\alpha+\beta-(r-s')}{2^n-q-p-i-r+s'}\right) \\
&\overset{(7)}{\geq} \prod_{i=0}^{s''+t-1} \left(1-\frac{2(\alpha+\beta)}{2^n}\right) \\
&\overset{(8)}{\geq} \prod_{i=0}^{s''+t-1} \left(1-\frac{2\sqrt{q}}{2^n}\right) \overset{(9)}{\geq} \left(1-\frac{2q^{3/2}}{2^n}\right), \qquad (4.55)
\end{aligned}$$

where (7) follows as $q+p+i+r-s' \leq 2^n/2$, (8) follows as $\alpha, \beta \leq q$ and (9) follows as $s''+t \leq q$.

Computing E.3.2.

$$\begin{aligned}
\mathsf{E.3.2} &= \frac{(2^n-q-p-\alpha-\beta-3t-r)_{s''}}{(2^n-q-p-\alpha-\beta-s'-t)_{s''}} \\
&= \prod_{i=0}^{s''-1} \left(1-\frac{2t+r-s'}{2^n-q-p-\alpha-\beta-s'-t-i}\right) \\
&\overset{(10)}{\geq} \left(1-\frac{2s''(2M+r-s')}{2^n}\right) \\
&\overset{(11)}{\geq} \left(1-\frac{6s''M}{2^n}\right) \overset{(12)}{\geq} \left(1-\frac{6q^2}{2^{4n/3}}\right), \qquad (4.56)
\end{aligned}$$

where (10) follows as $t \leq M$ and $q+p+\alpha+\beta+s'+t+i \leq 2^n/2$. (11) follows as $r-s' \leq M$ and (12) follows as $s'' \leq q$ and $M = q/2^{n/3}$. Therefore, from Eqn.s (4.55) and (4.56) and using the inequality $(1-a)(1-b) \geq (1-a-b)$ for $a, b \leq 1$, we have

$$\mathsf{E.3} \geq \left(1-\frac{2q^{3/2}}{2^n}-\frac{6q^2}{2^{4n/3}}\right). \qquad (4.57)$$

Now combining Eqn.s (4.52), (4.53) and (4.57) and using the inequality $(1-a)(1-b) \geq (1-a-b)$ for $a, b \leq 1$, we have

$$\mathsf{D.2} \geq \left(1-\frac{8q}{2^{2n/3}}-\frac{2q^{3/2}}{2^n}-\frac{6q^2}{2^{4n/3}}\right) \cdot \sum_{0 \leq t \leq M} \mathsf{E.2}. \qquad (4.58)$$

Note that for a fixed t, E.2 $= \mathbf{Hyp}(t)_{N',s',s'}$, where $N' = 2^n - q - p - \alpha - \beta$ with parameters N', s' and s'. It is a well-known result that the expectation of the hypergeometric distribution with parameters N', s', s' is s'^2/N'. Therefore, we have

$$\mathsf{D.2} \geq \left(1 - \frac{8q}{2^{2n/3}} - \frac{2q^{3/2}}{2^n} - \frac{6q^2}{2^{4n/3}}\right) \cdot \sum_{0 \leq t \leq M} \mathbf{Hyp}(t)_{N',s',s'}.$$

Now, using Markov's inequality, we have

$$\sum_{t > M} \mathbf{Hyp}(t)_{N',s',s'} \leq \frac{s'^2}{q2^{2n/3}} \leq \frac{q}{2^{2n/3}}, \tag{4.59}$$

where the first inequality appears due to the Markov's inequality and the second inequality follows as $s' \leq q$ and $M = q/2^{n/3}$. Therefore, from Eqn.s (4.58) and (4.59) and by the inequality $(1-a)(1-b) \geq (1-a-b)$ for $a, b \leq 1$, we have

$$\mathsf{D.2} \geq \left(1 - \frac{9q}{2^{2n/3}} - \frac{2q^{3/2}}{2^n} - \frac{6q^2}{2^{4n/3}}\right). \tag{4.60}$$

Finally, combining Eqn.s (4.50), (4.51) and (4.60) and using the inequality $(1-a)(1-b) \geq (1-a-b)$ for $a, b \leq 1$, we have

$$\rho(\tau) \geq 1 - \underbrace{\left(\frac{12q^2}{2^{4n/3}} + \frac{2pq}{2^{4n/3}} + \frac{13q}{2^{2n/3}} + \frac{2pt}{q^2} + \frac{2\sqrt{q}}{2^{n/3}} + \frac{28pq^2}{2^{2n}} + \frac{4p^2q}{2^{2n}} + \frac{24q^3}{2^{2n}} + \frac{2q^{3/2}}{2^n}\right)}_{\phi(\tau)}.$$

This completes the proof of Lemma 12. Now it only remains to compute the expectation of $\phi(\tau)$ as follows:

Computing the Expectation. We now compute the expectation of $\phi(\tau)$ over the randomness of the permutation π as follows:

$$\begin{aligned} \mathbf{E}_\pi[\phi(\tau)] &= \mathbf{E}_\pi\left[\tfrac{2pt}{q^2}\right] + \left(\tfrac{12q^2}{2^{4n/3}} + \tfrac{2pq}{2^{4n/3}} + \tfrac{13q}{2^{2n/3}} + \tfrac{2\sqrt{q}}{2^{n/3}} + \tfrac{28pq^2}{2^{2n}} + \tfrac{4p^2q}{2^{2n}} + \tfrac{24q^3}{2^{2n}} + \tfrac{2q^{3/2}}{2^n}\right) \\ &= \tfrac{2p}{q^2}\mathbf{E}_\pi[t] + \left(\tfrac{12q^2}{2^{4n/3}} + \tfrac{2pq}{2^{4n/3}} + \tfrac{13q}{2^{2n/3}} + \tfrac{2\sqrt{q}}{2^{n/3}} + \tfrac{28pq^2}{2^{2n}} + \tfrac{4p^2q}{2^{2n}} + \tfrac{24q^3}{2^{2n}} + \tfrac{2q^{3/2}}{2^n}\right). \end{aligned}$$

Now, it remains to compute the expectation of the random variable t over the randomness of the permutation π. Let t_i be the indicator random variable that takes the value 1 if $\pi(M_i \oplus \mathbf{k}_1) \oplus M_i \oplus \mathbf{k}_2 \in \mathcal{X}$, for $1 \leq i \leq M$. Therefore, it is easy to see that

$$\Pr[t_i = 1] = \Pr[\pi(M_i \oplus \mathbf{k}_1) \oplus M_i \oplus \mathbf{k}_2 \in \mathcal{X}] \leq \frac{q'}{2^n}.$$

Since $t = t_1 + \ldots + t_M$, due to the linearity of expectation, we have

$$\mathbf{E}_\pi[t] = \sum_{i=1}^{M} \mathbf{E}_\pi[t_i] = \sum_{i=1}^{M} \Pr[t_i = 1] \leq \frac{q'M}{2^n} \leq \frac{q^2}{2^{4n/3}}, \tag{4.61}$$

where the last inequality appears as $M = q/2^{n/3}$ and $q' \leq q$. Therefore, from Eqn. (4.61), we have

$$\mathbf{E}_\pi[\phi(\tau)] \leq \frac{12q^2}{2^{4n/3}} + \frac{2pq}{2^{4n/3}} + \frac{13q}{2^{2n/3}} + \frac{2\sqrt{q}}{2^{n/3}} + \frac{28pq^2}{2^{2n}} + \frac{4p^2q}{2^{2n}} + \frac{24q^3}{2^{2n}} + \frac{2q^{3/2}}{2^n} + \frac{2p}{2^{4n/3}}. \tag{4.62}$$

The result of Theorem 9 follows from the expectation method (1.1), Lemma 11 and Eqn. (4.62) which concludes the proof of the security result.

4.4. Summary

This chapter has proposed an inverse-free single permutation-based beyond the birthday bound secure PRF that requires $2n$ bit keys. The same goal may also be achieved using the single permutation-based tweakable Even-Mansour cipher [67]. However, this solution comes at the cost of implementing the costly universal hash functions. Furthermore, parallel modes like nEHtM$_p$, SoEM22 or DS-SoEM also achieve beyond the birthday bound PRF security, but again the former requires implementation of a universal hash function, SoEM22 requires two independent permutations and DS-SoEM takes an $(n-1)$-bit input. It will be interesting to study the sequential design of an inverse-free single permutation-based PRF with only an n bit key. We believe that pEDM can be turned into a single permutation-oriented beyond the birthday bound secure nonce based MAC by XORing an almost-XOR universal hash function in between the two permutation calls (in a similar vein as the EWCDM [56]).

5. Tight Security Analysis of the Public Permutation-Based PMAC_Plus

Abstract

Yasuda proposed a variable input-length PRF in CRYPTO 2011, called PMAC_Plus, based on an n-bit block cipher. PMAC_Plus is a rate-1 construction and inherits the well-known PMAC parallel network with a low additional cost. However, unlike PMAC, PMAC_Plus is secure roughly up to $2^{2n/3}$ queries. Zhang et al. proposed 3kf9 in ASIACRYPT 2012, Naito proposed LightMAC_Plus in ASIACRYPT 2017, and Iwata et al. proposed GCM-SIV2 in FSE 2017 – all of them secure up to around $2^{2n/3}$ queries. Their structural designs and corresponding security proofs were unified by Datta et al. in their framework *Double-block Hash-then-Sum* (DbHtS). Leurent et al. in CRYPTO 2018 and then Lee et al. in EUROCRYPT 2020 established a tight security bound of $2^{3n/4}$ on DbHtS. That PMAC_Plus provides security for roughly up to $2^{3n/4}$ queries is a consequence of this result. In this chapter, we propose a public permutation-based variable input-length PRF called pPMAC_Plus. We show that pPMAC_Plus is secure against all adversaries that make at most $2^{2n/3}$ queries. We also show that the bound is essentially tight. It is of note here that instantiation of each block cipher of pPMAC_Plus with the two-round iterated Even-Mansour cipher can yield a beyond the birthday bound secure PRF based on public permutations. Altogether, the solution incurs $(2\ell + 4)$ permutation calls, whereas our proposal requires only $(\ell + 2)$ permutation calls, ℓ being the maximum number of message blocks.

Keywords – PMAC_Plus, Public Permutation, PRF from PRP, Sum-Capture Lemma, Coefficients-H Technique

5.1. Introduction

Beyond the Birthday Bound PRFs. Over the years, there have been many proposals of beyond the birthday bound-secure PRFs. In [135], Yasuda proposed SUM-ECBC, a beyond the birthday bound-secure PRF. SUM-ECBC is a rate-1/2 sequential mode of construction with four block cipher keys that offers about $2n/3$-bit security. In [134], he proposed another beyond the birthday bound-secure PRF, called PMAC_Plus that also offers about $2n/3$-bit security. However, unlike SUM-ECBC, it is a rate-1 and parallel mode of construction with three block cipher keys. In the following year, Zhang et al. [136] proposed another candidate for a beyond the birthday bound-secure PRF, called 3kf9, which is a rate-1 sequential mode of construction with three block cipher keys and offers $2n/3$-bit security. Following these works, Naito proposed LightMAC_Plus in [114], the first beyond the birthday bound-secure PRF which is proven to have an ℓ independent beyond the birthday bound and hence effectively offers a better security than that of all the earlier three proposals. Datta et al. [63] proposed a single-keyed variant of the PMAC_Plus that offers a better security bound than that of PMAC_Plus. In [61], Datta et al. unified the design of all four beyond the birthday bound-secure PRFs (i.e., SUM-ECBC, PMAC_Plus, 3kf9, LightMAC_Plus) and gave a common security proof for all of them. They also proposed a two-keyed version of SUM-ECBC, PMAC_Plus, 3kf9, LightMAC_Plus and have shown that all of them achieve roughly $2n/3$-bit security. Interestingly, all these constructions share a similar structural design and offer the same level of security. All this motivated the unification of these designs and the provision of a common security proof for all of them in [61].

Double block-Hash-then-Sum. DbHtS [61] is a generic methodology for designing block cipher-based beyond the birthday bound-secure PRFs. It is a composition of two constituent elements: (i) a double block hash function that outputs a $2n$-bit hash value of the input message and (ii) a sum function used in the finalization phase that generates the final tag by XORing the encryption (via two independent block ciphers) of two n-bit hash values. The authors have shown that if the cover-free advantage (refers to the probability that for a triplet of messages M_i, M_j, M_k, the first halves (i.e. the leftmost n bits) of the hash values of M_i and M_j collide and the second halves (i.e. the rightmost n bits) of the hash values of M_i and M_k collide) and the block-wise universal advantage (refers to the probability of collision of either of the halves of the hash values of any pair of distinct messages) of the underlying double-block hash function is sufficiently low, then DbHtS is secure up to $2^{2n/3}$ adversarial queries. The authors have also shown the applicability of their result by instantiating the two-keyed variants of SUM-ECBC, PMAC_Plus, 3kf9, LightMAC_Plus and have proven $2n/3$-bit security for all of them. Using the generic result, authors have also improved the

Table 5.1.: Comparison table for permutation-based PRFs and MACs. n denotes the state size of the permutation, which we also call block size. The last row describes pPMAC_Plus, proposed in this work. The second and third columns, i.e. $\#(\pi)$ and $\#(k)$, respectively show the number of permutations and the number of keys required by the construction. i/p (resp. o/p) size denotes the bit size of the input (resp. output) to the construction. Constructions with a dagger symbol use keyed hash functions and the number of keys they require includes the hash key as well; they also take nonce as one of their inputs. Security bounds mentioned in green denote lower bounds for which a matching upper bound isn't yet proven, while blue denotes tight bounds and red denotes upper bounds.

Constructions	$\#(\pi)$	$\#(k)$	(i/p, o/p) size	Security
SoEM1 [53]	1	2	(n, n)	**n/2**
SoEM21 [53]	2	1	(n, n)	**n/2**
SoEM22 [53]	2	2	(n, n)	**2n/3**
SoKAC1 [53]	1	2	(n, n)	**2n/3** [47]
SoKAC21 [53]	2	1	(n, n)	**n/2** [118]
pEDM [73]	1	2	(n, n)	**2n/3** [73]
PDMMAC [47]	1	1	(n, n)	**2n/3**
DS-SoEM [28]	1	2	$(n-1, n)$	**2n/3**
CENCPP* [28]	$w+1$	2	(n, wn)	$2n/3-\log(w^2)$
DS-CENCPP* [28]	1	2	$(n-\log(w+1), wn)$	$2n/3-\log(w^4)$
(†) nEHtM$_p$ [70]	1	2	$(n-1+\ell n, n)$	$2n/3$
(†) PDM*MAC [47]	1	2	$(n+\ell n, n)$	$2n/3$
(†) 1K-PDM*MAC [47]	1	1	$(n+\ell n, n)$	$2n/3$
Chaskey [112]	1	1	$(\ell n, t)$	$n/2+2^{-t}$
pPMAC_Plus	1	3	$(\ell n, n)$	$2n/3$

security bound for SUM-ECBC and PMAC_Plus.

In [95], Leurent et al. have shown attacks on all these constructions with $2^{3n/4}$-query complexity. Recently, Kim et al. [92] have proven $3n/4$-bit security of DbHtS and hence established the tightness of the bound for SUM-ECBC, PMAC_Plus, 3kf9 and LightMAC_Plus.

Permutation-based cryptography. A block cipher is generally designed to be efficient in evaluating the input in both forward and backward directions. However, a closer inspection reveals that all the block cipher-based PRFs discussed so far do not require the inverse mapping of the block ciphers. Thus, a block cipher is an over-engineered primitive for block cipher-based PRF constructions that do not require the inverse function of their underlying primitives.

Concurrently with block ciphers, cryptographic permutations have evolved as useful primitives. The primary feature of a cryptographic permutation

is that it does not use any key and hence does not require any separate processing for it. The use of cryptographic permutations gained popularity during the SHA-3 competition [127] as several submitted candidates in the competition were based on this type of primitive. The selection of the permutation-based Keccak sponge function as the SHA-3 standard has further boosted the level of confidence of the community in using this primitive. Today, permutation-based sponge-based constructions have become a successful and full-fledged alternative to block cipher-based modes. In fact, in the first round of the ongoing NIST lightweight competition [119], 24 out of the 57 submitted constructions are based on cryptographic permutations, and out of these 24, 16 permutation-based proposals have qualified for round 2. These statistics depict the wide adoption of permutation-based designs [46, 20, 26, 48, 59, 64] in the community. A long line of research has also been carried out in the study of designing block ciphers and tweakable block ciphers out of public random permutations. Iterated Even Mansour (IEM) [51] and Tweakable Even-Mansour (TEM) [54] ciphers are notable approaches in this direction.

PRFs Built from Public Permutations. Variable input-length PRFs built using public permutations mostly follow sponge-type constructions. Inherent drawbacks of such designs are that (i) they do not use the full size of the permutation for guaranteeing security and (ii) they attain only birthday bound security in the size of their capacity c, (except Bettle [46], whose security bound is roughly the size of its capacity). It is obvious that the sponge-type designs offering $c/2$-bit security are good in practice when they are instantiated with large permutations such as Keccak [23]. However, just like large block ciphers, large permutations are not suitable for a resource-constrained environment. In such a scenario, lightweight permutations such as SPONGENT [40] and PHOTON [80] (whose state sizes go as low as 88 and 100 bits respectively) are preferred over large ones. The use of these lightweight permutations in birthday bound-secure sponge constructions offers a practically inadequate security. Thus, to utilize lightweight permutations in practice, the natural choice would be to design a beyond the birthday bound-secure mode. In this regard, Chen et al. [53] have proposed two instances of public permutation-based pseudo-random functions, namely SoEM22 and SoKAC1. Both of them map an n-bit input to an n-bit output and offer beyond the birthday bound security with respect to the state size of the permutation. However, Nandi [118] has shown a birthday bound attack on SoKAC1 and hence invalidated its beyond the birthday bound security claim. Bhattacharjee et al. [28] have shown a public permutation-based fixed input-length to variable output-length PRF called XORPP* and its domain-separated variant called DS-XORPP*. Both of these constructions are built with a CENC [90]-style design and both of them have

$2n/3$-bit security [28]. Chakraborti et al. [47] have proposed a beyond the birthday bound-secure public permutation-based fixed input-length PRF, called PDMMAC, a variable input-length PRF PDM*MAC and its single-keyed variant. Recently, Dutta et al. [73] have proposed another candidate for public permutation-based PRFs, called pEDM, and have shown a tight $2n/3$-bit security. This line of research has been further extended in [69] by Dutta and Nandi, where they have proposed a beyond the birthday bound-secure nonce-based MAC build on top of public permutations.

5.1.1. Our Contribution

Given the state of the art in permutation-based cryptography, it is natural to wonder whether we can design a variable input-length PRF based on some lower-level primitive like public permutations instead of block ciphers that offer beyond the birthday bound security. In this chapter, we provide an answer in the positive. To this end, we propose a permutation-based PMAC_Plus construction, which we call pPMAC_Plus. The permutation-based variant of PMAC_Plus is exactly similar to PMAC_Plus with the following exception: in the block cipher-based PMAC_Plus construction, the output t is defined as follows:

$$t = \mathsf{E}_{k_1}(\Sigma) \oplus \mathsf{E}_{k_2}(\Theta),$$

where (Σ, Θ) is the $2n$-bit output value of the underlying double-block hash function PMAC_Plus-Hash. For pPMAC_Plus, we mask Σ and Θ with k_2 and follow by a domain separation through $\mathsf{chop}_{\mathsf{LSB}}(\cdot)\|0$, $\mathsf{chop}_{\mathsf{LSB}}(\cdot)\|1$, respectively. Next, we replace both $\mathsf{E}_{k_1}(\cdot)$ and $\mathsf{E}_{k_2}(\cdot)$ by an n-bit public random permutation $\pi(\cdot)$ (where k_1 and k_2 are two independently sampled block cipher keys). While PMAC_Plus-Hash is built from a block cipher E_k (independent from E_{k_1} and E_{k_2}), E_k is also replaced by π in pPMAC_Plus-Hash, the α^{th} block of the input message masked with the string $(2^{\alpha}k_0 \oplus 2^{2\alpha}k_1)$, where k_0, k_1 and k_2 are three independently sampled n-bit strings.

One can directly replace each block cipher of PMAC_Plus with the two-round iterated Even-Mansour cipher [52] or Mennnink's SoEM22 construction [53] and obtain a beyond the birthday bound secure PRF based on public permutations. While both the solutions incur $(2\ell + 4)$ permutation calls, our proposal requires only $(\ell + 2)$ permutation calls, where ℓ is the maximum number of message blocks. Furthermore, unlike PMAC_Plus which has a tight $3n/4$-bit security, we have shown that pPMAC_Plus achieves a tight security bound of the order of $2^{2n/3}$.

5.2. pPMAC_Plus: A Public Permutation-Based BBB Secure MAC

In this section, we propose pPMAC_Plus, a public permutation-based beyond the birthday bound secure MAC. It takes an n-bit independent public permutation π and three independent n-bit keys k_0, k_1 and k_2. For processing a message $M \in \{0,1\}^*$, the padding function $\mathsf{pad} : \{0,1\}^* \to (\{0,1\}^n)^+$ is applied on M that parses M into l blocks $(M[1], M[2], \ldots, M[l])$ by concatenating $10*$ to the right so that for each $i \in [l-1]$, $|M[i]| = n$ and $1 \leq |M[l]| \leq n$.

$\textsc{pPMAC\_Plus}^{\pi}_{k_0,k_1,k_2}(M)$

1 : $(\Sigma,\Theta) \leftarrow \mathsf{pPMAC\_Plus\text{-}Hash}(M)$.
2 : **return** $T \leftarrow \pi(\Sigma) \oplus \pi(\Theta)$.

$\textsc{pPMAC\_Plus-Hash}^{\pi}_{k_0,k_1,k_2}(M)$

1 : $(M[1], M[2], \ldots, M[\ell]) \leftarrow \mathsf{pad}(M)$.
2 : $(\Sigma, \Theta) \leftarrow (0^n, 0^n)$.
3 : for $(i = 1; i \leq \ell; i++)$
4 : $\quad X \leftarrow 2^i k_0 \oplus 2^{2i} k_1 \oplus M[i]$.
5 : $\quad Y_i \leftarrow \pi_0(X)$.
6 : $\quad \Sigma \leftarrow \Sigma \oplus Y_i, \Theta \leftarrow \Theta \oplus 2^{l-i+1} \cdot Y_i$.
7 : **return** $(\mathsf{chop}_{\mathsf{LSB}}(\Sigma \oplus k_2) \| 0, \mathsf{chop}_{\mathsf{LSB}}(\Theta \oplus k_2) \| 1)$.

Figure 5.1.: pPMAC_Plus is depicted on the left, while a permutation-based DbH function of pPMAC_Plus is shown on the right.

For each $\alpha \in [l]$, the message block $M[\alpha]$ of M is masked with $2^{\alpha} k_0 \oplus 2^{2\alpha} k_1$ before passing it through the permutation π. Output blocks of the permutation are then XORed together, followed by masking with another key k_2 to generate an n-bit value Σ. Each output block of the hash permutation instances is simply XORed in one case and multiplied by 2 before XORing in another, and both are masked with the key k_2 to generate output values Σ and Θ, respectively. Finally, $\mathsf{chop}_{\mathsf{LSB}}(\Sigma \oplus k_2) \| 0$ and $\mathsf{chop}_{\mathsf{LSB}}(\Theta \oplus k_2) \| 1$ are passed through two copies of the same permutation π (as used in the hash function) and the XOR of their outputs produces the MAC T. An algorithmic description of pPMAC_Plus is given in Fig. 5.1, and a pictorial illustration in Fig. 5.2.

Remark 3. *Note that the structural design of pPMAC_Plus is similar to that of PMAC_Plus. The only difference of the former with the latter is that PMAC_Plus uses a block cipher E with three independent block cipher keys, whereas pPMAC_Plus replaces E by an n-bit random permutation π along with some masking elements and domain separation. It is easy to see that directly replacing E in PMAC_Plus*

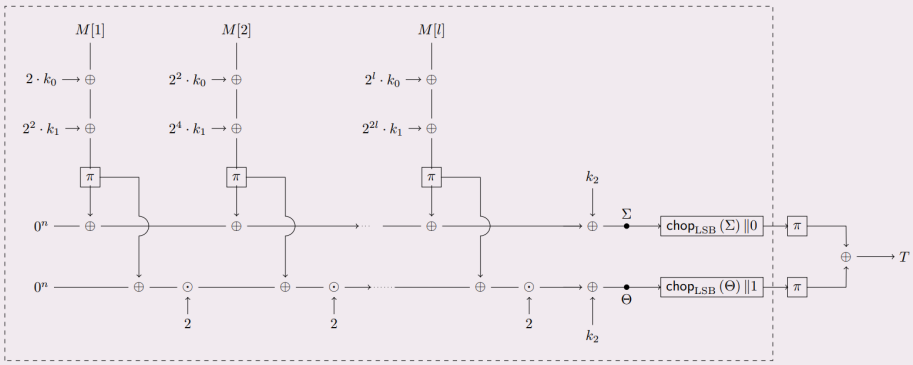


Figure 5.2.: A message with ℓ blocks (after padding) is input into $\mathsf{pPMAC\_Plus} - \mathsf{Hash}^{\pi}_{k_0,k_1,k_2}$, which produces outputs Σ and Θ. These outputs are masked with a random key k_2 before going through a domain-separating function, and the final pPMAC_Plus output is produced by passing these two values through two permutations and adding the resultants.

by the-two round iterated Even-Mansour cipher or SoEM22 construction [52] immediately leads the security of the resulting construction to beyond the birthday bound. However, both solutions pay a price for invoking the underlying permutation twice to process a single message block. Therefore, processing an ℓ-block message requires $2\ell + 4$ permutation calls in the former approach and only $\ell + 2$ permutation calls in ours.

5.2.1. Security of pPMAC_Plus

In this section, we state that pPMAC_Plus is secure against any information theoretic adversary that makes roughly $2^{2n/3}$ online and offline queries.

Theorem 10 (Security of pPMAC_Plus). *Let $\mathcal{M}$ be a non-empty finite set and π a uniformly sampled n-bit public permutation. Let A be any distinguisher that makes at most q construction queries and at most p primitive queries, and runs for at most time t. Then*

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{pPMAC\_Plus}}(\mathsf{A}) \leq \frac{2\sqrt{3nqp_1p_2}+4}{2^n} + \frac{q^3(5\ell^3+3\ell^2+8\ell+4)}{2^{2n}} + \frac{6qp^2l^2+2q^2pl+2q^2l}{2^{2n}} + \frac{4q^3+45q^2p+20qp^2+5q^2}{2^{2n}}.$$

The PRF security of pPMAC_Plus is roughly at most $2^{2n/3}$ when $q \approx p$.

5.3. A Key-Recovery Attack on pPMAC_Plus

In this section, we show a matching key-recovery attack on pPMAC_Plus with a total of $2^{2n/3+1}$ of each of construction and primitive queries. We refer the readers to the full attack in Fig. 5.3.

Backward attack. The attack proceeds by first making $2^{2n/3}$ construction queries of two-block messages $M_i[1]\|M_i[2]$ for $i \in [2^{2n/3}]$, and collects the responses T_i. Next, it makes two sets of $2^{2n/3}$ offline forward queries – one with least significant bit (LSB) 0 and the other with LSB 1 – to the primitive permutation π, and collects their corresponding responses in lists $\mathcal{L}_0$ and $\mathcal{L}_1$, respectively. All these $2p = 2^{2n/3+1}$ forward queries and their responses are also collected into a list of pairs $\mathcal{L} = \{(\tilde{x}_1, \tilde{y}_1), (\tilde{x}_2, \tilde{y}_2), \ldots, (\tilde{x}_a, \tilde{y}_a), \ldots, (\tilde{x}_{2p}, \tilde{y}_{2p})\}$. A check of pairs $(\tilde{v}_b, \tilde{z}_c) \in (\mathcal{L}_0 \times \mathcal{L}_1)$ such that $\tilde{v}_b \oplus \tilde{z}_c = T_i$ provides triples (i, b, c) collected in a set $\mathcal{S}_1$. Computing pairs $(\hat{\Sigma}, \hat{\Theta})$ for all pairs of second-coordinates $(\tilde{y}_{a_1}, \tilde{y}_{a_2}) \in \mathcal{L}\big|_2 \times \mathcal{L}\big|_2$ helps filter the elements of $\mathcal{S}_1$ by checking whether

$$\hat{\Sigma} \oplus \tilde{u}_b = \hat{\Theta} \oplus \tilde{w}_c,$$

where $\tilde{u}_b$ and $\tilde{w}_c$ are the preimages of $\tilde{v}_b$ and $\tilde{z}_c$ respectively. If this check passes, then the attack computes a candidate key $\hat{k}_2$, stores the corresponding triple in $\mathcal{S}_2$ and then for all elements of $\mathcal{S}_2$, computes a pair of candidate keys $(\hat{k}_0, \hat{k}_1)$.

Removing false positives. In order to remove the false positive keys from the set of candidates, the attack makes another $2^{2n/3}$ construction queries with messages of two blocks $M_i'[1]\|M_i'[2]$, where $M_i'[1] = (M_i[1] \oplus 1)$ and $M_i'[2] = M_i[2]$, and collects their corresponding responses T_i' for $i \in [2^{2n/3}]$. Next, it evaluates pPMAC_Plus on messages $M_i'[1]\|M_i'[2]$, $i \in [2^{2n/3}]$, with the candidate key-triple $\left(\hat{k}_0, \hat{k}_1, \hat{k}_2\right)$. If the computed values match with the received responses T_i', then this triple of keys $(\hat{k}_0, \hat{k}_1, \hat{k}_2)$ stays in the candidate key-list, otherwise, it is removed. We show that the true key belongs to the set of potential candidate keys with a high probability and that the size of the set of the candidate keys is not very large. We have thus described a deterministic adversary A that recovers the key of pPMAC_Plus by making a total of $2^{2n/3+1}$ construction queries and $2^{2n/3+1}$ primitive queries as shown in Fig. 5.3.

5.3.1. Analysis of the Attack

First observe that for internal values $x_i[1] = M_i[1] \oplus 2 \cdot k_0 \oplus 2^2 \cdot k_1$ and $x_i[2] = M_i[2] \oplus 2^2 \cdot k_0 \oplus 2^4 \cdot k_1$ $(M_i = M_i[1] \| M_i[2])$, $i \in [2^{2n/3}]$,

$$\mathbf{E}\left[\left|\{(i, a_1, a_2) \in [2^{2n/3}] \times [2^{2n/3+1}] \times [2^{2n/3+1}]\}\right| : (x_i[1] = \tilde{x}_{a_1}) \wedge (x_i[2] = \tilde{x}_{a_2})\right] = \mathcal{O}(1).$$

Next, for internal values $u_i = y_i[1] \oplus y_i[2] \oplus k_2$ and $w_i = 2^2 \cdot y_i[1] \oplus 2 \cdot y_i[2] \oplus k_2$, $i \in [2^{2n/3}]$,

$$\mathbf{E}\left[\left|\{(b, c) \in [2^{2n/3}] \times [2^{2n/3}]\}\right| : (u_i = \tilde{u}_b) \wedge (w_i = \tilde{w}_c)\right] = \mathcal{O}(1).$$

Thus, bounding the number of queries to the construction and each of the primitives by $\mathcal{O}\left(2^{2n/3}\right)$ ensures the presence of at least one tuple (a_1, a_2, b, c) of primitive query indices that matches with true internal values corresponding to a construction index i with high probability.

The backward attack checks for the validity of the equations induced by the construction. First consider the set $\mathcal{S}_1$. It is computed over sets of sizes q, p and p with a restriction of two conditions on n-bit strings. Therefore, $\mathbf{E}[|\mathcal{S}_1|] = \frac{qp^2}{2^n}$. Similarly, $\mathbf{E}[|\mathcal{S}_2|] = \frac{qp^4}{2^{2n}}$. Note here that only the indices b, c that appear in tuples $(i, b, c) \in \mathcal{S}_1$ are considered for the check in step 7 of the backward attack, and the corresponding construction query-index i is used next for computing guess values $(\hat{k}_0, \hat{k}_1)$ of the key-pair. Observe that the probability depends on the sampling of values $\tilde{y}_a$, and not on the keys, as the hash computation of the message is not even considered so far.

By the same formula, the expected size of $\mathcal{K}$ is $|\mathcal{S}_2| \times q \times \frac{1}{2^{2n}} = \frac{q^2p^4}{2^{4n}}$. Since q and p both have the same order $\mathcal{O}\left(2^{2n/3}\right)$, $\mathbf{E}[|\mathcal{K}|]$ is $\mathcal{O}(1)$ when $q = \mathcal{O}\left(2^{2n/3}\right)$. Finally, since the true key is in $\mathcal{K}$ with very high probability due to the choice of lengths of the query-lists, the true key (k_0, k_1, k_2) must belong to $\mathcal{K}$ with very high probability. This demonstrates that the above is indeed an $\mathcal{O}\left(2^{2n/3}\right)$ attack on pPMAC_Plus.

5.4. Proof of Theorem 10

In this section, we prove Theorem 10. We often denote pPMAC_Plus$[\pi, k_0, k_1, k_2]$ simply by pPMAC_Plus* when the primitives and the underlying keys are understood. We consider any information theoretic deterministic distinguisher A that has access to a triplet of oracles in the real and the ideal worlds: In the real world, it has access to the oracles $\mathbf{O}_{\text{re}} := (\text{pPMAC\_Plus}^*, \pi^+, \pi^-)$,

where π is a uniformly chosen random n-bit permutation and k_0, k_1, k_2 are three independently and uniformly chosen random n-bit keys. In the ideal world, it has access to the oracles $\mathbf{O}_{\mathrm{id}} := (\$, \pi^+, \pi^-)$, where π is again a uniformly chosen n-bit random permutation. Queries to the first oracle in either of the two worlds are called *construction queries* and queries to the remaining oracles are called *primitive queries*. Note that as the primitive π is a permutation, an adversary can make queries in the forward direction, which we call *forward primitive queries*, as well as in the inverse direction, which we call *backward primitive queries*. Throughout the proof, we assume that neither does an adversary A make duplicate or redundant queries nor does it make queries whose responses can be constructed from the previous query-responses. We call such an adversary a **non-trivial adversary**. We also assume that A makes q construction queries and p (forward and backward) primitive queries in either of the two worlds.

Once an adversary has finished making all its queries, the keys k_0, k_1, k_2 in the real world, and corresponding dummy values in the ideal world are released to the adversary. Furthermore, the intermediate values $((x_i[1], x_i[2], \ldots, x_i[l_i]), (y_i[1], y_i[2], \ldots, y_i[l_i]), u_i, v_i, w_i, z_i)$ for each construction query $i \in [q]$ are also released. These values represent the following:

$$\begin{aligned} x_i[\alpha] = M_i \oplus 2^{\alpha} k_0 \oplus 2^{1\alpha} k_1 \; \forall\, \alpha \in [l_i] \quad &, \quad y_i[\alpha] = \pi(x_i[\alpha]) \; \forall\, \alpha \in [l_i] \\ \Sigma_i = y_i[1] \oplus \ldots \oplus y_i[l_i] \quad &, \quad \Theta_i = 2^{l_i} \cdot y_i[1] \oplus \ldots \oplus 2 \cdot y_i[l_i] \\ u_i = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \,\|0 \quad &, \quad v_i = \pi(u_i) \\ w_i = \mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \,\|1 \quad &, \quad z_i = \pi(w_i). \end{aligned} \tag{5.1}$$

5.4.1. An Outline of the Proof

We begin the proof by providing well-defined algorithms for the interaction of an adversary with the real and ideal worlds. While the adversarial interaction with the real world only involves an online phase (since its responses are true to the construction), the ideal world also requires an offline phase for computation of certain output values so as to mimic the real world more closely. These algorithms are detailed in Fig.s 5.4–5.8.

Stage I of the offline phase of the ideal world (Fig. 5.6) lists certain events (which we call *bad events*), for which, the algorithm aborts; the probability of occurrence of these events is computed next. This is the bad event analysis, and can be found in Sect. 5.4.5.

The remaining cases are analyzed in the good transcript analysis (Sect. 5.5) by proving that the ideal interpolation probability is very close to the real interpolation probability. The computation for the real case is quite straightforward, and the bound is given by Eqn. (5.52).

For the ideal world, all queries made by the adversary to the online and offline oracles are indexed according to the respective algorithms. These

indices are first split into those corresponding to free (non-repeating hash output blocks) and single-colliding (collision in exactly one block of the hash output) indices. An equivalence relation is defined according to the collisions of the hash function outputs of the second category of indices so as to classify the output definitions for both inputs. These steps are detailed in Stages II (Fig. 5.7) and III (Fig. 5.8) of the offline phase of the ideal world. This partitions all queried indices into the following sets:

1. $\mathcal{F}$ is the set of indices corresponding to free queries,
2. $\mathcal{I}$ contains the indices corresponding to queries with one hash output block colliding with a primitive query input,
3. $\mathcal{P}^c$ is the set of indices corresponding to queries with one of their hash output blocks colliding with one of the hash-primitive inputs, and
4. $\mathcal{Q}^c$ is the set of indices corresponding to queries with one of their hash output blocks colliding with the corresponding block of the hash output of another query.

5.4.2. Real World and Ideal World

In the real world, when an adversary A makes a construction query with message M to pPMAC_Plus*, it receives the tag $T \leftarrow$ pPMAC_Plus$^*(M)$. In the ideal world, when A makes a construction query with message M to \$, it samples an n-bit tag $T \xleftarrow{\$} \{0,1\}^n$ and returns it to A. In both the worlds, A is allowed to make forward as well as backward primitive queries to π. When A makes the a^{th} forward query $\tilde{x}_a$ to π for $a \in [2p]$, it samples $\tilde{y}_a \xleftarrow{\$} \{0,1\}^n \setminus \{\tilde{y}_1, \ldots, \tilde{y}_{a-1}\}$ and returns it to the adversary. Similarly, for the a^{th} backward query $\tilde{y}_a$ to π, it returns $\tilde{x}_a \xleftarrow{\$} \{0,1\}^n \setminus \{\tilde{x}_1, \ldots, \tilde{x}_{a-1}\}$ and returns it to the adversary.
The behavior of the oracles in the real and ideal worlds is detailed in Fig.s 5.4 and 5.5. When all the queries and responses are finished, the real world returns the key (k_0, k_1, k_2) to A, whereas the ideal world behaves as depicted in Fig.s 5.6, 5.7 and 5.8.

