On the XOR of Multiple Random Permutations

Bart Mennink(✉) and Bart Preneel

Department of Electrical Engineering, ESAT/COSIC,
KU Leuven, and iMinds, Leuven, Belgium
{bart.mennink,bart.preneel}@esat.kuleuven.be

Abstract. A straightforward way of constructing an n-bit pseudorandom function is to XOR two or more pseudorandom permutations: $p_1 \oplus \ldots \oplus p_k$. This XOR construction has gained broad attention over the last two decades. In this work, we revisit the security of this well-established construction. We consider the case where the underlying permutations are considered secret, as well as the case where these permutations are publicly available to the adversary. In the secret permutation setting, we present a simple reduction showing that the XOR construction achieves optimal 2^n security for all $k \geq 2$, therewith improving a recent result of Cogliati et al. (FSE 2014). Regarding the public permutation setting, Mandal et al. (INDOCRYPT 2010) proved $2^{2n/3}$ security for the case $k = 2$, but we point out the existence of a non-trivial flaw in the proof. We re-establish and generalize the claimed security bound for general $k \geq 2$ using a different proof approach.

Keywords: XOR of permutations · Indifferentiability · Beyond birthday bound · H-coefficient technique

1 Introduction

A fundamental research question in cryptography is how to construct a pseudorandom function (PRF) from a pseudorandom permutation (PRP). The first to formally consider this problem were Bellare et al. [21]. They named the problem "Luby-Rackoff backwards", referring to the celebrated result by Luby and Rackoff who showed how to construct a PRP from a PRF [31]. Their PRF construction consisted of two sequential block cipher calls, where the output of the first call is the key input to the second one: $f(k, x) = E(E(k, x), x)$. This construction only achieves security up to the birthday bound on the output size.

Various methods to construct a PRF from a PRP have been presented that achieve security beyond the $2^{n/2}$ birthday bound, the most notable approach being the XOR of multiple n-bit permutations. In more detail, let $p_1, \ldots, p_k$ be $k \geq 1$ n-bit permutations, and define the following function:

$$f_k = p_1 \oplus \cdots \oplus p_k. \tag{1}$$

For $k = 1$, the security of f_1 is commonly known as the PRP-PRF switch, and primary analysis on this function has, among others, been performed by

T. Malkin et al. (Eds.): ACNS 2015, LNCS 9092, pp. 619–634, 2015.
DOI: 10.1007/978-3-319-28166-7_30

Impagliazzo and Rudich [25], Black et al. [19], Hall et al. [29], and Bellare and Rogaway [4]. For general $k \geq 1$, Lucks [9] proved that this function is a secure PRF up to about $2^{\frac{k}{k+1}n}$ queries. For $k = 2$, Bellare and Impagliazzo [18] proved security up to approximately $2^n/n^{2/3}$, and Patarin [30] improved this bound to approximately 2^n. The latter result is proven using the H-coefficient technique [10], a proof technique that has recently been revisited by Chen and Steinberger [27] and found adoption (among others) in the security of key alternating ciphers [27], cascade encryption [1], and MACs [5,24]. Using the same techniques, Cogliati et al. [28] recently improved the security bounds of f_k for $k \geq 3$, proving that it behaves like a PRF up to approximately $2^{\frac{2k+1}{2k+2}n}$ queries. The authors also mention that the bound could be improved to 2^n, via methods similar to the iterative method employed by Patarin [30], but no proof is given. The state of the art is summarized in Table 1.

All of above-mentioned results are in the secret permutation setting. In more detail, one considers an adversary that is given access to either f_k (using secret permutations), or a random function $\mathcal{R}$, and its goal is to *distinguish* both worlds. While to a certain degree it is possible to view the permutations as secret – one can consider them being instantiated as block ciphers with fixed and secret keys – a novel trend in cryptography is to view permutations as standalone and publicly available objects. For instance, various permutation-based hash functions have appeared over the last years [6,7,16,20,22,23] and the recently started CAESAR competition [11] received various permutation-based submissions, and all of these constructions have been analyzed in the public permutation model. If we wish to consider f_k in the case where the underlying permutations are publicly available, the indistinguishability model is deficient. An improved notion is the *indifferentiability* framework, introduced by Maurer et al. [15]. Informally, it gives a sufficient condition under which an ideal functionality $\mathcal{R}$ can be replaced by f_k using ideal, publicly available, primitives $p = (p_1, \ldots, p_k)$. Indifferentiability proofs consider the existence of a simulator $\mathcal{S}$ with access to $\mathcal{R}$ such that (f_k, p) on the one hand and $(\mathcal{R}, \mathcal{S})$ on the other hand are indistinguishable. In this indifferentiability model, Mandal et al. [2] proved that f_2 achieves $\mathcal{O}(2^{2n/3})$ security. The authors conjecture that their simulator allows to achieve optimal $\mathcal{O}(2^n)$ indifferentiability. An additional open problem is to generalize this result to $k > 2$ permutations. Table 1 also summarizes the state of the art for the public permutation setting.

A related result is the construction of a permutation XORed with its inverse, $p \oplus p^{-1}$, as introduced by Dodis et al. [12]. However, this construction is only proven to achieve indifferentiability security up to the birthday bound.

Our Contributions

We revisit the state of the art in both the secret permutation setting and the public permutation setting.

Starting with security in the secret permutation setting, we present an alternative and short proof showing that f_k indeed achieves 2^n indistinguishability

Table 1. State of the art for indistinguishability (first) and indifferentiability (second). Results in **bold** are derived in this work

	k	bound	reference	remark
indistinguishability (p_i secret)	≥ 1	$2^{\frac{k}{k+1}n}$	[9]	
	2	$2^n/n^{2/3}$	[18]	
	2	2^n	[30]	
	≥ 3	$2^{\frac{2k+1}{2k+2}n}$	[28]	conjectured 2^n
	$\boldsymbol{\geq 3}$	**$\boldsymbol{2^n}$**	**Sect. 3**	
indifferentiability (p_i public)	2	$2^{n/2}$	[2]	
	2	$2^{2n/3}$	[2]	**flawed (Sect. 4.3)**
	$\boldsymbol{\geq 2}$	**$\boldsymbol{2^{2n/3}}$**	**Sect. 4**	

security for all $k \geq 3$. The proof is fairly straightforward, consisting of a reduction of the security of f_{k+1} to f_k for all $k \geq 2$, and using Patarin's proof of 2^n security of f_2 [30]. The proof is simpler than the one suggested by Cogliati et al. to achieve 2^n security [28], but the price to pay is a slightly worse security bound. (The difference lies in the security exponent. Informally, this is a value c such that the security bound behaves like $(q/2^n)^c$. A larger c means a sharper curve for the security advantage, or in other words that the threshold value q_0 such that $(q_0/2^n)^c = 1/2$, is higher. The approach suggested in [28] is expected to result in a larger security exponent.)