5.4.3. Offline Phase of the Ideal World

After the query-response phase, the ideal world samples three n-bit dummy keys (k_0, k_1, k_2), uniformly and independently of all the previously sampled random variables. Then it starts computing the hash value of pPMAC_Plus-Hash* for all the q queried messages. During this hash computation, if any of the events mentioned in stage 1 of the game (shown in Fig. 5.6) occur, it is aborted. The first event Coll addresses collisions between two inputs to the hash-permutations of a particular construction query and inputs to any

forward primitive query. 3-Coll takes care of collisions of a hash-permutation input from one construction query with one input block each of hash-permutations involved in two other construction queries. (Bad_1-Bad_3) occur when there is a collision in both invocations of π involved in the sum function. Note that $\overline{\mathsf{Bad}_2}$ and $\overline{\mathsf{Bad}_3}$ guarantee that a collision of the value Σ_i of the i^{th} construction query with a primitive query $\tilde{x}_a$ ensures freshness of Θ_i, and by symmetry, the same for Σ_i due to a primitive-value collision of Θ_i. This makes certain that the output $T_i \oplus \tilde{y}_a$ of Θ_i through π remains fresh. However, if $T_i \oplus \tilde{y}_a$ collides with any $\tilde{y}_{a'}$ due to the sampling of T_i, then permutation compatibility is violated. A similar violation arises when $\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 0$ collides with a primitive query $\tilde{x}_a$, but the output of Σ_i is not fresh. This event is captured in Bad_4. The events $\overline{\mathsf{Bad}_1}$ and $\overline{\mathsf{Bad}_3}$ guarantee that a collision in exactly one half of the hash blocks of two construction queries implies freshness of the other half. This also means that their tags do not collide with each other. However, if they do happen to collide with each other through sampling of the tags, permutation compatibility is again violated, as captured in Bad_5. If the game is not aborted in stage I, it proceeds to stage II.

In this stage, there may exist a set of indices for which exactly one hash block collides with a primitive query. For example, if $\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0$ collides with $\tilde{u}_b$ for some $i \in [q]$ and for some $b \in [p]$, then we remove i from $\mathcal{I}$ and add Σ_i to $\widetilde{\Sigma}$ and Θ_i to $\widetilde{\Theta}$, as well as $\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1$ to the domain of π and $T_i \oplus \tilde{v}_b$ to the range of π. Similarly, if $\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 0$ collides with $\tilde{w}_c$ for some $i \in [q]$ and for some $c \in [p]$, we remove i from $\mathcal{I}$ and add Σ_i and Θ_i to $\widetilde{\Sigma}$ and $\widetilde{\Theta}$ respectively, as well as $\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0$ to the domain of π and $T_i \oplus \tilde{z}_c$ to the range of π. Note that if $\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0$ collides with $\tilde{u}_b$, then Θ_i is fresh as Bad_2 and Bad_3 do not occur. Moreover, $T_i \oplus \tilde{y}_a$ is also fresh as Bad_4 does not occur. Hence, the inclusion of Θ_i in the set $\mathsf{Dom}(\pi^1)$ and $T_i \oplus \tilde{y}_a$ in $\mathsf{Ran}(\pi^1)$ is sound. One can similarly argue that the inclusion of Σ_i in $\mathsf{Dom}(\pi^0)$ and $T_i \oplus y_j^2$ in $\mathsf{Ran}(\pi^0)$ is also sound.

For the remaining $q - |\mathcal{I}|$ indices, there may exist a set of free indices $\mathcal{F}$ for which both blocks of the hash value are fresh in the set of $2(q - |\mathcal{I}|)$ hash block values. The oracle samples outputs for these fresh hash values without replacement such that for any $i \in \mathcal{F}$, the sampled outputs v_i and z_i sum up to T_i.

The cases remaining in stage III are those for which exactly one block of the hash value collides with that of another construction query. For all $i \in [q] \setminus (\mathcal{F} \sqcup \mathcal{I})$, if the output of the colliding hash value, say Σ_i, is not yet sampled, then the oracle samples its output without replacement, say v_i and sets the output of the remaining block, i.e., the output of Θ_i as the sum of v_i and T_i (see line 2 of stage III). Else, the oracle sets the output of Σ_i to the already defined element and adjusts the output of the other block

accordingly (see line 3 of stage III). Note that in the latter case, the oracle does not sample the output. If the output of Θ_i (i.e., $T_i \oplus v_i$) happens to collide with any previously sampled output or any element of $\mathsf{Ran}(\pi^1)$ in the above argument, then RC_Σ is set to 1 (see line 4 of stage III) and aborts the game. Similarly, the oracle sets RC_Θ to 1 if the adjustment of the output of Σ_i causes a collision with any previously sampled output or any element of $\mathsf{Ran}(\pi^0)$. Note that these events cannot hold for the real oracle as at least one of Θ_i or Σ_i is always fresh in the tuple of $2(q - |\mathcal{I}|)$ hash block values. Finally, it returns all these sampled values along with the sampled hash key to the distinguisher A.

5.4.4. Attack transcript

Let $\tau_c := \{(M_1, T_1), (M_2, T_2), \ldots, (M_q, T_q)\}$ be the list of construction queries and responses made by A. We call τ_c the *construction query transcript*. Let $\tau_p := \{(\tilde{x}_1, \tilde{y}_1), \ldots, (\tilde{x}_{2p}, \tilde{y}_{2p})\}$ be the list of primitive queries and responses made to π by A. The pair (τ_c, τ_p) constitutes the query transcript of the attack. For convenience, we slightly modify the experiment by revealing the keys (k_0, k_1, k_2) and internal or random values to the distinguisher A (only after it completes making all its queries but before it outputs its decision) in addition to responses to the queries it makes. If A interacts with the real world, then the actual key of the construction is revealed along with the permutation outputs of the hash output blocks Σ and Θ, whereas for the ideal world, a triplet of dummy n-bit keys (k_0, k_1, k_2) is revealed. The construction query transcript of the attack is thus

$$\hat{\tau}_c = \big((M_1, T_1, v_1, z_1), (M_2, T_2, v_2, z_2), \ldots, (M_q, T_q, v_q, z_q), k_0, k_1, k_2\big)\,.$$

Therefore, the query transcript of the attack is $\tau = (\hat{\tau}_c, \tau_p)$, where τ_p can further be partitioned into $\tau_p^0 := \{(\tilde{u}_b, \tilde{v}_b) : \tilde{u}_b = \hat{u}_b \| 0 \text{ where } \hat{u}_b \in \{0,1\}^{n-1}, \forall b \in [p]\}$ and $\tau_p^1 := \{(\tilde{w}_c, \tilde{z}_c) : \tilde{w}_c = \hat{w}_c \| 1 \text{ where } \hat{w}_c \in \{0,1\}^{n-1}, \forall c \in [p]\}$. Note that if A interacts with the real world, then

$$\forall i \in [q], v_i := \pi^0\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_i \oplus k_2\right) \| 0\right) := \pi\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_i \oplus k_2\right) \| 0\right),$$
$$z_i := \pi^1\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_i \oplus k_2\right) \| 1\right) := \pi\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_i \oplus k_2\right) \| 1\right),$$

where $(\Sigma_i, \Theta_i) := \mathsf{pPMAC\_Plus\text{-}Hash}^{\pi}_{k_0,k_1,k_2}(M_i)$. Moreover, a transcript τ in the real world must satisfy the following conditions:

- $v_i \oplus z_i = T_i, \forall i \in [q]$.
- $\forall a \in [2p], \pi(\tilde{x}_a) = \tilde{y}_a$ such that $\forall b \in [p], \pi(\tilde{u}_b) = \tilde{v}_b$ and $\forall c \in [p], \pi(\tilde{w}_c) = \tilde{z}_c$, where $\tilde{u}_b = \hat{u}_b \| 0$ and $\tilde{w}_c = \hat{w}_c \| 1$ for $\hat{u}_b, \hat{w}_c \in \{0,1\}^{n-1}$.
- $\widetilde{\Sigma}$ is permutation compatible with $\widetilde{v}$ and $\widetilde{\Theta}$ is permutation compatible with $\widetilde{z}$ (note that $(\widetilde{\Sigma}, \widetilde{\Theta})$ is uniquely determined by the message tuple $(M_1, \ldots, M_q)$, the tuple of keys k_0, k_1, k_2 and the public random permutation π).

5.4.5. Definition and Probability of Bad Transcripts

Suppose $\mathcal{X}$ denotes the set of all attainable transcripts and D_{re} and D_{id} the random variables that take transcript τ induced in the real world and ideal world respectively. An attainable transcript $\tau \in \mathcal{X}$ is said to be *bad* if either of the following bad flags

$$\mathsf{Coll}, \mathsf{3\text{-}Coll}, \mathsf{Bad}_1, \mathsf{Bad}_2, \mathsf{Bad}_3, \mathsf{Bad}_4, \mathsf{Bad}_5, \mathsf{Bad}_6, \mathsf{RC}_\Sigma, \mathsf{RC}_\Theta$$

is set to 1 as defined in Fig. 5.6. We define the event Bad as

$$\mathsf{Coll} \vee \mathsf{3\text{-}Coll} \vee \left(\vee_{i=1}^{6} \underbrace{(\mathsf{Bad}_i \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}})}_{\mathsf{Bad}_i^*} \right) \vee \underbrace{(\mathsf{RC}_\Sigma \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}})}_{\mathsf{RC}_\Sigma^*} \vee \underbrace{(\mathsf{RC}_\Theta \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}})}_{\mathsf{RC}_\Theta^*}.$$

Thus, $\mathsf{BadT} := \{((M_i, T_i, v_i, z_i), (\tilde{x}_a, \tilde{y}_a)) \in (\hat{\tau}_c, \tau_p) : ((M_i, T_i, v_i, z_i), (\tilde{x}_a, \tilde{y}_a))$ satisfies at least one condition boxed in Fig. 5.6$\} \subseteq \mathcal{X}$ and $\mathsf{GoodT} := \mathcal{X} \setminus \mathsf{BadT}$ denote the set of bad and good transcripts, respectively. Having identified the bad transcripts, we bound the probability of realizing them in the ideal world in the following lemma.

Lemma 15. *Let* BadT *be the set of all attainable bad transcripts and* D_{id} *be the random variable that takes a transcript* τ *induced in the ideal world. Then*

$$\Pr[\mathsf{D}_{\mathrm{id}} \in \mathsf{BadT}] \leq \epsilon_{\mathrm{bad}} = \frac{2\sqrt{3nqp^2} + 4}{2^n} + \frac{q^3(10\ell^3 + 5\ell^2 + 4\ell + 8)}{2^{2n}} + \frac{q^2p(2\ell + 9)}{2^{2n}} + \frac{qp^2(11\ell^2 + 4\ell + 8)}{2^{2n}} + \frac{q^2(2\ell + 5)}{2^{2n}}.$$

proof. Bounding the probability of the bad transcripts in the ideal world is equivalent to bounding the probability of the event Bad in the ideal world. Due to the union bound,

$$\Pr[\mathsf{Bad}] \leq \Pr[\mathsf{Coll}] + \Pr[\mathsf{3\text{-}Coll}] + \sum_{i=1}^{6} \Pr[\mathsf{Bad}_i^*] + \Pr[\mathsf{RC}_\Sigma^*] + \Pr[\mathsf{RC}_\Theta^*]. \quad (5.2)$$

In the following, we separately bound each of the above terms. By a slight abuse of notation, we use the flag names to identify the corresponding event. Before we bound the terms, we set up a few notations.

Notational set-up. Let $\mathcal{U} = \{\tilde{x}_a \in \{0,1\}^n : (\tilde{x}_a, \tilde{y}_a) \in \tau_p\}$ and $\mathcal{V} = \{\tilde{y}_a \in \{0,1\}^n : (\tilde{x}_a, \tilde{y}_a) \in \tau_p\}$ be the domain and range of the transcript of π. Let $(M_1, \ldots, M_q)$ be a tuple of q distinct messages such that the i^{th} message

M_i has ℓ_i blocks with $\ell = \max\{l_1, \ldots, l_q\}$, being the maximum number of message blocks amongst all the q messages. For two distinct fixed indices $i_1, i_2 \in [q]$, we define the set

$$\mathsf{NEQ}_{i_1,i_2} = \{\alpha \in \min[l_{i_1}, l_{i_2}] : M_{i_1}[\alpha] \neq M_{i_2}[\alpha]\} \cup \{\alpha : \min[l_{i_1}, l_{i_2}] + 1 \leq \alpha \leq \max[l_{i_1}, l_{i_2}]\}.$$

In words, NEQ_{i_1,i_2} refers to the set of all positions at which inputs to the hash permutation π from message blocks of M_{i_1} and M_{i_2} differ. We denote the inputs (resp. outputs) of these permutation instances as x_i (resp. y_i). In particular, we write $x_i[\alpha]$ to denote the permutation input corresponding to the α^{th} block of the i^{th} message, i.e. $x_i[\alpha] = M_i[\alpha] \oplus 2^{\alpha}k_0 \oplus 2^{2\alpha}k_1$ and $y_i[\alpha] = \pi(x_i[\alpha])$.

Bounding Coll. For a fixed choice of $i \in [q], \alpha \neq \beta$ in $[l_i]$ and $a_1, a_2 \in [p]$, the system of equations

$$\begin{aligned} 2^{\alpha}k_0 \oplus 2^{2\alpha}k_1 &= M^i[\alpha] \oplus \tilde{x}_{a_1}, \\ 2^{\beta}k_0 \oplus 2^{2\beta}k_1 &= M^i[\beta] \oplus \tilde{x}_{a_2} \end{aligned}$$

has rank 2. Since k_0 and k_1 are two independent n-bit keys, varying over all possible choices of indices gives

$$\Pr[\mathsf{Coll}] \leq \frac{qp^2\ell^2}{2^{2n+1}}. \tag{5.3}$$

Bounding 3-Coll. For a fixed choice of $i_1, i_2, i_3 \in [q]$, and distinct $\alpha_1 \in [l_{i_1}], \alpha_2 \in [l_{i_2}], \alpha_3 \in [l_{i_3}]$, the system of equations

$$\begin{aligned} (2^{\alpha_1} \oplus 2^{\alpha_2})k_0 \oplus (2^{2\alpha_1} \oplus 2^{2\alpha_2})k_1 &= M_{i_1}[\alpha_1] \oplus M_{i_2}[\alpha_2], \\ (2^{\alpha_1} \oplus 2^{\alpha_3})k_0 \oplus (2^{2\alpha_1} \oplus 2^{2\alpha_3})k_1 &= M_{i_1}[\alpha_1] \oplus M_{i_3}[\alpha_3] \end{aligned}$$

has rank 2. Since k_0 and k_1 are two independent n-bit keys, varying over all possible choices of indices gives

$$\Pr[\text{3-Coll}] \leq \frac{q^3\binom{3\ell}{3}}{2^{2n}} \leq \frac{5q^3\ell^3}{2^{2n}}. \tag{5.4}$$

Bounding Event Bad_1^*. We fix three messages M_{i_1}, M_{i_2} and M_{i_3} where $i_1 \neq i_2, i_1 \neq i_3$, such that M_{i_1} has l_{i_1} blocks, M_{i_2} has l_{i_2} blocks and M_{i_3} has l_{i_3} blocks. Consider the event

$$\mathsf{CollX}^{(1)} : \left\{\exists\, j_1, j_2 \in \{i_1, i_2, i_3\} \text{ and } \alpha \in [l_{j_1}], \beta \in [l_{j_2}], \text{ such that } x_{j_1}[\alpha] = x_{j_2}[\beta]\right\}.$$

Therefore,

$$\Pr[\mathsf{Bad}_1^*] \leq \sum_{i_1,i_2,i_3} \Bigg(\underbrace{\Pr[\Theta_{i_1} = \Theta_{i_3} \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \mathsf{CollX}^{(1)}]}_{(1)} + \underbrace{\Pr[\Sigma_{i_1} = \Sigma_{i_2} \wedge \Theta_{i_1} = \Theta_{i_3} \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}^{(1)}}]}_{(2)} \Bigg). \quad (5.5)$$

Bounding (1): It is easy to see that for a fixed triplet of messages, the probability of $\mathsf{CollX}^{(1)}$ is at most $\binom{3\ell}{2}/2^n$. Under this condition, $\Theta_{i_1} = \Theta_{i_3}$ provides a non-trivial equation for some random variable $y_{i'}[\alpha']$. Assuming $l_{i_1} \leq l_{i_3}$, let $\alpha \in [l_{i_1}]$ (if it exists) be the largest index such that $M_{i_1}[\alpha] \neq M_{i_3}[\alpha]$. Then either $y_{i_1}[\alpha]$ or $y_{i_3}[\alpha]$ is fresh and the equation $\Theta_{i_1} = \Theta_{i_3}$ is non-trivial for this random variable. On the other hand, if no such index α exists(i.e. $M_{i_1}[\alpha] = M_{i_3}[\alpha]$ for all $\alpha \in [l_{i_1}]$ and $l_{i_1} < l_{i_3}$), we can obtain a freshly sampled random variable $y_{i_3}[\beta]$, for which $\Theta_{i_1} = \Theta_{i_3}$ becomes non-trivial. Therefore, the probability that this equation is satisfied is at most $1/(2^n - 2\ell) \leq 2/2^n$, assuming $\ell \leq 2^{n-2}$, giving (1) an upper bound of $\binom{3\ell}{2}/2^n \cdot 2/2^n \leq 9\ell^2/2^{2n}$:

$$\Pr[\Theta_i = \Theta_k \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \mathsf{CollX}^{(1)}] \leq \frac{9\ell^2}{2^{2n}}. \quad (5.6)$$

Bounding (2): We split this case into the following two subcases:

$i_2 = i_3$: Without loss of generality, assume $l_{i_1} \leq l_{i_2}$. Note that if $l_{i_1} = l_{i_2}$, then l_{i_2} must be at least 2 for $\Sigma_{i_1} = \Sigma_{i_2}$ to yield a non-trivial equation. In this case, we can easily find two freshly sampled random variables $y_{i_1}[\alpha]$ and $y_{i_2}[\beta]$ for which $(\Sigma_{i_1} = \Sigma_{i_2}) \wedge (\Theta_{i_1} = \Theta_{i_2})$ yields a system of equations of rank 2. Hence by the rank argument (i.e. Lemma 25),

$$\Pr[(\Sigma_{i_1} = \Sigma_{i_2}) \wedge (\Theta_{i_1} = \Theta_{i_2}) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}^{(1)}}] \leq \frac{1}{(2^n - 2\ell)_2}. \quad (5.7)$$

In the particular case when $l_{i_1} + 1 = l_{i_2}$ and $\mathsf{NEQ}_{i_1 i_2} = \{l_{i_2}\}$, if $x_{i_2}[l_{i_2}] = \tilde{x}_a$ for some $a \in [p]$, then $\Sigma_{i_1} = \Sigma_{i_2}$ and $\Theta_{i_1} = \Theta_{i_2}$ would boil down to

$$\begin{aligned} &\tilde{y}_a = 0^n, \\ &\left(2^{l_{i_1}} \oplus 2^{l_{i_1}+1}\right) y_{i_1}[1] \oplus \ldots \left(2 \oplus 2^2\right) y_{i_1}[l_{i_1}] \oplus 2\tilde{y}_a = 0^n. \end{aligned} \quad (5.8)$$

As the second equation in 5.8 is non-trivial, and $x_{i_2}[l_{i_2}] = \tilde{x}_a$ holds with probability at most $1/2^n$ (the number of choices for $\tilde{x}_a$ is 1),

$$\Pr[(\Sigma_{i_1} = \Sigma_{i_2}) \wedge (\Theta_{i_1} = \Theta_{i_2}) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}^{(1)}}] \leq \frac{1}{2^n(2^n - 2\ell)}. \quad (5.9)$$

In case $l_{i_2} \geq l_{i_1} + 2$, we either determine $\beta_1, \beta_2 \in \{l_{i_1}+1, \ldots, l_{i_2}\}$ or $\beta_1 \in [l_{i_1}], \beta_2 \in \{l_{i_1}+1, \ldots, l_{i_2}\}$ such that $y_{i_2}[\beta_1]$ and $y_{i_2}[\beta_2]$ are freshly sampled. In both instances, $\left(\Sigma_{i_1} = \Sigma_{i_2}\right) \wedge \left(\Theta_{i_1} = \Theta_{i_2}\right)$ would result in a system of equations having rank 2, and hence by the rank argument (i.e. Lemma 25),

$$\Pr[\left(\Sigma_{i_1} = \Sigma_{i_2}\right) \wedge \left(\Theta_{i_1} = \Theta_{i_2}\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}^{(1)}}] \leq \frac{1}{(2^n - 2\ell)_2}. \tag{5.10}$$

Combining Eqn.s (5.7), (5.9) and (5.10), and assuming $\ell + 1 \leq 2^{n-2}$, we have

$$\Pr[\left(\Sigma_{i_1} = \Sigma_{i_2}\right) \wedge \left(\Theta_{i_1} = \Theta_{i_2}\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}^{(1)}}] \leq \frac{10}{2^{2n}}. \tag{5.11}$$

$i_2 \neq i_3$: We approach this in five parts, the first four addressing cases when either M_{i_1} is a prefix of one of M_{i_2} and M_{i_3}, or one of M_{i_2} and M_{i_3} is a prefix of M_{i_1}, and the fifth when neither of the first four occur.

M_{i_1} **is a prefix of** M_{i_2}**:** Let $l_{i_2} = l_{i_1} + 1$ and $x_{i_2}[l_{i_2}] = \tilde{x}_a$ for some $a \in [p]$. Then $\Theta_{i_1} = \Theta_{i_3}$ becomes a non-trivial equation, contributing a term $1/(2^n - 3\ell)$ to the bound. An additional $1/2^n$ is contributed by the event $x_{i_2}[l_{i_2}] = \tilde{x}_a$ (as the number of choices for $\tilde{x}_a$ is 1). Assuming $\ell \leq 2^{n-1}/3$, the bound is thus $2/2^{2n}$. On the other hand, if $x_{i_2}[l_{i_2}]$ is fresh, then a freshly sampled random variable $y_{i_1}[\star]$ can be found such that $\Theta_{i_1} = \Theta_{i_3}$ becomes a non-trivial equation. Therefore, $\Sigma_{i_1} = \Sigma_{i_2}, \Theta_{i_1} = \Theta_{i_3}$ becomes a system of equations of rank 2 (in $y_{i_2}[l_{i_2}]$ and $y_{i_1}[\star]$), and hence by the rank argument (i.e. Lemma 25), we bound the probability of the event by $1/(2^n - 3\ell)_2 \leq 4/2^{2n}$, assuming $3\ell + 1 \leq 2^{n-1}$.

If $l_{i_2} \geq l_{i_1} + 2$, then it is easy to find an index $\beta \in \{l_{i_1}+1, \ldots, l_{i_2}\}$ such that $y_{i_2}[\beta]$ is freshly sampled. Moreover, we can find another index $\alpha \in [l_{i_1}]$ (or $\alpha \in [l_{i_3}]$) such that $y_{i_1}[\alpha]$ (or $y_{i_3}[\alpha]$) is freshly sampled. In both cases, $\Sigma_{i_1} = \Sigma_{i_2}, \Theta_{i_1} = \Theta_{i_3}$ becomes a system of equations of rank 2. Therefore, by the rank argument (i.e. Lemma 25) and assuming $3\ell + 1 \leq 2^{n-1}$, the probability of the event becomes at most $4/2^{2n}$. Thus,

$$\Pr[\Sigma_{i_1} = \Sigma_{i_2} \wedge \Theta_{i_1} = \Theta_{i_2} \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}^{(1)}}] \leq \frac{10}{2^{2n}}. \tag{5.12}$$

The other subcases can be argued similarly and their probabilities bounded above by $10/2^{2n}$.

We now assume that neither is M_{i_1} a prefix of M_{i_2} or M_{i_3}, and nor the reverse. In this case, we can find an index α such that $M_{i_1}[\alpha] \neq M_{i_2}[\alpha]$ and $y_{i_1}[\alpha]$ is freshly sampled. Moreover, we can find another index

β such that $M_{i_1}[\beta] \neq M_{i_3}[\beta]$ and $y_{i_3}[\beta]$ is freshly sampled. $\Sigma_{i_1} = \Sigma_{i_2}, \Theta_{i_1} = \Theta_{i_3}$ is a system of equations of rank 2 in these two variables, and hence by the rank argument (i.e. Lemma 25) and by assuming $3\ell + 1 \leq 2^{n-1}$,

$$\Pr[\Sigma_{i_1} = \Sigma_{i_2} \wedge \Theta_{i_1} = \Theta_{i_2} \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}^{(1)}}] \leq \frac{4}{2^{2n}}. \tag{5.13}$$

Therefore, combining Eqn.s (5.12) and (5.13), the assumption $3\ell + 1 \leq 2^{n-1}$ gives

$$\Pr[\Sigma_{i_1} = \Sigma_{i_2} \wedge \Theta_{i_1} = \Theta_{i_2} \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}^{(1)}}] \leq \frac{14}{2^{2n}}. \tag{5.14}$$

Finally, varying over all choices of $i_1, i_2, i_3 \in [q]$ and combining Eqn.s (5.5), (5.6), (5.11) and (5.14) with the assumption that $3\ell + 1 \leq 2^{n-1}$, we have

$$\Pr[\mathsf{Bad}_1^*] \leq \frac{3q^3\ell^2}{2^{2n+1}} + \frac{4q^3}{2^{2n}}. \tag{5.15}$$

Bounding Event Bad_2^*. For fixed indices $i \in [q]$ and $b, c \in [p]$, the event

$$(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0 = \tilde{u}_b) \wedge (\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1 = \tilde{w}_c)$$

can be reduced to the following system of equations:

$$\begin{aligned} y_i[1] \oplus y_i[2] \oplus \ldots \oplus y_i[l_i] \oplus k_2 &= \tilde{u}_b, \\ 2^{l_i} y_i[1] \oplus 2^{l_i - 1} y_i[2] \oplus \ldots \oplus 2 y_i[l_i] \oplus k_2 &= \tilde{w}_c. \end{aligned} \tag{5.16}$$

We split Bad_2^* into the following two cases:

Case (1): Suppose $l_i = 1$ and $M_i[1] \oplus 2k_0 \oplus 2^2 k_1$ collides with a primitive query input $\tilde{x}_{a'}$ for some $a' \in [2p]$. In this case, (5.16) boils down to $\{\tilde{y}_{a'} \oplus k_2 = \tilde{u}_b, 2\tilde{y}_{a'} \oplus k_2 = \tilde{w}_c\}$. The probability of occurrence of Bad_2^* can now be bounded using the events $M_i[1] \oplus 2k_0 \oplus 2^2 k_1 = \tilde{x}_{a'}$ and $k_2 = \tilde{y}_{a'} \oplus \tilde{u}_b$; the probability of the first event is bounded by 2^{-n} through the randomness of k_0, and the probability of the latter is bounded by 2^{-n} through the randomness of k_2. Note that the number of choices for a' is $2p$, that for b, c (each) is p, and that for i is q. For each of these choices of $\tilde{x}_{a'}$ and $\tilde{u}_b$, the number of choices for $\tilde{w}_c$ is 1. Hence,

$$\Pr[\mathsf{Bad}_2^*] \leq \frac{2qp^2}{2^{2n}}. \tag{5.17}$$

On the other hand, if $M_i[1] \oplus 2k_0 \oplus 2^2 k_1$ does not collide with any primitive query, then $y_i[1]$ is fresh. Thus, (5.16) boils down to $\{y_i[1] \oplus k_2 = \tilde{u}_b, 2y_i[1] \oplus$

$k_2 = \tilde{w}_c\}$. Note that the rank of this system of equations is 2. Varying over all possible choices of $b, c \in [p]$ and $i \in [q]$ gives

$$\Pr[\mathsf{Bad}_2^*] \leq \frac{qp^2}{2^n(2^n - \ell)}. \tag{5.18}$$

Case (2): In this case, we assume $l_i > 1$. Let $\mathsf{CollX}^{(2)}$ be the event that refers to the collision of any two input blocks, i.e.

$$\mathsf{CollX}^{(2)} : \{\exists\, \alpha_1, \alpha_2 \in [l_i], \text{ such that } \alpha_1 \neq \alpha_2, x_i[\alpha_1] = x_i[\alpha_2]\}\,.$$

Therefore, we write

$$\Pr[\mathsf{Bad}_2^*] \leq \sum_{i=1}^{q} \Bigg(\underbrace{\Pr[\text{Eqn.s (5.16) hold } \wedge \mathsf{CollX}^{(2)}]}_{(1)} + \underbrace{\Pr[\text{Eqn.s (5.16) hold } \wedge \overline{\mathsf{CollX}^{(2)}}]}_{(2)} \Bigg).$$

The joint event in (1) holds with probability at most $\binom{\ell}{2}/2^{2n}$ (in which the event $\mathsf{CollX}^{(2)}$ contrinutes the term $\binom{\ell}{2}/2^n$ and the randomness of k_2 contributes the term $1/2^n$). The event in (2) ensures the freshness of at least one of the variables $y_i[1], \ldots, y_i[l_i]$. Without loss of generality, let us assume $y_i[1]$ is fresh. Given the values of all the other random variables $y_i[\star]$ in (5.16), the reduced system of equations $\{y_i[1] \oplus k_2 = c,\ 2y_i[1] \oplus k_2 = c'\}$ with rank 2 results in an upper bound of $1/2^n(2^n - \ell)$. Varying (1) and (2) over all choices of $b, c \in [p]$ and $i \in [q]$ gives

$$\Pr[\mathsf{Bad}_2^*] \leq \frac{qp^2\ell^2}{2^{2n+1}} + \frac{qp^2}{2^n(2^n - \ell)}. \tag{5.19}$$

From Eqn.s (5.17), (5.18) and (5.19), and with the assumption that $\ell \leq 2^{n-1}$, we obtain

$$\Pr[\mathsf{Bad}_2^*] \leq \frac{5qp^2\ell^2}{2^{2n}} + \frac{2qp^2}{2^{2n}}. \tag{5.20}$$

Bounding Event Bad_3^*. This event can be split into the following two sub-events:

$$\begin{aligned}
(1) \ : \ & \{\exists\, i_1 \neq i_2 \text{ in } [q] : \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0\right) \\
& \wedge \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 1 \in \mathsf{Dom}(\pi^1)\right) \wedge \overline{\mathsf{Coll} \vee \text{3-}\mathsf{Coll}}\}, \\
(2) \ : \ & \{\exists\, i_1 \neq i_2 \text{ in } [q] : \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 1 = \mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_2} \oplus k_2\right) \| 1\right) \\
& \wedge \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 \in \mathsf{Dom}(\pi^0)\right) \wedge \overline{\mathsf{Coll} \vee \text{3-}\mathsf{Coll}}\}.
\end{aligned}$$

BOUNDING (1): For fixed $i_1 \neq i_2$ in $[q]$ and a fixed $c \in [p]$, the event is $\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0\right) \wedge \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 1 = \tilde{w}_c\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}$. Without loss of generality, assume $l_{i_1} \geq l_{i_2}$. Since the probability of (1) is zero for $l_{i_1} \leq 1$, assume $l_{i_1} \geq 2$. As before, we determine an index $\beta \in [l_{i_1} - 1]$: If $l_{i_1} > l_{i_2}$, then $\beta = l_{i_1}$; if $l_{i_1} = l_{i_2}$ and $\mathsf{NEQ}_{i_1 i_2} = \{l_{i_1}\}$, then the probability becomes zero – so we set $\beta = \max\left\{\alpha \in \mathsf{NEQ}_{i_1 i_2}\right\}$ $(\neq l_{i_1})$ when $l_{i_1} = l_{i_2}$. Let

$$\mathsf{CollX}_{\beta}^{(3)} : \left\{ \left(\exists\, \beta_1 \in [l_{i_1}] : \beta_1 \neq \beta, x_{i_1}[\beta] = x_{i_1}[\beta_1]\right) \vee \left(\exists\, \beta_2 \in [l_{i_2}] : x_{i_1}[\beta] = x_{i_2}[\beta_2]\right) \right\}$$

be the event that denotes the collision of $x_{i_1}[\beta]$ with at least one of the remaining input blocks. Also let E_β denote the event $\left\{\exists\, a \in [p] : x_{i_1}[\beta] = \tilde{x}_a\right\}$. Therefore, we write

$$\begin{aligned}
&\Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1}\oplus k_2\right)\|0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2}\oplus k_2\right)\|0\right) \wedge \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1}\oplus k_2\right)\|1 = \tilde{w}_c\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}\right] \\
&\quad \leq \Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right)\|1 = \tilde{w}_c\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \mathsf{CollX}_{\beta}^{(3)}\right] \\
&\qquad + \Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right)\|0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right)\|0\right) \wedge \right. \\
&\qquad \left. \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right)\|1 = \tilde{w}_c\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}_{\beta}^{(3)}}\right] \\
&\qquad \leq \Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right)\|1 = \tilde{w}_c\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \mathsf{CollX}_{\beta}^{(3)}\right] \\
&\qquad\quad + \Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right)\|0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right)\|0\right) \wedge \right. \\
&\qquad\quad \left. \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right)\|1 = \tilde{w}_c\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}_{\beta}^{(3)}} \wedge \mathsf{E}_\beta\right] \\
&\qquad\quad + \Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right)\|0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right)\|0\right) \wedge \right. \\
&\qquad \left. \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right)\|1 = \tilde{w}_c\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}_{\beta}^{(3)}} \wedge \overline{\mathsf{E}_\beta}\right]. \quad (5.21)
\end{aligned}$$

We break this down into three manageable chunks:

$$\mathsf{E.1} \;:=\; \Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1}\oplus k_2\right)\|1=\tilde{w}_c\right)\wedge\overline{\mathsf{Coll}}\wedge\overline{\mathsf{3\text{-}Coll}}\wedge\mathsf{CollX}_{\beta}^{(3)}\right],$$

$$\mathsf{E.2} \;:=\; \Pr\Bigg[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1}\oplus k_2\right)\|0=\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2}\oplus k_2\right)\|0\right)$$
$$\wedge\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1}\oplus k_2\right)\|1=\tilde{w}_c\right)\wedge\overline{\mathsf{Coll}}\wedge\overline{\mathsf{3\text{-}Coll}}\wedge\overline{\mathsf{CollX}_{\beta}^{(3)}}\wedge\mathsf{E}_{\beta}\Bigg],$$

$$\mathsf{E.3} \;:=\; \Pr\Bigg[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1}\oplus k_2\right)\|0=\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2}\oplus k_2\right)\|0\right)$$
$$\wedge\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1}\oplus k_2\right)\|1=\tilde{w}_c\right)\wedge\overline{\mathsf{Coll}}\wedge\overline{\mathsf{3\text{-}Coll}}\wedge\overline{\mathsf{CollX}_{\beta}^{(3)}}\wedge\overline{\mathsf{E}_{\beta}}\Bigg].$$

1. In the sub-event (E.1), since the equation $\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1}\oplus k_2\right)\|1=\tilde{w}_c$ is non-trivial, it can be bound by probability $2/2^n$ using the randomness of k_2, and $\mathsf{CollX}_{\beta}^{(3)}$ holds with probability at most $2\ell/2^n$. Thus, (E.1) can be bound by
$$4\ell/2^{2n}. \tag{5.22}$$
2. We first consider the case when $l_{i_1}=l_{i_2}+1$ and $\mathsf{NEQ}_{i_1i_2}=\{l_{i_1}\}$ in (E.2). Since $x_{i_1}[l_{i_1}]=\tilde{x}_a$, it boils the event
$\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1}\oplus k_2\right)\|0=\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2}\oplus k_2\right)\|0\right)$
$\wedge\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1}\oplus k_2\right)\|1=\tilde{w}_c\right)$ down to the following system of equations:
$$\begin{aligned}\tilde{y}_a &= 0^n,\\ 2^{l_{i_1}}y_{i_1}[1]\oplus 2^{l_{i_1}-1}y_{i_1}[2]\oplus\ldots\oplus 2y_{i_1}[l_{i_1}]\oplus k_2 &= \tilde{w}_c.\end{aligned} \tag{5.23}$$
As the equation $\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1}\oplus k_2\right)\|1=\tilde{w}_c$ is non-trivial, its probability can be at most $2/2^n$. Moreover, the probability that $x_{i_1}[l_{i_1}]=\tilde{x}_a$ is bounded above by $1/2^n$ (since the number of choices for a is 1). Thus, this case of (E.2) can be bound by $2/2^{2n}$.
We next consider the case when $l_{i_1}>l_{i_2}$ in (E.2). As E_{β} holds, $y_{i_1}[l_{i_1}]$ is not fresh. However, as Coll, 3-Coll and $\mathsf{CollX}_{\beta}^{(3)}$ also do not hold, at least one of the variables $y_{i_1}[\star]$ must be fresh, i.e. $\exists\,\alpha\in\mathsf{NEQ}_{i_1i_2}\setminus\{l_{i_1}\}$ such that $y_{i_1}[\alpha]$ is fresh. Without loss of generality, let us assume that $y_{i_1}[1]$ is fresh. Given all other random variables $y_{i_1}[\star]$ and $y_{i_2}[\star]$ in
$$\begin{aligned}y_{i_1}[1]\oplus y_{i_1}[2]\oplus\ldots y_{i_1}[l_{i_1}]\oplus y_{i_2}[1]\oplus y_{i_2}[2]\oplus\ldots\oplus y_{i_2}[l_{i_2}] &= 0^n,\\ 2^{l_{i_1}}y_{i_1}[1]\oplus 2^{l_{i_1}-1}y_{i_1}[2]\oplus\ldots 2y_{i_1}[l_{i_1}]\oplus k_2 &= \tilde{w}_c,\end{aligned} \tag{5.24}$$

we obtain $y_{i_1}[1] = d$ and $2^{l_{i_1}} y_{i_1}[1] \oplus k_2 = d'$, for constants d and d'. Hence,

$$\Pr[\text{Eqn. 5.24 holds }] \leq \frac{1}{2^n(2^n - 2\ell)} \leq \frac{8}{2^{2n}}, \text{ assuming } \ell \leq 2^{n-2}.$$

Combining the above bounds the probability of (E.2) by

$$\frac{10}{2^{2n}}. \tag{5.25}$$

3. In the event (E.3), it is easy to see that $y_{i_1}[l_{i_1}]$ is fresh. Hence, given all other random variables $y_{i_1}[\star]$ and $y_{i_2}[\star]$ in Eqn. (5.24), the system is reduced to $y_{i_1}[l_{i_1}] = d$, $2y_{i_1}[l_{i_1}] \oplus k_2 = d'$ for some constants d and d'. Hence, the probability of (E.3) has an upper bound of

$$\frac{4}{2^n(2^n - 2\ell)} \leq \frac{8}{2^{2n}}, \tag{5.26}$$

where the last inequality follows as $\ell \leq 2^{n-1}$.

Varying over all possible choices of $i_1 \neq i_2$ in $[q]$ and $c \in [p]$ and combining Eqn.s (5.21), (5.22), (5.25) and (5.26) gives

$$\Pr[(1)] \leq \frac{(4.5 + l)q^2 p}{2^{2n}}. \tag{5.27}$$

Bounding (2): This is symmetric to (1). Hence, it can be similarly bounded:

$$\Pr[(2)] \leq \frac{(4.5 + l)q^2 p}{2^{2n}}. \tag{5.28}$$

Therefore, from Eqn.s (5.27) and (5.28),

$$\Pr[\mathsf{Bad}_3^*] = \Pr[(1)] + \Pr[(2)] \leq \frac{(9 + 2l)q^2 p}{2^{2n}}. \tag{5.29}$$

Bounding Bad_4^*. This event can be split into the following two sub-events:

$$(1) \quad : \quad \left\{\exists\, i \in [q],\, b, c \in [p] : \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0\right) \wedge \left(T_i \oplus \tilde{v}_b = \tilde{z}_c\right) \wedge \overline{\mathsf{Coll} \vee \text{3-Coll}}\right\},$$

$$(2) \quad : \quad \left\{\exists\, i \in [q],\, b, c \in [p] : \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 1 = \mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_2} \oplus k_2\right) \| 1\right) \wedge \left(T_i \oplus \tilde{z}_c = \tilde{v}_b\right) \wedge \overline{\mathsf{Coll} \vee \text{3-Coll}}\right\}.$$

Bounding (1): We fix a message M_i consisting of l_i blocks. We also fix the indices b and c. Now, we analyze the probability of the event in two cases: (I) The i^{th} construction query occurs after the b^{th} and c^{th} primitive queries. (II) At least one of the primitive queries appears after the i^{th} construction query.

Case I: As T_i is distributed uniformly at random and since the distribution of k_2 is independent of all the other random variables, we bound the probability of the event by $1/2^{2n}$. Varying over all possible choices of $i \in [q]$ and $b, c \in [p]$, we have

$$\Pr[\mathsf{Bad}_4^*] \leq \frac{qp^2}{2^{2n}} \text{ in case (I).} \tag{5.30}$$

Case II: Suppose the b^{th} primitive query is the latest.

(a) If the primitive query is in the forward direction, then $\tilde{v}_b$ is randomly distributed. Hence by the randomness of k_2 and $\tilde{v}_b$, we bound the probability of the event to at most $2/2^n$. Varying over all possible choices of $i \in [q]$ and $b, c \in [p]$, we have

$$\Pr[\mathsf{Bad}_4^*] \leq \frac{2qp^2}{2^{2n}} \text{ in case (IIa).} \tag{5.31}$$

(b) If the b^{th} primitive query is in the inverse direction, then $\tilde{u}_b$ is random. We bound the event Bad_4^* given the complement of the event

$$\mathsf{E} : \left\{ |\{(T_i, \tilde{v}_b, \tilde{z}_c) \in [q] \times [p] \times [p] : T_i = \tilde{v}_b \oplus \tilde{z}_c\}| \geq \frac{qp^2}{2^n} + \sqrt{3nqp^2} \right\}.$$

As $\Pr[\mathsf{Bad}_4^*] \leq \Pr[\mathsf{Bad}_4^* \mid \overline{\mathsf{E}}] + \Pr[\mathsf{E}]$ and as $\Pr[\mathsf{E}] \leq 2/2^n$ according to the sum-capture Corollary 5, for a fixed choice of i, b and c such that $T_i = \tilde{v}_b \oplus \tilde{z}_c$, the probability of the event $\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0 = \tilde{u}_b$ is at most $1/2^n$ by the randomness of k_2. As the number of choices for i, b and c is at most $qp^2/2^n + \sqrt{3nqp^2}$,

$$\Pr[\mathsf{Bad}_4^*] \leq \frac{qp^2}{2^{2n}} + \frac{\sqrt{3nqp^2}}{2^n} + \frac{2}{2^n} \text{ in case (IIb).} \tag{5.32}$$

The analysis is exactly the same when the c^{th} primitive query is the latest. Therefore,

$$\Pr[\mathsf{Bad}_4^*] \leq \frac{qp^2}{2^{2n}} + \frac{\sqrt{3nqp^2}}{2^n} + \frac{2}{2^n}. \tag{5.33}$$

Bounding (2): The analysis for bounding this sub-event is exactly identical to that of Bad_4^*. Thus

$$\Pr[\mathsf{Bad}_4^*] \leq \frac{2qp^2}{2^{2n}} + \frac{2\sqrt{3nqp^2}}{2^n} + \frac{4}{2^n}. \tag{5.34}$$

Bounding Bad_5^*. We again begin by partitioning the event into two sub-events:

$$(1) \;:\; \{\exists\, i_1 \neq i_2 \text{ in } [q] : \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0\right) \wedge \left(T_{i_1} = T_{i_2}\right) \wedge \overline{\mathsf{Coll} \vee \mathsf{3\text{-}Coll}}\}$$

$$(2) \;:\; \{\exists\, i_1 \neq i_2 \text{ in } [q] : \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_2} \oplus k_2\right) \| 1\right) \wedge \left(T_{i_1} = T_{i_2}\right) \wedge \overline{\mathsf{Coll} \vee \mathsf{3\text{-}Coll}}\}.$$

Bounding (1): For the two fixed distinct messages M_{i_1} and M_{i_2}, the event $\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0$ is reduced to the following equations:

$$y_{i_1}[1] \oplus y_{i_1}[2] \oplus \ldots \oplus y_{i_1}[l_{i_1}] \oplus y_{i_2}[1] \oplus y_{i_2}[2] \oplus \ldots \oplus y_{i_2}[l_{i_2}] = 0^n. \tag{5.35}$$

Without loss of generality, assume $l_{i_1} \geq l_{i_2}$. The probability of the event is zero for $l_{i_1} \leq 1$. Thus, we assume $l_{i_1} \geq 2$. As before, we determine an index $\beta \in [l_{i_1} - 1]$ as follows: if $l_{i_1} > l_{i_2}$, then $\beta = l_{i_1}$. If $l_{i_1} = l_{i_2}$ and $\mathsf{NEQ}_{i_1 i_2} = \{l_{i_1}\}$, then the probability of the event is again zero. So we set $\beta = \max \mathsf{NEQ}_{i_1 i_2}$ when $l_{i_1} = l_{i_2}$. Note the following event:

$$\mathsf{CollX}_\beta^{(4)} \;:\; \{\; \left(\exists\, \beta_1 \in [l_{i_1}] : \beta_1 \neq \beta, x_{i_1}[\beta] = x_{i_1}[\beta_1]\right) \vee \left(\exists\, \beta_2 \in [l_{i_2}] \text{ such that } x_{i_1}[\beta] = x_{i_2}[\beta_2]\right) \;\}.$$

$$\text{Therefore, } \Pr\left[\mathsf{chop}_{\mathsf{LSB}}\left(\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}\right] \leq \underbrace{\Pr\left[\mathsf{CollX}_\beta^{(5)}\right]}_{\mathsf{E.4}} +$$

$$\underbrace{\Pr\left[\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0 \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}_\beta^{(5)}}\right]}_{\mathsf{E.5}} \tag{5.36}$$

Due to the randomness of k_0 and k_1, the first term (i.e. E.4) in Eqn. (5.36) is bound by $(\ell - 1 + \ell)/2^n \leq 2\ell/2^n$. We split the analysis of E.5 into the following two cases:

Case I: When $l_{i_1} = l_{i_2} + 1$ and $\mathsf{NEQ}_{i_1 i_2} = \{l_{i_1}\}$, if $x_{i_1}[l_{i_1}] = \tilde{x}_a$ for some $a \in [p]$, then $\tilde{y}_a = 0^n$. Therefore, the event occurs with a probability of at most $1/2^n$ due to the randomness of k_0 and k_1 (note that the number of choices for $\tilde{x}_a$ is 1). On the other hand, if $x_{i_1}[l_{i_1}]$ is fresh, then $y_{i_1}[l_{i_1}]$ is freshly sampled and hence for this random variable, the rank 1 equation $\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0$ ensures a probability bound of $1/(2^n - 2\ell)$, by the rank argument (i.e. Lemma 25).

Case-II: When $l_{i_1} \geq l_{i_2} + 2$, at least one $\beta \in \{l_{i_2}+1, \ldots, l_{i_1}\}$ can be certainly found such that $x_{i_1}[\beta]$ is fresh and hence $y_{i_1}[\beta]$ is freshly sampled. For this random variable $y_{i_1}[\beta]$, the rank 1 equation $\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0$ ensures a probability bound of $1/(2^n - 2\ell)$, by the rank argument (i.e. Lemma 25).

Combining the above two cases and by assuming $\ell \leq 2^{n-2}$gives

$$\Pr\left[\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0 \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \wedge \overline{\mathsf{CollX}_{\beta}^{(5)}}\right] \leq \frac{5}{2^n}. \quad (5.37)$$

Therefore from Eqn.s (5.36) and (5.37), and by the assumption $\ell \leq 2^{n-2}$, we have

$$\Pr\left[\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0 \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}\right] \leq \frac{2\ell + 5}{2^n}. \quad (5.38)$$

Finally, from Eqn. (5.39), the fact that the event $T_{i_1} = T_{i_2}$ is independent of the event $\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}$, and that for a fixed choice of i_1 and i_2, the probability that $T_{i_1} = T_{i_2}$ holds is 2^{-n}, we have

$$\begin{aligned} & \Pr[(1)] \\ = & \sum_{i_1, i_2} \left(\Pr\left[T_{i_1} = T_{i_2}\right] \cdot \right. \\ & \left. \Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}\right] \right) \\ \leq & \frac{\frac{1}{2}q^2(2\ell+5)}{2^{2n}} \leq \frac{q^2\ell + 2.5q^2}{2^{2n}}. \end{aligned} \quad (5.39)$$

Bounding (2): This event is symmetric to the first, and thus has the same bound:

$$\Pr[(2)] \leq \frac{q^2\ell + 2.5q^2}{2^{2n}}. \quad (5.40)$$

Therefore, from Eqn.s (5.39) and (5.40), we have

$$\Pr[\mathsf{Bad}_5^*] = \Pr[(1)] + \Pr[(2)] \leq \frac{2q^2\ell + 5q^2}{2^{2n}}. \quad (5.41)$$

Bounding Bad_6^*. Consider the sub-event $\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = x_{i_1}[\alpha]\right) \wedge \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 1 = x_{i_3}[\beta]\right) \wedge \overline{\mathsf{Coll} \vee \mathsf{3\text{-}Coll}}$. This event can be expanded

in terms of XOR operations on the hash permutation outputs as follows (where $\alpha \in [l_{i_2}]$ and $\beta \in [l_{i_3}]$ are arbitrary indices):

$$\begin{aligned}\Pr\left[\mathsf{Bad}_6^*\right] &= \Pr\Big[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = M_{i_2}[\alpha] \oplus 2^{\alpha}k_0 \oplus 2^{2\alpha}k_1\right) \\ &\wedge \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 1 = M_{i_3}[\beta] \oplus 2^{\beta}k_0 \oplus 2^{2\beta}k_1\right) \wedge \overline{\mathsf{Coll} \vee \mathsf{3\text{-}Coll}}\Big] \\ &\leq \Pr\left[\pi\left(M_{i_1}[1]\right) \oplus \ldots \oplus \pi\left(M_{i_1}[l_i]\right) = M_{i_2}[\alpha] \oplus 2^{\alpha}k_0 \oplus 2^{2\alpha}k_1\right] \times \\ &\qquad \Pr\left[2^{l_i}\pi\left(M_{i_1}[1]\right) \oplus \ldots \oplus 2\pi\left(M_{i_1}[l_i]\right) = M_{i_3}[\beta] \oplus 2^{\beta}k_0 \oplus 2^{2\beta}k_1\right].\end{aligned}$$

For fixed indices i_1, i_2, i_3, the above probability is clearly $(2^{-n})^2$, by the randomness of keys k_0 and k_1. Similarly, the probability of occurrence of the remaining four sub-events is also $(2^{-n})^2$. Counting the choices for each index thus gives

$$\Pr[\mathsf{Bad}_6^*] \leq \frac{q^3l^2 + 2qp^2 + 2q^3}{2^{2n}}. \tag{5.42}$$

Bounding $\mathsf{RC}_{\widetilde{\Sigma}}^*$. Recall the offline phase of the ideal oracle (Fig.s 5.6-5.8). Denote the number of elements removed from the construction transcript of an adversary in step 3 of stage II by s_1, and the number of elements removed in step 2 of stage III by s_2. Thus $\widehat{q^0} := q - (s_1 + s_2 + f)$ denotes the number of elements left in $\widetilde{\Sigma}$ at the end of the offline phase, f as in step 10 of stage III. Also let $\widehat{p^0} := \left|\mathsf{Dom}\left(\pi^0\right)\right|$, where the set $\mathsf{Dom}\left(\pi^0\right)$ is as it stands at the end of the offline phase. Thus $\widehat{p^0} = p + (s_1 + s_2)$ (since p is the number of primitive queries with LSB 0). $\widehat{q^1}$ and $\widehat{p^1}$ can be similarly defined. The bad event occurs if for some $i' \neq i$ in $[\widehat{q^0}]$, one of the following occurs:

$$\begin{aligned}(1) \quad &: \left\{\exists\, c \in [\widehat{p^0}]\ \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_i \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i'} \oplus k_2\right) \| 0\right) \wedge \left(z_i = \tilde{z}_c\right)\right\} \\ (2) \quad &: \left\{\exists\, j \in [\widehat{q^0}]\ \left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_i \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i'} \oplus k_2\right) \| 0\right) \wedge \left(z_i = z_j\right)\right\},\end{aligned}$$

where $v_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\pi^0)$.

Bounding (1): The sub-event $z_i = \tilde{z}_c$, i.e. $v_i = T_i \oplus \tilde{z}_c$ is a result of the lazy sampling of v_i, independent of the sub-event $\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_i \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i'} \oplus k_2\right) \| 0$. For a particular choice of i,i' and c,

$$\begin{aligned}&\Pr\left[\left(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_i \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i'} \oplus k_2\right) \| 0\right) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}\right] \\ &\times \Pr\left[z_i = \tilde{z}_c\right] \leq \frac{2\ell + 5}{2^n} \times \Pr\left[z_i = \tilde{z}_c\right] \text{ (as already computed in Eqn. (5.39))} \\ &\qquad \leq \frac{2\ell + 5}{2^n} \cdot \frac{1}{2^n - \widehat{q^0}},\end{aligned}$$

where ℓ denotes the maximum number of message blocks amongst all q queries. Summing over all choices of i, i' and c bounds the probability to

$$\widehat{q^0}(\widehat{q^0}-1)\widehat{p^1}\cdot\frac{(2\ell+5)}{2^{2n}}. \tag{5.43}$$

Bounding (2): We split this bad event into the following cases:

Case I: Suppose $i' \neq j$. As in (1), the sub-event $z_i = z_j$ is a result of the lazy sampling of z_i, independent of the sub-event $\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i'} \oplus k_2)\|0$. Thus, the probability of this case for a particular choice of i,i' and c is

$$\begin{aligned}\mathsf{P}_{i,i',c} &= \Pr[\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i'} \oplus k_2)\|0 \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}] \\ &\times \Pr[z_i = z_j] \leq \frac{2\ell+5}{2^n} \times \frac{1}{2^n - \widehat{q^0}} \text{ (as already computed in Eqn. (5.39))} \\ &\leq \frac{4\ell+10}{2^{2n}} \text{ (since } \widehat{q^0} \leq 2^{n-1}).\end{aligned}$$

Summing over all possible choices of i,i' and c, we obtain an upper bound

$$\widehat{q^0}(\widehat{q^0}-1)(\widehat{q^0}-2)\cdot\frac{4\ell+10}{2^{2n}}. \tag{5.44}$$

Case II: Now suppose $i' = i$. $v_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\pi^0)$ is thus sampled first and $z_{i'}$ is then set to v_i. This case eventually boils down to the joint event $\left(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i'} \oplus k_2)\|0\right) \wedge \left(z_i = z_j\right)$. If $T_i = T_{i'}$, then $z_i = z_j$ is implied by the first sub-event. Therefore,

$$\begin{aligned}&\Pr\left[T_i = T_{i'} \wedge (\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i'} \oplus k_2)\|0) \right. \\ &\qquad \left. \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}}\right] \\ &= \Pr\left[(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i'} \oplus k_2)\|0) \wedge \overline{\mathsf{Coll}} \wedge \overline{\mathsf{3\text{-}Coll}} \right. \\ &\qquad \left. \mid T_i = T_{i'}\right] \cdot \Pr\left[T_i = T_{i'}\right] \\ &\leq \frac{2\ell+5}{2^n} \times \Pr\left[T_i = T_{i'}\right] \text{ (computed in Eqn. (5.39))} \leq \frac{2\ell+5}{2^{2n}}, \end{aligned} \tag{5.45}$$

as all the $\widehat{q^0}$ messages are fixed given $T_1, \ldots, T_{\widehat{q^0}}$. On the other hand, if $T_i \neq T_{i'}$ then $z_i \neq v_{i'} \oplus T_{i'}$ and hence the probability becomes zero. Summing over all (i, i', j) with $i < i'$, the probability for this case is bounded by

$$\frac{\widehat{q^0}(\widehat{q^0}-1)}{2}\cdot\frac{2\ell+5}{2^{2n}}. \tag{5.46}$$

Combining cases I and II, we have

$$\Pr[(2)] \leq \widehat{q^0}(\widehat{q^0}-1)(2\widehat{q^0}-3)\cdot\frac{2\ell+5}{2^{2n}}. \tag{5.47}$$

$$\begin{aligned}\text{Therefore,}\qquad \Pr[\mathsf{RC}^*_\Sigma] &\leq \Pr[(1)]+\Pr[(2)]\\ &\leq \frac{\widehat{q^0}(\widehat{q^0}-1)\cdot(2\ell+5)}{2^{2n}}\left(2\widehat{q^0}-3+\widehat{p^1}\right)\\ &\leq \frac{2q(q-1)(2\ell+5)}{2^{2n}}(q+p)\,, \end{aligned}\tag{5.48}$$

since $\widehat{q^0}\leq q$ and $\widehat{p^1}\leq 2p$.

Bounding RC^*_Θ. The event RC^*_Θ can be bound identically as RC^*_Σ. Hence,

$$\Pr[\mathsf{RC}^*_\Theta] \leq \frac{2q(q-1)(2\ell+5)}{2^{2n}}(q+p)\,. \tag{5.49}$$

The final bound follows from Eqn.s (5.2)–(5.49). □

5.5. Analysis of Good Transcripts

In this section, we show that realizing a good transcript $\tau = (\hat{\tau}_c, \tau_p)$ is almost as likely in the real world as in the ideal world. For each $i \in \mathcal{F}$, both $\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\,\|0$ and $\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2)\,\|1$ are fresh for elements (Σ_i, Θ_i) in $\widetilde{\Sigma}\times\widetilde{\Theta}$, as shown in line 9 of stage II of Fig. 5.7. Due to the changes made in lines 2 and 6 of the same stage, repeating elements of $\widetilde{\Sigma}$ (resp. $\widetilde{\Theta}$) are moved to τ_p, and each such index i is added to $\mathcal{I}$. Since these alterations do not create any inconsistencies, the cardinality of τ_p increases. Assuming that s_1+s_2 elements are added to τ_p in step 3 and t_1+t_2 elements in step 7, the size of the modified transcript τ'_p is $p' := 2p+s_1+s_2+t_1+t_2 = p'_0+p'_1$ (where $p'_0 := p+s_1+s_2$ and $p'_1 := p+t_1+t_2$). Therefore, the number of elements in the modified collections $\widetilde{\Sigma}$ and $\widetilde{\Theta}$, which we denote by $\widetilde{\Sigma}_*$ and $\widetilde{\Theta}_*$ (resp.), is $q' := q-s_1-s_2-t_1-t_2$ at the end of stage II.
Moreover, as the transcript τ is good, for every $i \notin \mathcal{F}\sqcup\mathcal{I}$, exactly one of $\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i\oplus k_2)\,\|0$ and $\mathsf{chop}_{\mathsf{LSB}}(\Theta_i\oplus k_2)\,\|1$ is fresh in $(\widetilde{\Sigma}_*, \widetilde{\Theta}_*)$. Thus, there are exactly $(q'+f)$ fresh blocks ($2f$ fresh blocks corresponding to all indices belonging to $\mathcal{F}$ and $(2q'-2f)/2$ additional fresh blocks), and $q'-f$ repeated blocks.
Let $\mathcal{P}^c$ be the set of all indices corresponding to queries with one of their hash output blocks colliding with one of the hash primitive inputs. We define a relation $\sim$ on $\mathcal{Q}^c := [q]\setminus\mathcal{F}\sqcup\mathcal{I}\sqcup\mathcal{P}^c$ as $i_1\sim i_2$ if $\left(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1}\oplus k_2)\,\|0\right.$

$= \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0\big) \ \vee \ \big(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 1 \ = \ \mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_2} \oplus k_2\right) \| 1\big)$, where Σ_{i_1} is an element of $\widetilde{\Sigma}_*$ and Θ_{i_1} is an element of $\widetilde{\Theta}_*$. Note that as τ is good, for any $i_1 \sim i_2$, exactly one of the following two occurs:

$$\begin{aligned} &\text{(i)} \quad \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_1} \oplus k_2\right) \| 0 = \mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_{i_2} \oplus k_2\right) \| 0, \\ &\text{(ii)} \quad \mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_1} \oplus k_2\right) \| 1 = \mathsf{chop}_{\mathsf{LSB}}\left(\Theta_{i_2} \oplus k_2\right) \| 1. \end{aligned}$$

Furthermore, if i_1 and i_2 are related through (i), then any other index $j \in \mathcal{Q}^c$ cannot be related to i_1 or i_2 through (ii), and vice versa. Clearly, $\sim$ is an equivalence relation. Thus, it partitions $\mathcal{Q}^c$, which in turn induces a partition on $\widetilde{\Sigma}_*$ and $\widetilde{\Theta}_*$. Let r_0 be the number of equivalence classes of $\widetilde{\Sigma}_*$ and r_1 the number of equivalence classes of $\widetilde{\Theta}_*$. Let d_i^0 be the number of elements in the i^{th} equivalence class of $\widetilde{\Sigma}_*$ and d_i^1 the number of elements in the i^{th} equivalence class of $\widetilde{\Theta}_*$. For each equivalence class of $\widetilde{\Sigma}_*$ or $\widetilde{\Theta}_*$, we sample an output for the least-indexed element, thus determining the (common) output for all other elements in that class (see lines 4 and 11 of stage III in Fig. 5.8). Due to the definition of $\mathcal{S}$ in line 12 of stage II, and due to lines $4, 5, 11$ and 12 of stage III $\forall\ i \in [q']$, $v_i \oplus z_i = T_i$ holds. Also, RC_Σ or RC_Θ are not set to 1 (as τ is good), ensuring no range collision for two different inputs. This proves the following result:

For a good transcript τ, the q' tuples of input and output blocks of π^0 and π^1 are permutation compatible, i.e. $\widetilde{\Sigma}_*$ is permutation compatible with $\mathsf{Ran}(\Pi^0) \cup \mathsf{Ran}(\pi^0)$ and $\widetilde{\Theta}_*$ is permutation compatible with $\mathsf{Ran}(\Pi^1) \cup \mathsf{Ran}(\pi^1)$.

This is useful for computing the ratio of the real to ideal interpolation probabilities of a good transcript τ through the following lemma:

Lemma 16. *Let $\tau = (\hat{\tau}_q, \tau_p)$ be a good transcript. Then*

$$\frac{\Pr\left[\mathsf{D}_{\text{re}} = \tau\right]}{\Pr\left[\mathsf{D}_{\text{id}} = \tau\right]} \geq 1 - \frac{16qp^2 + 16q^2p + 4q^3}{2^{2n}}.$$

proof. Ideal Interpolation Probability. Observe that the keys (k_0, k_1, k_2), the response tuple $\widetilde{T}$, and the (lazily sampled) π^0, π^1, Π^0 and Π^1 are jointly independent as each T_i is distributed independent of all the previously sampled values of T, all outputs of π^0 and π^1, the keys k_0, k_1 and k_2 as well as Π^0 and Π^1 (in the offline phase of the game). Let B denote the event $\left\{\left(\Pi^0(\mathsf{chop}_{\mathsf{LSB}}\left(\Sigma_i \oplus k_2\right) \| 0) = v_i\right) \wedge \left(\Pi^1(\mathsf{chop}_{\mathsf{LSB}}\left(\Theta_i \oplus k_2\right) \| 1) = z_i\right) \ \forall i \in \mathcal{F}\right\}$.

Therefore,

$$\Pr[\mathsf{D}_{\mathrm{id}} = \tau] = \frac{1}{2^{3n}} \cdot \frac{1}{2^{nq}} \cdot \frac{1}{(2^n)_{p'_0}} \cdot \frac{1}{(2^n)_{p'_1}} \cdot \Pr\Big[\Pi^0(\mathsf{chop}_{\mathsf{LSB}}\,(\Sigma_i \oplus k_2)\,\|0) = v_i \wedge \Pi^1(\mathsf{chop}_{\mathsf{LSB}}\,(\Theta_i \oplus k_2)\,\|1) = z_i\ \forall i \in [q]\Big]$$
$$= \frac{1}{2^{3n}} \cdot \frac{1}{2^{nq}} \cdot \frac{1}{(2^n)_{p'_0}} \cdot \frac{1}{(2^n)_{p'_1}} \cdot \Pr[\mathsf{B}] \cdot \Pr\Big[\Pi^0(\mathsf{chop}_{\mathsf{LSB}}\,(\Sigma_i \oplus k_2)\,\|0) = v_i \wedge \Pi^1(\mathsf{chop}_{\mathsf{LSB}}\,(\Theta_i \oplus k_2)\,\|1) = z_i\ \forall i \in \mathcal{Q}^c \mid \mathsf{B}\Big]$$
$$= \frac{1}{2^{3n}} \cdot \frac{1}{2^{nq}} \cdot \frac{1}{(2^n)_{p'_0}} \cdot \frac{1}{(2^n)_{p'_1}} \cdot \frac{1}{|\mathcal{S}|} \cdot \frac{1}{(2^n - f - p'_0)_{r_0}} \cdot \frac{1}{(2^n - f - p'_1)_{r_1}}. \quad (5.50)$$

Recall here that Π^0 and Π^1 are defined in two steps:

1. Elements of $\mathcal{S}$ are sampled randomly for all free indices $i \in \mathcal{F}$ (line 13 of stage II in Fig. 5.7) and thus $\Pr[\mathsf{B}] = |\mathcal{S}|^{-1}$.
2. The remaining input-output values of Π^0 and Π^1 are defined through lazy sampling (lines 4, 5, 11 and 12 of stage III in Fig. 5.8).

In the second step of the sampling process, the oracle samples permutation outputs for r_0 and r_1 distinct values in such a manner that neither do they collide with the values sampled in the first step, nor with the values in the modified list τ'_p. Hence, we have

$$\Pr\Big[\Big(\Pi^0(\mathsf{chop}_{\mathsf{LSB}}\,(\Sigma_i \oplus k_2)\,\|0) = v_i\Big) \wedge \Big(\Pi^1(\mathsf{chop}_{\mathsf{LSB}}\,(\Theta_i \oplus k_2)\,\|1) = z_i\Big)\ \forall i \in \mathcal{Q}^c \mid \mathsf{B}\Big] = \frac{1}{(2^n - f - p'_0)_{r_0}} \cdot \frac{1}{(2^n - f - p'_1)_{r_1}}.$$

Real Interpolation Probability. From the claim (5.5) stated previously in this section, it is obvious that $\widetilde{\Sigma}_*$ is permutation compatible with $\mathsf{Ran}(\Pi^0) \cup \mathsf{Ran}(\pi^0)$ and $\widetilde{\Theta}_*$ is permutation compatible with $\mathsf{Ran}(\Pi^1) \cup \mathsf{Ran}(\pi^1)$. Therefore,

$$\sum_{i=1}^{r_0} d_i^0 + \sum_{i=1}^{r_1} d_i^1 = |\mathcal{Q}^c| = (q' - f), \quad (5.51)$$

since the number of non-fresh blocks is $(q' - f)$. We define two sets:

$$\mathcal{U}_0 := \{i \in \mathcal{Q}^c : \mathsf{chop}_{\mathsf{LSB}}\,(\Sigma_i \oplus k_2)\,\|0 \text{ is fresh in } \widetilde{\Sigma}_*\},$$
$$\mathcal{U}_1 := \{i \in \mathcal{Q}^c : \mathsf{chop}_{\mathsf{LSB}}\,(\Theta_i \oplus k_2)\,\|1 \text{ is fresh in } \widetilde{\Theta}_*\}.$$

$$\text{Clearly, } u_0 := |\mathcal{U}_0| = r_0 + f + \sum_{i=1}^{r_1} d_i^1, \quad u_1 := |\mathcal{U}_1| = r_1 + f + \sum_{i=1}^{r_0} d_i^0.$$

One can easily verify that the number of distinct inputs to π^b ($b \in \{0,1\}$) is $\overline{u}_b := u_b + p'_b$. Hence,

$$\Pr[\mathsf{D}_{\mathrm{re}} = \tau] \quad = \quad \frac{1}{2^{3n}} \cdot \frac{1}{(2^n)_{\overline{u}_0}} \cdot \frac{1}{(2^n)_{\overline{u}_1}}. \tag{5.52}$$

Computing the ratio. From Eqn.s (5.52) and (5.50),

$$\begin{aligned} \frac{\Pr[\mathsf{D}_{\mathrm{re}} = \tau]}{\Pr[\mathsf{D}_{\mathrm{id}} = \tau]} &\overset{(2)}{=} \frac{2^{nq} \cdot \left((2^n)_p\right)^2 \cdot (2^n - f - p'_0)_{r_0} \cdot (2^n - f - p'_1)_{r_1} \cdot |\mathcal{S}|}{(2^n)_{\overline{u}_0} \cdot (2^n)_{\overline{u}_1}} \\ &\overset{(3)}{\geq} 2^{n(q-f)} \cdot \mathsf{A}_1 \cdot \mathsf{A}_2 \cdot \left(1 - \frac{4fp'_0p'_1 + 4f^2(p'_0 + p'_1) + 4f^3}{2^{2n}}\right), \\ \text{where} \qquad & \mathsf{A}_1 := \left(\frac{(2^n - p'_0)_f \cdot (2^n - f - p'_0)_{r_0}}{(2^n - p)_{u_1 + s_1}}\right),\ \mathsf{A}_2 := \left(\frac{(2^n - p'_1)_f \cdot (2^n - f - p'_1)_{r_1}}{(2^n - p)_{u_2 + t_1}}\right). \end{aligned} \tag{5.53}$$

Note that (3) follows from $p'_0 = p + s_1$ and $p'_1 = p + t_1$ and the following result from Corollary 5:

$$|\mathcal{S}| \geq \frac{(2^n - p'_0)_f \cdot (2^n - p'_1)_f}{2^{nf}} \cdot \underbrace{\left(1 - \frac{4fp'_0p'_1 + 4f^2(p'_0 + p'_1) + 4f^3}{2^{2n}}\right)}_{\Delta},$$

where we assume that $f + p'_0 \leq 2^{n-1}$ and $f + p'_1 \leq 2^{n-1}$. Furthermore,

$$\mathsf{A}_1 \quad = \quad \left(\frac{(2^n - p'_0)_{f+r_0}}{(2^n - p)_{s_1} \cdot (2^n - p'_0)_{f+r_0} \cdot (2^n - p'_0 - f - r_0)_{\sum_{i=1}^{r_1} d_i^1}}\right)$$

and

$$\mathsf{A}_2 \quad = \quad \left(\frac{(2^n - p'_1)_{f+r_1}}{(2^n - p)_{t_1} \cdot (2^n - p'_1)_{f+r_1} \cdot (2^n - p'_1 - f - r_1)_{\sum_{i=1}^{r_0} d_i^0}}\right)$$

Therefore, from Eqn. (5.53),

$$\mathsf{P} = \frac{2^{n(q-f)} \cdot \Delta}{(2^n - p)_{s_1} \cdot (2^n - p'_0 - f - r_0)_{\sum_{i=1}^{r_1} d_i^1} \cdot (2^n - p)_{t_1} \cdot (2^n - p'_1 - f - r_1)_{\sum_{i=1}^{r_0} d_i^0}}.$$

Due to Eqn. (5.51), the total number of terms in the denominator of P is

$$\sum_{i=1}^{r_0} d_i^0 + \sum_{i=1}^{r_1} d_i^1 + s_1 + t_1 = q' - f + s_1 + t_1 = q - f,$$

as $q' = q - s_1 - t_1$. Not only does this number match exactly with the number of terms in its numerator (except the constant Δ), but also each term of the numerator (except Δ) is greater than each term of the denominator. Thus the term-by-term ratio is at least 1 and hence $\mathsf{P} \geq \Delta$. Finally, the inequalities $f \leq q$, $p'_0 \leq 2p$ and $p'_1 \leq 2p$ prove the result. □

5.6. Summary

This chapter gives a tight security bound of the public permutation-based pPMAC_Plus construction. Unlike PMAC_Plus, which is tightly secure for $2^{3n/4}$ queries, the public permutation-based pPMAC_Plus is tightly secure for $2^{2n/3}$ queries. Similar to pPMAC_Plus, analysing the security of the public permutation-based LightMAC_Plus construction is an interesting open problem.

Construction and Primitive Queries

1: **choose distinct** $(M_i[1]\|M_i[2]) \in \{0,1\}^{2n}\ \forall i \in [2^{2n/3}]$
2: $T_i \leftarrow \mathsf{pPMAC\_Plus}\,(M_i[1]\|M_i[2])\ \forall i \in [2^{2n/3}]$.
3: **choose distinct** $\tilde{u}_b \in \{0,1\}^n\ \forall b \in [2^{2n/3}]$
4: $\tilde{v}_b \leftarrow \pi_1\,(\tilde{u}_b)\ \forall b \in [2^{2n/3}]$.
5: **choose distinct** $\tilde{w}_c \in \{0,1\}^n\ \forall c \in [2^{2n/3}]$
6: $\tilde{z}_c \leftarrow \pi_2\,(\tilde{w}_c)\ \forall c \in [2^{2n/3}]$.
7: $\{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{2^{2n/3+1}}\} \leftarrow \{\tilde{u}_b\}_{b=1}^{2^{2n/3}} \cup \{\tilde{w}_c\}_{c=1}^{2^{2n/3}}$.
8: $\tilde{y}_a \leftarrow \pi_0\,(\tilde{x}_a)\,, a \in [2^{2n/3+1}]$.

Backward Attack

1: $\mathcal{S}_1 \leftarrow \phi$.
2: $\forall (i,b,c) \in [2^{2n/3}] \times [2^{2n/3}] \times [2^{2n/3}]$, if $\tilde{v}_b \oplus \tilde{z}_c = T_i$
3: then $\mathcal{S}_1 \leftarrow \mathcal{S}_1 \cup \{(i,b,c)\}$
4: $\mathcal{S}_2 \leftarrow \phi$.
5: $\forall (a_1, a_2, (i,b,c)) \in [2^{2n/3+1}] \times [2^{2n/3+1}] \times \mathcal{S}_1$,
6: **compute** $\hat{\Sigma}^{(a_1,a_2,(i,b,c))} \leftarrow \tilde{y}_{a_1} \oplus \tilde{y}_{a_2}$
7: **compute** $\hat{\Theta}^{(a_1,a_2,(i,b,c))} \leftarrow 2^2 \cdot \tilde{y}_{a_1} \oplus 2 \cdot \tilde{y}_{a_2}$.
8: if $\hat{\Sigma}_2^{(a_1,a_2,(i,b,c))} \oplus \tilde{u}_b = \hat{\Theta}_2^{(a_1,a_2,(i,b,c))} \oplus \tilde{w}_c$ then
9: $\hat{k}_2^{(a_1,a_2,(i,b,c))} \leftarrow \hat{\Sigma}_2^{(a_1,a_2,(i,b,c))} \oplus \tilde{u}_b$
10: $\mathcal{S}_2 \leftarrow \mathcal{S}_2 \cup \{(a_1, a_2, (i,b,c))\}$.
11: $\forall (a_1, a_2, (i,b,c)) \in \mathcal{S}_2$
12: **compute** $\hat{k}_0^{(a_1,a_2,(i,b,c))} \leftarrow (2^3 \oplus 2^4)^{-1}\,(2 \cdot M_i[1] \oplus M_i[2] \oplus 2 \cdot \tilde{x}_{a_1} \oplus \tilde{x}_{a_2})\,,$
13: **compute** $\hat{k}_1^{(a_1,a_2,(i,b,c))} \leftarrow (2^2 \oplus 2^3)^{-1}\,(2^2 \cdot M_i[1] \oplus M_i[2] \oplus 2^2 \cdot \tilde{x}_{a_1} \oplus \tilde{x}_{a_2})\,.$

Removing False Positives

1: $T_i' \leftarrow \mathsf{pPMAC\_Plus}\,((M_i[1] \oplus 1)\|M_i[2])\ \ \forall i \in [2^{2n/3}]$.
2: $\mathcal{K} \leftarrow \phi$
3: $\forall (a_1, a_2, (i,b,c)) \in \mathcal{S}_2$,
4: if $T_i' = \mathsf{pPMAC\_Plus}^{\left(\hat{k}_0^{(a_1,a_2,(i,b,c))}, \hat{k}_1^{(a_1,a_2,(i,b,c))}, \hat{k}_2^{(a_1,a_2,(i,b,c))}\right)}((M_i[1]\oplus 1)\|M_i[2])\forall i \in [2^{2n/3}]$,
5: then $\mathcal{K} \leftarrow \mathcal{K} \cup \{\hat{k}_0^{(a_1,a_2,(i,b,c))}, \hat{k}_1^{(a_1,a_2,(i,b,c))}, \hat{k}_2^{(a_1,a_2,(i,b,c))}\}$.
6: **return** $\mathcal{K}$.