Regarding security in the public permutation setting, we revisit the work of Mandal et al. [2] and note that the proof contains a subtle but non-negligible flaw. The bug appears in the technical part of the proof, it is not straightforwardly fixable, and thus invalidates the security result, leaving the indifferentiability of f_2 *beyond* $2^{n/2}$ as an open problem. Nevertheless, the mistake does not have a direct influence on the proposed simulator. For a generalization of their simulator to $k \geq 2$ rounds, we next restore the claimed security bound. In more detail, we re-confirm that f_k achieves at least $2^{2n/3}$ indifferentiability security. The security result is obtained by following a different proof approach and avoiding the flawed part all the way. The new proof particularly relies on a result from the area of Fourier theory proven by Babai [3], Steinberger [13], and Chen et al. [14], that (informally) bounds the number of solutions to $a \oplus b = c$ for $(a, b, c) \in A \times B \times C$, where C is a set of random elements and A and B are two arbitrarily chosen sets of size $|C|$ (details follow in Sect. 4.4). This problem found earlier adoption in the area of permutation-based hashing [23], digital signatures [17], and the security of Even-Mansour [14].

The new results are also included in Table 1.

Outline

We introduce some mathematical preliminaries and discuss the indistinguishability and indifferentiability models in Sect. 2. We present our short and alternative proof for the indistinguishability of f_k in Sect. 3. A new indifferentiability proof for $k \geq 2$, using a generalization of the simulator of Mandal et al. [2], is given in Sect. 4. The work is concluded in Sect. 5. In this section, we also elaborate on possible improvements of our result to 2^n security.

2 Preliminaries

Let $n \geq 1$ be an integer. By $\mathsf{Func}(n)$ we denote the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$ and by $\mathsf{Perm}(n)$ the set of all permutations on $\{0,1\}^n$. For a set $\mathcal{X}$, we denote by $x \xleftarrow{\$} \mathcal{X}$ the uniformly random sampling of an element from $\mathcal{X}$. If x and y are two bitstrings of the same size, $x \oplus y$ denotes their bitwise XOR.

Throughout, a distinguisher $\mathcal{D}$ is a computationally unbounded probabilistic algorithm that has oracle access to one or more oracles $\mathcal{O}$. The distinguisher can make a certain amount of oracle queries to $\mathcal{O}$, and after this interaction $\mathcal{D}^{\mathcal{O}}$ outputs a 0 or a 1.

Definition 1 (Indistinguishability). *For an integer $k \geq 1$, consider f_k of (1) based on $p = (p_1, \ldots, p_k) \xleftarrow{\$} \mathsf{Perm}(n)^k$. Let $\mathcal{R} \xleftarrow{\$} \mathsf{Func}(n)$. The distinguishing advantage of $\mathcal{D}$ against f_k is defined as*

$$\mathsf{Adv}_{f_k}^{\mathrm{dist}}(\mathcal{D}) = \left|\mathbf{P}\left(\mathcal{D}^{f_k} = 1\right) - \mathbf{P}\left(\mathcal{D}^{\mathcal{R}} = 1\right)\right|,$$

where the probabilities are taken over the randomness of p, $\mathcal{R}$, and $\mathcal{D}$.

Maurer et al. [15] introduced indifferentiability as an extension of indistinguishability, more suitable for the case the underlying primitives are publicly available. Indifferentiability of a function f_k from a random function $\mathcal{R}$, intuitively, means that f_k shows no structural design flaws and that it can replace $\mathcal{R}$ in any construction, up to the indifferentiability security bound of f_k. We employ the adaption and simplification by Coron et al. [26], rewritten in our own terminology.

Definition 2 (Indifferentiability). *For an integer $k \geq 1$, consider f_k of (1) based on $p = (p_1, \ldots, p_k) \xleftarrow{\$} \mathsf{Perm}(n)^k$. Let $\mathcal{R} \xleftarrow{\$} \mathsf{Func}(n)$. Let $\mathcal{S}$ be a simulator with the same interface as p and with oracle access to $\mathcal{R}$. The differentiating advantage of $\mathcal{D}$ against f_k for simulator $\mathcal{S}$ is defined as*

$$\mathsf{Adv}_{f_k,\mathcal{S}}^{\mathrm{diff}}(\mathcal{D}) = \left|\mathbf{P}\left(\mathcal{D}^{f_k,p} = 1\right) - \mathbf{P}\left(\mathcal{D}^{\mathcal{R},\mathcal{S}} = 1\right)\right|,$$

where the probabilities are taken over the randomness of p, $\mathcal{R}$, $\mathcal{S}$, and $\mathcal{D}$.

The indistinguishability and indifferentiability definitions are depicted in Fig. 1.

3 Indistinguishability of f_k

We present a short proof for the indistinguishability of f_k from a random function $\mathcal{R}$ from $\mathsf{Func}(n)$, in accordance with Definition 1. We start with a security reduction of f_{k+1} to f_k for all $k \geq 2$.

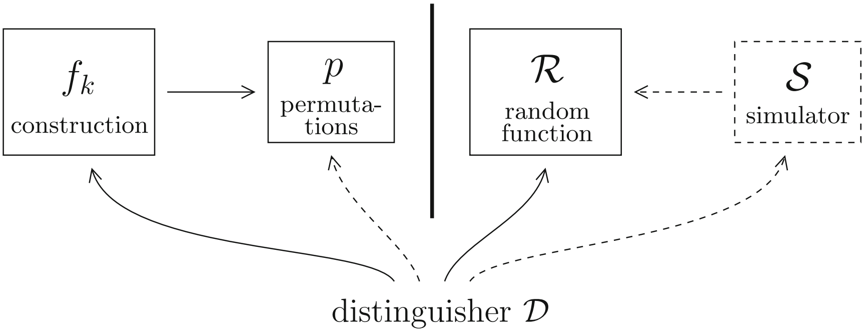

Fig. 1. Indistinguishability (without dashed elements) and indifferentiability (with dashed elements)

Theorem 1. *For all $k \geq 2$, for any distinguisher $\mathcal{D}$, we have $\mathsf{Adv}^{\text{dist}}_{f_{k+1}}(\mathcal{D}) \leq \mathsf{Adv}^{\text{dist}}_{f_k}(\mathcal{D})$.*

Proof. We consider a distinguisher $\mathcal{D}^{\mathcal{O}}$ that has access to an oracle $\mathcal{O}$, either f_{k+1} or $\mathcal{R} \xleftarrow{\$} \mathsf{Func}(n)$, and makes q queries to this oracle. If $\mathcal{D}^{\mathcal{O},\mathcal{O}'}$ is given access to an additional oracle $\mathcal{O}'$ with the same domain as $\mathcal{O}$, this means that for every query $\mathcal{D}$ makes to $\mathcal{O}$, it gets the same query to $\mathcal{O}'$ for free. In other words, if $\mathcal{D}^{\mathcal{O},\mathcal{O}'}$ queries x to its oracle $\mathcal{O}$, it gets as response the values $\mathcal{O}(x)$ *and* $\mathcal{O}'(x)$.

For brevity, we denote $\mathbf{P}\left(\mathcal{D}^{\mathcal{O}} = 1\right) = \mathbf{P}(\mathcal{O})$ and $\mathbf{P}\left(\mathcal{D}^{\mathcal{O},\mathcal{O}'} = 1\right) = \mathbf{P}(\mathcal{O},\mathcal{O}')$. Recall that $f_{k+1} = p_1 \oplus \cdots \oplus p_{k+1}$. By construction:

$$\begin{aligned}\mathsf{Adv}^{\text{dist}}_{f_{k+1}}(\mathcal{D}) &= \left|\mathbf{P}(p_1 \oplus \cdots \oplus p_{k+1}) - \mathbf{P}(\mathcal{R})\right| \\ &\overset{(a)}{\leq} \left|\mathbf{P}(p_1 \oplus \cdots \oplus p_{k+1}, p_{k+1}) - \mathbf{P}(\mathcal{R}, p_{k+1})\right| \\ &\overset{(b)}{=} \left|\mathbf{P}(p_1 \oplus \cdots \oplus p_k, p_{k+1}) - \mathbf{P}(\mathcal{R}, p_{k+1})\right| \\ &\overset{(c)}{=} \left|\mathbf{P}(p_1 \oplus \cdots \oplus p_k) - \mathbf{P}(\mathcal{R})\right| = \mathsf{Adv}^{\text{dist}}_{f_k}(\mathcal{D}),\end{aligned}$$

where (a) holds as extra access may only increase the advantage, (b) holds as $(p_1 \oplus \cdots \oplus p_k, p_{k+1})$ can be computed from $(p_1 \oplus \cdots \oplus p_{k+1}, p_{k+1})$ and vice versa, and (c) holds as p_{k+1} is an independent permutation. □

Next, we recall the result of Patarin [30] on the indistinguishability of f_2.

Lemma 1 (Patarin [30]). *For any $\mathcal{D}$ making q oracle queries, we have $\mathsf{Adv}^{\text{dist}}_{f_2}(\mathcal{D}) = \mathcal{O}(q/2^n)$.*

From Theorem 1 and Lemma 1, the following corollary immediately follows, showing that f_k is indistinguishable up to about 2^n queries, for all $k \geq 2$.

Corollary 1. *For all $k \geq 2$, for any $\mathcal{D}$ making q oracle queries, we have $\mathsf{Adv}^{\text{dist}}_{f_k}(\mathcal{D}) = \mathcal{O}(q/2^n)$.*

4 Indifferentiability of f_k

In this section we consider the indifferentiability of f_k (cf. Definition 2), in case the underlying permutations are public. We prove the following result.

Theorem 2. *For all $k \geq 2$, there exists a simulator $\mathcal{S}$ such that for any $\mathcal{D}$ making $q \geq 9n$ oracle queries,*

$$\mathsf{Adv}^{\mathrm{diff}}_{f_k,\mathcal{S}}(\mathcal{D}) \leq \frac{4q^3}{2^{2n}} + \frac{3n^{1/2}q^{3/2}}{2^n} + \frac{2}{2^n} \,.$$

The simulator makes at most $2q$ queries to $\mathcal{R}$.

While the theorem is stated for general k, the bound is independent of k. This is caused by the fact that we consider a direct generalization of the simulator of Mandal et al. [2], but the core problems that determine the bound find their roots in the basic case of $k = 2$. We refer to Sect. 5 for a more detailed discussion.

The remainder of the section is organized as follows. Firstly, we describe a generalization of the simulator $\mathcal{S}$ introduced by Mandal et al. [2] to $k \geq 2$ (in Sect. 4.1). Secondly, we present Patarin's H-coefficient technique upon which the proof is based, along with some preliminary observations (in Sect. 4.2). These follow [2] with the difference that we use the re-formalization of Patarin's technique by Chen and Steinberger [27]. Thirdly, we discuss the original indifferentiability proof of [2] (in Sect. 4.3). Fourthly, we present our new proof (in Sect. 4.4).

4.1 Simulator

We describe the simulator used in our work. It is a direct generalization of the simulator $\mathcal{S}$ of Mandal et al. [2] to a general number of $k \geq 2$ permutations.

The goal of the simulator $\mathcal{S}$ is to mimic the permutations $p = (p_1, \ldots, p_k)$ in such a way that (f_k, p) and $(\mathcal{R}, \mathcal{S})$ look indistinguishable. $\mathcal{S}$ therefore has the same interface as p, and we write $\mathcal{S} = (\mathcal{S}_1, \ldots, \mathcal{S}_k)$. The distinguisher can make forward and inverse queries to each of these functionalities, which means that it can query $\mathcal{S}$ in $2k$ ways. However, the simulator should look like $\mathcal{R} = \mathcal{S}_1 \oplus \cdots \oplus \mathcal{S}_k$, and if a distinguisher would, for instance, query $\mathcal{S}_1(x)$, it very likely also wishes to know $\mathcal{S}_2(x), \ldots, \mathcal{S}_k(x)$. To suit the analysis, we model the simulator in such a way that on a forward query x, the distinguisher is given all values $\mathcal{S}(x) = (\mathcal{S}_1(x), \ldots, \mathcal{S}_k(x))$. This simplification essentially corresponds to giving the distinguisher $k - 1$ "free" queries. It also means that $\mathcal{S}$ has only one interface for forward queries.

A similar issue arises for inverse queries. If the distinguisher makes a query to $\mathcal{S}_\ell^{-1}$ for $\ell \in \{1, \ldots, k\}$, the simulator will not only output a preimage x, but also the corresponding range values $\mathcal{S}_1(x), \ldots, \mathcal{S}_{\ell-1}(x), \mathcal{S}_{\ell+1}(x), \ldots, \mathcal{S}_k(x)$. Also here, the distinguisher essentially gets $k - 1$ queries for free.

The simulator maintains a sequence of responses $\{(x_i, y_i^1, \ldots, y_i^k)\}_{i=1}^q$, where q denotes the number of queries to $\mathcal{S}$. These tuples correspond to the evaluations

$$\mathcal{S}(x_i) = (\mathcal{S}_1(x_i), \ldots, \mathcal{S}_k(x_i)) = (y_i^1, \ldots, y_i^k) \,,$$

for $i = 1, \ldots, q$. Note that every forward query as well as every inverse query to $\mathcal{S}$ results in exactly one such tuple. Here and throughout, we assume $\mathcal{D}$ never repeats an old query, e.g., in a forward query $\mathcal{S}(x_i)$, we have $x_i \notin \{x_1, \ldots, x_{i-1}\}$.

The simulator $\mathcal{S}$ is defined as follows. We consider its description for the ith query, for $i \in \{1, \ldots, q\}$. We describe the simulator for forward queries $\mathcal{S}(x_i)$, and for inverse queries $\mathcal{S}_\ell^{-1}(y_i^\ell)$ for $\ell \in \{1, \ldots, k\}$.