Figure 5.3.: An attack on pPMAC_Plus, where a computationally unbounded adversary makes $\mathcal{O}\left(2^{2n/3}\right)$ queries to the construction and primitives.

Real-Online

1 : $\mathsf{Dom}(\pi^0) \leftarrow \phi, \mathsf{Dom}(\pi^1) \leftarrow \phi, \mathsf{Ran}(\pi^0) \leftarrow \phi, \mathsf{Ran}(\pi^1) \leftarrow \phi.$

2 : $\forall i \in [q]$, on query M_i, output $T_i \leftarrow \mathsf{pPMAC\_Plus}^*(M_i)$.

3 : $\forall b \in [p]$ such that $\tilde{u}_b = \hat{u}_b \| 0$, on query $(\tilde{u}_b, +)$ to π^0,

4 : output $\tilde{v}_b \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\pi^0) \cup \mathsf{Ran}(\pi^1)$.

5 : $\mathsf{Dom}(\pi^0) \leftarrow \mathsf{Dom}(\pi^0) \cup \{\tilde{u}_b\}$.

6 : $\mathsf{Ran}(\pi^0) \leftarrow \mathsf{Ran}(\pi^0) \cup \{\tilde{v}_b\}$.

7 : $\forall c \in [p]$ such that $\tilde{w}_c = \hat{w}_c \| 1$, on query $(\tilde{w}_c, +)$ to π^1,

8 : output $\tilde{z}_c \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\pi^0) \cup \mathsf{Ran}(\pi^1)$.

9 : $\mathsf{Dom}(\pi^1) \leftarrow \mathsf{Dom}(\pi^1) \cup \{\tilde{w}_c\}$.

10 : $\mathsf{Ran}(\pi^1) \leftarrow \mathsf{Ran}(\pi^1) \cup \{\tilde{z}_c\}$.

11 : $\forall a \in [p]$, on query $(\tilde{y}_a, -)$ to π such that
$\tilde{y}_a \notin \mathsf{Ran}(\pi^0) \cup \mathsf{Ran}(\pi^1)$, output $\tilde{x}_a \xleftarrow{\$} \{0,1\}^n \setminus \left(\mathsf{Dom}(\pi^0) \cup \mathsf{Dom}(\pi^1)\right)$.

12 : if $\mathsf{LSB}(\tilde{x}_a) = 0$,

13 : then $\mathsf{Dom}(\pi^0) \leftarrow \mathsf{Dom}(\pi^0) \cup \{\tilde{x}_a\}, \mathsf{Ran}(\pi^0) \leftarrow \mathsf{Ran}(\pi^0) \cup \{\tilde{y}_a\}$.

14 : else $\mathsf{Dom}(\pi^1) \leftarrow \mathsf{Dom}(\pi^1) \cup \{\tilde{x}_a\}, \mathsf{Ran}(\pi^1) \leftarrow \mathsf{Ran}(\pi^1) \cup \{\tilde{y}_a\}$.

15 : $\mathsf{Dom}(\pi) \leftarrow \mathsf{Dom}(\pi^0) \sqcup \mathsf{Dom}(\pi^1), \mathsf{Ran}(\pi) \leftarrow \mathsf{Ran}(\pi^0) \sqcup \mathsf{Ran}(\pi^1)$.

Figure 5.4.: Description of the online phase of the real world. π^0 is the restriction of the permutation π to the domain $\{\hat{u}\|0 : \hat{u} \in \{0,1\}^{n-1}\}$, and similarly, π^1 is the restriction of the permutation π to the domain $\{\hat{w}\|1 : \hat{w} \in \{0,1\}^{n-1}\}$.

IDEAL-ONLINE

1 : $\mathsf{Dom}(\pi^0) \leftarrow \phi, \mathsf{Dom}(\pi^1) \leftarrow \phi, \mathsf{Ran}(\pi^0) \leftarrow \phi, \mathsf{Ran}(\pi^1) \leftarrow \phi.$

2 : $\forall i \in [q]$, on query M_i, output $T_i \xleftarrow{\$} \{0,1\}^n$.

3 : $\forall b \in [p]$ such that $\tilde{u}_b = \hat{u}_b \| 0$, on query $(\tilde{u}_b, +)$ to π,

4 : output $\tilde{v}_b \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\pi^0) \cup \mathsf{Ran}(\pi^1)$.

5 : $\mathsf{Dom}(\pi^0) \leftarrow \mathsf{Dom}(\pi^0) \cup \{\tilde{u}_b\}$.

6 : $\mathsf{Ran}(\pi^0) \leftarrow \mathsf{Ran}(\pi^0) \cup \{\tilde{v}_b\}$.

7 : $\forall c \in [p]$ such that $\tilde{w}_c = \hat{w}_c \| 1$, on query $(\tilde{w}_c, +)$ to π,

8 : output $\tilde{z}_c \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\pi^0) \cup \mathsf{Ran}(\pi^1)$.

9 : $\mathsf{Dom}(\pi^1) \leftarrow \mathsf{Dom}(\pi^1) \cup \{\tilde{w}_c\}$.

10 : $\mathsf{Ran}(\pi^1) \leftarrow \mathsf{Ran}(\pi^1) \cup \{\tilde{z}_c\}$.

11 : $\forall a \in [p]$, on query $(\tilde{y}_a, -)$ to π such that
$\tilde{y}_a \notin \mathsf{Ran}(\pi^0) \cup \mathsf{Ran}(\pi^1)$, output $\tilde{x}_a \xleftarrow{\$} \{0,1\}^n \setminus \left(\mathsf{Dom}(\pi^0) \cup \mathsf{Dom}(\pi^1)\right)$.

12 : if $\mathsf{LSB}(\tilde{x}_a) = 0$,

13 : then $\mathsf{Dom}(\pi^0) \leftarrow \mathsf{Dom}(\pi^0) \cup \{\tilde{x}_a\}, \mathsf{Ran}(\pi^0) \leftarrow \mathsf{Ran}(\pi^0) \cup \{\tilde{y}_a\}$.

14 : else $\mathsf{Dom}(\pi^1) \leftarrow \mathsf{Dom}(\pi^1) \cup \{\tilde{x}_a\}, \mathsf{Ran}(\pi^1) \leftarrow \mathsf{Ran}(\pi^1) \cup \{\tilde{y}_a\}$.

15 : $\mathsf{Dom}(\pi) \leftarrow \mathsf{Dom}(\pi^0) \sqcup \mathsf{Dom}(\pi^1), \mathsf{Ran}(\pi) \leftarrow \mathsf{Ran}(\pi^0) \sqcup \mathsf{Ran}(\pi^1)$.

Figure 5.5.: Description of the online phase of the ideal world. π^0 is the restriction of the permutation π to the domain $\{\hat{u}\|0 : \hat{u} \in \{0,1\}^{n-1}\}$, and similarly, π^1 is the restriction of the permutation π to the domain $\{\hat{w}\|1 : \hat{w} \in \{0,1\}^{n-1}\}$.

IDEAL-OFFLINE: STAGE I

1 : $(k_0, k_1, k_2) \xleftarrow{\$} (\{0,1\}^n)^3$.

2 : if $\exists i \in [q], \alpha \neq \beta$ in $[l_i]$ and $a_1 \neq a_2$ for which $\tilde{x}_{a_1}, \tilde{x}_{a_2} \in \mathsf{Dom}(\pi)$:
$\left(M_i[\alpha] \oplus 2^{\alpha} k_0 \oplus 2^{2\alpha} k_1 = \tilde{x}_{a_1}\right) \wedge \left(M_i[\beta] \oplus 2^{\beta} k_0 \oplus 2^{2\beta} k_1 = \tilde{x}_{a_2}\right)$,
then $\boxed{\mathsf{Coll} \leftarrow 1,}$ $\perp$.

3 : if $\exists i_1, i_2, i_3 \in [q]$, and distinct $\alpha_1 \in [l_{i_1}], \alpha_2 \in [l_{i_2}], \alpha_3 \in [l_{i_3}]$:
$\left(M_{i_1}[\alpha_1] \oplus 2^{\alpha_1} k_0 \oplus 2^{2\alpha_1} k_1 = M_{i_2}[\alpha_2] \oplus 2^{\alpha_2} k_0 \oplus 2^{2\alpha_2} k_1\right)$
$\wedge \left(M_{i_1}[\alpha_1] \oplus 2^{\alpha_1} k_0 \oplus 2^{2\alpha_1} k_1 = M_{i_3}[\alpha_3] \oplus 2^{\alpha_3} k_0 \oplus 2^{2\alpha_3} k_1\right)$,
then $\boxed{\mathsf{3\text{-}Coll} \leftarrow 1,}$ $\perp$.

4 : $\forall i \in [q]$, $(\Sigma_i, \Theta_i) \leftarrow \mathsf{pPMAC\_Plus\text{-}Hash}^{\pi}_{k_0,k_1,k_2}(M_i)$. / * Subroutine 5.1 * /

5 : $\widetilde{\Sigma} \leftarrow \{\Sigma_1, \ldots, \Sigma_q\}$, $\widetilde{\Theta} \leftarrow \{\Theta_1, \ldots, \Theta_q\}$.

6 : if $\exists i_1, i_2, i_3 \in [q]$ with $i_2 \neq i_1, i_3 \neq i_1$:
$\left(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_2} \oplus k_2) \| 0\right)$
$\wedge \left(\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 = \mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_3} \oplus k_2) \| 1\right)$,
then $\boxed{\mathsf{Bad}_1 \leftarrow 1,}$ $\perp$.

7 : if $\exists i \in [q]$:
$\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0 \in \mathsf{Dom}(\pi^0) \wedge \mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1 \in \mathsf{Dom}(\pi^1)$,
then $\boxed{\mathsf{Bad}_2 \leftarrow 1,}$ $\perp$.

8 : if $\exists i_1 \neq i_2 \in [q]$:
$\left[\left(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_2} \oplus k_2) \| 0\right) \wedge \left(\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 \in \mathsf{Dom}(\pi^1)\right)\right]$
$\vee \left[\left(\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 = \mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_2} \oplus k_2) \| 1\right) \wedge \left(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 \in \mathsf{Dom}(\pi^0)\right)\right]$,
then $\boxed{\mathsf{Bad}_3 \leftarrow 1,}$ $\perp$.

9 : if $\exists i \in [q], b, c \in [p]$:
$\left[\left(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0 = \tilde{u}_b\right) \wedge \left(T_i \oplus \tilde{v}_b = \tilde{z}_c\right)\right]$
$\vee \left[\left(\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1 = \tilde{w}_c\right) \wedge \left(T_i \oplus \tilde{z}_c = \tilde{v}_b\right)\right]$,
then $\boxed{\mathsf{Bad}_4 \leftarrow 1,}$ $\perp$.

10 : if $\exists$ distinct $i_1, i_2 \in [q]$:
$\left[\left(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_2} \oplus k_2) \| 0\right) \wedge \left(T_{i_1} = T_{i_2}\right)\right]$
$\vee \left[\left(\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 = \mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_2} \oplus k_2) \| 1\right) \wedge \left(T_{i_1} = T_{i_2}\right)\right]$,
then $\boxed{\mathsf{Bad}_5 \leftarrow 1,}$ $\perp$.

11 : if $\exists i_1, i_2, i_3 \in [q], b, c \in [p]$ and $\alpha in [l_{i_2}], \beta \in [l_{i_3}]$:
$\left(\left[\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 = x_{i_2}[\alpha]\right] \wedge \left[\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 = x_{i_3}[\beta]\right]\right) \vee$
$\left(\left[\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 = x_{i_2}[\alpha]\right] \wedge \left[\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 = \tilde{w}_c\right]\right) \vee$
$\left(\left[\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 = x_{i_2}[\alpha]\right] \wedge \left[\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 = \mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_3} \oplus k_2) \| 1\right]\right) \vee$
$\left(\left[\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 = \tilde{u}_b\right] \wedge \left[\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 = x_{i_3}[\beta]\right]\right) \vee$
$\left(\left[\mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0 = \mathsf{chop}_{\mathsf{LSB}}(\Sigma_{i_1} \oplus k_2) \| 0\right] \wedge \left[\mathsf{chop}_{\mathsf{LSB}}(\Theta_{i_1} \oplus k_2) \| 1 = x_{i_3}[\beta]\right]\right)$,
then $\boxed{\mathsf{Bad}_6 \leftarrow 1,}$ $\perp$.

12 : go to stage II .

Figure 5.6.: Stage I of the offline phase of the ideal oracle. The internal values $x_i[\alpha]$ are as defined in Eqn. (5.1).

IDEAL-OFFLINE: STAGE II

1 : $\forall i \in [q]$ if $(\exists b \in [p] : \mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0 = \tilde{u}_b) \vee$
$(\exists i_2 \in [q]$ and $\alpha \in [l_{i_2}] : \mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0 = x_{i_2}[\alpha])$, then

2 : $\widetilde{\Sigma} \leftarrow \widetilde{\Sigma} \setminus \Sigma_i$ and $\mathcal{I} \leftarrow \mathcal{I} \cup \{i\}$.

3 : $\mathsf{Dom}(\pi^1) \leftarrow \mathsf{Dom}(\pi^1) \cup \{\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1\}$.

4 : $\mathsf{Ran}(\pi^1) \leftarrow \mathsf{Ran}(\pi^1) \cup \{T_i \oplus \tilde{v}_b\}$.

5 : $\forall i \in [q]$if $(\exists c \in [p] : \mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1 = \tilde{w}_c) \vee$
$(\exists i_2 \in [q]$ and $\alpha \in [l_{i_2}] : \mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1 = x_{i_2}[\alpha])$, then

6 : $\widetilde{\Theta} \leftarrow \widetilde{\Theta} \setminus \Theta_i$ and $\mathcal{I} \leftarrow \mathcal{I} \cup \{i\}$.

7 : $\mathsf{Dom}(\pi^0) \leftarrow \mathsf{Dom}(\pi^0) \cup \{\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0\}$.

8 : $\mathsf{Ran}(\pi^0) \leftarrow \mathsf{Ran}(\pi^0) \cup \{T_i \oplus \tilde{z}_c\}$.

9 : $\mathcal{F} \leftarrow \{i \in [q] \setminus \mathcal{I} : (\Sigma_i \neq \Sigma_{i'}) \wedge (\Theta_i \neq \Theta_{i''})$ for any $i', i'' \neq i$ in $[q] \setminus \mathcal{I}\}$.

10 : $f \leftarrow |\mathcal{F}|$.

11 : $v_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\pi^0)\ \forall i \in \mathcal{F}$.

12 : $\mathcal{S} \leftarrow \{(v_i, z_i) \in \{0,1\}^n \setminus \mathsf{Ran}(\pi^0) \times \{0,1\}^n \setminus \mathsf{Ran}(\pi^1) : v_i \oplus z_i = T_i\}_{i \in \mathcal{F}}$.

13 : `for` $(v_i, z_i) \xleftarrow{\$} \mathcal{S}$:

14 : set $\Pi^0(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0) \leftarrow v_i, \Pi^1(\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1) \leftarrow z_i$.

15 : $\mathsf{Dom}(\Pi^0) \leftarrow \mathsf{Dom}(\Pi^0) \cup \{\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2) \| 0, \}$,
$\mathsf{Ran}(\Pi^0) \leftarrow \mathsf{Ran}(\Pi^0) \cup \{v_i\}$.

16 : $\mathsf{Dom}\left(\Pi^1\right) \leftarrow \mathsf{Dom}\left(\Pi^1\right) \cup \{\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2) \| 1\}$,
$\mathsf{Ran}(\Pi^0) \leftarrow \mathsf{Ran}(\Pi^0) \cup \{v_i\}$.

17 : go to stage III.

Figure 5.7.: Stage II of the offline phase of the ideal oracle.

IDEAL-OFFLINE: STAGE III

1 : $\forall i \in [q]$,
$\mathsf{Dom}(\pi) \leftarrow \mathsf{Dom}(\pi^0) \cup \mathsf{Dom}(\pi^1) \cup \left\{M_i[\alpha] \oplus 2^{\alpha}k_0 \oplus 2^{2\alpha}k_1 \,:\, \alpha \in [l_i]\right\}$,
$\mathsf{Ran}(\pi) \leftarrow \mathsf{Ran}(\pi^0) \cup \mathsf{Ran}(\pi^1) \cup \left\{\pi\left(M_i[\alpha] \oplus 2^{\alpha}k_0 \oplus 2^{2\alpha}k_1\right) \,:\, \alpha \in [l_i]\right\}$.

2 : $\forall i \in [q] \setminus (\mathcal{F} \sqcup \mathcal{I})$ such that $\exists \Sigma_{i'} \in \widetilde{\Sigma}$ with $\Sigma_i = \Sigma_{i'}$,

3 : if $\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0 \notin \mathsf{Dom}(\pi) \cup \mathsf{Dom}(\Pi^0)$,

4 : then $\Pi^0(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0) \leftarrow v_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\Pi^0) \cup \mathsf{Ran}(\pi^0)$ and $z_i \leftarrow T_i \oplus v_i$.

5 : else $v_i \leftarrow \Pi^0(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2\|0)$ and $z_i \leftarrow T_i \oplus v_i$.

6 : $\mathsf{Dom}\left(\Pi^0\right) \leftarrow \mathsf{Dom}\left(\Pi^0\right) \cup \{\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0,\}$,
$\mathsf{Ran}\left(\Pi^0\right) \leftarrow \mathsf{Ran}\left(\Pi^0\right) \cup \{v_i\}$.

7 : $\mathsf{Dom}\left(\Pi^1\right) \leftarrow \mathsf{Dom}\left(\Pi^1\right) \cup \{\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2)\|1\}$,
$\mathsf{Ran}\left(\Pi^1\right) \leftarrow \mathsf{Ran}\left(\Pi^1\right) \cup \{v_i\}$.

8 : if $z_i \in \mathsf{Ran}(\Pi^1) \cup \mathsf{Ran}(\pi^1)$, then $\boxed{\mathsf{RC}_{\Sigma} \leftarrow 1}$,
$\Pi^1(\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2)\|1) \leftarrow z_i, \perp$.

9 : $\forall i \in [q] \setminus (\mathcal{F} \sqcup \mathcal{I})$ such that $\exists \Theta_{i''} \in \widetilde{\Theta}$ with $\Theta_i = \Theta_{i''}$,

10 : if $\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2)\|1 \notin \mathsf{Dom}(\pi) \cup \mathsf{Dom}(\Pi^1)$,

11 : then $\Pi^1(\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2)\|1) \leftarrow z_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\Pi^1) \cup \mathsf{Ran}(\pi^1)$ and $z_i \leftarrow T_i \oplus z_i$.

12 : else $z_i \leftarrow \Pi^1(\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2\|1)$ and $v_i \leftarrow T_i \oplus z_i$.

13 : $\mathsf{Dom}\left(\Pi^0\right) \leftarrow \mathsf{Dom}\left(\Pi^0\right) \cup \{\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0,\}$,
$\mathsf{Ran}\left(\Pi^0\right) \leftarrow \mathsf{Ran}\left(\Pi^0\right) \cup \{v_i\}$.

14 : $\mathsf{Dom}\left(\Pi^1\right) \leftarrow \mathsf{Dom}\left(\Pi^1\right) \cup \{\mathsf{chop}_{\mathsf{LSB}}(\Theta_i \oplus k_2)\|1\}$,
$\mathsf{Ran}\left(\Pi^0\right) \leftarrow \mathsf{Ran}\left(\Pi^0\right) \cup \{v_i\}$.

15 : if $v_i \in \mathsf{Ran}(\Pi^0) \cup \mathsf{Ran}(\pi^0)$, then $\boxed{\mathsf{RC}_{\Theta} \leftarrow 1}$,
$\Pi^0(\mathsf{chop}_{\mathsf{LSB}}(\Sigma_i \oplus k_2)\|0) \leftarrow v_i, \perp$.

16 : $\mathsf{Dom}(\pi) \leftarrow \mathsf{Dom}(\pi^0) \sqcup \mathsf{Dom}(\pi^1)$, $\mathsf{Ran}(\pi) \leftarrow \mathsf{Ran}(\pi^0) \sqcup \mathsf{Ran}(\pi^1)$.

17 : $\mathsf{Dom}(\Pi) \leftarrow \mathsf{Dom}(\Pi^0) \sqcup \mathsf{Dom}(\Pi^1)$, $\mathsf{Ran}(\Pi) \leftarrow \mathsf{Ran}(\Pi^0) \sqcup \mathsf{Ran}(\Pi^1)$.

Figure 5.8.: Stage III of the offline phase of the ideal oracle. Boxed statements denote bad events. Whenever a bad event is set to 1, the game gets immediately aborted (denoted $\perp$) and returns the remaining values of the transcript arbitrarily.

6. Tight Multi-User Security Bound of DbHtS

Abstract

In CRYPTO'21, Shen et al. have proved in the ideal cipher model that the Two-Keyed-DbHtS construction is secure up to $2^{2n/3}$ queries in the multi-user setting independent of the number of users, where the underlying double-block hash function H of Two-Keyed-DbHtS is realized as the concatenation of two independent n-bit keyed hash functions $(\mathsf{H}_{K_h,1}, \mathsf{H}_{K_h,2})$ such that each of the n-bit keyed hash functions is $O(2^{-n})$ universal and regular. They have also demonstrated the applicability of their result to the key-reduced variants of DbHtS MACs, including 2K-SUM-ECBC, 2K-PMAC_Plus and 2K-LightMAC_Plus without requiring domain separation techniques and proved $2n/3$-bit multi-user security of these constructions in the ideal cipher model. Recently, Guo and Wang invalidated the security claim of Shen et al.'s result by exhibiting three constructions, which are the instantiations of the Two-Keyed-DbHtS framework, such that each of their n-bit keyed hash functions is $O(2^{-n})$ universal and regular, while the constructions themselves are secure only up to the birthday bound. In this work, we show a sufficient condition on the underlying Double-block Hash (DbH) function, under which we prove $3n/4$-bit multi-user security of the Two-Keyed-DbHtS construction in the ideal-cipher model. As an instantiation, we show that the two-keyed Polyhash-based DbHtS construction is multi-user secure up to $2^{3n/4}$ queries in the ideal-cipher model. Furthermore, due to the generic attack on DbHtS constructions by Leurent et al. in CRYPTO'18, our derived bound for the construction is tight.

Keywords – DbHtS, PRF, Polyhash, Coefficients-H Technique, Mirror Theory.

6.1. Introduction

Multi-user security of DbHtS. The security bounds of DbHtS constructions discussed in the previous chapter(s) are those in which adversaries are given access to some keyed oracles for a single unknown randomly sampled key. Such a model is known as the *single-user security model*, i.e. when the adversary interacts with one specific machine in which the cryptographic algorithm is deployed and tries to compromise its security. However, in practice, cryptographic algorithms are usually deployed in more than one machine. For example, AES-GCM [101, 74] is now widely used in the TLS protocol to protect web traffic and is currently used by billions of users daily. Thus, the security of DbHtS constructions in the *multi-key setting* is worth investigating; we ask, *"to what extent the number of users will affect the security of* DbHtS *constructions?"*, where adversaries are successful if they compromise the security of one out of many users. Thus, the adversary's winning condition is a disjunction of single-key winning conditions.

The notion of multi-user (mu) security was introduced by Biham [32] in symmetric-key cryptanalysis and by Bellare, Boldyreva, and Micali [11] in the context of public-key encryption. In the multi-user setting, attackers have access to multiple machines such that a particular cryptographic algorithm F is deployed in each machine with independent secret keys. An attacker can adaptively distribute its queries across multiple machines with independent keys. Multi-user security considers attackers that succeed in compromising the security of at least one machine.

Multi-user security for block ciphers is different from multi-user security for modes. In the single-key setting, the best attacks against block cipher such as AES do not improve with increased data complexity. However, in the multi-key environment, they do, as first observed by Biham [32] and later refined as a time-memory-data trade-off by Biryukov et al. [33]. These results demonstrate how one can take advantage of the fact that recovering a block cipher key out of a large group of keys is much easier than targeting a specific key. The same observation can be applied to any deterministic symmetric-key algorithm, as done for MACs by Chatterjee et al.[50]. A more general result guarantees that the *multi-user advantage of an adversary for a cryptographic algorithm is at most u times its single user advantage.* Therefore, for any cryptographic algorithm, a multi-user security bound involving a factor u is easily established using a hybrid argument that shows the upper bound of the adversarial success probability to be roughly u times its single-user security advantage. Bellare and Tackmann [18] first formalized a multi-user secure authenticated encryption scheme and also analyzed countermeasures against multi-key attacks in the context of TLS 1.3. However, they derived a security bound that also contained the factor u. Such a bound implies a significant security drop of the construction when

the number of users is large, and in fact, this is precisely the situation faced in large-scale deployments of AES-GCM such as TLS.

As evident from [14, 18, 42, 85, 86, 99, 111], it is a challenging problem to study the security degradation of cryptographic primitives with the number of users, even when its security is known in the single-user setting. Studies of multi-user security of MACs are somewhat scarce in the literature except for the work of Chatterjee et al. [50], and a very recent work of Andrew et al. [110], and Bellare et al. [14]. The first two consider a generic reduction for MACs, in which the security of the primitive in the multi-user setting is derived by multiplying the number of users u by the single-user security.

In CRYPTO'21, Shen et al. [128] have analyzed the security of DbHtS in the multi-user setting. It is worth noting here that by applying the generic reduction from the single-user to the multi-user setting, the security bound of DbHtS would have capped at worse than the birthday bound, i.e. $uq^{4/3}/2^n$, when each user made a single query and the number of users reached q. Thus, a direct analysis was needed for deriving the multi-user bound of the construction. Shen et al. [128] have shown that in the multi-user setting, the two-keyed [1] DbHtS paradigm,

$$\mathsf{Two\text{-}Keyed\text{-}DbHtS}(M) := \mathsf{E}_K(\mathsf{H}_{K_h,1}(M)) \oplus \mathsf{E}_K(\mathsf{H}_{K_h,2}(M)),$$

is secure up to $2^{2n/3}$ queries in the ideal-cipher model when the $2n$-bit double-block hash function is the concatenation of two independent n-bit keyed hash functions $\mathsf{H}_{K_h,1}$ and $\mathsf{H}_{K_h,2}$. In particular, they have shown that if both $\mathsf{H}_{K_h,1}$ and $\mathsf{H}_{K_h,2}$ are $O(2^{-n})$-regular and $O(2^{-n})$-universal [2], then the multi-user security bound of the two-keyed DbHtS is of the order of

$$\frac{qp\ell}{2k+n} + \frac{q^3}{2^{2n}} + \frac{q^2p + qp^2}{2^{2k}},$$

where q is the total number of MAC queries across all u users, p is the total number of ideal cipher queries, ℓ is the maximum number of message blocks among all queries and n, k are the block size and the key size of the block cipher respectively. Note that the above bound is independent of the number of users u, which can be adaptively chosen by the adversary and grows as large as q. Besides this result, Shen et al. have also shown that 2K-SUM-ECBC [61], 2K-PMAC_Plus [61] and 2K-LightMAC_Plus [61] are all secure roughly up to $2^{2n/3}$ queries (including all MAC and ideal cipher queries) in the multi-user setting independent of the number of users, where these constructions do not employ domain separation techniques.

[1] two-keyed stands for one hash key and one block cipher key.

[2] A family of keyed hash function is said to be ϵ_1-regular if for any x and y, the probability that a randomly sampled hash function from the family maps x to y is ϵ_1; it is said to be ϵ_2-universal if for any distinct x, x', the probability that a randomly sampled hash function from the family yields a collision on the pair (x, x') is ϵ_2.

Remark 4. *In their paper [61], Datta et al. named the two-keyed variants of SUM-ECBC, PMAC_Plus and LightMAC_Plus as 2K-SUM-ECBC, 2K-PMAC_Plus and 2K-LightMAC_Plus respectively, where for each of these constructions, the domain separation technique ensured disjointness of the set of values of Σ and Θ. However, in [128], Shen et al. considered the same constructions but without any domain separation, and refer to them using the same names. Henceforth, we shall implicitly mean the non domain-separated variants only (unless otherwise stated) when referring to the two-keyed constructions 2K-SUM-ECBC, 2K-PMAC_Plus and 2K-LightMAC_Plus.*

6.1.1. Issue with the CRYPTO'21 Paper [128]

In this section, we discuss three issues with [128]. The first two issues examine flaws in the security analysis of the construction and the last issue points out a flawed security claim of the construction. We begin by identifying the first issue. The Two-Keyed-DbHtS framework was proven to be multi-user secure up to $2^{2n/3}$ queries in the ideal-cipher model [128] under the assumption that each of the underlying n-bit independent keyed hash functions is $O(2^{-n})$-universal and regular. As an instantiation of the framework, [128] showed $2n/3$-bit multi-user security of 2K-SUM-ECBC, 2K-LightMAC_Plus and 2K-PMAC_Plus in the ideal-cipher model. In the secuirty proof of these instantiated constructions, they only bounded the regular and the universal advantages of the corresponding hash functions (i.e., the DbH of 2K-SUM-ECBC, 2K-LightMAC_Plus and 2K-PMAC_Plus) up to $O(\ell/2^n)$, where ℓ is the maximum number of message blocks amongst all queries. However, the regular and universal advantages of the underlying double block hash functions of the above three constructions were not proven in the ideal-cipher model; instead, the authors bounded them in the standard model, where the adversary is not allowed to query the underlying block ciphers of the corresponding hash functions. In other words, considering the example of 2K-LightMAC_Plus, while bounding the probability of the event $\Sigma_i = \Sigma_j$ (where $\Sigma_i = \Sigma_j \Rightarrow Y_1^i \oplus Y_2^i \oplus \ldots \oplus Y_{\ell_i}^i = Y_1^j \oplus Y_2^j \oplus \ldots \oplus Y_{\ell_j}^j$ and $Y_a^i = \mathsf{E}_K(M_a^i \| \langle a \rangle_s)$), the authors have simply assumed that at least one of variables Y in the above equation will be fresh, thus providing sufficient entropy for bounding the event. However, the authors have conspicuously missed the fact that existence of such a variable Y may not always be guaranteed in the ideal cipher model. For example, suppose an adversary makes the following three forward primitive queries:

1. forward query with $(x \| \langle 1 \rangle_s)$ and obtains y_1
2. forward query with $(x' \| \langle 1 \rangle_s)$ and obtains y_2
3. forward query with $(x'' \| \langle 2 \rangle_s)$ and obtains y_3

Let us assume that the (albeit probabilistic) event $y_1 \oplus y_2 \oplus y_3 = 0$ occurs. Suppose the adversary makes two more queries: the first, a construction query with (x) and the second, a construction query with $(x'\|x'')$. Then, one cannot find any fresh variable Y in the following equations:

$$Y_1^1 = Y_1^2 \oplus Y_2^2.$$

Therefore, to prove the security of such block cipher-based DbHtS constructions in the ideal-cipher model, one needs to consider the fact that the regular or universal advantage of the underlying double block hash functions must be bounded under the assumption that the adversary makes primitive queries to the underlying block cipher. We therefore believe that to prove the security of the constructions in the ideal-cipher model for the block cipher-based DbH function, one needs to provide a generalized definition of the universal and regular advantages in the ideal-cipher model and prove their security under this model, which was missing in [128].

The second issue is regarding the good transcript analysis of the Two-Keyed-DbHtS construction. In Fig. 4 of [128], the authors have identified the set of $(i,a) \in [u] \times [q_i]$, which they denoted as $F(J)$, such that both Σ_a^i and Θ_a^i are fresh. They have also defined a set $S(J)$,

$$S(J) := \{(W_a^i, X_a^i) \in \{0,1\}^n \setminus \mathsf{Ran}(\Phi_j)^{(2|F(J)|)} : W_a^i \oplus X_a^i = T_a^i\}.$$

Then for all $(i,a) \in F(J)$, (U_a^i, V_a^i) is sampled from $S(J)$ and is set as the permutation output of Σ_a^i and Θ_a^i, respectively. Finally, they have provided a lower bound on the cardinality of the set $S(J)$ from Lemma 2. Noting that Lemma 2 proves the cardinality of the set

$$S := \{(U_i, V_i) \in (\{0,1\}^n)^{(2q)} : U_i \oplus V_i = T_i\}$$

to be at least $2^n(2^n-1)\dots(2^n-2q+1)/2^{nq} \cdot (1-6q^3/2^{2n})$, which is used to obtain a lower bound on $|S(J)|$, reveals a fallacy as the two sets S and $S(J)$ are not isomorphic to each other.

The third issue is regarding the flawed security claim of the Two-Keyed-DbHtS construction in [128]. In Theorem 1 of [128], Shen et al. show that when the underlying double block hash function of the Two-Keyed-DbHtS construction is the concatenation of two independent n-bit keyed hash functions such that each is $O(2^{-n})$-universal and $O(2^{-n})$-regular, Two-Keyed-DbHtS achieves $2n/3$-bit multi-user security in the ideal-cipher model. In a recent work by Guo and Wang [83], the authors came up with three concrete constructions that are instantiations of the Two-Keyed-DbHtS paradigm such that the underlying double block hash function of each of the three constructions is the concatenation of two independent n-bit keyed hash functions. Guo and Wang also show that each of the n-bit hash functions for these three

constructions meets the $O(2^{-n})$-universal and $O(2^{-n})$-regular advantages. However, the constructions have a birthday bound distinguishing attack. As a consequence, the security bound of Two-Keyed-DbHtS as proven in Theorem 1 of [128] stands flawed. We would like to mention here that the attack holds only for those instances of Two-Keyed-DbHtS where the underlying DbH is the concatenation of two independent n-bit hash functions and it does not have any domain separation. In fact, authors of [83] were not able to show any birthday bound attack on 2K-PMAC_Plus and 2K-LightMAC_Plus as the underlying DbH function of these two constructions is not the concatenation of two independent n-bit keyed hash functions. However, it is to be noted that as the double block hash function for 2K-SUM-ECBC is the concatenation of two independent n-bit CBC functions, the attack of [83] holds for it.

6.1.2. Our Contribution

In this chapter, we prove that the Two-Keyed-DbHtS construction is multi-user secure up to $2^{3n/4}$ queries in the ideal-cipher model. To prove it, we first define the notion of a **good** double-block hash function, which informally means that the concatenation of two independent n-bit keyed hash functions is "good" if each has negligible universal and regular advantages, and the probability that the outputs of two hash function colliding for any pair of messages M, M' is zero. Then, we prove that if the underlying $2n$-bit DbH function of the Two-Keyed-DbHtS construction is *good*, such that each of the n-bit keyed hash functions is ϵ_{reg}-regular and ϵ_{univ}-universal, then the multi-user security of our construction in the ideal-cipher model is of the order

$$\frac{9q^{4/3}}{8\cdot 2^n}+\frac{3q^{8/3}}{2\cdot 2^{2n}}+\frac{q^2}{2^{2n}}+\frac{9q^{7/3}}{8\cdot 2^{2n}}+\frac{8q^4}{3\cdot 2^{3n}}+\frac{q}{2^n}+\frac{2u^2}{2^{k_h+k}}+\frac{2q^2}{2^{n+k}}$$
$$+\frac{2qp\epsilon_{\text{reg}}}{2^k}+\frac{q^2\epsilon_{\text{univ}}}{2^n}+\frac{2q^2\epsilon_{\text{reg}}}{2^{k_h}}+3q^{4/3}\epsilon_{\text{univ}}+\frac{q^2\epsilon_{\text{univ}}^2}{2}+\frac{2qp}{2^{n+k}},$$

where q is the total number of MAC queries across all u users, p is the total number of ideal-cipher queries, n is the block size of the block cipher, k_h is the size of the hash key and k is the key size of the block cipher of the construction. As an instantiation of the Two-Keyed-DbHtS framework, we have proved that $\mathsf{C}_2[\text{PH-DbH}, E]$, the Polyhash-based Two-Keyed-DbHtS construction which was proposed in [61] and proven to be secure up to $2^{2n/3}$ queries in the single-user setting, is multi-user secure up to $2^{3n/4}$ queries in the ideal-cipher model. The security proof of the construction crucially depends on a refined result of mirror theory over an abelian group $(\{0,1\}^n, \oplus)$, where one systematically estimates the number of solutions to a system of equations to prove the security of the finalization function of

the construction up to $2^{3n/4}$ queries. Due to the attack result of Leurent et al. [95] on the DbHtS paradigm with $2^{3n/4}$ queries, the multi-user security bound of our construction is tight.