Forward Query $\boldsymbol{\mathcal{S}(x_i)}$. For $\ell = 3, \ldots, k$, the simulator draws y_i^ℓ uniformly randomly from $\{0,1\}^n \backslash \{y_1^\ell, \ldots, y_{i-1}^\ell\}$. Then, it queries x_i to $\mathcal{R}$ and generates y_i^1 uniformly randomly from

$$\{0,1\}^n \backslash \{y_1^1, \ldots, y_{i-1}^1, \bar{y}_i \oplus y_1^2, \ldots, \bar{y}_i \oplus y_{i-1}^2\} \tag{2}$$

where we define $\bar{y}_i = \mathcal{R}(x_i) \oplus y_i^3 \oplus \ldots \oplus y_i^k$. Finally, it sets $y_i^2 = \bar{y}_i \oplus y_i^1$.

Informally, $\mathcal{S}(x_i)$ selects random y_i^ℓ for $\ell = 3, \ldots, k$, and uses y_i^1 and y_i^2 to make sure that $\mathcal{R}(x_i) = y_i^1 \oplus \ldots \oplus y_i^k$. Note that, due to the drawing of y_i^1 from (2), we have $y_i^2 \notin \{y_1^2, \ldots, y_{i-1}^2\}$.

Inverse Query $\boldsymbol{\mathcal{S}_\ell^{-1}(y_i^\ell)}$. The simulator generates its response as follows.

(1) Draw $y_i^{\ell'}$ uniformly randomly from $\{0,1\}^n \backslash \{y_1^{\ell'}, \ldots, y_{i-1}^{\ell'}\}$ for $\ell' \in \{\ell + 1, \ldots, \ell + k - 2\}$;[1]
(2) Draw x_i uniformly randomly from $\{0,1\}^n \backslash \{x_1, \ldots, x_{i-1}\}$ and query x_i to $\mathcal{R}$;
(3) Set $y_i^{\ell-1} = \mathcal{R}(x_i) \oplus y_i^\ell \oplus \ldots \oplus y_i^{\ell+k-2}$. If $y_i^{\ell-1} \in \{y_1^{\ell-1}, \ldots, y_{i-1}^{\ell-1}\}$, return to (2).

We call a drawing x_i such that the resulting value $y_i^{\ell-1}$ in step (3) is not new a "failed guess". As in [2], in the proof we will limit the simulator to make at most 2 attempts (and thus at most 1 failed guess) per query. The simulator will abort once it exceeds this bound for some query.

4.2 Patarin's Technique

Fix any distinguisher $\mathcal{D}$ making q queries. As it is computationally unbounded, without loss of generality we can assume it is deterministic. We summarize the interaction of $\mathcal{D}$ with its oracles by a transcript τ, which consists of all query-response tuples $\mathcal{D}$ sees during its interaction with its oracles. We assume $\mathcal{D}$ never makes duplicate queries. The set of all possible transcripts is denoted by $\mathcal{T}$. Denote by X (resp. Y) the probability distribution of transcripts in the ideal (resp. simulated) world, for the fixed deterministic distinguisher $\mathcal{D}$.

Patarin's H-coefficient technique [27,30] states the following.[2]

[1] Here and throughout, all indices are taken modulo k and in the range $\{1, \ldots, k\}$.

[2] The H-coefficient technique in fact applies to indistinguishability in general, but to suit the presentation, we introduce it in the context of the indifferentiability of f_k.

Lemma 2 (H-coefficient Technique [27,30]). *Consider a fixed deterministic distinguisher $\mathcal{D}$. Let $\mathcal{T} = \mathcal{T}_{\mathrm{good}} \cup \mathcal{T}_{\mathrm{bad}}$ be a partition of the set of transcripts. Let ε be such that for all $\tau \in \mathcal{T}_{\mathrm{good}}$,*

$$\mathbf{P}\left(Y = \tau\right) \geq \mathbf{P}\left(X = \tau\right) \cdot (1 - \varepsilon)\,. \tag{3}$$

Then, $\mathsf{Adv}^{\mathrm{diff}}_{f_k,\mathcal{S}}(\mathcal{D}) \leq \varepsilon + \mathbf{P}\left(X \in \mathcal{T}_{\mathrm{bad}}\right)$.

Proof. The proof is fairly straightforward, and we include it for completeness. We refer to [27] for a more detailed discussion.

We consider a deterministic distinguisher $\mathcal{D}$, and as such, its differentiating advantage equals the statistical distance between the distributions of transcripts in the ideal and simulated world:

$$\begin{aligned}
\mathsf{Adv}^{\mathrm{diff}}_{f_k,\mathcal{S}}(\mathcal{D}) &= \frac{1}{2} \sum_{\tau \in \mathcal{T}} \left| \mathbf{P}\left(X = \tau\right) - \mathbf{P}\left(Y = \tau\right) \right| \\
&\overset{(a)}{=} \sum_{\tau \in \mathcal{T} : \mathbf{P}(X=\tau) > \mathbf{P}(Y=\tau)} \left(\mathbf{P}\left(X = \tau\right) - \mathbf{P}\left(Y = \tau\right) \right) \\
&\overset{(b)}{=} \sum_{\tau \in \mathcal{T} : \mathbf{P}(X=\tau) > \mathbf{P}(Y=\tau)} \mathbf{P}\left(X = \tau\right) \left(1 - \frac{\mathbf{P}\left(Y = \tau\right)}{\mathbf{P}\left(X = \tau\right)} \right) \\
&\overset{(c)}{\leq} \sum_{\tau \in \mathcal{T}_{\mathrm{good}}} \mathbf{P}\left(X = \tau\right) \varepsilon + \sum_{\tau \in \mathcal{T}_{\mathrm{bad}}} \mathbf{P}\left(X = \tau\right) \\
&\leq \varepsilon + \mathbf{P}\left(X \in \mathcal{T}_{\mathrm{bad}}\right),
\end{aligned}$$

where (a) holds by symmetry, (b) as $\mathbf{P}\left(X = \tau\right) > 0$ by construction, and (c) holds by (3). □

The main idea of the technique is exposed in the last step: for almost all transcripts (the good ones), the ratio of (3) will be rather close to one and for these transcripts we can take ε close to 0. For the few bad transcripts, ε may become large (even close to 1). Additionally, the technique allows us to focus on fixed transcripts and compute the probability of such a transcript to occur.