6.2. Mirror Theory

Recall from Sect. 2.3 that a graph $G = (\mathcal{V}, \mathcal{S})$ inducing a system of affine equations over $\{0,1\}^n, \oplus$ is a *good graph* if it has no cycles and if no path label is equal to $\mathbf{0}$. Furthermore, If G is a bipartite graph with a vertex set $\mathcal{V} = P \sqcup Q$ such that:

- each vertex from P represents the left variable in one of the affine equations and the vertex in Q to which an edge joins it represents the right variable in the equation, and
- there are no isolated vertices,

then we shall call G a *good bipartite graph*. Note that a good bipartite graph $\mathcal{G}$ contains no cycle. Therefore, $\mathcal{G}$ can be decomposed into its connected components, all of which are trees; let

$$\mathcal{G} = \mathcal{C}_1 \sqcup \mathcal{C}_2 \sqcup \ldots \sqcup \mathcal{C}_\alpha \sqcup \mathcal{D}_1 \sqcup \mathcal{D}_2 \sqcup \ldots \sqcup \mathcal{D}_\beta$$

for some $\alpha, \beta \geq 0$, where $\mathcal{C}_i$ denotes a component of size greater than 2, and $\mathcal{D}_i$ denotes a component size of 2. We write $\mathcal{C} = \mathcal{C}_1 \sqcup \mathcal{C}_2 \sqcup \ldots \sqcup \mathcal{C}_\alpha$ and $\mathcal{D} = \mathcal{D}_1 \sqcup \mathcal{D}_2 \sqcup \ldots \sqcup \mathcal{D}_\beta$.

Assigning any value to a vertex in P allows the labeled edges to uniquely determine the values of all the other vertices in the component containing P, since $\mathcal{G}$ contains no cycle. The values in the same component are all distinct as $\lambda(\mathcal{P}) \neq 0^n$ for any path $\mathcal{P}$. The number of possible assignments of distinct values to the vertices in $\mathcal{G}$ is $(2^n)_{(|\mathcal{P}|+|\mathcal{Q}|)}$. One may expect that when such an assignment is chosen uniformly at random, it would satisfy all the equations in $\mathcal{G}$ with probability 2^{-nq}, where q denotes the number of edges (i.e., equations) in $\mathcal{G}$. Indeed, we can prove that the number of solutions is closed to $(2^n)_{(|\mathcal{P}|+|\mathcal{Q}|)} / 2^{nq}$, up to a certain error. Formally, we have the following result:

Lemma 17. *Let $\mathcal{G}$ be a good bipartite graph, and let q and q^c denote the number of edges of $\mathcal{G}$ and $\mathcal{C}$, respectively. Let v be the number of vertices of $\mathcal{G}$. If $q < 2^n/8$, then the number of solutions to $\mathcal{G}$, denoted $h(\mathcal{G})$, satisfies*

$$\frac{h(\mathcal{G})2^{nq}}{(2^n)_v} \geq \left(1 - \frac{9(q^c)^2}{8 \cdot 2^n} - \frac{3q^c q^2}{2 \cdot 2^{2n}} - \frac{q^2}{2^{2n}} - \frac{9(q^c)^2 q}{8 \cdot 2^{2n}} - \frac{8q^4}{3 \cdot 2^{3n}}\right).$$

We refer the reader to [92] for a proof of the lemma.

6.3. The Two-Keyed DbHtS Construction

In this section, we describe the Two-Keyed Double-block Hash-then-Sum (Two-Keyed-DbHtS) construction to build a beyond the birthday bound secure variable input-length PRF. Let $\mathsf{H}^1 : \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^n$ and $\mathsf{H}^2 : \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^n$ be two keyed hash functions. Based on H^1 and H^2, we define the Double-block Hash (DbH) function $\mathsf{H} : \mathcal{K}_h \times \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^{2n}$ as follows:

$$\mathsf{H}_{(L_1,L_2)}(M) = (\mathsf{H}^1_{L_1}(M), \mathsf{H}^2_{L_2}(M)). \tag{6.1}$$

We compose this DbH function with a very simple and efficient single-keyed XOR function $\mathsf{XOR}_K(x,y) = \mathsf{E}_K(x) \oplus \mathsf{E}_K(y)$, where E_K is an n-bit block cipher and the block cipher key K is independent of the hash key (L_1, L_2), to realize the two-Keyed-DbHtS construction as follows (For the sake of brevity, we refer to Two-Keyed-DbHtS by simply C_2):

$$\mathsf{C}_2[\mathsf{H},\mathsf{E}](M) := \mathsf{XOR}_K(\mathsf{H}^1_{L_1}(M), \mathsf{H}^2_{L_2}(M)).$$

We use the name Two-Keyed-DbHtS, counting the hash key as one key and the XOR function key (independent of the hash key) as the other. Most beyond the birthday bound secure variable input-length PRFs like 2K-SUM-ECBC, 2K-PMAC_Plus and 2K-LightMAC_Plus are specific instantiations of the Two-Keyed-DbHtS paradigm. These constructions have been proven secure up to $2^{2n/3}$ queries in the standard model [61] for a single-user setting. In [128], all these three constructions have been proven secure up to $2^{2n/3}$ queries in the ideal-cipher model for a multi-user setting. We note here that as the XOR function is not a PRF over two blocks, we can not apply the traditional *Hash-the-PRP* composition result directly on the security analysis of the two-keyed DbHtS. Thus, we need a different type of composition result that utilizes higher security properties of its underlying DbH function instead of merely the universal or regular property.

Definition 5. *Let* $\mathsf{H}^1 : \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^n$ *and* $\mathsf{H}^2 : \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^n$ *be two n-bit keyed hash functions. We say that the double-block hash function* $\mathsf{H} : \mathcal{K}_h \times \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^{2n}$ *defined in Eqn. (6.1) is* good *if it satisfies the following conditions:*

- H^1 *is a family of* ϵ_{reg}*-regular and* ϵ_{univ}*-universal functions.*
- H^2 *is a family of* ϵ_{reg}*-regular and* ϵ_{univ}*-universal functions.*
- *For every* $M, M' \in \{0,1\}^*$,
  $\Pr[L_1 \xleftarrow{\$} \mathcal{K}_h, L_2 \xleftarrow{\$} \mathcal{K}_h : \mathsf{H}^1_{L_1}(M) = \mathsf{H}^2_{L_2}(M')] = 0.$

The first two conditions imply that the regular and universal advantages of both the hash functions should be negligible, whereas the last condition

indicates that the first hash output for any message cannot collide with the second hash output. Having defined the Two-Keyed-DbHtS construction, we now state and prove its security.

Theorem 11. *Let $\mathcal{K}, \mathcal{K}_h$ and $\mathcal{M}$ be three non-empty finite sets. Let $\mathsf{E} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be an n-bit block cipher. Let $\mathsf{H}^1 : \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^n$ and $\mathsf{H}^2 : \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^n$ be two n-bit keyed hash functions such that each is ϵ_{reg}-regular and ϵ_{univ}-universal. Let $\mathsf{H} : \mathcal{K}_h \times \mathcal{K}_h \times \{0,1\}^* \to \{0,1\}^{2n}$ be a* good *double-block hash function as defined in Eqn. (6.1). Then any computationally unbounded distinguisher making a total of q construction queries across all u users and a total of p primitive queries to the block cipher E can distinguish C_2 from an n-bit uniform random function with prf advantage*

$$\begin{aligned}\mathbf{Adv}^{\text{mPRF}}_{\mathsf{C}_2}(u,q,p,\ell) \;\leq\; & \frac{9q^{4/3}}{8\cdot 2^n} + \frac{3q^{8/3}}{2\cdot 2^{2n}} + \frac{q^2}{2^{2n}} + \frac{9q^{7/3}}{8\cdot 2^{2n}} + \frac{8q^4}{3\cdot 2^{3n}} + \frac{q}{2^n} \\ & + \frac{2u^2}{2^{k_h+k}} + \frac{2q^2}{2^{n+k}} + \frac{2qp\epsilon_{\text{reg}}}{2^k} + \frac{q^2\epsilon_{\text{univ}}}{2^n} + \frac{2q^2\epsilon_{\text{reg}}}{2^{k_h}} \\ & + 3q^{4/3}\epsilon_{\text{univ}} + \frac{q^2\epsilon^2_{\text{univ}}}{2} + \frac{2qp}{2^{n+k}}.\end{aligned}$$

6.4. Proof of Theorem 11

We consider a computationally unbounded non-trivial deterministic distinguisher A that interacts with a pair of oracles in either the real world or the ideal world, described as follows: in the real world, A is given access to u independent instances of C_2, i.e., to a tuple of u oracles $(\mathsf{C}_2[(L^i_1, L^i_2, K^i)])_{i\in[u]}$, where each (L^i_1, L^i_2) is independent of (L^j_1, L^j_2), K^i is independent of K^j and $\mathsf{E} \xleftarrow{\$} \mathsf{BC}(\mathcal{K}, \{0,1\}^n)$ is an ideal block cipher. Additionally, A has access to the oracle $\mathsf{E}^{\pm}$, underneath the construction C_2. In the ideal world, A is given access to (i) a tuple of u independent random functions $(\mathsf{RF}_1, \ldots, \mathsf{RF}_u)$, where each RF_i is the random function over $\{0,1\}^*$ to $\{0,1\}^n$ that can be equivalently described as a procedure that returns an n-bit uniform string on input of any arbitrary message, and (ii) the oracle $\mathsf{E}^{\pm}$, where $\mathsf{E} \xleftarrow{\$} \mathsf{BC}(\mathcal{K}, \{0,1\}^n)$ is an ideal block cipher, sampled independently of the sequence of u independent random functions. In both worlds, the first oracle is called the *construction oracle* and the latter, the *ideal cipher oracle*. Using the ideal cipher oracle, a distinguisher A can evaluate any query x under its chosen key J. A query to the construction oracle is called a *construction query* and to that of the ideal cipher oracle is called an *ideal cipher query*. Note that A can make either *forward* (i.e. queries to E with a chosen key and input), or *inverse* (i.e. queries to E^{-1} with a chosen key and input) ideal cipher queries. The ideal oracle is depicted in Fig.s 6.1 and 6.2.

6.4.1. Description of the Ideal World

The ideal world consists of two phases: (i) the online and (ii) the offline phase. Before the game begins, we sample u independent functions $f_1, f_2, \ldots, f_u$ uniformly at random from the set of all functions $\mathsf{Func}(\{0,1\}^*, \{0,1\}^n)$ that map an arbitrary-length string to an n-bit string. We also sample an n-bit block cipher E from the set of all block ciphers with a k-bit key and an n-bit input. In the online phase, when the distinguisher makes the a^{th} construction query for the i^{th} user — M_a^i — to the construction oracle, it returns $T_a^i \leftarrow f_i(M_a^i)$. Similarly, if the distinguisher makes a forward (resp. inverse) primitive query with a chosen block cipher key J and an input x to the ideal cipher oracle, it returns $\mathsf{E}(J,x)$ (resp. $\mathsf{E}^{-1}(J,x)$). However, if any response of the construction queries is an all-zero string 0^n, then the bad flag Bad-Tag is set to 1 and the game is aborted. After this interaction is over,

ONLINE PHASE OF $\mathcal{O}_{\text{ideal}}$

$\mathsf{E} \xleftarrow{\$} \mathsf{BC}(\mathcal{K}, \{0,1\}^n)$;

CONSTRUCTION QUERY:

On a^{th} query of i^{th} user M_a^i, **return** $T_a^i \xleftarrow{\$} \{0,1\}^n$;

if $\exists (i,a) : T_a^i = \mathbf{0}$ then $\boxed{\mathsf{Bad\text{-}Tag} \leftarrow 1}$, $\perp$;

PRIMITIVE QUERY:

On j^{th} forward query with chosen key J^j and input u_α^j,
return $v_\alpha^j \leftarrow \mathsf{E}_{J^j}(u_\alpha^j)$;

On j^{th} backward query with chosen key J^j and input v_α^j,
return $u_\alpha^j \leftarrow \mathsf{E}_{J^j}^{-1}(v_\alpha^j)$;

$\mathsf{Dom}(\mathsf{E}_{J^j}) \leftarrow \mathsf{Dom}(\mathsf{E}_{J^j}) \cup \{u_\alpha^j\}$, $\mathsf{Ran}(\mathsf{E}_{J^j}) \leftarrow \mathsf{Ran}(\mathsf{E}_{J^j}) \cup \{v_\alpha^j\}$;

Figure 6.1.: Online Phase of the Ideal oracle \$: Boxed statements denote bad events. Whenever a bad event is set to 1, the ideal oracle immediately aborts (denoted as $\perp$) and returns the remaining values of the transcript in an arbitrary manner. So, if the game aborts for some bad event, then its previous bad events must not have occurred.

the offline phase begins. In this phase, we sample u pairs of dummy hash keys $(L_1^i, L_2^i)_{i\in[u]} \xleftarrow{\$} \mathcal{K}_h \times \mathcal{K}_h$ and u dummy block cipher keys $(K^i)_{i\in[u]} \xleftarrow{\$} \mathcal{K}$, where L_1^i (resp. L_2^i) is the *left (resp. right) hash key* for the i^{th} user and K^i is its *block cipher key*. If the block cipher key and a left (resp. right) hash key of the i_1^{th} user collides with the block cipher key and left (resp. right) hash key of the i_2^{th} user, then we set the flag BadK to 1 and abort the game. If the game

is not aborted, then we can compute a pair of $2n$-bit hash values (Σ_a^i, Θ_a^i) for all queries across u users, where we often refer to $\Sigma_a^i \leftarrow \mathsf{H}^1_{L_1^i}(M_a^i)$ as the *left hash output* and to $\Theta_a^i \leftarrow \mathsf{H}^2_{L_2^i}(M_a^i)$ as the *right hash output* for the a^{th} query of the i^{th} user.

Now, if the block cipher key of the i^{th} user and the left hash or right hash output for its a^{th} query collides with some chosen ideal cipher key and one of the corresponding inputs of the forward ideal cipher query, then we set the bad flag Bad1 to 1 and abort the game.

For the i^{th} user, if the left or right hash outputs for two of its queries collide and the corresponding responses also collide with each other (i.e., $\Sigma_a^i = \Sigma_b^i, T_a^i = T_b^i$), then we consider it to be a bad event. Similarly, for a pair of users i_1 and i_2, if their left or right hash outputs collide with each other and the corresponding responses also collide with each other, then we again consider it to be a bad event. If at least one of the above bad events occurs, we set Bad2 to 1 and abort the game. We also set another flag Bad3 to 1 and abort the game if for the i^{th} user, the number of the pairs of queries whose either left or right hash outputs collide with each other is at least $q_i^{2/3}$, where q_i is the number of queries made by the i^{th} user. Finally, we set the flag Bad4 to 1 if at least one of the following events holds: (a) for the i^{th} user, two left hash outputs collide and their corresponding right hash outputs also collide, or (b) for the i^{th} user, there exists a tuple of four query indices a, b, c, d such that either (i) $\Sigma_a^i = \Sigma_b^i, \Theta_b^i = \Theta_c^i, \Sigma_c^i = \Sigma_d^i$ or (ii) $\Theta_a^i = \Theta_b^i, \Sigma_b^i = \Sigma_c^i, \Theta_c^i = \Theta_d^i$ holds. As the DbH function H is *good*, Σ_a^i cannot collide with Θ_b^i.

If the game is not aborted at this stage, then it follows that none of the bad events have occurred. All the query-response pairs belong to exactly one of the sets $\mathcal{Q}^{=}$ or $\mathcal{Q}^{\neq}$ as defined in lines 1 and 11 of Fig. 6.3, where $\mathcal{Q}^{=}$ is the set of all queries across all users such that the block cipher key of the i^{th} user collides with an ideal cipher key, but none of its hash outputs collide with any ideal cipher query, and $\mathcal{Q}^{\neq}$ is the set of all queries across all users such that the block cipher key of the i^{th} user does not collide with any ideal cipher key. We also define two additional sets: $\mathcal{I}^{=}$ and $\mathcal{I}^{\neq}$ for $\mathcal{Q}^{=}$ and $\mathcal{Q}^{\neq}$, where $\mathcal{I}^{=}$ (resp. $\mathcal{I}^{\neq}$) is the set of all i such that $(i, \star) \in \mathcal{Q}^{=}$ (resp. $(i, \star) \in \mathcal{Q}^{\neq}$). We partition $\mathcal{I}^{=}$ into r non-empty equivalence classes $\mathcal{I}_1^{=}, \mathcal{I}_2^{=}, \ldots, \mathcal{I}_r^{=}$ based on the relation that the i^{th} user-key K^i collides with J^j if and only if $i \in \mathcal{I}_j^{=}$. Similarly, we partition $\mathcal{I}^{\neq}$ into s equivalence classes based on the equivalence relation $i \sim j$ if and only if $K^i = K^j$. Now, for the j^{th} equivalence class of $\mathcal{I}^{=}$, we consider the tuple

$$\widetilde{\Sigma}_j := \bigcup_{i \in \mathcal{I}_j^{=}} \{(\Sigma_1^i, \Sigma_2^i, \ldots, \Sigma_{q_i}^i)\}, \quad \widetilde{\Theta}_j := \bigcup_{i \in \mathcal{I}_j^{=}} \{(\Theta_1^i, \Theta_2^i, \ldots, \Theta_{q_i}^i)\}.$$

Note that due to the event in line number 7.(b) (resp. 7.(d)) of Fig. 6.2, we

Offline Phase of $\mathcal{O}_{\text{ideal}}$

1 : $(L_1^i, L_2^i)_{i\in[u]} \xleftarrow{\$} \mathcal{K}_h \times \mathcal{K}_h$; $(K^i)_{i\in[u]} \xleftarrow{\$} \mathcal{K}$;

2 : if $\exists b \in \{1,2\}$ and $i_1, i_2 \in [u]$ such that $K^{i_1} = K^{i_2} \wedge L_b^{i_1} = L_b^{i_2}$;

3 : then $\boxed{\mathsf{BadK} \leftarrow 1}$, $\perp$;

4 : $\forall i \in [u], \forall a \in [q_i]$: $(\Sigma_a^i, \Theta_a^i) \leftarrow (\mathsf{H}^1_{L_1^i}(M_a^i), \mathsf{H}^2_{L_2^i}(M_a^i))$;

5 : if one of the following holds:

(a) $\exists i \in [u], j \in [s], u[0]^j_\alpha \in \mathsf{Dom}(\mathsf{E}_{J^j})$, such that $K^i = J^j \wedge \Sigma_a^i = u[0]^j_\alpha$;

(b) $\exists i \in [u], j \in [s], u[1]^j_\alpha \in \mathsf{Dom}(\mathsf{E}_{J^j})$, such that $K^i = J^j \wedge \Theta_a^i = u[1]^j_\alpha$;

6 : then $\boxed{\mathsf{Bad1} \leftarrow 1}$, $\perp$;

7 : if one of the following holds:

(a) $\exists i \in [u], a, b \in [q_i]$, such that $\Sigma_a^i = \Sigma_b^i \wedge T_a^i = T_b^i$;

(b) $\exists i_1, i_2 \in [u], a \in [q_{i_1}], b \in [q_{i_2}]$, such that $K^{i_1} = K^{i_2} \wedge \Sigma_a^{i_1} = \Sigma_b^{i_2}$;

(c) $\exists i \in [u], a, b \in [q_i]$, such that $\Theta_a^{i_1} = \Theta_b^{i_1} \wedge T_a^{i_1} = T_b^{i_1}$;

(d) $\exists i_1, i_2 \in [u], a \in [q_{i_1}], b \in [q_{i_2}]$, such that $K^{i_1} = K^{i_2} \wedge \Theta_a^{i_1} = \Theta_b^{i_2}$;

8 : then $\boxed{\mathsf{Bad2} \leftarrow 1}$, $\perp$;

9 : if one of the following holds:

(a) $\exists i \in [u]$, such that $\left|\left\{(a,b) : \Sigma_a^i = \Sigma_b^i\right\}\right| \geq q_i^{2/3}$;

(b) $\exists i \in [u]$, such that $\left|\left\{(a,b) : \Theta_a^i = \Theta_b^i\right\}\right| \geq q_i^{2/3}$;

10 : then $\boxed{\mathsf{Bad3} \leftarrow 1}$, $\perp$;

11 : if one of the following holds:

(a) $\exists i \in [u], a, b \in [q_i]$ such that $\Sigma_a^i = \Sigma_b^i \wedge \Theta_a^i = \Theta_b^i$;

(b) $\exists i \in [u], a, b, c, d \in [q_i]$ such that $\Sigma_a^i = \Sigma_b^i \wedge \Theta_b^i = \Theta_c^i \wedge \Sigma_c^i = \Sigma_d^i$;

(c) $\exists i \in [u], a, b, c, d \in [q_i]$ such that $\Theta_a^i = \Theta_b^i \wedge \Sigma_b^i = \Sigma_c^i \wedge \Theta_c^i = \Theta_d^i$;

12 : then $\boxed{\mathsf{Bad4} \leftarrow 1}$, $\perp$;

13 : go to subroutine 6.3;

Figure 6.2.: Offline Phase of the Ideal oracle \$: Boxed statements denote bad events. Whenever a bad event is set to 1, the ideal oracle immediately aborts (denoted as $\perp$) and returns the remaining values of the transcript in an arbitrary manner. So, if the game aborts for some bad event, then we may assume that the previous bad events have not occurred.

Offline Phase of $\mathcal{O}_{\text{ideal}}$, Sampling Phase

1 : $\mathcal{Q}^{=} := \{(i,a) \in [u] \times [q_i] : \exists j \in [s],\ K^i = J^j,\ \Sigma^i_a \notin \mathsf{Dom}(\mathsf{E}_{J^j}),\ \Theta^i_a \notin \mathsf{Dom}(\mathsf{E}_{J^j})\}$;

2 : $\mathcal{I}^{=} := \{i \in [u] : (i,\star) \in \mathcal{Q}^{=}\} = \mathcal{I}^{=}_1 \sqcup \mathcal{I}^{=}_2 \sqcup \ldots \sqcup \mathcal{I}^{=}_r$; // $i \in \mathcal{I}^{=}_j \Leftrightarrow K^i = J^j$

3 : $\forall j \in [r] : \widetilde{\Sigma}^j = \bigcup_{i \in \mathcal{I}^{=}_j} \{(\Sigma^i_1, \Sigma^i_2, \ldots, \Sigma^i_{q_i})\},\ \widetilde{\Theta^j} = \bigcup_{i \in \mathcal{I}^{=}_j} \{(\Theta^i_1, \Theta^i_2, \ldots, \Theta^i_{q_i})\}$;

4 : $\forall j \in [r]$ do the following steps:

5 : $\quad \forall i \in \mathcal{I}^{=}_j$ let Σ^i_a be not fresh in $(\Sigma^i_1, \Sigma^i_2, \ldots, \Sigma^i_{q_i})$;

6 : $\quad$ if $\Sigma^i_a \notin \mathsf{Dom}(\mathsf{E}_{J^j})$,

$\quad$ then $\Psi(\Sigma^i_a) \leftarrow Z^i_{1,a} \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\mathsf{E}_{J^j})$, $Z^i_{2,a} \leftarrow Z^i_{1,a} \oplus T^i_a$;

7 : $\quad$ else $Z^i_{1,a} \leftarrow \Psi(\Sigma^i_a)$, $Z^i_{2,a} \leftarrow Z^i_{1,a} \oplus T^i_a$;

8 : $\quad$ if $Z^i_{2,a} \in \mathsf{Ran}(\mathsf{E}_{J^j})$ then $\boxed{\mathsf{Bad\text{-}Samp} \leftarrow 1}$, $\perp$;

9 : $\quad$ else $\mathsf{Dom}(\mathsf{E}_{J^j}) \leftarrow \mathsf{Dom}(\mathsf{E}_{J^j}) \cup \{(\Sigma^i_a, \Theta^i_a)\}$,

$\quad$ $\mathsf{Ran}(\mathsf{E}_{J^j}) \leftarrow \mathsf{Ran}(\mathsf{E}_{J^j}) \cup \{(Z^i_a, Z^i_a \oplus T^i_a)\}$;

10 : $\forall (i,a) \in \mathcal{Q}^{=} : \Psi(\Sigma^i_a) \leftarrow Z^i_{1,a},\ \Psi(\Theta^i_a) \leftarrow Z^i_{2,a}$;

11 : $\mathcal{Q}^{\neq} := \{(i,a) \in [u] \times [q_i] : \forall j \in [s],\ K^i \neq J^j\}$;

12 : $\mathcal{I}^{\neq} := \{i \in [u] : (i,\star) \in \mathcal{Q}^{\neq}\} = \mathcal{I}^{\neq}_1 \sqcup \mathcal{I}^{\neq}_2 \sqcup \ldots \sqcup \mathcal{I}^{\neq}_{r'}$; // $i \in \mathcal{I}^{\neq}_j \Leftrightarrow K^i = K^j$

13 : $\forall j \in [r'] : f_j :=$ distinct number of elements in the tuple $\widetilde{\Sigma}_j \cup \widetilde{\Theta}_j$;

14 : $\forall j \in [r'] : (Z^i_{1,a}, Z^i_{2,a})_{i \in \mathcal{I}^{\neq}_j, a \in [q_i]} \xleftarrow{\$} \mathcal{S}_j$,

$\quad \mathcal{S}_j := \{(Q^i_a, R^i_a)_{i \in \mathcal{I}^{\neq}_j, a \in [q_i]} \in (\{0,1\}^n)^{(f_j)} : Q^i_a \oplus R^i_a = T^i_a\}$;

15 : $\forall j \in [r'] :$ do the following steps:

16 : $\quad \mathsf{Dom}(\mathsf{E}_J) \leftarrow \mathsf{Dom}(\mathsf{E}_J) \cup \{(\Sigma^i_a, \Theta^i_a) : i \in \mathcal{I}^{\neq}_j, a \in [q_i]\}$;

$\quad \mathsf{Ran}(\mathsf{E}_J) \leftarrow \mathsf{Ran}(\mathsf{E}_J) \cup \{(Z^i_{1,a}, Z^i_{2,a}) : i \in \mathcal{I}^{\neq}_j, a \in [q_i]\}$;

17 : $\forall (i,a) \in \mathcal{Q}^{\neq} : \Psi(\Sigma^i_a) \leftarrow Z^i_{1,a}, \Psi(\Theta^i_a) \leftarrow Z^i_{2,a}$;

18 : **return** $(\Sigma^i_a, \Theta^i_a, Z^i_{1,a}, Z^i_{2,a})_{(i,a) \in [u] \times [q_i]}$;

Figure 6.3.: Offline Phase of the Ideal oracle \$, where we sample the output of the hash values.

have $\Sigma^{i_1}_a \neq \Sigma^{i_2}_b$ (resp. $\Theta^{i_1}_a \neq \Theta^{i_2}_b$) for $i_1, i_2 \in \mathcal{I}^{=}_j$ and $a \in [q_{i_1}], b \in [q_{i_2}]$. If Σ^i_a is not fresh in the tuple $(\Sigma^i_1, \Sigma^i_2, \ldots, \Sigma^i_{q_i})$ for some $(i,a) \in \mathcal{I}^{=}_j \times [q_i]$ and the output of Σ^i_a has not been sampled yet, then we sample the its output $Z^i_{1,a}$ from outside the range of E_{J^j} and set the output of Θ^i_a as the xor of Z^i_a and

T_a^i (see line 6 of Fig. 6.3). Otherwise, we set the output of Σ_a^i to the already defined element and adjust the output of the other hash value accordingly (see line 7 of Fig. 6.3). Note that in the latter case, we do not sample the output. In the above adjustment, if the output of Θ_a^i happens to collide with any previously sampled output, then we set flag Bad-Samp to 1 and abort the game (see line 8 of Fig. 6.3). This event cannot hold for the real oracle, as Θ_a^i is fresh in $(\Theta_1^i, \Theta_2^i, \ldots, \Theta_{q_i}^i)$ for $i \in \mathcal{I}_j^=$ and $a \in [q_i]$. If the above flag is not set to 1, then the sampling for the output of Σ_a^i, where $(i,a) \in \mathcal{Q}^=$ preserves permutation compatibility. Finally, for all other $(i,a) \in \mathcal{Q}^{\neq}$, we sample $Z_{1,a}^i$ and $Z_{2,a}^i$ such that $Z_{1,a}^i \oplus Z_{2,a}^i = T_a^i$.

6.4.2. Attack Transcript

We summarize here, the interaction between the distinguisher and the challenger in a transcript. The set of all construction queries for u instances are summarized in a transcript $\tau_c = \tau_c^1 \cup \tau_c^2 \cup \ldots \cup \tau_c^u$, where $\tau_c^i = \{(M_1^i, T_1^i), \ldots, (M_{q_i}^i, T_{q_i}^i)\}$ denotes the query-response transcript generated from the i^{th} instance of the construction. Moreover, we assume that A has chosen s distinct ideal cipher keys $J^1, \ldots, J^s$ such that it makes p_j ideal cipher queries to the block cipher with the chosen key J^j. We summarize the ideal cipher queries in a transcript $\tau_p = \tau_p^1 \cup \tau_p^2 \cup \ldots \cup \tau_p^s$, where $\tau_p^j = \{(u_1^j, v_1^j), \ldots, (u_{p_j}^j, v_{p_j}^j), J^j\}$ denotes the transcript of the ideal cipher queries when the chosen ideal cipher key is J^j. We assume that A makes q_i construction queries for the i^{th} instance and p_j ideal cipher queries (including forward and inverse queries) with chosen ideal cipher key J^j. We also assume that the total number of construction queries across u instances is q, i.e., $q = (q_1 + \ldots + q_u)$ and the total number of ideal cipher queries is $p = (p_1 + \ldots + p_s)$. Since A is non-trivial, none of the transcripts contain any duplicate elements.

We modify the experiment by releasing internal information to A after it has finished its interaction but has not yet output the decision bit. In the real world, we reveal all the keys (L_1^i, L_2^i, K^i) for all u instances used in the construction. In the ideal world, we sample them uniformly at random from their respective key spaces and reveal them to the distinguisher. Once the keys are revealed to the distinguisher, A can compute $(\Sigma_a^i, \Theta_a^i, \Psi(\Sigma_a^i), \Psi(\Theta_a^i))$, where the function Ψ, defined for the ideal world, is given in Fig. 6.3, whereas for the real world, we define Ψ as follows:

$$\Psi(\Sigma_a^i) = \mathsf{E}_{K^i}(\Sigma_a^i), \quad \Psi(\Theta_a^i) = \mathsf{E}_{K^i}(\Theta_a^i).$$

Therefore, each transcript τ_i^c is now modified to include the corresponding intermediate input-output values for the i^{th} instance of the construction.

Thus,

$$\tau_c^i = \{(M_1^i, T_1^i, \Sigma_1^i, \Theta_1^i, \Psi(\Sigma_1^i), \Psi(\Theta_1^i)), \ldots, (M_{q_i}^i, T_{q_i}^i, \Sigma_{q_i}^i, \Theta_{q_i}^i, \Psi(\Sigma_{q_i}^i), \Psi(\Theta_{q_i}^i))\}.$$

In all the following, the complete construction query transcript is

$$\tau_c = \bigcup_{i=1}^{u} \tau_c^i$$

and the complete transcript is $\tau = \tau_c \cup \tau_p$. The modified experiment only makes the distinguisher more powerful and hence the distinguishing advantage of A in this experiment is no less than its distinguishing advantage in the former.

Therefore, to prove the security of the construction using the coeffients-H technique (Theorem 1), we need to identify the set of bad transcripts and compute an upper bound for their probability in the ideal world. Then we find a lower bound for the ratio of the real to ideal interpolation probability for a good transcript. We have already identified the bad transcripts in Fig.s 6.1, 6.2 and 6.3. Therefore, it only remains to bound the probability of bad transcripts in the ideal world and provide a lower bound for the ratio of the real to ideal interpolation probability for a good transcript. Having explained the coeffients-H technique in the view of our construction, it follows that for each $i \in [u]$, $\mathsf{C}_2[\mathsf{E}, (L_1^i, L_2^i), K^i] \mapsto \tau_c^i$ denotes the following:

1. $\Sigma_a^i = (\mathsf{H}^1_{L_1^i}(M_a^i)), \Theta_a^i = (\mathsf{H}^2_{L_2^i}(M_a^i))$,
2. $\mathsf{E}_{K^i}(\Sigma_a^i) = \Psi(\Sigma_a^i), \mathsf{E}_{K^i}(\Theta_a^i) = \Psi(\Theta_a^i)$, and
3. $\mathsf{E}_{K^i}(\Sigma_a^i) \oplus \mathsf{E}_{K^i}(\Theta_a^i) = T_a^i$.

6.4.3. Bounding the Probability of Bad Transcripts

We call a transcript $\tau = (\tau_c, \tau_p)$ **bad** if at least one of the flags is set to 1 during the generation of the transcript in Fig.s 6.1, 6.2 and 6.3. Recall that $\mathsf{Bad\text{-}Tag} \subseteq \Theta$ is the set of all attainable bad transcripts and $\mathsf{GoodT} = \Theta \setminus \mathsf{Bad\text{-}Tag}$ is the set of all attainable good transcripts. We bound the probability of bad transcripts in the ideal world as follows.

Lemma 18. *Let $\tau = (\tau_c, \tau_p)$ be any attainable transcript. Let X_{id} and Θ_{b} be defined as above. Then*

$$\begin{aligned}\Pr[\mathsf{X}_{\mathrm{id}} \in \mathit{Bad\text{-}Tag}] \leq\ & \frac{q}{2^n} + \frac{2u^2}{2^{k_h+k}} + \frac{2qp\epsilon_{\mathrm{reg}}}{2^k} + \frac{q^2\epsilon_{\mathrm{univ}}}{2^n} + \frac{2q^2\epsilon_{\mathrm{reg}}}{2^{k_h}} \\ & + 3q^{4/3}\epsilon_{\mathrm{univ}} + \frac{q^2\epsilon_{\mathrm{univ}}^2}{2} + \frac{2qp}{2^{n+k}} + \frac{2q^2}{2^{n+k}}.\end{aligned}$$

Proof. Using the union bound, we write

$$\Pr[\mathsf{X}_{\mathsf{id}} \in \mathsf{Bad\text{-}Tag}] \leq \Pr[\mathsf{Bad\text{-}Tag}] + \Pr[\mathsf{BadK}] + \sum_{i=1}^{4} \Pr[\mathsf{Badi}] + \Pr[\mathsf{Bad\text{-}Samp}]. \tag{6.2}$$

We individually bound each bad event and then use Eqn. (6.2) to derive the result. In the subsequent analysis, we assume that $|\mathcal{K}_h| = k_h$ and $|\mathcal{K}| = k$.

Bounding Event Bad-Tag

▷ Bounding B.13: For a fixed choice of indices, the probability of the event can be bound by $1/2^n$ as the outputs of the construction queries are sampled uniformly and independently of other random variables. Therefore, by summing over all possible choices of indices, we have

$$\Pr[\mathsf{Bad\text{-}Tag}] \leq \frac{q}{2^n}. \tag{6.3}$$

Bounding Event BadK

▷ Bounding BadK.1: For a fixed choice of indices, the probability of the event can be bound by $1/2^{k_h+k}$ as the event $K^{i_1} = K^{i_2}$ is independent of $L_b^{i_1} = L_b^{i_2}$ for each $b \in \{1,2\}$. Therefore, summing over all possible choices of indices, we have

$$\Pr[\mathsf{BadK}] \leq \frac{2u^2}{2^{k_h+k}}. \tag{6.4}$$

Bounding Event Bad1 | $\overline{\text{BadK}}$

We say that the event Bad1 | $\overline{\mathsf{BadK}}$ holds if either of the events defined in line 5.(a) or in line 5.(b) of Fig. 6.2 holds. We refer to the event defined in line 5.(a) as B.11 and refer to the event defined in line 5.(b) as B.12

▷ Bounding B.11 | $\overline{\mathsf{BadK}}$: For a fixed choice of indices, $\Sigma_a^i = u[0]_\alpha^j$ is bound by the regular advantage of the hash function $\mathsf{H}^1_{L_1^i}$. As the hash key L_1^i is independent of the block cipher key K^i, we have

$$\begin{aligned}\Pr[\mathsf{B.11} \mid \overline{\mathsf{BadK}}] &\leq \sum_{\substack{i\in[u]\\ a\in[q_i]}} \sum_{\substack{j\in[s]\\ \alpha\in[p_j]}} \Pr[K^i = J^j] \cdot \Pr[\Sigma_a^i = u[0]_\alpha^j] \\ &= \sum_{\substack{i\in[u]\\ a\in[q_i]}} \sum_{\substack{j\in[s]\\ \alpha,\beta\in[p_j]}} \epsilon_{\mathrm{reg}} \cdot \frac{1}{2^k} \overset{(1)}{\leq} \frac{qp\epsilon_{\mathrm{reg}}}{2^k},\end{aligned} \tag{6.5}$$

where (1) holds due to the fact that $(q_1 + \ldots + q_u) = q$ and $(p_1^2 + \ldots + p_s^2) \leq p^2$.

▷ Bounding B.12 | $\overline{\text{BadK}}$: With an identical argument, one can show that the probability of the event B.12 can be bounded by $\frac{qp\epsilon_{\text{reg}}}{2^k}$, i.e.,

$$\Pr[\text{B.12} \mid \overline{\text{BadK}}] \leq \frac{qp\epsilon_{\text{reg}}}{2^k}. \tag{6.6}$$

Therefore, combining Eqn. (6.5) and Eqn. (6.6), we have

$$\Pr[\text{Bad1} \mid \overline{\text{BadK}}] = \Pr[\text{B.11} \mid \overline{\text{BadK}} \vee \text{B.12} \mid \overline{\text{BadK}}] \leq \frac{2qp\epsilon_{\text{reg}}}{2^k}. \tag{6.7}$$

Bounding Event Bad2 | $\overline{\text{BadK}}$

We say that the event Bad2 | $\overline{\text{BadK}}$ holds if either of the events defined in line 7.(a) or in line 7.(b) or line 7.(c) or in line 7.(d) of Fig. 6.2 holds. We refer to the event defined in line 7.(a) as B.21, in line 7.(b) as B.22, in line 7.(c) as B.23 and finally in line 7.(d) as B.24

▷ Bounding B.21 | $\overline{\text{BadK}}$: For a fixed choice of indices, we analyze the probability of the event

$$\Sigma_a^i = \Sigma_b^i \wedge T_a^i = T_b^i.$$

Due to independence of the hash key L_1^i and T_a^i, the probability of this joint event can be bound by the universal property of the H^1 hash function and the randomness of T_a^i. Therefore,

$$\Pr[\text{B.21} \mid \overline{\text{BadK}}] \leq \sum_{i\in[u],\ a,b\in[q_i]} \Pr[\Sigma_a^i = \Sigma_b^i \wedge T_a^i = T_b^i] \leq \frac{q^2\epsilon_{\text{univ}}}{2^{n+1}}. \tag{6.8}$$

▷ Bounding B.22 | $\overline{\text{BadK}}$: We bound the event given $\overline{\text{BadK}}$, i.e. even if the block cipher keys for users i_1 and i_2 collide, their corresponding hash keys, i.e., $L_1^{i_1}$ and $L_2^{i_2}$ do not collide. Given this event, for a fixed choice of indices, we bound $\Sigma_a^{i_1} = \Sigma_b^{i_2}$ using the regular property of the hash function H^1 with the randomness of the hash key $L_1^{i_1}$. Moreover, the first event is independent of the second event and can thus be bound exactly by 2^{-k_h}. Therefore,

$$\Pr[\text{B.22} \mid \overline{\text{BadK}}] \leq \sum_{\substack{i_1,i_2\in[u]\\ a\in[q_{i_1}],b\in[q_{i_2}]}} \epsilon_{\text{reg}} \cdot \frac{1}{2^{k_h}} \leq \frac{q^2\epsilon_{\text{reg}}}{2^{k_h}}. \tag{6.9}$$

▷ Bounding B.23 | $\overline{\text{BadK}}$ and B.24 | $\overline{\text{BadK}}$: Bounding B.23 | $\overline{\text{BadK}}$ and B.24 | $\overline{\text{BadK}}$ is identical to bounding B.21 | $\overline{\text{BadK}}$ and B.22 | $\overline{\text{BadK}}$ respectively. Hence,

$$\Pr[\text{B.23} \mid \overline{\text{BadK}}] \leq \frac{q^2\epsilon_{\text{univ}}}{2^{n+1}}, \quad \Pr[\text{B.24} \mid \overline{\text{BadK}}] \leq \frac{q^2\epsilon_{\text{reg}}}{2^{k_h}}. \tag{6.10}$$

Therefore, combining Eqn. (6.8)-Eqn. (6.10),

$$\Pr[\mathsf{Bad2} \mid \overline{\mathsf{BadK}}] \leq \Pr[\mathsf{B.21} \mid \overline{\mathsf{BadK}}] + \Pr[\mathsf{B.22} \mid \overline{\mathsf{BadK}}] + \Pr[\mathsf{B.23} \mid \overline{\mathsf{BadK}}] + \Pr[\mathsf{B.24} \mid \overline{\mathsf{BadK}}] \leq \frac{q^2\epsilon_{\text{univ}}}{2^n} + \frac{2q^2\epsilon_{\text{reg}}}{2^{k_h}}. \tag{6.11}$$

Bounding Event Bad3 | $\overline{\text{BadK}}$

We say that the event Bad3 | $\overline{\text{BadK}}$ holds if either of the events defined in line 9.(a) or in line 9.(b) of Fig. 6.2 holds. We refer to the event defined in line 9.(a) as B.31 and in line 9.(b) as B.32

▷ Bounding B.31 | $\overline{\text{BadK}}$ and B.32 | $\overline{\text{BadK}}$: We first bound the event B.31 | $\overline{\text{BadK}}$. For a fixed choice of indices, we define an indicator random variable $\mathbb{I}^i_{a,b}$ which takes the value 1 if $\Sigma^i_a = \Sigma^i_b$, and 0 otherwise. Let $\mathbb{I}^i = \sum_{a,b} \mathbb{I}^i_{a,b}$. By linearity of expectation,

$$\mathbf{E}[\mathbb{I}^i] = \sum_{a,b} \mathbf{E}[\mathbb{I}^i_{a,b}] = \sum_{a,b} \Pr[\Sigma^i_a = \Sigma^i_b] \leq \frac{q_i^2\epsilon_{\text{univ}}}{2}.$$

Now,

$$\Pr[\mathsf{B.31} \mid \overline{\mathsf{BadK}}] \leq \sum_{i\in[u]} \Pr[|\{(a,b) \in [q_i]^2 : \Sigma^i_a = \Sigma^i_b\}| \geq q_i^{2/3}] = \sum_{i=1}^{u} \Pr[\mathbb{I}^i \geq q_i^{2/3}] \overset{(1)}{=} \sum_{i=1}^{u} \frac{q_i^2\epsilon_{\text{univ}}}{2q_i^{2/3}} \leq \frac{q^{4/3}\epsilon_{\text{univ}}}{2}, \tag{6.12}$$

where (1) holds due to the Markov inequality.
Similar to B.31 | $\overline{\text{BadK}}$, we bound B.32 | $\overline{\text{BadK}}$ as follows:

$$\Pr[\mathsf{B.32} \mid \overline{\mathsf{BadK}}] \leq \frac{q^{4/3}\epsilon_{\text{univ}}}{2}. \tag{6.13}$$

Therefore, combining Eqn. (6.12) and Eqn. (6.13), we have

$$\Pr[\mathsf{Bad3} \mid \overline{\mathsf{BadK}}] = \Pr[\mathsf{B.31} \mid \overline{\mathsf{BadK}} \vee \mathsf{B.32} \mid \overline{\mathsf{BadK}}] \leq q^{4/3}\epsilon_{\text{univ}}. \tag{6.14}$$

Bounding Event Bad4 | $\overline{\text{BadK}}$

We say that the event Bad4 | $\overline{\text{BadK}}$ holds if either of the events defined in line 11.(a) or in line 11.(b) or in line 11.(c) of Fig. 6.2 holds. We refer to the event defined in line 11.(a) as B.41, line 11.(b) as B.42 and in line 11.(c) as B.43.

▷ Bounding B.41 | $\overline{\text{BadK}}$: Due to independence of the hash key L^i_1 and L^i_2, for a fixed choice of indices, the probability of this joint event can be

bound by the universal property of the individual hash functions H^1 and H^2. Therefore, varying over all possible choices of indices, we have

$$\begin{aligned}\Pr[\mathsf{B}.41 \mid \overline{\mathsf{BadK}}] &\leq \sum_{\substack{i\in[u]\\ a,b\in[q_i]}} \Pr[\Sigma_a^i = \Sigma_b^i \wedge \Theta_a^i = \Theta_b^i] \\ &= \sum_{\substack{i\in[u]\\ a,b\in[q_i]}} \Pr[\Sigma_a^i = \Sigma_b^i] \cdot \Pr[\Theta_a^i = \Theta_b^i] \\ &\leq \frac{q^2\epsilon_{\mathrm{univ}}^2}{2}. \end{aligned} \tag{6.15}$$

▷ Bounding $\mathsf{B}.42 \mid \overline{\mathsf{BadK}}$ and $\mathsf{B}.43 \mid \overline{\mathsf{BadK}}$: We first bound the event $\mathsf{B}.42 \mid \overline{\mathsf{BadK}}$. We bound this event given $\overline{\mathsf{B}.31}$. This results in the fact that for a fixed $i \in [u]$, the number of quadruples (a, b, c, d) such that $\Sigma_a^i = \Sigma_b^i$, $\Sigma_c^i = \Sigma_d^i$ holds is at most $q_i^{4/3}$. For a fixed choice of such quadruples, the event $\Theta_b^i = \Theta_c^i$ holds with probability at most ϵ_{univ} due to the universal property of the hash function H^2. Therefore,

$$\Pr[\mathsf{B}.42 \mid \overline{\mathsf{B}.31} \wedge \overline{\mathsf{BadK}}] \leq \sum_{i\in[u]} q_i^{4/3}\epsilon_{\mathrm{univ}} \leq q^{4/3}\epsilon_{\mathrm{univ}}. \tag{6.16}$$

Similar to B.42, we bound B.43 as follows:

$$\Pr[\mathsf{B}.43 \mid \overline{\mathsf{B}.31} \wedge \overline{\mathsf{BadK}}] \leq q^{4/3}\epsilon_{\mathrm{univ}}. \tag{6.17}$$

Combining Eqn. (6.15), Eqn. (6.16) and Eqn. (6.17), we have

$$\Pr[\mathsf{Bad4} \mid \overline{\mathsf{BadK}}] \leq \frac{q^2\epsilon_{\mathrm{univ}}^2}{2} + 2q^{4/3}\epsilon_{\mathrm{univ}}. \tag{6.18}$$

Bounding Event Bad-Samp $\mid \overline{\mathsf{BadK}}$

We consider bounding this event as a union of several events, namely for a fixed $i \in [u], j \in [s]$ and $a \in [q_i]$, we define

$$\mathsf{BS}_{i,j,a} := K^i = J^j \wedge Z_a^i \oplus T_a^i \in \mathsf{Ran}(\mathsf{E}_{J^j}).$$

Then we say that the event Bad-Samp $\mid \overline{\mathsf{BadK}}$ holds if there exists an $i \in [u]$ and $j \in [s]$ such that $\mathsf{BS}_{i,j,a}$ holds, where $Z_a^i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{Ran}(\mathsf{E}_{J^j})$. We first fix an index $j \in [s]$, which determines $\mathcal{I}_j^=$, an index $i \in \mathcal{I}_j^=$ and $a \in [q_i]$. For this choice of indices, the probability that $K^i = J^j \wedge Z_{1,a}^i \oplus T_a^i \in \mathsf{Ran}(\mathsf{E}_{J^j})$ holds is at most $2^{-(k+n)} \cdot (p_j + q_j)$. This is due to the fact that the cardinality of $\mathsf{Ran}(\mathsf{E}_{J^j})$ is bounded above by $(p_j + q_j)$, where q_j is the number of tuples $(\Sigma_a^i, \Theta_a^i)_{i\in\mathcal{I}_j^=, a\in[q_i]}$ which have been added into the set $\mathsf{Dom}(\mathsf{E}_{J^j})$ such that

$K^i = J^j$. Moreover, as the event $K^i = J^j$ is independent of $Z^i_{1,a} \oplus T^i_a \in \mathsf{Ran}(\mathsf{E}_{J^j})$, by taking the union bound, we have

$$\Pr[\text{Bad-Samp}] \leq \sum_{j=1}^{s} \sum_{i \in \mathcal{I}_j^{=}} \sum_{a \in [q_i]} \frac{1}{2^k} \cdot \frac{p_j + q_j}{2^n - (p_j + q_j)} \leq \frac{2qp + 2q^2}{2^{n+k}}. \tag{6.19}$$

Note that the number of choices for (i, a) is at most q and the number of choices for j is s. Thus, summing over all possible choices of (i, j, a) and by assuming $p_j \leq p$ and $p \leq 2^{n-1}$ gives Eqn. (6.19).

Finally, the result follows combining Eqn.s (6.3)-(6.19). □

6.4.4. Analysis of Good Transcripts

In this section, we compute a lower bound for the ratio of the real to ideal interpolation probability for a good transcript. We first consider the set of transcripts $\mathcal{Q}^{=}$. For each $j \in [s]$ and for each $i \in \mathcal{I}_j^{=}$, we consider the sequence

$$\widetilde{\Sigma}^i := (\Sigma^i_1, \Sigma^i_2, \ldots, \Sigma^i_{q_i}), \widetilde{\Theta}^i := (\Theta^i_1, \Theta^i_2, \ldots, \Theta^i_{q_i}).$$

From this sequence, we construct a bipartite graph G_i, where the nodes in one partition represent values Σ^i_a and the nodes in other, Θ^i_a; an edge connects the nodes Σ^i_a and Θ^i_a. If $\Sigma^i_a = \Sigma^i_b$, then we merge the corresponding nodes into a single node, and similarly for $\Theta^i_a = \Theta^i_b$. This allows us to break the graph into w_i components. As the transcript is good, it is easy to see that each component is acyclic and contains a path of length at most 3. Let v_i be the total number of nodes of the graph G_i. Similar to $\mathcal{Q}^{=}$, we consider $\mathcal{Q}^{\neq}$. For each $j \in [r']$ and for each $i \in \mathcal{I}_j^{\neq}$, consider the sequence

$$\widetilde{\Sigma}^i := (\Sigma^i_1, \Sigma^i_2, \ldots, \Sigma^i_{q_i}), \widetilde{\Theta}^i := (\Theta^i_1, \Theta^i_2, \ldots, \Theta^i_{q_i}).$$

Similar to G_i, we construct a bipartite graph H_i, one of whose partitions represents the nodes corresponding to Σ^i_a and the other, the nodes corresponding to Θ^i_a; an edge connects the nodes corresponding to Σ^i_a and Θ^i_a. If two nodes represent the same values, we merge them into a single node. Let w'_i be the number of components of H_i and v'_i be the total number of vertices. Then for a good transcript $\tau = (\tau_c, \tau_p)$, realizing τ is almost as likely in the real world as in the ideal world:

Lemma 19 (Good Lemma). *Let $\tau = (\tau_c, \tau_p) \in \mathsf{GoodT}$ be a good transcript. Let X_{re} and X_{id} be defined as above. Then*

$$\frac{\Pr[\mathsf{X}_{\text{re}} = \tau]}{\Pr[\mathsf{X}_{\text{id}} = \tau]} \geq 1 - \frac{9q^{4/3}}{8 \cdot 2^n} - \frac{3q^{8/3}}{2 \cdot 2^{2n}} - \frac{q^2}{2^{2n}} - \frac{9q^{7/3}}{8 \cdot 2^{2n}} - \frac{8q^4}{3 \cdot 2^{3n}}.$$

Proof. We are now ready to calculate the real interpolation probability. For this, we must bound the total number of input-output pairs on which the block cipher E — with different keys — is executed. As the transcript releases the $2k_h$-bit hash keys and the k-bit block cipher key for each user, it contributes to a term $2^{-(2k_h+k)}$ in the real interpolation probability calculation. Now, for each $j \in [r]$, the block cipher E with key J^j is evaluated on a total of

$$p_j + \sum_{i \in \mathcal{I}_j^{=}} v_i$$

input-output pairs. For the remaining ideal cipher keys, with which none of the users' block cipher keys have collided, we have p_j input-output pairs, which are fixed due to the evaluation of the block cipher with those ideal cipher keys. Moreover, for each $j \in [r']$, the block cipher E is evaluated on a total of $\sum_{i \in \mathcal{I}_j^{\neq}} v'_i$ input-output pairs with key K^j. Summarizing the above,

$$\begin{aligned} \Pr[\mathsf{X}_{\mathrm{re}} = \tau] \;=\; & \prod_{i=1}^{u} \frac{1}{2^{2k_h+k}} \cdot \left(\prod_{j=1}^{r} \frac{1}{(2^n)_{p_j + \sum_{i \in \mathcal{I}_j^{=}} v_i)}} \right) \\ & \cdot \prod_{j \in [s] \setminus [r]} \frac{1}{(2^n)_{p_j)}} \cdot \left(\prod_{j=1}^{r'} \frac{1}{(2^n)_{\sum_{i \in \mathcal{I}_j^{\neq}} v'_i)}} \right). \end{aligned} \tag{6.20}$$

Ideal Interpolation Probability: The term $\prod_{i=1}^{u} 2^{-nq_i}$, which is contributed to the ideal interpolation probability due to the sampling of responses of the adversarial query, samples $2k_h$-bit hash keys and k-bit block cipher keys for all u users. For each $j \in [r]$, and for each $i \in \mathcal{I}_j^{=}$, we construct the graph G_i as defined above. It is easy to see that for each $j \in [r]$ and for each $i \in \mathcal{I}_j^{=}$, the graph G_i *good*. Next, for each $j \in [r]$ and for each $i \in \mathcal{I}_j^{=}$, we sample the value of a node for each component of the graph G_i. Hence, for $j \in [r]$, the total number of sampled points is

$$p_j + \sum_{i \in \mathcal{I}_j^{=}} w_i.$$

Moreover, for each $j \in [s] \setminus [r]$, the total number of sample points is p_j. Subsequently, we consider the set of transcripts $\mathcal{Q}^{\neq}$. For each $j \in [r']$, and for each $i \in \mathcal{I}_j^{\neq}$, we construct the graph H_i as defined above, and compute the set $\mathcal{S}_j$ for each $j \in [r']$ as defined in line 14 of Fig. 6.3 (which is defined

as the number of tuples (Q_a^i, R_a^i) such that $Q_a^i \oplus R_a^i = T_a^i$ for all $i \in \mathcal{I}_j^{\neq}$ and for all $a \in [q_i]$). In summary,

$$\Pr[\mathsf{X}_{\mathrm{id}} = \tau] = \prod_{i=1}^{u} \frac{1}{2^{nq_i}} \cdot \prod_{i=1}^{u} \frac{1}{2^{2k_h+k}} \cdot \left(\prod_{j=1}^{r} \frac{1}{(2^n)_{p_j + \sum_{i \in \mathcal{I}_j^{=}} w_i)}} \right) \cdot \prod_{j \in [s] \setminus [r]} \frac{1}{(2^n)_{p_j)}} \cdot \left(\prod_{j=1}^{r'} \frac{1}{|\mathcal{S}_j|} \right). \tag{6.21}$$

Calculation of the ratio: By plugging in the value of $|\mathcal{S}_j|$ from Lemma 17 into Eqn. (6.21) and then taking the ratio of Eqn. (6.20) to Eqn. (6.21), we

have

$$
\begin{aligned}
\mathsf{p}(\tau) &= \prod_{i=1}^{u} 2^{nq_i} \cdot \prod_{j=1}^{r} \frac{(2^n)_{p_j + \sum_{i \in \mathcal{I}_j^{=}} w_i}}{(2^n)_{p_j + \sum_{i \in \mathcal{I}_j^{=}} v_i}} \cdot \prod_{j=1}^{r'} \frac{|\mathcal{S}_j|}{(2^n)_{\sum_{i \in \mathcal{I}_j^{\neq}} v'_i}} \\
&= \prod_{i=1}^{u} 2^{nq_i} \cdot \prod_{j=1}^{r} \frac{1}{\left(2^n - p_j - \sum_{i \in \mathcal{I}_j^{=}} w_i\right)_{\sum_{i \in \mathcal{I}_j^{=}} (v_i - w_i)}} \\
&\quad \cdot \prod_{j=1}^{r'} \frac{(2^n)_{\sum_{i \in \mathcal{I}_j^{\neq}} v'_i \cdot (1-\epsilon_j)}}{(2^n)_{\left(\sum_{i \in \mathcal{I}_j^{\neq}} v'_i \cdot 2^{\left(n \cdot \sum_{i \in \mathcal{I}_j^{\neq}} (v'_i - w'_i)\right)}\right)}} \\
&= \prod_{i=1}^{u} 2^{nq_i} \cdot \prod_{j=1}^{r} \frac{1}{\left(2^n - p_j - \sum_{i \in \mathcal{I}_j^{=}} w_i\right)_{\sum_{i \in \mathcal{I}_j^{=}} (v_i - w_i)}} \\
&\quad \cdot \prod_{j=1}^{r'} \frac{1}{2^{\left(n \cdot \sum_{i \in \mathcal{I}_j^{\neq}} (v'_i - w'_i)\right)}} \cdot \prod_{j=1}^{r'} (1 - \epsilon_j) \\
&= \prod_{j=1}^{r} \underbrace{\frac{2^{n \sum_{i \in \mathcal{I}_j^{=}} q_i}}{\left(2^n - p_j - \sum_{i \in \mathcal{I}_j^{=}} w_i\right)_{\sum_{i \in \mathcal{I}_j^{=}} (v_i - w_i))}}}_{\geq 1} \cdot \prod_{j=1}^{r'} \underbrace{\frac{2^{\left(n \cdot \sum_{i \in \mathcal{I}_j^{\neq}} q_i\right)}}{2^{\left(n \cdot \sum_{i \in \mathcal{I}_j^{\neq}} (v'_i - w'_i)\right)}}}_{\geq 1} \cdot \prod_{j=1}^{r'} (1 - \epsilon_j) \\
&\geq \left(1 - \sum_{j=1}^{r'} \epsilon_j\right) \\
&\geq 1 - \sum_{j=1}^{r'} \sum_{i \in \mathcal{I}_j^{\neq}} \left(\frac{9(q^{\mathsf{c}})_i^2}{8 \cdot 2^n} + \frac{3q_i^{\mathsf{c}} q_i^2}{2 \cdot 2^{2n}} + \frac{q_i^2}{2^{2n}} + \frac{9(q^{\mathsf{c}})_i^2 q_i}{8 \cdot 2^{2n}} + \frac{8q_i^4}{3 \cdot 2^{3n}}\right)
\end{aligned}
$$

$$\geq 1 - \sum_{j=1}^{r'} \sum_{i \in \mathcal{I}_j^{\neq}} \left(\frac{9q_i^{4/3}}{8 \cdot 2^n} + \frac{3q_i^{8/3}}{2 \cdot 2^{2n}} + \frac{q_i^2}{2^{2n}} + \frac{9q_i^{7/3}}{8 \cdot 2^{2n}} + \frac{8q_i^4}{3 \cdot 2^{3n}} \right)$$

$$\geq 1 - \left(\frac{9q^{4/3}}{8 \cdot 2^n} + \frac{3q^{8/3}}{2 \cdot 2^{2n}} + \frac{q^2}{2^{2n}} + \frac{9q^{7/3}}{8 \cdot 2^{2n}} + \frac{8q^4}{3 \cdot 2^{3n}} \right),$$

since $q_i^c \leq q_i^{2/3}$ for all $i \in \mathcal{I}_j^{\neq}$ such that $j \in [r']$. Note that for each $j \in [r]$, $\sum_{i \in \mathcal{I}_j^{=}} (v_i - w_i)$ denotes the total number of edges in the graph $\bigcup_{i \in \mathcal{I}_j^{=}} G_i$, which is $\sum_{i \in \mathcal{I}_j^{=}} q_i$. Similarly, for each $j \in [r']$, $\sum_{i \in \mathcal{I}_j^{\neq}} (v'_i - w'_i)$ denotes the total number of edges in the graph $\bigcup_{i \in \mathcal{I}_j^{\neq}} H_i$, which is $\sum_{i \in \mathcal{I}_j^{\neq}} q_i$.

6.5. Instantiation of the Two-Keyed-DbHtS with PolyHash

PolyHash [37, 31, 131] is a very efficient algebraic hash function. For a fixed natural number n, it first samples an n-bit key L uniformly at random from $\{0,1\}^n$. To apply this function on a message $M \in \{0,1\}^*$, we first apply an injective padding function 10^* (i.e. append a bit 1 followed by a minimum number of zeroes to the message M so that the total number of bits in the padded message becomes a multiple of n). Let the padded message be $M^* = M_1 \| M_2 \| \ldots \| M_l$, where l is the number of n-bit blocks in it. Then, we define the PolyHash function as follows:

$$\mathsf{PH}(L, M^*) := M_1 \cdot L^l \oplus M_2 \cdot L^{l-1} \oplus \ldots \oplus M_l \cdot L,$$

where l is the number of blocks of M. If the size of the message M is a multiple of n, then we do not apply the padding function and apply the PolyHash function on the message M itself as follows:

$$\mathsf{PH}(L, M) = M_1 \cdot L^l \oplus M_2 \cdot L^{l-1} \oplus \ldots \oplus M_l \cdot L.$$

In both equations, the multiplications are defined in the field $\mathrm{GF}(2^n)$. Then PolyHash [108] is $\ell/2^n$-regular, $\ell/2^n$-axu and $\ell/2^n$-universal, where ℓ is the maximum number of message blocks (the proof of the lemma is related to a result on the number of distinct roots of a polynomial):

Lemma 20. *Let* PH *be the PolyHash function as defined above. Then* PH *is* $\ell/2^n$*-regular,* $\ell/2^n$*-almost-xor universal and* $\ell/2^n$*-universal.*

Proof. We first compute the regular advantage of the hash function. Clearly, $\mathsf{PH}(L, M) = \Delta$ is a polynomial in L with constant term $\Delta$with degree at most ℓ. $\mathsf{PH}(L, M) \oplus \Delta$ is a non-zero polynomial and hence, $\epsilon_{\text{reg}} = \ell/2^n$, where ℓ is the maximum number of message blocks amongst all q messages, as the maximum number of roots for the polynomial $\mathsf{PH}(L, M) \oplus \Delta$ is ℓ. Moreover, for any two distinct messages M and M', $\mathsf{PH}(L, M) \oplus \mathsf{PH}(L, M') \oplus \Delta$ is a non-zero polynomial in L with degree at most ℓ, and hence the maximum number of roots this polynomial can have is ℓ. Therefore, the almost-xor-universal advantage of PH is $\ell/2^n$. □

From Lemma 20, a simple corollary immediately follows:

Corollary 3. *Let* $\mathsf{fix}_b(\mathsf{PH})$ *be the variant of the PolyHash function in which the least significant bit of the n-bit output of the function is fixed to bit b. Then,* $\mathsf{fix}_b(\mathsf{PH})$ *is a* $2\ell/2^n$*-regular,* $2\ell/2^n$*-almost-xor universal and* $2\ell/2^n$*-universal hash function.*

We now define the PolyHash-based double-block hash function, (PH-DbH function):

$$\mathsf{PH\text{-}DbH}(L_1, L_2, M) := \left(\underbrace{\mathsf{fix}_0(\mathsf{PH}(L_1, M))}_{\mathsf{H}^1_{L_1}}, \underbrace{\mathsf{fix}_1(\mathsf{PH}(L_2, M))}_{\mathsf{H}^2_{L_2}} \right). \tag{6.22}$$

Thus, two independent instances of the PolyHash function keyed with two independent keys L_1 and L_2 are applied separately to a message M, and the least significant bit of their output is replaced with bits 0 and 1 respectively. The PolyHash-based DbHtS construction can now be defined directly from the Two-Keyed-DbHtS construction as follows: encrypt $\mathsf{fix}_0(\mathsf{PH}(K_1, M))$ and $\mathsf{fix}_1(\mathsf{PH}(K_2, M))$ through a block cipher E_K and xor the result together to produce the output. An algorithmic description of the construction is shown in Fig. 6.4. Clearly, the PH-DbH function is a good double-block hash function

$\mathsf{PH\text{-}DbHtS}(K_1, K_2, K, M)$

1: $\Sigma = \mathsf{fix}_0(\mathsf{PH}(K_1, M))$;
2: $\Theta = \mathsf{fix}_1(\mathsf{PH}(K_2, M))$;
3: $T = \mathsf{E}_K(\Sigma) \oplus \mathsf{E}_K(\Theta)$;
return T;

$\mathsf{PH}(L, M)$

1: $M_1 \| \ldots \| M_\ell \xleftarrow{n} M \| 10^*$;
2: $Y = M_1 \cdot L^\ell \oplus M_2 \cdot L^{\ell-1} \oplus \cdots \oplus M_\ell \cdot L$;
return Y;

Figure 6.4.: The PH-DbHtS construction with PH-DbH as the underlying double-block hash function. $M_1 \| M_2 \| \ldots \| M_\ell \xleftarrow{n} M \| 10^*$ denotes the parsing of message $M \| 10^*$ into n bit strings.

as the individual hash functions H^1 and H^2 are both $2\ell/2^n$-regular and universal. Furthermore, for a randomly chosen pair of keys L_1, L_2, and for any pair of messages $M, M' \in \{0,1\}^*$,

$$\Pr[\mathsf{fix}_0(\mathsf{PH}(L_1, M)) = \mathsf{fix}_1(\mathsf{PH}(L, M'))] = 0.$$

Therefore, combining the Corollary 3 with Theorem 11, we derive the following security of PolyHash-based DbHtS (for the sake of brevity, we write Π to denote the PH-DbHtS construction):

Theorem 12. *Let $\mathcal{K}$ be a non-empty finite set. Let $\mathsf{E} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be an n-bit block cipher and $\mathsf{PH\text{-}DbH} : (\{0,1\}^n \times \{0,1\}^n) \times \{0,1\}^* \to (\{0,1\}^n)^2$ be the PolyHash-based double-block hash function as defined above. Then any computationally unbounded distinguisher making a total of q construction queries across all u users such that each queried message is at most ℓ blocks long with $\ell \leq 2^{n-2}$ and a total of p primitive queries to the block cipher E can distinguish Π from an n-bit uniform random function with advantage*

$$\begin{aligned}\mathbf{Adv}^{\mathrm{mPRF}}_{\Pi}(u,q,p,\ell) \leq\ & \frac{9q^{4/3}}{8\cdot 2^n} + \frac{3q^{8/3}}{2\cdot 2^{2n}} + \frac{q^2}{2^{2n}} + \frac{9q^{7/3}}{8\cdot 2^{2n}} + \frac{8q^4}{3\cdot 2^{3n}} + \frac{q}{2^n} \\ & + \frac{2u^2}{2^{n+k}} + \frac{4qp\ell}{2^{n+k}} + \frac{4q^2\ell}{2^{2n}} + \frac{4q^2\ell}{2^{n+k}} + \frac{8q^{4/3}\ell}{2^n} + \frac{4q^2\ell^2}{2^{2n}} + \frac{2qp}{2^{n+k}} + \frac{2q^2}{2^{n+k}}.\end{aligned}$$

Remark 5. *We would like to mention that the definition of the PolyHash function used in this chapter is different from that used in [83]. Nevertheless, one can also establish the $3n/4$-bit multi-user security of the PolyHash-based DbHtS construction with the PolyHash function used in [83].*

6.6. Summary

In this chapter, we have shown that the Two-Keyed DbHtS construction is multi-user secure up to $2^{3n/4}$ queries in the ideal cipher model. As an instantiation of the result, we have shown that Polyhash-based DbHtS provides $3n/4$-bit multi-user security in the ideal cipher model. Combining it with the generic result on the attack complexity of DbHtS makes the bound tight. However, we cannot apply this result to analyze the security of 2K-SUM-ECBC, 2K-PMAC_Plus and 2K-LightMAC_Plus. This is because their underlying DbH functions are based on block ciphers, and our proof technique does not support analysis of their security in the ideal cipher model as the underlying DbH function of these constructions is built from block ciphers. We believe that proving $3n/4$-bit security of the DbHtS construction based on block cipher-based double-block hash functions needs a careful study.

7. Conclusion

Various designs of message authentication codes that possess beyond the birthday bound security were detailed in this work.

Chapter 1 proposed a nonce-based, block cipher-based construction nEHtM (nonce-based Enhanced Hash-then-Mask), which is $2n/3$-bit secure when the nonce is respected. It also introduced a concept of faulty nonces and showed that the security of nEHtM degrades gracefully with the number of faulty nonces. Chapter 2 proposed a permutation-based MAC dubbed PDMMAC (Permutation-based Davies-Meyer MAC) and its variants, thus obtaining pseudorandom functions from PRPs. It proved $2n/3$-bit tight security for these constructions. In continuation, chapter 3 proposed the pEDM (permutation-based Encrypted Davies-Meyer) MAC, which is an inverse-free permutation-based MAC with a single instance of the permutation. This construction was shown to possess tight $2n/3$-bit security. Next, chapter 4 proposed another permutation-based construction called the p-DbHtS (permutation-based Double-block Hash-then-Sum), which also has tight $2n/3$-bit security. Chapter 5 continued exploration of the DbHtS construction, proving tight $3n/4$-bit security of the block cipher-based DbHtS in the multi-user setting. It also described a PolyHash-based instantiation of the construction with the same security.

Furthermore, this work also extended Patarin's Mirror Theory with some new theorems, as well as other results such as multicollision theorems, variants of the sum-capture lemma and other counting results.

Bibliography

[1] A.Joux. "Comments on the draft GCM specification – authentication failures in NIST version of GCM." In: () (cit. on p. 28).

[2] Elena Andreeva et al. "Security of Keyed Sponge Constructions Using a Modular Proof Approach." In: *FSE 2015*. 2015, pp. 364–384. DOI: 10.1007/978-3-662-48116-5_18. URL: https://doi.org/10.1007/978-3-662-48116-5%5C_18 (cit. on p. 65).

[3] Kazumaro Aoki and Kan Yasuda. "The Security and Performance of "GCM" when Short Multiplications Are Used Instead." In: *Information Security and Cryptology - 8th International Conference, Inscrypt 2012, Beijing, China, November 28-30, 2012, Revised Selected Papers*. 2012, pp. 225–245 (cit. on pp. 3, 14, 34).

[4] Tomer Ashur, Orr Dunkelman, and Atul Luykx. "Boosting Authenticated Encryption Robustness with Minimal Modifications." In: *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*. 2017, pp. 3–33 (cit. on pp. 3, 14, 34, 35).

[5] B.Smith. "Pull request: Removing the AEAD explicit IV. Mail to IETF TLS Working Group." In: (2015) (cit. on pp. 2, 14).

[6] László Babai. "The Fourier Transform and Equations over Finite Abelian Groups (Lec- ture Notes, version 1.3)." In: (2002). URL: http://people.cs.uchicago.edu/~laci/reu02/%20fourier.pdf (cit. on p. 211).

[7] Subhadeep Banik et al. "GIFT: A Small Present - Towards Reaching the Limit of Lightweight Encryption." In: *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*. Ed. by Wieland Fischer and Naofumi Homma. Vol. 10529. Lecture Notes in Computer Science. Springer, 2017, pp. 321–345 (cit. on pp. 2, 3, 15, 19).

[8] Ray Beaulieu et al. "The SIMON and SPECK Families of Lightweight Block Ciphers." In: *IACR Cryptology ePrint Archive* 2013 (2013), p. 404. URL: http://dblp.uni-trier.de/db/journals/iacr/iacr2013.html#BeaulieuSSTWW13 (cit. on p. 15).

[9] Christof Beierle et al. "The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS." In: *CRYPTO 2016, Part II*. 2016, pp. 123–153 (cit. on p. 15).

[10] M. Bellare and R. Impagliazzo. *A tool for obtaining tighter security analyses of pseudorandom function based constructions, with applications to PRP to PRF conversion*. Cryptology ePrint Archive, Report 1999/024. http://eprint.iacr.org/1999/024. 1999 (cit. on pp. 15, 19).

[11] Mihir Bellare, Alexandra Boldyreva, and Silvio Micali. "Public-Key Encryption in a Multi-user Setting: Security Proofs and Improvements." In: *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*. Ed. by Bart Preneel. Vol. 1807. Lecture Notes in Computer Science. Springer, 2000, pp. 259–274 (cit. on p. 168).

[12] Mihir Bellare, Joe Kilian, and Phillip Rogaway. "The Security of Cipher Block Chaining." In: *Advances in Cryptology - CRYPTO '94, 14th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1994, Proceedings*. Ed. by Yvo Desmedt. Vol. 839. Lecture Notes in Computer Science. Springer, 1994, pp. 341–358 (cit. on p. 19).

[13] Mihir Bellare, Joe Kilian, and Phillip Rogaway. "The Security of the Cipher Block Chaining Message Authentication Code." In: *J. Comput. Syst. Sci.* 61.3 (2000), pp. 362–399 (cit. on pp. 2, 3, 15).

[14] Mihir Bellare, Ted Krovetz, and Phillip Rogaway. "Luby-Rackoff Backwards: Increasing Security by Making Block Ciphers Non-invertible." In: *EUROCRYPT '98, Proceeding*. 1998, pp. 266–280 (cit. on pp. 15, 18, 19, 169).

[15] Mihir Bellare and Chanathip Namprempre. "Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm." In: *Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings*. 2000, pp. 531–545 (cit. on pp. 3, 14, 32).

[16] Mihir Bellare and Phillip Rogaway. "Random Oracles Are Practical: A Paradigm for Designing Efficient Protocols." In: *Proceedings of the 1st ACM Conference on Computer and Communications Security*. CCS '93. Fairfax, Virginia, USA: Association for Computing Machinery, 1993, pp. 62–73. ISBN: 0897916298. DOI: 10.1145/168588.168596. URL: https://doi.org/10.1145/168588.168596 (cit. on p. 8).

[17] Mihir Bellare and Phillip Rogaway. "The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs." In: *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings*. Ed. by Serge Vaudenay.

Vol. 4004. Lecture Notes in Computer Science. Springer, 2006, pp. 409–426 (cit. on pp. 15, 19).

[18] Mihir Bellare and Björn Tackmann. "The Multi-user Security of Authenticated Encryption: AES-GCM in TLS 1.3." In: *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part I*. 2016, pp. 247–276 (cit. on pp. 2, 14, 168, 169).

[19] Daniel J. Bernstein. "The Poly1305-AES Message-Authentication Code." In: *Fast Software Encryption: 12th International Workshop, FSE 2005, Paris, France, February 21-23, 2005, Revised Selected Papers*. 2005, pp. 32–49 (cit. on pp. 2, 14).

[20] Daniel J. Bernstein et al. "Gimli : A Cross-Platform Permutation." In: *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*. 2017, pp. 299–320 (cit. on pp. 18, 131).

[21] Guido Bertoni et al. "Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications." In: *Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*. 2011, pp. 320–337 (cit. on p. 65).

[22] Guido Bertoni et al. "Farfalle: parallel permutation-based cryptography." In: *IACR Cryptol. ePrint Arch.* 2016 (2016), p. 1188 (cit. on p. 18).

[23] Guido Bertoni et al. "Keccak." In: *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*. 2013, pp. 313–314 (cit. on pp. 18, 131).

[24] Guido Bertoni et al. "Keccak." In: *IACR Cryptology ePrint Archive* 2015 (2015), p. 389. URL: http://eprint.iacr.org/2015/389 (cit. on p. 65).

[25] Guido Bertoni et al. *On the Security of the Keyed Sponge Construction. In* Symmetric Key Encryption Workshop. 2011 (cit. on p. 65).

[26] Tim Beyne et al. "Elephant." In: *NIST LWC* (2019). URL: https://csrc.nist.gov/CSRC/media/Projects/Lightweight-Cryptography/documents/round-1/spec-doc/elephant-spec.pdf (cit. on pp. 18, 19, 131).