We build the following distinguisher $\mathcal{D}'$ on top of $\mathcal{D}$. Distinguisher $\mathcal{D}'$ operates as $\mathcal{D}$, and particularly outputs the same decision. However, at the end $\mathcal{D}'$ will make an additional amount of q_1 primitive queries to $p/\mathcal{S}$ as follows: for each of the queries to $f_k/\mathcal{R}$ it has made, $\mathcal{D}'$ makes the same query to $p/\mathcal{S}$, except if this would imply a duplicate primitive query in which case $\mathcal{D}'$ may replace it with a random non-repeating query. Clearly, $\mathcal{D}$ and $\mathcal{D}'$ always output the same decision, and hence $\mathsf{Adv}^{\mathrm{diff}}_{f_k,\mathcal{S}}(\mathcal{D}) = \mathsf{Adv}^{\mathrm{diff}}_{f_k,\mathcal{S}}(\mathcal{D}')$. Also, if $\mathcal{D}$ makes q_1 queries to its construction oracle and q_2 queries to its primitive, then $\mathcal{D}'$ makes exactly q_1 additional queries to its primitive. Note that, particularly, $\mathcal{D}'$ makes q queries to the primitive. In a transcript of $\mathcal{D}'$, all queries to the construction oracle (f_k or $\mathcal{R}$) are encapsulated in the queries to the primitive oracle (p or $\mathcal{S}$). Therefore, this approach reduces our problem to the problem of comparing $(p_1, \ldots, p_k)$ with $(\mathcal{S}_1, \ldots, \mathcal{S}_k)$, the former called the ideal and the latter the simulated world.

Finally, recall that the simulator may abort. This is formalized by including in the transcript a dedicated symbol $b \in \{\top, \bot\}$. In the ideal world, we always have $b = \top$, and in the simulated world, $b = \top$ *unless* the simulator aborted. Note that if $b = \bot$, the distinguisher succeeds with probability 1. In fact, in this case the transcript will be considered a bad transcript, and due to Lemma 2, the technical part of the work centers around good transcripts.

Let $\tau = \big(\{(x_i, y_i^1, \dots, y_i^k)\}_{i=1}^q, b\big) \in \mathcal{T}$ be any transcript that can be seen by distinguisher $\mathcal{D}'$. Note that, as $\mathcal{D}'$ makes no duplicate queries, we have $x_i \neq x_{i'}$ and $y_i^\ell \neq y_{i'}^\ell$ for all i, i', ℓ. For arbitrary $z \in \{0,1\}^n$, we define

$$N(z) = \{(j, j') \in \{1, \dots, q\}^2 \mid y_j^1 \oplus y_{j'}^2 = z\}\,. \tag{4}$$

4.3 Intermezzo: Proof of Mandal et al. [2]

The skeleton of our proof is similar to the one of [2]. Differences arise at the definition of the bad event, and the remainder of the proof. Before proceeding with our proof, we revisit the one of [2] at a high level (in our terminology), point out the presence of a flaw, and briefly discuss to what extent our proposed fix differs. Recall that the proof of [2] is for $k = 2$.

In the original proof, a transcript[3] $\tau = \{(x_i, y_i^1, y_i^2)\}_{i=1}^q$ is called "bad" if $N(z) > \frac{24q^2}{2^n - q}$ for some $z \in \{0,1\}^n$. In [2, Theorem 5], it is then proven that

$$\mathbf{P}\left(X \in \mathcal{T}_{\text{bad}}\right) = \mathbf{P}\left(\exists z \in \{0,1\}^n \;:\; N(z) > \frac{24q^2}{2^n - q}\right) \leq 1/2^{11n}.$$

The proof assumes randomness of $\{(y_i^1, y_i^2)\}_{i=1}^q$, but if an adversary makes an inverse query to one of its primitive oracles, it can freely choose y_i^1 or y_i^2. Inspired by this, we can consider an adversary that operates as follows (define $q' = q/2$):

- Choose $z \in \{0,1\}^n$;
- Query $y_i^1 \xrightarrow{p_1^{-1}} x_i, y_i^2$ for $i = 1, \dots, q'$, all distinct values;
- Query $y_i^2 = y_{i-q'}^1 \oplus z \xrightarrow{p_2^{-1}} x_i, y_i^1$ for $i = q' + 1, \dots, 2q' = q$, all distinct values.

Then, we have $y_i^1 \oplus y_{i+q'}^2 = z$ for all $i = 1, \dots, q'$. In other words, $N(z) \geq q/2$ after q queries, invalidating the claim for any $2 \leq q \leq 2^n/49$. (In a personal communication, the authors of [2] have confirmed the presence of this flaw.)

We note that a straightforward fix of the proof of [2], consisting of imposing $N(z) \leq \text{const} \cdot q$ for good transcripts, does not work: it only results in $\mathcal{O}(2^{n/2})$ security of the construction. This issue is resolved in our proof by using a structurally different bad event, and relying on existing results from the area of Fourier theory [3,13,14]. Naturally, the employment of a different bad event also leaves its traces in the analysis of good transcripts, as becomes clear from the proof.

[3] The abortion bit b is absent in the original proof.

4.4 Proof of Theorem 2

The proof of Theorem 2 roughly consists of four steps: (i) we define what type of transcripts we consider "bad", (ii) we bound the probability a bad transcript occurs, (iii) we derive a bound on the ratio a good transcript is seen in the real and ideal world, and (iv) the pieces are connected and the proof of Theorem 2 is completed.

The proof differs from the one of [2] in the definition of bad transcripts and the probability analysis thereof, and in the analysis of forward queries for good transcripts.

Bad Transcripts

Let $\tau = \big(\{(x_i, y_i^1, , \ldots, y_i^k)\}_{i=1}^q, b\big) \in \mathcal{T}$ be any attainable transcript. Recall the definition of $N(z)$ for arbitrary $z \in \{0,1\}^n$, Eq. (4). Transcript τ is called *bad* if $b = \bot$, or if

$$\sum_{i=1}^{q} |N(y_i^1 \oplus y_i^2)| > C \tag{5}$$

for some to-be-determined $C > 0$. Next, we upper bound the probability a bad transcript is obtained in the ideal world, $\mathbf{P}\left(X \in \mathcal{T}_{\text{bad}}\right)$, and lower bound the ratio $\mathbf{P}\left(Y = \tau\right) / \mathbf{P}\left(X = \tau\right)$ for $\tau \in \mathcal{T}_{\text{good}}$.

Upper Bounding $\mathbf{P}\left(X \in \mathcal{T}_{\text{bad}}\right)$

The ideal world never aborts, hence $b = \top$ by construction. Consequently, the badness of transcripts is solely defined based on the values (y_i^1, y_i^2). We isolate the problem, and consider an adversary whose sole objective is to maximize $\sum_{i=1}^q |N(y_i^1 \oplus y_i^2)|$.