[27] Srimanta Bhattacharya and Mridul Nandi. "Revisiting Variable Output Length XOR Pseudorandom Function." In: *IACR Trans. Symmetric Cryptol.* 2018.1 (2018), pp. 314–335 (cit. on p. 34).

[28] Arghya Bhattarcharjee et al. "CENCPP - Beyond-birthday-secure Encryption from Public Permutations." In: *IACR Cryptol. ePrint Arch.* 2020 (2020), p. 602 (cit. on pp. 96, 98, 130–132).

[29] Ritam Bhaumik and Mridul Nandi. "OleF: an Inverse-Free Online Cipher. An Online SPRP with an Optimal Inverse-Free Construction." In: *IACR Trans. Symmetric Cryptol.* 2016.2 (2016), pp. 30–51 (cit. on p. 97).

[30] Ritam Bhaumik et al. "The Iterated Random Function Problem." In: *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Vol. 10625. Lecture Notes in Computer Science. Springer, 2017, pp. 667–697 (cit. on p. 18).

[31] Jürgen Bierbrauer et al. "On Families of Hash Functions via Geometric Codes and Concatenation." In: *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*. 1993, pp. 331–342 (cit. on p. 190).

[32] Eli Biham. "How to decrypt or even substitute DES-encrypted messages in 2^{28} steps." In: *Inf. Process. Lett.* 84.3 (2002), pp. 117–124 (cit. on p. 168).

[33] Alex Biryukov, Sourav Mukhopadhyay, and Palash Sarkar. "Improved Time-Memory Trade-Offs with Multiple Data." In: *Selected Areas in Cryptography, 12th International Workshop, SAC 2005, Kingston, ON, Canada, August 11-12, 2005, Revised Selected Papers*. Ed. by Bart Preneel and Stafford E. Tavares. Vol. 3897. Lecture Notes in Computer Science. Springer, 2005, pp. 110–127 (cit. on p. 168).

[34] John Black. *The Ideal-Cipher Model, Revisited: An Uninstantiable Blockcipher-Based Hash Function*. Cryptology ePrint Archive, Paper 2005/210. https://eprint.iacr.org/2005/210. 2005. URL: https://eprint.iacr.org/2005/210 (cit. on p. 8).

[35] John Black and Phillip Rogaway. "A Block-Cipher Mode of Operation for Parallelizable Message Authentication." In: *Advances in Cryptology - EUROCRYPT 2002, International Conference on the Theory and Applications of Cryptographic Techniques, Amsterdam, The Netherlands, April 28 - May 2, 2002, Proceedings*. Ed. by Lars R. Knudsen. Vol. 2332. Lecture Notes in Computer Science. Springer, 2002, pp. 384–397 (cit. on pp. 2, 3, 19).

[36] Hanno Böck et al. "Nonce-Disrespecting Adversaries: Practical Forgery Attacks on GCM in TLS." In: *10th USENIX Workshop on Offensive Technologies, WOOT 16, Austin, TX, USA, August 8-9, 2016*. 2016 (cit. on pp. 29, 30).

[37] Bert den Boer. "A Simple and Key-Economical Unconditional Authentication Scheme." In: *Journal of Computer Security* 2 (1993), pp. 65–72 (cit. on p. 190).

[38] Andrey Bogdanov et al. "PRESENT: An Ultra-Lightweight Block Cipher." In: *CHES 2007, Proceedings*. 2007, pp. 450–466 (cit. on pp. 2, 3, 15, 19).

[39] Andrey Bogdanov et al. "spongent: A Lightweight Hash Function." In: *CHES 2011*. 2011, pp. 312–325. DOI: 10.1007/978-3-642-23951-9_21. URL: https://doi.org/10.1007/978-3-642-23951-9%5C_21 (cit. on p. 65).

[40] Andrey Bogdanov et al. "SPONGENT: The Design Space of Lightweight Cryptographic Hashing." In: *IEEE Trans. Computers* 62.10 (2013), pp. 2041–2053 (cit. on pp. 18, 96, 131).

[41] Julia Borghoff et al. "PRINCE - A Low-Latency Block Cipher for Pervasive Computing Applications - Extended Abstract." In: *ASIACRYPT 2012*. 2012, pp. 208–225 (cit. on p. 15).

[42] Priyanka Bose, Viet Tung Hoang, and Stefano Tessaro. "Revisiting AES-GCM-SIV: Multi-user Security, Faster Key Derivation, and Better Bounds." In: *Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part I*. 2018, pp. 468–499 (cit. on pp. 24, 51, 169).

[43] "CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness." In: () (cit. on pp. 2, 14).

[44] Ran Canetti and Hugo Krawczyk. "Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels." In: *Advances in Cryptology - EUROCRYPT 2001, International Conference on the Theory and Application of Cryptographic Techniques, Innsbruck, Austria, May 6-10, 2001, Proceeding*. 2001, pp. 453–474 (cit. on pp. 3, 14, 32).

[45] Larry Carter and Mark N. Wegman. "Universal Classes of Hash Functions." In: *J. Comput. Syst. Sci.* 18.2 (1979), pp. 143–154. DOI: 10.1016/0022-0000(79)90044-8. URL: https://doi.org/10.1016/0022-0000(79)90044-8 (cit. on p. 15).

[46] Avik Chakraborti et al. "Beetle Family of Lightweight and Secure Authenticated Encryption Ciphers." In: *IACR Trans. Cryptogr. Hardw. Embed. Syst.* 2018.2 (2018), pp. 218–241 (cit. on pp. 18, 19, 65, 96, 131).

[47] Avik Chakraborti et al. "On the Composition of Single-Keyed Tweakable Even-Mansour for Achieving BBB Security." In: *IACR Trans. Symmetric Cryptol.* 2020.2 (2020), pp. 1–39 (cit. on pp. vii, 96–98, 130, 132).

[48] Bishwajit Chakraborty and Mridul Nandi. "ORANGE." In: *NIST LWC* (2019). URL: https://csrc.nist.gov/CSRC/media/Projects/Lightweight-Cryptography/documents/round-1/spec-doc/orange-spec.pdf (cit. on pp. 18, 131).

[49] Donghoon Chang and Mridul Nandi. "A Short Proof of the PRP/PRF Switching Lemma." In: *IACR Cryptol. ePrint Arch.* 2008 (2008), p. 78 (cit. on pp. 15, 19).

[50] Sanjit Chatterjee, Alfred Menezes, and Palash Sarkar. "Another Look at Tightness." In: *Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*. Ed. by Ali Miri and Serge Vaudenay. Vol. 7118. Lecture Notes in Computer Science. Springer, 2011, pp. 293–319 (cit. on pp. 168, 169).

[51] Shan Chen and John P. Steinberger. "Tight Security Bounds for Key-Alternating Ciphers." In: *Advances in Cryptology - EUROCRYPT 2014*, 2014, pp. 327–350 (cit. on p. 131).

[52] Shan Chen et al. "Minimizing the Two-Round Even-Mansour Cipher." In: *Advances in Cryptology - CRYPTO 2014*, 2014, pp. 39–56 (cit. on pp. 132, 134, 211, 214).

[53] Yu Long Chen, Eran Lambooij, and Bart Mennink. "How to Build Pseudorandom Functions from Public Random Permutations." In: *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part I*. 2019, pp. 266–293 (cit. on pp. vii, 18, 22, 65–67, 75, 93, 96–98, 130–132).

[54] Benoît Cogliati, Rodolphe Lampe, and Yannick Seurin. "Tweaking Even-Mansour Ciphers." In: *CRYPTO 2015, Part I*. 2015, pp. 189–208. DOI: 10.1007/978-3-662-47989-6_9. URL: https://doi.org/10.1007/978-3-662-47989-6%5C_9 (cit. on pp. 76, 131).

[55] Benoıt Cogliati and Yannick Seurin. "Analysis of the single-permutation encrypted Davies-Meyer construction." In: *Des. Codes Cryptogr.* 86.12 (2018), pp. 2703–2723 (cit. on pp. 15, 16, 20, 67, 118, 215).

[56] Benoıt Cogliati and Yannick Seurin. "EWCDM: An Efficient, Beyond-Birthday Secure, Nonce-Misuse Resistant MAC." In: *CRYPTO 2016, Proceedings, Part I*. 2016, pp. 121–149 (cit. on pp. 17, 19, 29, 35, 51, 126).

[57] Jean-Sébastien Coron, Jacques Patarin, and Yannick Seurin. "The Random Oracle Model and the Ideal Cipher Model Are Equivalent." In: *Advances in Cryptology – CRYPTO 2008*. Ed. by David Wagner. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–20. ISBN: 978-3-540-85174-5 (cit. on p. 8).

[58] Joan Daemen, Bart Mennink, and Gilles Van Assche. "Full-State Keyed Duplex with Built-In Multi-user Support." In: *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Vol. 10625. Lecture Notes in Computer Science. Springer, 2017, pp. 606–637 (cit. on p. 96).

[59] Joan Daemen et al. "Xoodyak, a lightweight cryptographic scheme." In: *IACR Trans. Symmetric Cryptol.* 2020.S1 (2020), pp. 60–87 (cit. on pp. 18, 131).

[60] Wei Dai, Viet Tung Hoang, and Stefano Tessaro. "Information-Theoretic Indistinguishability via the Chi-Squared Method." In: *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*. 2017, pp. 497–523 (cit. on pp. 15, 19, 102).

[61] Nilanjan Datta et al. "Double-block Hash-then-Sum: A Paradigm for Constructing BBB Secure PRF." In: *IACR Transactions on Symmetric Cryptology* 2018.3 (2018), pp. 36–92 (cit. on pp. vii, 4, 19, 29, 57, 129, 169, 170, 172, 174).

[62] Nilanjan Datta et al. "Encrypt or Decrypt? To Make a Single-Key Beyond Birthday Secure Nonce-Based MAC." In: *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*. 2018, pp. 631–661 (cit. on pp. vii, 16, 17, 19, 29, 40–42, 51, 66, 67, 69, 70, 97).

[63] Nilanjan Datta et al. "Single Key Variant of PMAC_Plus." In: *IACR Trans. Symmetric Cryptol.* 2017.4 (2017), pp. 268–305 (cit. on pp. 29, 129, 222, 223).

[64] Christoph Dobraunig et al. *Ascon v1.2. Submission to CAESAR*. https://competitions.cr.yp.to/round3/asconv12.pdf. 2016 (cit. on pp. 18, 19, 65, 131).

[65] Christoph Dobraunig et al. "Isap v2.0." In: *IACR Trans. Symmetric Cryptol.* 2020.S1 (2020), pp. 390–416 (cit. on p. 18).

[66] Yevgeniy Dodis and Prashant Puniya. "On the Relation Between the Ideal Cipher and the Random Oracle Models." In: *Theory of Cryptography*. Ed. by Shai Halevi and Tal Rabin. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 184–206. ISBN: 978-3-540-32732-5 (cit. on p. 8).

[67] Avijit Dutta. "Minimizing the Two-Round Tweakable Even-Mansour Cipher." In: *IACR Cryptol. ePrint Arch.* 2020 (2020), p. 1076 (cit. on pp. 126, 210).

[68] Avijit Dutta, Ashwin Jha, and Mridul Nandi. "Tight Security Analysis of EHtM MAC." In: *IACR Trans. Symmetric Cryptol.* 2017.3 (2017), pp. 130–150 (cit. on pp. 8, 30, 37, 51, 86).

[69] Avijit Dutta and Mridul Nandi. *BBB Secure Nonce Based MAC Using Public Permutations*. Cryptology ePrint Archive, Report 2020/509. https://eprint.iacr.org/2020/509. 2020 (cit. on p. 132).

[70] Avijit Dutta and Mridul Nandi. "BBB Secure Nonce Based MAC Using Public Permutations." In: *Progress in Cryptology - AFRICACRYPT 2020 - 12th International Conference on Cryptology in Africa, Cairo, Egypt, July 20-22, 2020, Proceedings*. Ed. by Abderrahmane Nitaj and Amr M. Youssef. Vol. 12174. Lecture Notes in Computer Science. Springer, 2020, pp. 172–191 (cit. on pp. 97, 98, 130).

[71] Avijit Dutta, Mridul Nandi, and Abishanka Saha. "Proof of Mirror Theory for $\xi_{\max} = 2$." In: *IACR Cryptol. ePrint Arch.* 2020 (2020), p. 669 (cit. on pp. 20, 102).

[72] Avijit Dutta, Mridul Nandi, and Suprita Talnikar. "Beyond Birthday Bound Secure MAC in Faulty Nonce Model." In: *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I*. 2019, pp. 437–466 (cit. on pp. 68, 69).

[73] Avijit Dutta, Mridul Nandi, and Suprita Talnikar. "Permutation Based EDM: An Inverse Free BBB Secure PRF." In: *IACR Trans. Symmetric Cryptol.* 2021.2 (2021), pp. 31–70 (cit. on pp. 130, 132).

[74] Morris Dworkin. *Recommendation for block cipher modes of operation: Galois/Counter Mode (GCM) and GMAC*. NIST Special Publication 800-38D, csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf. 2011 (cit. on p. 168).

[75] *Etude de Géńerateurs de Permutations Basés sur les Schémas du DES*. Ph. Thesis. Inria, Domaine de Voluceau, France. 1991 (cit. on pp. 80, 86).

[76] Shimon Even and Yishay Mansour. "A Construction of a Cipher from a Single Pseudorandom Permutation." In: *J. Cryptology* 10.3 (1997), pp. 151–162 (cit. on pp. 75, 96).

[77] Shoni Gilboa and Shay Gueron. "The Advantage of Truncated Permutations." In: *CoRR* abs/1610.02518 (2016). arXiv: 1610.02518. URL: http://arxiv.org/abs/1610.02518 (cit. on pp. 15, 19).

[78] Aldo Gunsing and Bart Mennink. "The Summation-Truncation Hybrid: Reusing Discarded Bits for Free." In: *Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part I*. Ed. by Daniele Micciancio and Thomas Ristenpart. Vol. 12170. Lecture Notes in Computer Science. Springer, 2020, pp. 187–217 (cit. on p. 20).

[79] Chun Guo et al. "Beyond-birthday secure domain-preserving PRFs from a single permutation." In: *Des. Codes Cryptogr.* 87.6 (2019), pp. 1297–1322 (cit. on p. 20).

[80] Jian Guo, Thomas Peyrin, and Axel Poschmann. "The PHOTON Family of Lightweight Hash Functions." In: *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*. 2011, pp. 222–239 (cit. on pp. 65, 96, 131).

[81] Jian Guo et al. "The LED Block Cipher." In: *CHES 2011*. 2011, pp. 326–341 (cit. on pp. 2, 3).

[82] Jian Guo et al. "The LED Block Cipher." In: *IACR Cryptology ePrint Archive* 2012 (2012), p. 600 (cit. on p. 2).

[83] Tingting Guo and Peng Wang. *A Note on the Security Framework of Two-key DbHtS MACs*. Cryptology ePrint Archive, Report 2022/375. 2022 (cit. on pp. 171, 172, 192).

[84] Chris Hall et al. "Building PRFs from PRPs." In: *CRYPTO 1998, Proceedings*. 1998, pp. 370–389 (cit. on pp. 15, 19).

[85] Viet Tung Hoang and Stefano Tessaro. "Key-Alternating Ciphers and Key-Length Extension: Exact Bounds and Multi-user Security." In: *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part I*. 2016, pp. 3–32 (cit. on pp. 24, 51, 169).

[86] Viet Tung Hoang and Stefano Tessaro. "The Multi-user Security of Double Encryption." In: *Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part II*. 2017, pp. 381–411 (cit. on pp. 24, 169).

[87] Tetsu Iwata. "Authenticated Encryption Mode for Beyond the Birthday Bound Security." In: *Progress in Cryptology - AFRICACRYPT 2008, First International Conference on Cryptology in Africa, Casablanca, Morocco, June 11-14, 2008. Proceedings*. 2008, pp. 125–142 (cit. on pp. 3, 14).

[88] Tetsu Iwata. "New Blockcipher Modes of Operation with Beyond the Birthday Bound Security." In: *Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers*. 2006, pp. 310–327 (cit. on pp. 3, 14, 19, 34).

[89] Tetsu Iwata and Kaoru Kurosawa. "OMAC: One-Key CBC MAC." In: *FSE*. 2003, pp. 129–153 (cit. on pp. 2, 3).

[90] Tetsu Iwata, Bart Mennink, and Damian Vizár. "CENC is Optimally Secure." In: *IACR Cryptology ePrint Archive* 2016 (2016), p. 1087 (cit. on p. 131).

[91] Jérémy Jean, Ivica Nikolic, and Thomas Peyrin. "Tweaks and Keys for Block Ciphers: The TWEAKEY Framework." In: *ASIACRYPT 2014*. 2014, pp. 274–288 (cit. on p. 76).

[92] Seongkwang Kim, ByeongHak Lee, and Jooyoung Lee. "Tight Security Bounds for Double-Block Hash-then-Sum MACs." In: *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I*. Ed. by Anne Canteaut and Yuval Ishai. Vol. 12105. Lecture Notes in Computer Science. Springer, 2020, pp. 435–465 (cit. on pp. 4, 5, 130, 173).

[93] Tadayoshi Kohno, John Viega, and Doug Whiting. "CWC: A High-Performance Conventional Authenticated Encryption Mode." In: *Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers*. 2004, pp. 408–426 (cit. on pp. 3, 14, 34, 35).

[94] Will Landecker, Thomas Shrimpton, and R. Seth Terashima. "Tweakable Blockciphers with Beyond Birthday-Bound Security." In: *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*. 2012, pp. 14–30 (cit. on pp. 29, 35).

[95] Gaëtan Leurent, Mridul Nandi, and Ferdinand Sibleyras. "Generic Attacks Against Beyond-Birthday-Bound MACs." In: *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*. Ed. by Hovav Shacham and Alexandra Boldyreva. Vol. 10991. Lecture Notes in Computer Science. Springer, 2018, pp. 306–336 (cit. on pp. 5, 130, 173).

[96] Moses Liskov, Ronald L. Rivest, and David A. Wagner. "Tweakable Block Ciphers." In: *CRYPTO 2002*. 2002, pp. 31–46 (cit. on p. 76).

[97] Michael Luby and Charles Rackoff. "How to Construct Pseudorandom Permutations from Pseudorandom Functions." In: *SIAM J. Comput.* 17.2 (1988), pp. 373–386 (cit. on pp. 15, 65).

[98] Stefan Lucks. "The Sum of PRPs Is a Secure PRF." In: *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*. Ed. by Bart Preneel. Vol. 1807. Lecture Notes in Computer Science. Springer, 2000, pp. 470–484 (cit. on pp. 15, 19).

[99] Atul Luykx, Bart Mennink, and Kenneth G. Paterson. "Analyzing Multi-key Security Degradation." In: *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Vol. 10625. Lecture Notes in Computer Science. Springer, 2017, pp. 575–605 (cit. on p. 169).

[100] Atul Luykx et al. "A MAC Mode for Lightweight Block Ciphers." In: *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*. 2016, pp. 43–59 (cit. on pp. 2, 3, 19).

[101] David A. McGrew and John Viega. "The Security and Performance of the Galois/Counter Mode (GCM) of Operation." In: *Progress in Cryptology - INDOCRYPT 2004, 5th International Conference on Cryptology in India, Chennai, India, December 20-22, 2004, Proceedings*. 2004, pp. 343–355 (cit. on pp. 2, 14, 19, 34, 168).

[102] Bart Mennink. "XPX: Generalized Tweakable Even-Mansour with Improved Security Guarantees." In: *CRYPTO 2016, Part I*. 2016, pp. 64–94. DOI: 10.1007/978-3-662-53018-4_3. URL: https://doi.org/10.1007/978-3-662-53018-4%5C_3 (cit. on p. 76).

[103] Bart Mennink and Samuel Neves. *Encrypted Davies-Meyer and Its Dual: Towards Optimal Security Using Mirror Theory*. Cryptology ePrint Archive, Report 2017/473. 2017 (cit. on pp. 15, 16, 19, 29).

[104] Bart Mennink and Samuel Neves. "Encrypted Davies-Meyer and Its Dual: Towards Optimal Security Using Mirror Theory." In: *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*. 2017, pp. 556–583 (cit. on p. 41).

[105] Bart Mennink and Samuel Neves. "Optimal PRFs from Blockcipher Designs." In: *IACR Trans. Symmetric Cryptol.* 2017.3 (2017), pp. 228–252 (cit. on p. 18).

[106] Bart Mennink, Reza Reyhanitabar, and Damian Vizár. "Security of Full-State Keyed Sponge and Duplex: Applications to Authenticated Encryption." In: *ASIACRYPT 2015, Part II*. 2015, pp. 465–489. DOI: 10.1007/978-3-662-48800-3_19. URL: https://doi.org/10.1007/978-3-662-48800-3%5C_19 (cit. on p. 65).

[107] Kazuhiko Minematsu. "How to Thwart Birthday Attacks against MACs via Small Randomness." In: *Fast Software Encryption, FSE 2010*. 2010, pp. 230–249 (cit. on pp. 30, 36).

[108] Kazuhiko Minematsu and Tetsu Iwata. "Building Blockcipher from Tweakable Blockcipher: Extending FSE 2009 Proposal." In: *Cryptography and Coding - 13th IMA International Conference, IMACC 2011, Oxford, UK, December 12-15, 2011. Proceedings*. 2011, pp. 391–412 (cit. on pp. 29, 78, 190).

[109] Michael Mitzenmacher and Eli Upfal. *Probability and Computing - Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005 (cit. on p. 218).

[110] Andrew Morgan, Rafael Pass, and Elaine Shi. "On the Adaptive Security of MACs and PRFs." In: *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I*. Ed. by Shiho Moriai and Huaxiong Wang. Vol. 12491. Lecture Notes in Computer Science. Springer, 2020, pp. 724–753 (cit. on p. 169).

[111] Nicky Mouha and Atul Luykx. "Multi-key Security: The Even-Mansour Construction Revisited." In: *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*. Ed. by Rosario Gennaro and Matthew Robshaw. Vol. 9215. Lecture Notes in Computer Science. Springer, 2015, pp. 209–223 (cit. on p. 169).

[112] Nicky Mouha et al. "Chaskey: An Efficient MAC Algorithm for 32-bit Microcontrollers." In: *Selected Areas in Cryptography – SAC 2014*. Ed. by Antoine Joux and Amr Youssef. Cham: Springer International Publishing, 2014, pp. 306–323. ISBN: 978-3-319-13051-4 (cit. on p. 130).

[113] Valérie Nachef, Jacques Patarin, and Emmanuel Volte. *Feistel Ciphers - Security Proofs and Cryptanalysis*. Springer, 2017. ISBN: 978-3-319-49528-6. DOI: 10.1007/978-3-319-49530-9. URL: https://doi.org/10.1007/978-3-319-49530-9 (cit. on pp. 15, 67).

[114] Yusuke Naito. "Blockcipher-Based MACs: Beyond the Birthday Bound Without Message Length." In: *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part III*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Vol. 10626. Lecture Notes in Computer Science. Springer, 2017, pp. 446–470 (cit. on pp. 4, 129).

[115] Yusuke Naito et al. "SAEB: A Lightweight Blockcipher-Based AEAD Mode of Operation." In: *IACR Cryptol. ePrint Arch.* 2019 (2019), p. 700 (cit. on p. 18).

[116] Chanathip Namprempre, Phillip Rogaway, and Thomas Shrimpton. "Reconsidering Generic Composition." In: *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*. 2014, pp. 257–274 (cit. on pp. 3, 14).

[117] Mridul Nandi. *Birthday Attack on Dual EWCDM*. Cryptology ePrint Archive, Report 2017/579. https://eprint.iacr.org/2017/579. 2017 (cit. on pp. 2, 3, 29).

[118] Mridul Nandi. "Mind the Composition: Birthday Bound Attacks on EWCDMD and SoKAC21." In: *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I*. Ed. by Anne Canteaut and Yuval Ishai. Vol. 12105. Lecture Notes in Computer Science. Springer, 2020, pp. 203–220 (cit. on pp. 22, 66, 67, 96, 98, 130, 131).

[119] NIST. *Lightweight Cryptography*. Online: https://csrc.nist.gov/Projects/Lightweight-Cryptography. Accessed: August 01, 2019. 2018 (cit. on pp. 18, 131).

[120] Jacques Patarin. "Introduction to Mirror Theory: Analysis of Systems of Linear Equalities and Linear Non Equalities for Cryptography." In: *IACR Cryptology ePrint Archive* 2010 (2010), p. 287 (cit. on pp. 15, 40, 41, 67).

[121] Jacques Patarin. "Mirror Theory and Cryptography." In: *IACR Cryptology ePrint Archive* 2016 (2016), p. 702 (cit. on pp. 15, 40, 67).

[122] Jacques Patarin. "On Linear Systems of Equations with Distinct Variables and Small Block Size." In: *Information Security and Cryptology - ICISC 2005, 8th International Conference, Seoul, Korea, December 1-2, 2005, Revised Selected Papers*. 2005, pp. 299–321 (cit. on pp. 15, 67).

[123] Jacques Patarin. "Security in $O(2^n)$ for the Xor of Two Random Permutations - Proof with the standard H technique." In: *IACR Cryptology ePrint Archive* 2013 (2013), p. 368 (cit. on p. 40).

[124] Jacques Patarin. "The "Coefficients H" Technique." In: *Selected Areas in Cryptography, 15th International Workshop, SAC 2008, Sackville, New Brunswick, Canada, August 14-15, Revised Selected Papers*. 2008, pp. 328–345 (cit. on pp. 24, 25).

[125] Thomas Peyrin and Yannick Seurin. "Counter-in-Tweak: Authenticated Encryption Modes for Tweakable Block Ciphers." In: *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part I*. 2016, pp. 33–63 (cit. on p. 30).

[126] Phillip Rogaway. "Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC." In: *Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5-9, 2004, Proceedings*. 2004, pp. 16–31 (cit. on pp. 28, 35, 76).

[127] Phillip Rogaway, Mihir Bellare, and John Black. "SHA-3 standard." In: *ACM Transactions on Information and System Security (TISSEC)* 6.3 (2003), pp. 365–403 (cit. on pp. 18, 131).

[128] Yaobin Shen et al. "Revisiting the Security of DbHtS MACs: Beyond-Birthday-Bound in the Multi-user Setting." In: *Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III*. Ed. by Tal Malkin and Chris Peikert. Vol. 12827. Lecture Notes in Computer Science. Springer, 2021, pp. 309–336 (cit. on pp. 169–172, 174).

[129] Victor Shoup. "A Composition Theorem for Universal One-Way Hash Functions." In: *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*. Ed. by Bart Preneel. Vol. 1807. Lecture Notes in Computer Science. Springer, 2000, pp. 445–452 (cit. on p. 3).

[130] Victor Shoup. "On Fast and Provably Secure Message Authentication Based on Universal Hashing." In: *Advances in Cryptology - CRYPTO 1996, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*. 1996, pp. 313–328 (cit. on pp. 28, 35).

[131] Richard Taylor. "An Integrity Check Value Algorithm for Stream Ciphers." In: *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*. 1993, pp. 40–48 (cit. on p. 190).

[132] Serge Vaudenay. "Decorrelation: A Theory for Block Cipher Security." In: *J. Cryptology* 16.4 (2003), pp. 249–286 (cit. on pp. 25, 80, 86).

[133] Mark N. Wegman and Larry Carter. "New Hash Functions and Their Use in Authentication and Set Equality." In: *J. Comput. Syst. Sci.* 22.3 (1981), pp. 265–279 (cit. on pp. 3, 14, 15, 19, 34, 49).

[134] Kan Yasuda. "A New Variant of PMAC: Beyond the Birthday Bound." In: *CRYPTO 2011*. 2011, pp. 596–609 (cit. on pp. 4, 129).

[135] Kan Yasuda. "The Sum of CBC MACs Is a Secure PRF." In: *CT-RSA 2010*. 2010, pp. 366–381 (cit. on pp. 4, 129).

[136] Liting Zhang et al. "3kf9: Enhancing 3GPP-MAC beyond the Birthday Bound." In: *ASIACRYPT 2012*. 2012, pp. 296–312 (cit. on pp. 4, 129).

[137] Ping Zhang, Honggang Hu, and Qian Yuan. "Close to Optimally Secure Variants of GCM." In: *Security and Communication Networks* 2018 (2018), 9715947:1–9715947:12 (cit. on pp. 3, 14, 34).

Appendices

Appendix A. A Simple Result on Probability

In this section, we recall two simple probability results from [67] that shall prove useful in the proof of security of the pEDM construction (4).

Proposition 4. *Let* $\widetilde{Q} = (\mathcal{Q}_1, \ldots, \mathcal{Q}_{s+1})$ *be an* $(s+1)$*-tuple of ordered pairs such that for* $j \in [s+1]$, $\mathcal{Q}_j := ((x_1^j, y_1^j), \ldots, (x_{q_j}^j, y_{q_j}^j))$. *Moreover, for each* $j, j' \in [s+1]$, *Let* $\mathsf{Dom}(\mathcal{Q}_j) \cap \mathsf{Dom}(\mathcal{Q}_{j'}) = \phi$ *and* $\mathsf{Ran}(\mathcal{Q}_j) \cap \mathsf{Ran}(\mathcal{Q}_{j'}) = \phi$. *Therefore,* $\mathfrak{X} = (\mathsf{Dom}(\mathcal{Q}_1), \ldots, \mathsf{Dom}(\mathcal{Q}_{s+1}))$ *and* $\mathfrak{Y} = (\mathsf{Ran}(\mathcal{Q}_1), \ldots, \mathsf{Ran}(\mathcal{Q}_{s+1}))$ *are two disjoint collections of finite sets such that for each* $j \in [s+1]$, $|\mathsf{Dom}(\mathcal{Q}_j)| = |\mathsf{Ran}(\mathcal{Q}_j)| = q_j$. *Then,*

$$\Pr\left[\pi \xleftarrow{\$} \mathit{Perm}(n) : \mathfrak{X} \setminus \mathsf{Dom}(\mathcal{Q}_{s+1}) \xrightarrow{\pi} \mathfrak{Y} \setminus \mathsf{Ran}(\mathcal{Q}_{s+1}) \mid \pi \longrightarrow \mathcal{Q}_{s+1}\right] = \frac{1}{(N - q_{s+1})_{q_1 + \ldots + q_s}}.$$

Setting $s = 1$ in the above proposition gives the following simple corollary:

Corollary 4. *For two sets* $\mathcal{Q}_1 = ((x_1^1, y_1^1), \ldots, (x_{q_1}^1, y_{q_1}^1))$ *of cardinality* q_1 *and* $\mathcal{Q}_2 = ((x_1^2, y_1^2), \ldots, (x_{q_2}^2, y_{q_2}^2))$ *of cardinality* q_2 *such that* $\mathsf{Dom}(\mathcal{Q}_1) \cap \mathsf{Dom}(\mathcal{Q}_2) = \phi$ *and* $\mathsf{Ran}(\mathcal{Q}_1) \cap \mathsf{Ran}(\mathcal{Q}_2) = \phi$,

$$\Pr[\pi \xleftarrow{\$} \mathit{Perm}(n) : \pi \longrightarrow \mathcal{Q}_1 \mid \pi \longrightarrow \mathcal{Q}_2] = \frac{1}{(N - q_2)_{q_1}}.$$

Appendix B. Variants of the Sum Capture Lemma

In this section, we state a few variants of the sum capture lemma [6] used in [52]. Informally stated, the result bounds the value

$$\mu(\mathcal{A}) \coloneqq \max_{\mathcal{B},\mathcal{C}\subseteq \mathrm{GF}(2^n)} |\{(a,b,c) \in \mathcal{A}\times\mathcal{B}\times\mathcal{C} : a = b \oplus c\}|$$

by at most $q|\mathcal{B}||\mathcal{C}|/2^n$ (except in cases with negligible probability of occurrence), when choosing a random subset $\mathcal{A}$ of $\mathrm{GF}(2^n)$ (or more generally, of any abelian group). Chen et al. [52] proved the result in a different setting, in which $\mathcal{A}$ arises from the interaction of an adversary with a random permutation π $\big(\mathcal{A} = x \oplus y : (x,y) \in \mathcal{Q}$, where $\mathcal{Q}$ is the transcript of interaction between the adversary and the permutation$\big)$. We present here, a similar lemma.

Let us first recall some results in Fourier analysis over $\mathbb{Z}_2^n$ of size $q = 2^n$.

Notation. Given a subset $\mathcal{S} \subset \{0,1\}^n$, the *characteristic function* of $\mathcal{S}$ is the function $\mathbb{I}_{\mathcal{S}} : \{0,1\}^n \to \{0,1\}$ such that $\mathbb{I}_{\mathcal{S}}(s) = 1$ if and only if $s \in \mathcal{S}$. Given two real-valued functions $f, g : \{0,1\}^n \to \mathbb{R}$, *the inner product of f and g* is given by

$$\langle f, g\rangle = \mathbf{E}[fg] = \frac{1}{2^n}\sum_{x\in\{0,1\}^n} f(x)g(x),$$

and the convolution of f and g is given by

$$(f \star g)(x) = \sum_{y\in\{0,1\}^n} f(y)g(x\oplus y),\ \forall\ x \in \{0,1\}^n.$$

For $\alpha \in \{0,1\}^n$, the *character associated with α* is the function

$$\chi_\alpha : \{0,1\}^n \to \{+1,-1\},\ x \longrightarrow (-1)^{\alpha\cdot x}.$$

χ_0 is called the *principal character* and all other χ_α $(\neq 1)$ for $\alpha \neq 0$ are called *non-principal characters*. For $\alpha \in \{0,1\}^n$, we define the α^{th} Fourier coefficient of a real-valued function $f : \{0,1\}^n \to \mathbb{R}$ as

$$\widehat{f}(\alpha) \coloneqq \langle f, \chi_\alpha\rangle = \frac{1}{2^n}\sum_{x\in\{0,1\}^n} f(x)(-1)^{\alpha\cdot x}.$$

The coefficient corresponding to $\alpha = 0$ is called the *principal Fourier coefficient* and all other coefficients are *non-principal Fourier coefficients*. Note that the principal Fourier coefficient for a characteristic function $\mathbb{I}_{\mathcal{S}}$ of a set $\mathcal{S}$ is

$$\widehat{\mathbb{I}}_{\mathcal{S}}(0) = \frac{|\mathcal{S}|}{2^n}.$$

Having defined the necessary notations, one may now recall the following three important results for functions $f, g : \{0,1\}^n \to \mathbb{R}$, a constant $\alpha \in \{0,1\}^n$ and a set $\mathcal{S} \subseteq \{0,1\}^n$:

$$\sum_{x\in\{0,1\}^n} f(x)g(x) = 2^n \sum_{\alpha\in\{0,1\}^n} \widehat{f}(\alpha)\widehat{g}(\alpha), \tag{B.1}$$

$$\widehat{(f \star g)}(\alpha) = 2^n \widehat{f}(\alpha)\widehat{g}(\alpha), \tag{B.2}$$

$$\sum_{\alpha\in\{0,1\}^n} |\widehat{\mathbb{I}}_{\mathcal{S}}(\alpha)|^2 = \frac{|\mathcal{S}|}{2^n}. \tag{B.3}$$

Finally, note the following two definitions of parameters associated with $\mathcal{Q}$:

$$\Phi_{\alpha,\beta}(\mathcal{Q}) := 2^{2n}|\widehat{\mathbb{I}}_{\mathcal{Q}}(\alpha,\beta)| = |\sum_{(x,y)\in\mathcal{Q}} (-1)^{\alpha\cdot x\oplus\beta\cdot y}|,$$
$$\Phi(\mathcal{Q}) := \max_{\alpha\neq 0,\beta\neq 0} \Phi_{\alpha,\beta}(\mathcal{Q}).$$

We are now ready to state our sum-capture lemma:

Lemma 21. *Let* $\mathsf{RF} \in \mathsf{Func}(\{0,1\}^n)$ *and* D *be a probabilistic distinguisher that makes* q *adaptive queries to* RF. *Let* $\mathcal{Q} = ((x_1,y_1),\ldots,(x_q,y_q))$ *denote the transcript of interaction of* D *with* RF. *For any two subsets* $\mathcal{U}$ *and* $\mathcal{V}$ *of* $\{0,1\}^n$, *let*

$$\mu(\mathcal{Q},\mathcal{U},\mathcal{V}) = |\{((x,y),u,v) \in \mathcal{Q}\times\mathcal{U}\times\mathcal{V} : x\oplus u = y\oplus v\}|.$$

Then assuming $9n \le q \le 2^n/2$,

$$\Pr_{\mathsf{RF},\omega}\left[\exists \mathcal{U},\mathcal{V} \subseteq \{0,1\}^n : \mu(\mathcal{Q},\mathcal{U},\mathcal{V}) \ge \frac{q|\mathcal{U}||\mathcal{V}|}{2^n} + 3\sqrt{nq|\mathcal{U}||\mathcal{V}|}\right] \le \frac{2}{2^n}, \tag{B.4}$$

where the probability is over random choices of RF *and random coins* ω *of* D.