In a forward query, the adversary chooses x_i and receives randomly drawn y_i^1 and y_i^2. In an inverse query, it may choose either of the y_i-values and receives a randomly drawn opposite. Therefore, the adversary will be most successful if it only makes inverse queries to p_ℓ^{-1} for $\ell \in \{1, 2\}$. In light of this, we consider an adversary engaged in the following game. For $i = 1, \ldots, q$, either choose a y_i^1 to receive $y_i^2 = p_2 \circ p_1^{-1}(y_i^1)$, or choose a y_i^2 to receive $y_i^1 = (p_2 \circ p_1^{-1})^{-1}(y_i^2)$. Define $z_i = y_i^1 \oplus y_i^2$. The adversary's goal is to maximize

$$\sum_{i=1}^{q} |N(y_i^1 \oplus y_i^2)| = \sum_{i=1}^{q} |N(z_i)| = \left|\{(j, j', i) \in \{1, \ldots, q\}^3 \mid y_j^1 \oplus y_{j'}^2 = z_i\}\right|.$$

Note that, as $p_1, p_2 \xleftarrow{\$} \mathsf{Perm}(n)$, also $\pi = p_2 \circ p_1^{-1}$ behaves like a random permutation. We generalize the game as follows. Let $\pi \xleftarrow{\$} \mathsf{Perm}(n)$. The adversary can query π adaptively and in both directions to obtain two lists $Y^1 = \{y_1^1, \ldots, y_q^1\}$ and $Y^2 = \{y_1^2, \ldots, y_q^2\}$ such that $y_i^2 = p_2 \circ p_1^{-1}(y_i^1)$ for $i = 1, \ldots, q$. Write

$Z = \{z_1, \dots, z_q\}$, with the z_i's as before. Then, its goal is now to find two lists U and V of q elements that maximize

$$\lambda(U, V, Z) = \left|\left\{(u, v, z) \in U \times V \times Z \mid u \oplus v = z\right\}\right| .$$

Note that, by construction,

$$\sum_{i=1}^{q} |N(y_i^1 \oplus y_i^2)| = \lambda(Y^1, Y^2, Z) \leq \max_{U,V:|U|=|V|=q} \lambda(U, V, Z) =: \mu(Z) .$$

We therefore obtain:

$$\mathbf{P}\left(X \in \mathcal{T}_{\mathrm{bad}}\right) \leq \mathbf{P}\left(\sum_{i=1}^{q} |N(y_i^1 \oplus y_i^2)| > C\right) \leq \mathbf{P}\left(\mu(Z) > C\right) .$$

The problem of bounding $\mu(Z)$ appeared before in works on permutation-based hashing by Mennink and Preneel [23], on digital signatures by Kiltz et al. [17], and on the security of Even-Mansour by Chen et al. [14]. It is also known as the "sum-capture problem". We follow Chen et al. [14, Theorem 1], which in turn builds upon Babai [3] and Steinberger [13]:

Lemma 3 (Sum-Capture Problem [14]). *Let $\pi \xleftarrow{\$} \mathsf{Perm}(n)$ be a random permutation. Let $\mathcal{A}$ be some adversary that makes q two-sided adaptive queries to π, resulting in transcript $\{(y_1^1, y_1^2), \dots, (y_q^1, y_q^2)\}$. Write $Z = \{z_1, \dots, z_q\}$, where $z_i = y_i^1 \oplus y_i^2$ for $i = 1, \dots, q$. Then, assuming $9n \leq q \leq 2^n/2$,*

$$\mathbf{P}\left(\mu(Z) > 3q^3/2^n + 3n^{1/2}q^{3/2}\right) \leq \frac{2}{2^n} .$$

We, logically, define $C = 3q^3/2^n + 3n^{1/2}q^{3/2}$.

Lower Bounding Ratio $\mathbf{P}\left(Y = \tau\right) / \mathbf{P}\left(X = \tau\right)$

Let $\tau = \left(\{(x_i, y_i^1, \dots, y_i^k)\}_{i=1}^q, b\right) \in \mathcal{T}_{\mathrm{good}}$ be a good transcript. This particularly implies that $b = \top$ and that the simulator never aborts, and we omit this symbol in the remaining analysis. Note that in the ideal world $p_1, \dots, p_k$ are ideal permutations, and $\mathbf{P}\left(X = \tau\right) = \prod_{i=1}^q 1/\left(2^n - (i-1)\right)^k$. In the remainder, we will compute $\mathbf{P}\left(Y = \tau\right)$. For $\ell = 1, \dots, q$, we denote by e_ℓ the event that the failed guess in the ℓth query (if any) does not equal any $x_1, \dots, x_q$ and has not occurred before (the same condition was posed by Mandal et al. [2]). We write $\mathsf{E}_\ell = \mathsf{e}_1 \wedge \dots \wedge \mathsf{e}_\ell$. Clearly,

$$\mathbf{P}\left(Y = \tau\right) \geq \mathbf{P}\left(Y = \tau \wedge \mathsf{E}_q\right) , \tag{6}$$

and we focus on the latter probability. Denote $\tau_i = (x_i, y_i^1, \dots, y_i^k)$ for $i = 1, \dots, q$. Similarly for random variable Y, denote by Y_i the random variable corresponding to the ith tuple. We have

$$\mathbf{P}\left(Y = \tau \wedge \mathsf{E}_q\right) = \prod_{i=1}^{q} \underbrace{\mathbf{P}\left(Y_i = \tau_i \wedge \mathsf{E}_i \,\middle|\, \forall_{j=1}^{i-1} Y_j = \tau_j \wedge \mathsf{E}_{i-1}\right)}_{\mathbf{P}_i} . \tag{7}$$

We proceed with the analysis of $\mathbf{P}_i$ for $i \in \{1, \dots, q\}$.

Forward Query $\boldsymbol{\mathcal{S}(x_i)}$. Due to attainability of the transcript, x_i is distinct of $x_1, \ldots, x_{i-1}$. Additionally, E_{i-1} implies that x_i has not been queried to $\mathcal{R}$ before. Therefore, the response value $\mathcal{R}(x_i) = y_i^1 \oplus \cdots \oplus y_i^k$ is randomly drawn from a set of size 2^n. The values $y_i^3, \ldots, y_i^k$ are all drawn from a set of size $2^n - (i-1)$. Finally, y_i^1 is uniformly randomly drawn from the set (2) which is of size at most $2^n - 2(i-1) + |N(\bar{y}_i)|$, where $\bar{y}_i = \mathcal{R}(x_i) \oplus y_i^3 \oplus \ldots \oplus y_i^k$. Indeed, the sets

$$\{y_1^1, \ldots, y_{i-1}^1\} \qquad \text{and} \qquad \{\bar{y}_i \oplus y_1^2, \ldots, \bar{y}_i \oplus y_{i-1}^2\}$$

have an overlap of at most $|N(\bar{y}_i)|$. For forward queries we thus have

$$\begin{aligned}\mathbf{P}_i &\geq \frac{1}{\big(2^n - (i-1)\big)^{k-2}} \cdot \frac{1}{2^n} \cdot \frac{1}{2^n - 2(i-1) + |N(\bar{y}_i)|} \\ &\geq \frac{1}{\big(2^n - (i-1)\big)^{k}} \cdot \left(1 - \frac{|N(\bar{y}_i)|}{2^n}\right),\end{aligned}$$

which follows from the fact that (writing $B = |N(\bar{y}_i)|$)

$$\begin{aligned}\frac{1}{2^n} \cdot \frac{1}{2^n - 2(i-1) + B} &= \frac{1}{\big(2^n - (i-1)\big)^2} \cdot \frac{(2^n - (i-1))^2}{2^n(2^n - 2(i-1) + B)} \\ &= \frac{1}{\big(2^n - (i-1)\big)^2} \cdot \left(1 - \frac{B}{2^n} \cdot \frac{2^n - (i-1)^2/B}{2^n - 2(i-1) + B}\right) \\ &\geq \frac{1}{\big(2^n - (i-1)\big)^2} \cdot \left(1 - \frac{B}{2^n}\right),\end{aligned}$$

where in the last step we use that $2^n - (i-1)^2/B \leq 2^n - 2(i-1) + B$ as $(i-1)^2/B - 2(i-1) + B = (i-1-B)^2/B \geq 0$.

Finally, as $\bar{y}_i = y_i^1 \oplus y_i^2$ by construction, we have $B = |N(\bar{y}_i)| = |N(y_i^1 \oplus y_i^2)|$.

Inverse Query $\boldsymbol{\mathcal{S}_\ell^{-1}(y_i^\ell)(\ell \in \{1, \ldots, k\})}$. Regarding x_i, the simulator may make 2 trials in order to find a successful x_ℓ. For $\beta = 1, 2$, denote by $\mathsf{succ}(\beta)$ the event that attempts $1, \ldots, \beta - 1$ failed but attempt β succeeds. Then,

$$\mathbf{P}_i \geq \sum_{\beta=1}^{2} \underbrace{\mathbf{P}\left(Y_i = \tau_i \wedge \mathsf{E}_i \wedge \mathsf{succ}(\beta) \;\middle|\; \forall_{j=1}^{i-1} Y_j = \tau_j \wedge \mathsf{E}_{i-1}\right)}_{\mathbf{P}_{i,\beta}}. \tag{8}$$

Now, $\mathbf{P}_{i,\beta}$ covers the case that (i) the drawings $y_i^{\ell+1}, \ldots, y_i^{\ell+k-2}$ are all correct, (ii) the first guess fails (if $\beta = 2$), and (iii) the βth succeeds. Firstly, $y_i^{\ell+1}, \ldots, y_i^{\ell+k-2}$ are all randomly drawn from a set of size $2^n - (i-1)$. Secondly (if $\beta = 2$), the first guess fails with probability at least

$$\left(1 - \frac{(q - (i-1)) + (i-1)}{2^n - (i-1)}\right) \cdot \frac{i-1}{2^n},$$

where the first fraction comes from the number of invalid guesses x_i (which would violate the conditions in e_i), and the second fraction is because every guess corresponds to a random draw from $\{0,1\}^n$ (by $\mathcal{R}$) and it fails if $\mathcal{R}(x_i) \in \{\bar{y}_i \oplus y_1^{\ell-1}, \dots, \bar{y}_i \oplus y_{i-1}^{\ell-1}\}$, for $\bar{y}_i = y_i^\ell \oplus \cdots \oplus y_i^{\ell+k-2}$. The βth attempt succeeds with probability $\dfrac{1}{2^n-(i-1)} \cdot \dfrac{1}{2^n}$, where x_i is again taken from a set of size $2^n-(i-1)$ and $y_i^{\ell-1}$ is defined as the outcome of $\mathcal{R}$. Therefore, from (8):

$$\begin{aligned}
\mathbf{P}_i &\geq \frac{1}{\big(2^n-(i-1)\big)^{k-1}} \cdot \frac{1}{2^n} \cdot \left(\underbrace{1}_{\beta=1} + \underbrace{\frac{2^n-q-(i-1)}{2^n-(i-1)} \cdot \frac{i-1}{2^n}}_{\beta=2} \right) \\
&= \frac{1}{\big(2^n-(i-1)\big)^{k}} \cdot \frac{2^n(2^n-(i-1))+(2^n-q-(i-1))(i-1)}{2^{2n}} \\
&= \frac{1}{\big(2^n-(i-1)\big)^{k}} \cdot \left(1 - \frac{(q+(i-1))(i-1)}{2^{2n}}\right) \\
&\geq \frac{1}{\big(2^n-(i-1)\big)^{k}} \cdot \left(1 - \frac{2(i-1)q}{2^{2n}}\right).
\end{aligned}$$

Combination. Combining forward and inverse queries, we find that

$$\mathbf{P}_i \geq \frac{1}{\big(2^n-(i-1)\big)^{k}} \cdot \left(1 - \frac{|N(y_i^1 \oplus y_i^2)|}{2^n} - \frac{2(i-1)q}{2^{2n}}\right),$$

and thus, via (6–7):

$$\begin{aligned}
\mathbf{P}\left(Y=\tau\right) &\geq \mathbf{P}\left(X=\tau\right) \cdot \prod_{i=1}^{q} \left(1 - \frac{|N(y_i^1 \oplus y_i^2)|}{2^n} - \frac{2(i-1)q}{2^{2n}}\right) \\
&\geq \mathbf{P}\left(X=\tau\right) \cdot \left(1 - \sum_{i=1}^{q} \frac{|N(y_i^1 \oplus y_i^2)|}{2^n} - \sum_{i=1}^{q} \frac{2(i-1)q}{2^{2n}}\right) \\
&\geq \mathbf{P}\left(X=\tau\right) \cdot \left(1 - \sum_{i=1}^{q} \frac{|N(y_i^1 \oplus y_i^2)|}{2^n} - \frac{q^3}{2^{2n}}\right).
\end{aligned}$$

As τ is a good transcript, we know that $\sum_{i=1}^{q} |N(y_i^1 \oplus y_i^2)| \leq C = 3q^3/2^n + 3n^{1/2}q^{3/2}$, and hence we obtain,

$$\varepsilon = \frac{4q^3}{2^{2n}} + \frac{3n^{1/2}q^{3/2}}{2^n}. \tag{9}$$

Conclusion of Proof

Using Lemma 2, the value ε of (9) and Lemma 3 for a bound on the probability of a bad transcript combine to

$$\mathsf{Adv}^{\mathrm{diff}}_{f_k,\mathcal{S}}(\mathcal{D}) = \mathsf{Adv}^{\mathrm{diff}}_{f_k,\mathcal{S}}(\mathcal{D}') \leq \frac{4q^3}{2^{2n}} + \frac{3n^{1/2}q^{3/2}}{2^n} + \frac{2}{2^n}.$$

This completes the proof of Theorem 2.

5 Conclusions

Since their first appearance in [18], XOR constructions have received broad attention in the cryptographic community [2,9,10,18,28,30]. As a matter of fact, the security of the XOR construction in the secret permutation setting is well-studied, as reflected in Table 1, and our proof of Corollary 1 closes the case. On the other hand, for the more relevant case of security in the public permutation setting, the only result in this direction [2] claimed $2^{2n/3}$ security. We pointed out a bug in their analysis, and also our proof only guarantees security as long as the number of queries does not exceed this bound.

The original simulator of [2], and more generally the simulator of Sect. 4.1 for $k \geq 2$ is conjectured to allow for security up to $q \ll 2^n$ queries. We expect this to be a highly non-trivial exercise. Our generalized proof clearly shows the bottleneck (in the proof of Mandal [2] this was a bit less clear): while the analysis of the ratio $\mathbf{P}\left(Y=\tau\right)/\mathbf{P}\left(X=\tau\right)$ and the description of bad transcripts as imposed by our analysis leaves little room for tightening, the lossiness of the bound seems to originate from the analysis of $\mathbf{P}\left(X \in \mathcal{T}_{\mathrm{bad}}\right)$, or in more detail that the quantity of (5) is bounded by $\mathcal{O}(q^3/2^n)$. The bound we derive on this probability, however, relies on various well-established results from Fourier theory [3,13,14].