Proof. To bound $\mu(\mathcal{Q},\mathcal{U},\mathcal{V})$, consider the following two sets: $\mathcal{N} = \mathcal{U}\times\mathcal{V} = \{(u,v) : u\in\mathcal{U}, v\in\mathcal{V}\}$ and $\mathcal{K} = \{(k,k) : k\in\{0,1\}^n\}$. Note that $((x,y),u,v) \in \mathcal{Q}\times\mathcal{U}\times\mathcal{V}$ if and only if $\exists k \in \{0,1\}^n$ such that $(x,y)\oplus(u,v) = (k,k)$. Therefore,

$$\begin{aligned}
\mu(\mathcal{Q},\mathcal{U},\mathcal{V}) &= \sum_{(x,y)\in(\{0,1\}^n)^2}\ \sum_{(u,v)\in(\{0,1\}^n)^2} \mathbb{I}_{\mathcal{Q}}(x,y)\mathbb{I}_{\mathcal{N}}(u,v)\mathbb{I}_{\mathcal{K}}(x\oplus u, y\oplus v)\\
&= \sum_{(x,y)\in(\{0,1\}^n)^2} \mathbb{I}_{\mathcal{Q}}(x,y) \sum_{(u,v)\in(\{0,1\}^n)^2} \mathbb{I}_{\mathcal{N}}(u,v)\mathbb{I}_{\mathcal{K}}(x\oplus u, y\oplus v)\\
&= \sum_{(x,y)\in(\{0,1\}^n)^2} \mathbb{I}_{\mathcal{Q}}(x,y)(\mathbb{I}_{\mathcal{N}}\star\mathbb{I}_{\mathcal{K}})(x,y)\\
&= 2^{2n}\sum_{(\alpha,\beta)\in(\{0,1\}^n)^2} \widehat{\mathbb{I}}_{\mathcal{Q}}(\alpha,\beta)\widehat{(\mathbb{I}_{\mathcal{N}}\star\mathbb{I}_{\mathcal{K}})}(\alpha,\beta) \text{ (by Eqn. (B.1))}\\
&= 2^{4n}\sum_{(\alpha,\beta)\in(\{0,1\}^n)^2} \widehat{\mathbb{I}}_{\mathcal{Q}}(\alpha,\beta)\widehat{\mathbb{I}}_{\mathcal{N}}(\alpha,\beta)\widehat{\mathbb{I}}_{\mathcal{K}}(\alpha,\beta) \text{ (by Eqn. (B.2))}\\
&= 2^{4n}\frac{|\mathcal{Q}|}{2^{2n}}\frac{|\mathcal{N}|}{2^{2n}}\frac{|\mathcal{K}|}{2^{2n}} + 2^{4n}\sum_{(\alpha,\beta)\neq(0,0)} \widehat{\mathbb{I}}_{\mathcal{Q}}(\alpha,\beta)\widehat{\mathbb{I}}_{\mathcal{N}}(\alpha,\beta)\widehat{\mathbb{I}}_{\mathcal{K}}(\alpha,\beta)\\
&\quad \text{(separating principal Fourier coefficients from non-principal ones)}\\
&= \frac{q|\mathcal{U}||\mathcal{V}|}{2^n} + 2^{4n}\sum_{(\alpha,\beta)\neq(0,0)} \widehat{\mathbb{I}}_{\mathcal{Q}}(\alpha,\beta)\widehat{\mathbb{I}}_{\mathcal{N}}(\alpha,\beta)\widehat{\mathbb{I}}_{\mathcal{K}}(\alpha,\beta),\\
&\quad \text{(follows from the cardinality of } \mathcal{Q},\mathcal{N} \text{ and } \mathcal{K}). \qquad \text{(B.5)}\\
\text{Now, } \widehat{\mathbb{I}}_{\mathcal{N}}(\alpha,\beta) &= \frac{1}{2^{2n}}\sum_{(u,v)\in(\{0,1\}^n)^2} \mathbb{I}_{\mathcal{N}}(u,v)(-1)^{\alpha\cdot u\oplus\beta\cdot v}\\
&= \frac{1}{2^{2n}}\sum_{(u,v)\in(\{0,1\}^n)^2} \mathbb{I}_{\mathcal{U}}(u)\mathbb{I}_{\mathcal{U}}(u)(-1)^{\alpha\cdot u\oplus\beta\cdot v}\\
&= \frac{1}{2^{2n}}\left(\sum_{u\in(\{0,1\}^n)} \mathbb{I}_{\mathcal{U}}(u)(-1)^{\alpha\cdot u}\right)\left(\sum_{v\in(\{0,1\}^n)} \mathbb{I}_{\mathcal{V}}(v)(-1)^{\beta\cdot v}\right)\\
&= \widehat{\mathbb{I}}_{\mathcal{U}}(\alpha)\widehat{\mathbb{I}}_{\mathcal{V}}(\beta), \qquad \text{(B.6)}\\
\text{and } \widehat{\mathbb{I}}_{\mathcal{K}}(\alpha,\beta) &= \frac{1}{2^{2n}}\sum_{(x,y)\in(\{0,1\}^n)^2} \mathbb{I}_{\mathcal{K}}(x,y)(-1)^{\alpha\cdot x\oplus\beta\cdot y}\\
&= \frac{1}{2^{2n}}\sum_{y\in(\{0,1\}^n)} (-1)^{\alpha\cdot y\oplus\beta\cdot y}. \qquad \text{(B.7)}
\end{aligned}$$

Eqn. (B.7) evaluates to 0 if $\beta = \alpha$. Therefore, from Eqn.s (B.5), (B.6) and (B.7),

$$\begin{aligned}\mu(\mathcal{Q},\mathcal{U},\mathcal{V}) &= \frac{q|\mathcal{U}||\mathcal{V}|}{2^n} + 2^{3n}\sum_{\alpha\neq 0}\widehat{\mathbb{I}}_{\mathcal{Q}}(\alpha,\alpha)\widehat{\mathbb{I}}_{\mathcal{U}}(\alpha)\widehat{\mathbb{I}}_{\mathcal{V}}(\alpha)\\ &\leq \frac{q|\mathcal{U}||\mathcal{V}|}{2^n} + 2^{3n}\sum_{\alpha\neq 0}|\widehat{\mathbb{I}}_{\mathcal{Q}}(\alpha,\alpha)|\cdot|\widehat{\mathbb{I}}_{\mathcal{U}}(\alpha)|\cdot|\widehat{\mathbb{I}}_{\mathcal{V}}(\alpha)|\\ &\leq \frac{q|\mathcal{U}||\mathcal{V}|}{2^n} + 2^{n}\Phi(\mathcal{Q})\sum_{\alpha\neq 0}|\widehat{\mathbb{I}}_{\mathcal{U}}(\alpha)|\cdot|\widehat{\mathbb{I}}_{\mathcal{V}}(\alpha)|, \qquad \text{(B.8)}\end{aligned}$$

where the last inequality follows due to the definition $\Phi(\mathcal{Q}) := |\widehat{\mathbb{I}}_{\mathcal{Q}}(\alpha,\alpha)| \leq \Phi(\mathcal{Q})/2^{2n}$. Next, by the Cauchy-Schwartz inequality,

$$\sum_{\alpha\neq 0}|\widehat{\mathbb{I}}_{\mathcal{U}}(\alpha)|\cdot|\widehat{\mathbb{I}}_{\mathcal{V}}(\alpha)| \leq \sqrt{\sum_{\alpha\neq 0}|\widehat{\mathbb{I}}_{\mathcal{U}}(\alpha)|^2}\cdot\sqrt{\sum_{\alpha\neq 0}|\widehat{\mathbb{I}}_{\mathcal{V}}(\beta)|^2} \leq \frac{1}{2^n}\sqrt{|\mathcal{U}||\mathcal{V}|}. \qquad \text{(B.9)}$$

Plugging Eqn. (B.9) into Eqn. (B.7) then gives

$$\mu(\mathcal{Q},\mathcal{U},\mathcal{V}) \leq \frac{q|\mathcal{U}||\mathcal{V}|}{2^n} + \Phi(\mathcal{Q})\sqrt{|\mathcal{U}||\mathcal{V}|}. \qquad \text{(B.10)}$$

From Lemma 5 of [52] we know that if $A_1,\ldots,A_q$ is a sequence of random variables taking values in $\{+1,-1\}$ such that for all $i \in [q]$ and all $(a_1,\ldots,a_{i-1}) \in (\{+1,-1\})^{i-1}$,

$$\Pr[A_i = 1 \mid (A_1,\ldots,A_{i-1}) = (a_1,\ldots,a_{i-1})] \leq \frac{1}{2} + \epsilon, \qquad \text{(B.11)}$$

for some $\epsilon \in [0,1/2]$, then for any $\delta \in [0,1]$,

$$\Pr\left[\sum_{i=1}^{q} A_i \geq q(2\epsilon+\delta)\right] \leq e^{\frac{q\delta^2}{12}}.$$

We claim the following using this lemma:

Claim. *Assume $9n \leq q \leq 2^n/2$, and let* D *be a probabilistic distinguisher that makes q adaptive queries to* RF. *Let $\mathcal{Q} = ((x_1,y_1),\ldots,(x_q,y_q))$ denote the transcript of interaction with* RF *to* D. *Then*

$$\Pr_{\mathsf{RF},\omega}[\Phi(\mathcal{Q}) \geq 3\sqrt{nq}] \leq \frac{2}{2^n},$$

where the probability is over the randomness of RF *and the random coin ω of the distinguisher* D.

The proof of this claim is similar to that of Lemma 6 of [52]. Define random variables $A_i = (-1)^{\alpha\cdot x_i \oplus \beta\cdot y_i}$ $((x_i,y_i) \in \mathcal{Q})$ i. Then $|(A_1 + A_2 + \ldots + A_q)| =$

$\Phi_{\alpha,\beta}(\mathcal{Q})$. For the i^{th} query with input x_i, the output y_i is a uniform random variable over a set of size 2^n. Moreover, once x_i is fixed, there are exactly $2^n/2$ values y_i such that $A_i = 1$, since $\beta \neq 0$. Therefore,

$$\Pr[A_i = 1 \mid (A_1, \ldots, A_{i-1}) = (a_1, \ldots, a_{i-1})] = \frac{2^n/2}{2^n} = \frac{1}{2}.$$

Hence, $\epsilon = 0$ in Eqn. (B.11), and so,

$$\Pr[\sum_{i=1}^{q} A_i \geq q\delta] \leq e^{-\frac{q\delta^2}{12}}.$$

Setting $A'_i = -A_i$, a similar reasoning gives

$$\Pr[\sum_{i=1}^{q} A_i \leq -q\delta] \leq e^{-\frac{q\delta^2}{12}}.$$

Combining these two equations, we obtain

$$\Pr[\Phi(\mathcal{Q}) \geq q\delta] \leq 2e^{-\frac{q\delta^2}{12}},$$

and the result follows when $\delta = \sqrt{12 \log 2^n/q}$. This makes $q \geq 9n$, which implies $\delta \leq 1$ and $\sqrt{12 \log 2} \leq 3$. □

Let $T^\star$ be a multiset of size q.Recall lemma 1 of [55]:
Let T^* be a multiset of $q \geq 1$ uniformly random and independently chosen elements of $\{0,1\}^n$. Then-

$$\Pr\left[\mu(T^*) \geq \frac{q^3}{2^n} + q\sqrt{3nq}\right] \leq \frac{2}{2^n}.$$

The following is also another corollary of this lemma:

Corollary 5. *Let T^* be a multiset of q (≥ 1) uniformly random and independently chosen elements of $\{0,1\}^n$. Then assuming $9n \leq q \leq 2^{n-1}$,*

$$\Pr_{T^*}\left[\exists \mathcal{U}, \mathcal{V} \subseteq \{0,1\}^n : \mu(T^*, \mathcal{U}, \mathcal{V}) \geq \frac{q|\mathcal{U}||\mathcal{V}|}{2^n} + 3\sqrt{nq|\mathcal{U}||\mathcal{V}|}\right] \leq \frac{2}{2^n}, \quad \text{(B.12)}$$

where the probability is taken over the uniform distribution of the multiset T^*.

This lemma can be slightly altered by simply taking the sizes of the multiset T^* and the sets A, B to be q, p_1, p_2, respectively:

Lemma 22. *Let $T^\star = \{T_1, \ldots, T_q\}$ be the multiset of all the tags received through q (≥ 1) distinct queries to the construction.*

$$\Pr\left[\mu\left(T^\star\right) \geq \frac{p_1 p_2 q}{2^n} + \sqrt{3np_1p_2q}\right] \leq \frac{2}{2^n}. \quad \text{(B.13)}$$

If the set A is replaced by a multiset A^*, then this result is further modified into the following lemma:

Lemma 23. *Let T^*, A^* be multisets of $\{0,1\}^n$ and $B \subseteq \{0,1\}^n$. Define-*

$$\mu(T^*, A^*, B) = |\{(t,a,b) \in T^* \times A^* \times B : t = a \oplus b\}| \text{ and}$$

$$\mu(T^*) = \max_{\substack{A^*,B \\ |T^*|=q_1, |A^*|=q_2, |B|=p}} \mu(T^*, A^*, B).$$

If T^*, A^* are multisets of respectively q_1, q_2 uniformly random and independently chosen elements of $\{0,1\}^n$ and B is a subset of $\{0,1\}^n$ of size p, then

$$\Pr\left[\mu(T^*) \geq \frac{q_1 q_2 p}{2^n} + \sqrt{\frac{3np(q_1+q_2)}{2^n}}\right] \leq \frac{2}{2^n}. \tag{B.14}$$

Proof.

$$\begin{aligned}
\mu(T^*, A^*, B) &= \sum_{t,a \in \{0,1\}^n} \delta_{T^*}(t)\delta_{A^*}(a)\mathbb{1}_B(b) \\
&= \sum_{t \in \{0,1\}^n} \delta_{T^*}(t)\left(\delta_{A^*} \star \mathbb{1}_B\right)(t) \\
&= 2^n \sum_{\alpha \in \{0,1\}^n} \widehat{\delta_{T^*}}(\alpha)\left(\widehat{\delta_{A^*} \star \mathbb{1}_B}\right)(\alpha) \\
&= 2^{2n} \sum_{\alpha \in \{0,1\}^n} \widehat{\delta_{T^*}}(\alpha)\widehat{\delta_{A^*}}(\alpha)\widehat{\mathbb{1}_B}(\alpha) \\
&= 2^{2n}\widehat{\delta_{T^*}}(0)\widehat{\delta_{A^*}}(0)\widehat{\mathbb{1}_B}(0) + 2^{2n}\sum_{\alpha \neq 0} \widehat{\delta_{T^*}}(\alpha)\widehat{\delta_{A^*}}(\alpha)\widehat{\mathbb{1}_B}(\alpha), \\
&\text{where } \widehat{\delta_{T^*}}(0) = \frac{|T^*|}{2^n},\ \widehat{\delta_{A^*}}(0) = \frac{|A^*|}{2^n},\ \widehat{\mathbb{1}_B}(0) = \frac{|B|}{2^n} \text{ imply} \\
\mu(T^*, A^*, B) &= \frac{q_1 q_2 p}{2^n} + 2^{2n}\sum_{\alpha \neq 0} \widehat{\delta_{T^*}}(\alpha)\widehat{\delta_{A^*}}(\alpha)\widehat{\mathbb{1}_B}(\alpha) \\
&\leq \frac{q_1 q_2 p}{2^n} + 2^{2n}\sum_{\alpha \neq 0} \left|\widehat{\delta_{T^*}}(\alpha)\right| \left|\widehat{\delta_{A^*}}(\alpha)\right| \left|\widehat{\mathbb{1}_B}(\alpha)\right| \\
&\leq \frac{q_1 q_2 p}{2^n} + \Phi(T^*)\Phi(A^*)\sum_{\alpha \neq 0} \left|\widehat{\mathbb{1}_B}(\alpha)\right|, \\
&\text{where } \Phi(T^*) = \max_{\alpha \neq 0}\left\{2^n \left|\widehat{\delta_{T^*}}(\alpha)\right|\right\} \\
&\text{and } \Phi(A^*) = \max_{\alpha \neq 0}\left\{2^n \left|\widehat{\delta_{A^*}}(\alpha)\right|\right\}.
\end{aligned}$$

Now,

$$\sum_{\alpha} \in \{0,1\}^n \left|\widehat{\mathbb{1}_B}(\alpha)\right|^2 \geq \left(\sum_{\alpha} \neq 0 \left|\widehat{\mathbb{1}_B}(\alpha)\right|\right)^2 - 2 \cdot \sum_{0 \leq \alpha < \beta < 2^n} \left|\widehat{\mathbb{1}_B}(\alpha)\right| \cdot \left|\widehat{\mathbb{1}_B}(\beta)\right|$$

$$\implies \sum_{\alpha \neq 0} \left|\widehat{\mathbb{1}_B}(\alpha)\right| \leq \sqrt{\frac{|B|}{2^n} + 2 \sum_{0 \leq \alpha < \beta < 2^n} \left|\widehat{\mathbb{1}_B}(\alpha)\right| \cdot \left|\widehat{\mathbb{1}_B}(\beta)\right|} \leq \sqrt{\frac{|B|}{2^n}}.$$

Therefore, $\mu(T^*, A^*, B) \leq \frac{q_1 q_2 p}{2^n} + \Phi(T^*)\Phi(A^*) \cdot \sqrt{\frac{p}{2^n}}$. Since this holds for any $A^*, B \subseteq \{0,1\}^n$, it follows that

$$\frac{q_1 q_2 p}{2^n} + \sqrt{\frac{p}{2^n}} \cdot C \leq \mu(T^*) \leq \frac{q_1 q_2 p}{2^n} + \Phi(T^*)\Phi(A^*) \cdot \sqrt{\frac{p}{2^n}}$$

for some appropriate value of C, which implies $\Pr\left[\mu(T^*) \geq \frac{q_1 q_2 p}{2^n} + \sqrt{\frac{p}{2^n}} \cdot C\right] \leq \Pr\left[\Phi(T^*)\Phi(A^*) \geq C\right]$. Denote $T^* = \{t_1, \ldots, t_{q_1}\}$ and $A^* = \{a_1, \ldots, a_{q_2}\}$ using arbitrary orders. Then-

$$\begin{aligned}
\Phi(T^*) &= \max_{\alpha \neq 0} \left\{ 2^n \cdot \left|\widehat{\delta_{T^*}}(\alpha)\right| \right\} \\
&= \max_{\alpha \neq 0} \left\{ \left| \sum_{x \in \{0,1\}^n} \delta_{T^*}(x) \cdot (-1)^{\alpha \cdot x} \right| \right\} \\
&= \max_{\alpha \neq 0} \left\{ \left| \sum_{x \in \{0,1\}^n} \sum_{i=1}^{q_1} \mathbb{1}_{\{t_i\}}(x) \cdot (-1)^{\alpha \cdot x} \right| \right\} \\
&= \max_{\alpha \neq 0} \left\{ \left| \sum_{i=1}^{q_1} (-1)^{\alpha \cdot t_i} \right| \right\}.
\end{aligned}$$

Similarly, $\Phi(A^*) = \max_{\alpha \neq 0} \left\{ \left| \sum_{j=1}^{q_2} (-1)^{\alpha \cdot a_j} \right| \right\}$.

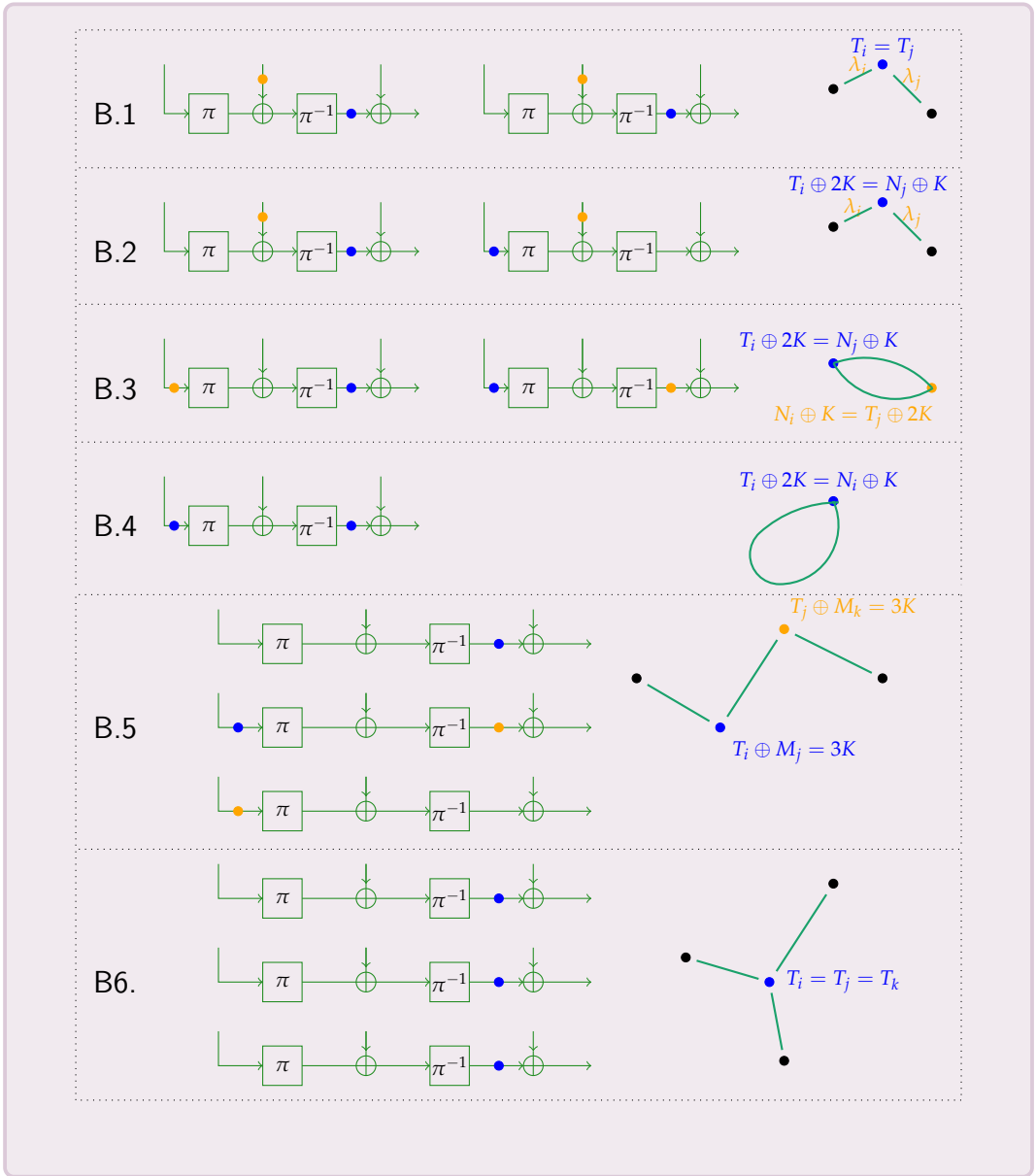
$$\begin{aligned}
\therefore \Phi(T^*)\Phi(A^*) &= \max_{\alpha \neq 0} \left\{ \left| \sum_{i=1}^{q_1} (-1)^{\alpha \cdot t_i} \right| \right\} \cdot \max_{\alpha \neq 0} \left\{ \left| \sum_{j=1}^{q_2} (-1)^{\alpha \cdot a_j} \right| \right\} \\
&= \max_{\alpha \neq 0} \left\{ \left| \sum_{i=1}^{q_1} (-1)^{\alpha \cdot t_i} \cdot \sum_{j=1}^{q_2} (-1)^{\alpha \cdot a_j} \right| \right\} \\
&= \max_{\alpha \neq 0} \left\{ \left| \sum_{(i,j) \in [q_1] \times [q_2]} (-1)^{\alpha \cdot (t_i + a_j)} \right| \right\}
\end{aligned}$$

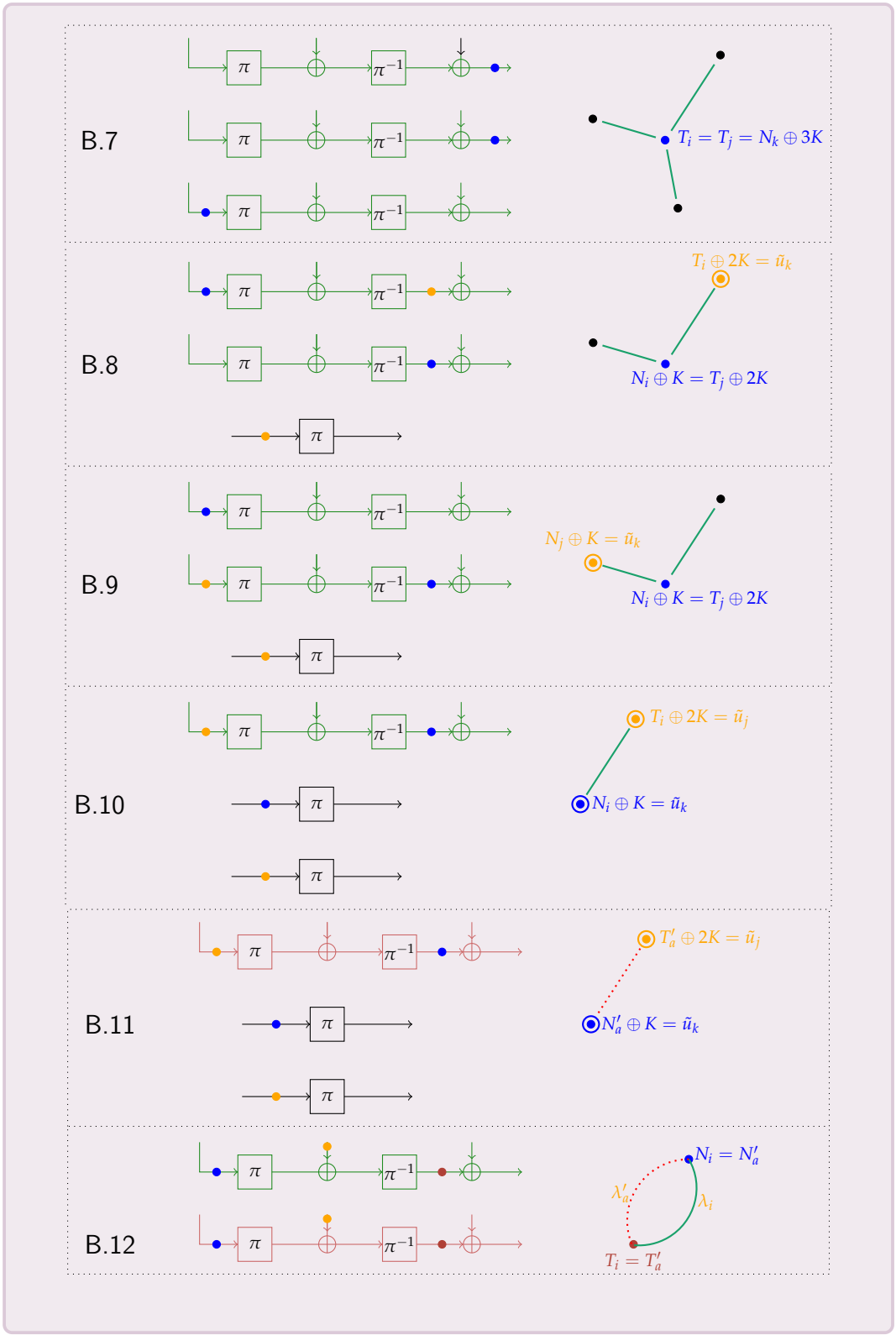
For $\alpha \neq 0$, denoting $A^{(\alpha)}_{(i,j)} = (-1)^{\alpha\cdot(t_i+a_j)}$ and $A^{(\alpha)} = \sum_{(i,j)\in[q_1]\times[q_2]} (-1)^{\alpha\cdot(t_i+a_j)}$, one obtains $\Phi(T^*)\Phi(A^*) = \max_{\alpha\neq 0}\left\{\left|A^{(\alpha)}\right|\right\}$. The random variable $A^{(\alpha)}$ is the sum of q_1+q_2 independent random variables $A^{(\alpha)}_{(i,j)}$ such that $\Pr\left[A^{(\alpha)}_{(i,j)} = 1\right] = \Pr\left[A^{(\alpha)}_{(i,j)} = -1\right] = \frac{1}{2}$. Therefore, by the Chernoff bound given in Corollary 4.8 of [109], for any $a > 0$, $\Pr\left[\left|A^{(\alpha)}\right| \geq a\right] \leq 2e^{-a^2/2(q_1+q_2)}$.
Let $C \geq \sqrt{3n(q_1+q_2)}$. Then $\Pr\left[\left|A^{(\alpha)}\right| \geq C\right] \leq 2e^{-C^2/2(q_1+q_2)}$

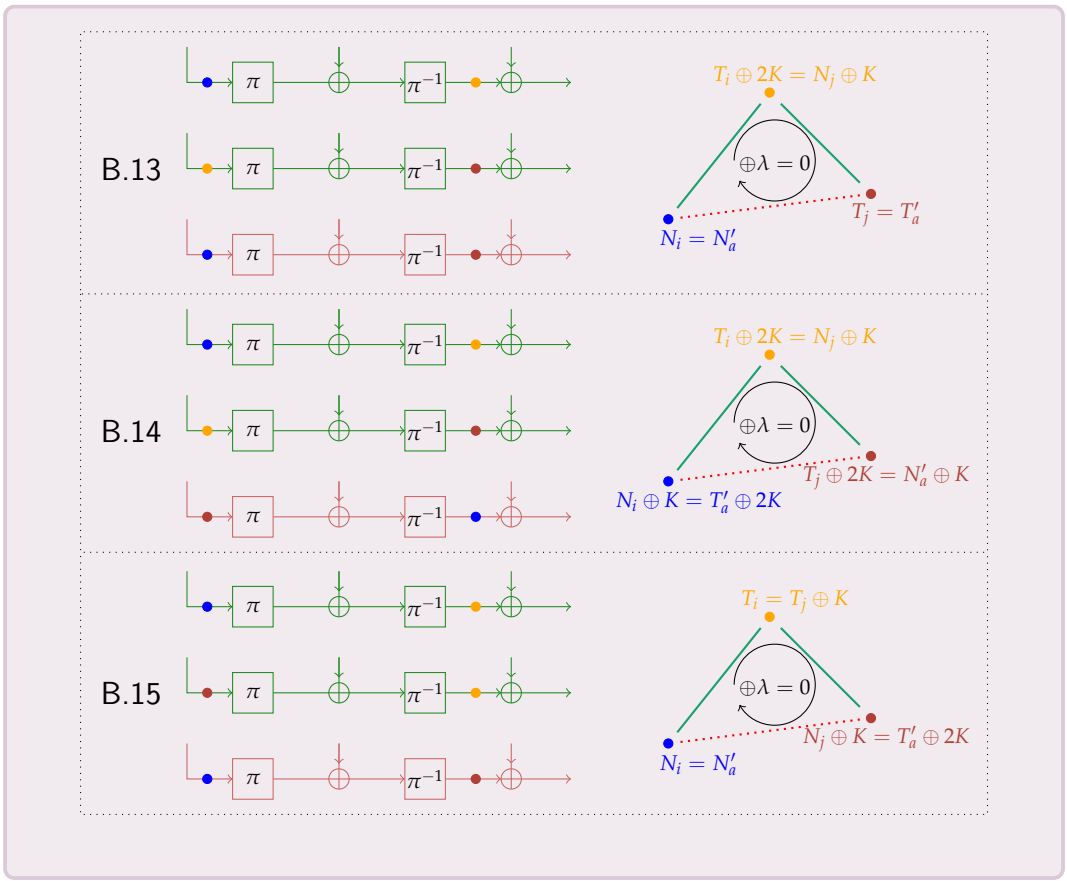
$$
\begin{aligned}
\implies \Pr\left[\mu(T^*) \geq \frac{q_1 q_2 p}{2^n} + \sqrt{\frac{p}{2^n}}\cdot C\right] &\leq \Pr\left[\Phi(T^*)\Phi(A^*) \geq C\right] \\
&= \Pr\left[\max_{\alpha\neq 0}\left\{\left|A^{(\alpha)}\right|\right\} \geq C\right] \\
&\leq \sum_{\alpha\neq 0} \Pr\left[\left|A^{(\alpha)}\right| \geq C\right] \\
&\leq 2e^{-C^2/2(q_1+q_2)} \leq \frac{2}{2^n}, \\
&\text{since } e^{3/4} \geq 2.
\end{aligned}
$$

□

Appendix C. Figures Describing Bad Events for PDM*MAC




B.7
$T_i = T_j = N_k \oplus 3K$
B.8
$T_i \oplus 2K = \tilde{u}_k$
$N_i \oplus K = T_j \oplus 2K$
B.9
$N_j \oplus K = \tilde{u}_k$
$N_i \oplus K = T_j \oplus 2K$
B.10
$T_i \oplus 2K = \tilde{u}_j$
$N_i \oplus K = \tilde{u}_k$
B.11
$T'_a \oplus 2K = \tilde{u}_j$
$N'_a \oplus K = \tilde{u}_k$
B.12
$N_i = N'_a$
λ'_a
λ_i
$T_i = T'_a$


B.13
π
π^{-1}
$T_i \oplus 2K = N_j \oplus K$
$\oplus\lambda = 0$
$T_j = T'_a$
$N_i = N'_a$
B.14
$T_i \oplus 2K = N_j \oplus K$
$\oplus\lambda = 0$
$T_j \oplus 2K = N'_a \oplus K$
$N_i \oplus K = T'_a \oplus 2K$
B.15
$T_i = T_j \oplus K$
$\oplus\lambda = 0$
$N_j \oplus K = T'_a \oplus 2K$
$N_i = N'_a$

Appendix D. Sum of Two Independent Random Permutations Under a Conditional Distribution

Let E be a block cipher over n-bits. Based on E, we define the *sum function* as follows:

$$\mathsf{sum}_{k_1,k_2}(x) := \mathsf{E}_{k_1}(x) \oplus \mathsf{E}_{k_2}(x),\ x \in \{0,1\}^n.$$

The security of the sum of two identical random permutations (i.e., when $k_1 = k_2$) under conditional distribution has been studied in [63]. This chapter requires the same result with the change that instead of two identical random permutations, it considers the permutations to be independent (i.e., k_1 and k_2 are independently sampled). Proof of the lemma is straightforward and similar to that of Theorem 2 of [63]. Hence we omit the proof.

Lemma 24. *Let $\mathcal{Y}_1 \subseteq \{0,1\}^n$ and $\mathcal{Y}_2 \subseteq \{0,1\}^n$ be two sets of size s_1 and s_2 respectively. Let $\widetilde{t} := (t_1, \ldots, t_r)$ be a block tuple of length r. We define the following set:*

$$\mathcal{H} := \{(h_i^1, h_i^2)_i : h_i^1 \oplus h_i^2 = t_i\ \forall i \in [r],\ (h_i^b)_i \in (\{0,1\}^n \setminus \mathcal{Y}_b)^{(r)}\ \forall b \in [2]\}.$$

Then we have the following lower bound on the cardinality of $\mathcal{H}$:

$$|\mathcal{H}| \geq \frac{(2^n - s_1)_r (2^n - s_2)_r}{2^{nr}} \left(1 - \frac{rs_1s_2 + r^2(s_1+s_2) + r^3}{(2^n - s_1 - r)(2^n - s_2 - r)}\right).$$

Moreover, if $s_1 + r \leq 2^{n-1}$ and $s_2 + r \leq 2^{n-1}$, then we have

$$|\mathcal{H}| \geq \frac{(2^n - s_1)_r (2^n - s_2)_r}{2^{nr}} \left(1 - \frac{4rs_1s_2 + 4r^2(s_1+s_2) + 4r^3}{2^{2n}}\right).$$

Appendix E. Some Results on Linear Algebra

Let A be a matrix of dimension $s \times t$ defined over $\{0,1\}^n$. A_{ij} denotes the element in its i^{th} row and j^{th} column. For a column vector $\widetilde{C}$ of dimension $s \times 1$, $A \| \widetilde{C}$ denotes the augmented matrix of dimension $s \times (t+1)$. For any row vector $\widetilde{R} := (r_1, \ldots, r_t)$ of dimension $1 \times t$, transpose of row vector $\widetilde{R}$, denoted as $\widetilde{R}^{\mathsf{T}}$, denotes the column vector

$$\widetilde{R}^{\mathsf{T}} := \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_t \end{pmatrix}$$

of dimension $t \times 1$. One can represent any system of s linear equations with t unknowns $\widetilde{Y} := (Y_1, \ldots, Y_t)$ defined over $\{0,1\}^n$, denoted as $\mathcal{L}$, as a matrix A of dimension $s \times t$, where the i^{th} equation $\mathcal{L}_i := a_{i1} \cdot Y_1 \oplus \ldots \oplus a_{it} \cdot Y_t = c_i$, where $c_i \in \{0,1\}^n$, corresponds to the i^{th} row vector of A as $\widetilde{a}_i := (a_{i1}, \ldots, a_{it})$. We say $\mathcal{L}$ is *consistent* if it has at least one solution, otherwise we call it *inconsistent*. For $\mathcal{L}$ to be consistent, one must have $\mathsf{rank}(A) = \mathsf{rank}(A \| \widetilde{C})$, where the rank of a matrix A is defined as the maximum number of linearly independent columns of A and $\widetilde{C} = (c_1, \ldots, c_s)^{\mathsf{T}}$. $\mathcal{L}$ has a unique solution if $\mathsf{rank}(A) = t$ and it has many solutions if $\mathsf{rank}(A) < t$.

Let $A \cdot \widetilde{Y}^{\mathsf{T}} = \widetilde{C}$ represent a system of s linear equations with t unknowns $\widetilde{Y}$ defined over $\{0,1\}^n$, where $\mathsf{rank}(A) = r$ and the elements of A are from $\{0,1\}^n$. Let $\widetilde{Y} \xleftarrow{\text{wor}} \mathcal{Y} \subseteq \{0,1\}^n$ and $\widetilde{C}$ is any arbitrary column vector of dimension $s \times 1$ with its elements from $\{0,1\}^n$. Thus, the probability of realizing a particular solution is at most $\frac{1}{(|\mathcal{Y}|-t+r)_r}$ as stated formally in the following lemma, proof of which can be found in [63].

Lemma 25. *Let $\widetilde{Y} := (Y_1, \ldots, Y_t)$ be without replacement samples from a set $\mathcal{Y} \subseteq \{0,1\}^n$ and A be a matrix of dimension $s \times t$ defined over $\{0,1\}^n$. Then, for any given column vector $\widetilde{C}$ of dimension $s \times 1$ over $\{0,1\}^n$, we have*

$$\Pr[(A)_{s \times t} \cdot \widetilde{Y}^T = \widetilde{C}] \leq \frac{1}{(|\mathcal{Y}| - t + r)_r},$$

where $r = \mathsf{rank}(A)$.