A possible alternative improvement lies in the description of the simulator. Indeed, the presented simulator is constructed to effectively use two out of k of its responses to comply with $\mathcal{R}$. It may be possible to generate its responses so as to minimize the quantity of (5) or a generalized variant thereof. This, however, leads to a simulator that is significantly harder to analyze, and it may additionally influence the ratio for good transcripts. We recall that, already for the case $k = 2$, optimal security is conjectured.

Acknowledgments. The authors would like to thank Elena Andreeva, Rodolphe Lampe, Atul Luykx, Avradip Mandal, and Jacques Patarin for their comments. This work was supported in part by the Research Fund KU Leuven, OT/13/071, and in part by the Research Council KU Leuven: GOA TENSE (GOA/11/007). Bart Mennink is a Postdoctoral Fellow of the Research Foundation – Flanders (FWO).

References

1. Assche, G., Andreeva, E., Mennink, B., Daemen, J.: Security of keyed sponge constructions using a modular proof approach. In: Leander, G. (ed.) FSE 2015. LNCS, vol. 9054, pp. 364–384. Springer, Heidelberg (2015)
2. Babai, L.: The Fourier Transform and Equations over Finite Abelian Groups (Lecture Notes, version 1.3) (2002). http://people.cs.uchicago.edu/laci/reu02/fourier.pdf
3. Bellare, M., Impagliazzo, R.: 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 (1999)
4. Bellare, M., Kilian, J., Rogaway, P.: The security of cipher block chaining. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 341–358. Springer, Heidelberg (1994)

5. Bellare, M., Krovetz, T., Rogaway, P.: Luby-Rackoff backwards: increasing security by making block ciphers non-invertible. In: Nyberg, K. (ed.) EUROCRYPT 1998. LNCS, vol. 1403, pp. 266–280. Springer, Heidelberg (1998)
6. Bellare, M., Rogaway, P.: The security of triple encryption and a framework for code-based game-playing proofs. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 409–426. Springer, Heidelberg (2006)
7. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Sponge functions. In: ECRYPT Hash Function Workshop (2007). http://sponge.noekeon.org/SpongeFunctions.pdf
8. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the security of the keyed sponge construction. In: Symmetric Key Encryption Workshop (2011)
9. CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness (2014). http://competitions.cr.yp.to/caesar.html
10. Chen, S., Lampe, R., Lee, J., Seurin, Y., Steinberger, J.: Minimizing the two-round even-mansour cipher. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 39–56. Springer, Heidelberg (2014)
11. Chen, S., Steinberger, J.: Tight security bounds for key-alternating ciphers. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 327–350. Springer, Heidelberg (2014)
12. Cogliati, B., Lampe, R., Patarin, J.: The indistinguishability of the XOR of k permutations. In: Cid, C., Rechberger, C. (eds.) FSE 2014. LNCS, vol. 8540, pp. 285–302. Springer, Heidelberg (2015)
13. Coron, J.-S., Dodis, Y., Malinaud, C., Puniya, P.: Merkle-Damgård revisited: how to construct a hash function. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 430–448. Springer, Heidelberg (2005)
14. Dai, Y., Lee, J., Mennink, B., Steinberger, J.P.: The security of multiple encryption in the ideal cipher model. In: Garay and Gennaro [31], pp. 20–38
15. Dodis, Y., Pietrzak, K., Puniya, P.: A new mode of operation for block ciphers and length-preserving MACs. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 198–219. Springer, Heidelberg (2008)
16. Garay, J.A., Gennaro, R. (eds.): CRYPTO 2014, Part I. LNCS, vol. 8616. Springer, Heidelberg (2014)
17. Gauravaram, P., Knudsen, L.R., Matusiewicz, K., Mendel, F., Rechberger, C., Schläffer, M., Thomsen, S.: Grøstl - a SHA-3 candidate (2009). Submission to NIST's SHA-3 competition
18. Hall, C., Wagner, D., Kelsey, J., Schneier, B.: Building PRFs from PRPs. In: Krawczyk, H. (ed.) CRYPTO 1998. LNCS, vol. 1462, pp. 370–389. Springer, Heidelberg (1998)
19. Impagliazzo, R., Rudich, S.: Limits on the provable consequences of one-way permutations. In: Goldwasser, S. (ed.) CRYPTO 1988. LNCS, vol. 403, pp. 8–26. Springer, Heidelberg (1990)
20. Kiltz, E., Pietrzak, K., Szegedy, M.: Digital signatures with minimal overhead from indifferentiable random invertible functions. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 571–588. Springer, Heidelberg (2013)
21. Luby, M., Rackoff, C.: How to construct pseudorandom permutations from pseudorandom functions. SIAM J. Comput. **17**, 373–386 (1988)
22. Lucks, S.: The sum of PRPs is a secure PRF. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 470–484. Springer, Heidelberg (2000)
23. Mandal, A., Nachef, V., Patarin, J.: Indifferentiability beyond the birthday bound for the XOR of two public random permutations. In: Gong, G., Gupta, K.C. (eds.) INDOCRYPT 2010. LNCS, vol. 6498, pp. 69–81. Springer, Heidelberg (2010)

24. Maurer, U.M., Renner, R.S., Holenstein, C.: Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 21–39. Springer, Heidelberg (2004)
25. Mennink, B., Preneel, B.: Hash functions based on three permutations: a generic security analysis. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 330–347. Springer, Heidelberg (2012)
26. Mouha, N., Mennink, B., Van Herrewege, A., Watanabe, D., Preneel, B., Verbauwhede, I.: Chaskey: an efficient MAC algorithm for 32-bit microcontrollers. In: Joux, A., Youssef, A. (eds.) SAC 2014. LNCS, vol. 8781, pp. 306–323. Springer, Heidelberg (2014)
27. Patarin, J.: A proof of security in $O(2^n)$ for the XOR of two random permutations. In: Safavi-Naini, R. (ed.) ICITS 2008. LNCS, vol. 5155, pp. 232–248. Springer, Heidelberg (2008)
28. Patarin, J.: The "Coefficients H" technique. In: Avanzi, R.M., Keliher, L., Sica, F. (eds.) SAC 2008. LNCS, vol. 5381, pp. 328–345. Springer, Heidelberg (2009)
29. Steinberger, J.P., Rogaway, P.: Constructing cryptographic hash functions from fixed-key blockciphers. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 433–450. Springer, Heidelberg (2008)
30. Steinberger, J.: The Sum-Capture Problem for Abelian Groups (2014). arxiv.org/abs/1309.5582
31. Wu, H.: The Hash Function JH (2009). Submission to NIST's SHA-3 Competition