Security of Keyed Sponge Constructions Using a Modular Proof Approach

Elena Andreeva[1(✉)], Joan Daemen[2], Bart Mennink[1], and Gilles Van Assche[2]

[1] Department of Electrical Engineering, ESAT/COSIC, KU Leuven, iMinds, Leuven, Belgium
{elena.andreeva,bart.mennink}@esat.kuleuven.be
[2] STMicroelectronics, Diegem, Belgium
{joan.daemen,gilles.vanassche}@st.com

Abstract. Sponge functions were originally proposed for hashing, but find increasingly more applications in keyed constructions, such as encryption and authentication. Depending on how the key is used we see two main types of keyed sponges in practice: *inner*- and *outer*-keyed. Earlier security bounds, mostly due to the well-known sponge indifferentiability result, guarantee a security level of $c/2$ bits with c the capacity. We reconsider these two keyed sponge versions and derive improved bounds in the classical indistinguishability setting as well as in an extended setting where the adversary targets multiple instances at the same time. For cryptographically significant parameter values, the expected workload for an attacker to be successful in an n-target attack against the outer-keyed sponge is the minimum over $2^k/n$ and $2^c/\mu$ with k the key length and μ the total maximum multiplicity. For the inner-keyed sponge this simplifies to $2^k/\mu$ with maximum security if $k = c$. The multiplicity is a characteristic of the data available to the attacker. It is at most twice the data complexity, but will be much smaller in practically relevant attack scenarios. We take a modular proof approach, and our indistinguishability bounds are the sum of a bound in the PRP model and a bound on the PRP-security of Even-Mansour type block ciphers in the ideal permutation model, where we obtain the latter result by using Patarin's H-coefficient technique.

Keywords: Sponge construction · Keyed sponge · (Authenticated) encryption · Indistinguishability

1 Introduction

Sponge functions are versatile cryptographic primitives that can be used for hashing, but also in a wide range of keyed applications, such as message authentication codes (MAC), stream encryption, authenticated encryption, and pseudo-random sequence generation [5,7,8]. This fact is illustrated by the large number of sponge based candidates in the CAESAR competition for authenticated encryption schemes [12]: Artemia [1], Ascon [15], ICEPOLE [23], Ketje [10],

G. Leander (Ed.): FSE 2015, LNCS 9054, pp. 364–384, 2015.
DOI: 10.1007/978-3-662-48116-5_18

Keyak [11], NORX [3], π-Cipher [19], PRIMATEs [2], Prøst [21] and STRIBOB [28]. More recently, Rivest and Schuldt [27] presented an update of the RC4 stream cipher with the name Spritz, also adopting a keyed sponge construction.

The sponge function consists of the application of the sponge construction to a fixed-length permutation (or transformation) f. It is a function that maps an input string of variable length to an output of arbitrary length. The *duplex construction* also makes use of a fixed-length permutation but results in a *stateful object* that can be fed with short input strings and from which short output strings can be extracted [8]. The above mentioned authenticated encryption schemes are for example based on the duplex construction. In [8] Bertoni et al. prove the security of the duplex construction equivalent to the security of the sponge construction, which means that any security result on the sponge construction is automatically valid for the duplex construction.

We can identify two types of keyed sponge functions, both of which we see applied in practice [1–3,10,11,15,19,21,23,25,28,29]. The first type applies the key by taking it as the first part of the sponge input and we call it the *outer-keyed sponge*. The second *inner-keyed sponge* applies the key on the inner part of the initial state, and can be viewed as successive applications of the Even-Mansour [16,17] type block cipher, which in turn calls an unkeyed permutation.

One way to argue security of the keyed sponge constructions is via the indifferentiability result of [6]. This result guarantees that the keyed sponge constructions can replace random oracles in any single-stage cryptographic system [22,26] as long as the total complexity of the adversary is less than $2^{(c+1)/2}$. Bertoni et al. [9] derived an improved bound on the distinguishing advantage against the outer-keyed sponge by separating the total complexity into time and data complexity. However, their proof contains a subtle error: [9, Lemma 1] proves that the keyed sponge output is uniformly and independently distributed if certain conditions are fulfilled, whereas the proof requires uniformity of the *joint* keyed sponge output and queries to f, which does exhibit a bias. Regarding the inner-keyed sponge, Chang et al. considered security of the construction in the so-called standard model in [13]. Central in their reasoning is the clever trick to describe the keyed sponge as the sponge construction calling an Even-Mansour block cipher. Their bound does however not go beyond the generic sponge indifferentiability bound of [6] as their main intention appears to have been to prove security in the standard model rather than the ideal permutation model.

1.1 Our Contribution

We prove bounds on the generic security of both types of keyed sponge constructions in the single-target and multi-target scenarios. In the single-target scenario, we bound the success probability of distinguishing a single instance of the construction from a random oracle for a given attack complexity and providing the adversary with additional access to the underlying permutation f. In the multi-target scenario, the adversary targets multiple instances of the keyed sponge at the same time. In practice, many systems support multiple users using the same algorithm and the adversary may be willing to leverage his resources

to break at least one of the users' account. It can be regarded as important to the system provider who wants to avoid losing credibility in such a case. For the multi-target analysis, we introduce a generalized version of indistinguishability security.

Our proofs are performed in two steps. Firstly, considering the keyed sponge constructions to be implicitly based on an underlying block cipher, we derive a bound for the distinguishing advantage of the constructions in the PRP model. Secondly, we deal with the PRP security of the Even-Mansour construction in the ideal permutation model using Patarin's H-coefficient technique [14,24]. This modular proof approach results in compact proofs that are easy to verify.

When estimating the required capacity c to achieve a required security level, the important term in all of the bounds is of the form $\frac{M^2+\mu N}{2^c}$. Here, M is the data complexity, N the time complexity, and μ is the so-called *total maximum multiplicity*. The multiplicity is determined by the keyed sponge outputs available to the adversary and is a function of M. It first appeared in Bertoni et al. [7], and allows us to achieve bounds that significantly improve over the earlier single-target bounds of [9,13]. The multiplicity makes the bound widely applicable, as it allows to take into account the restrictions an adversary faces in a concrete use case. In more detail, in the worst case the multiplicity equals twice the data complexity but in many attack scenarios it is orders of magnitude smaller, leading to a tighter bound. For cryptographically significant parameter values, the dominant term in the bound is the time complexity by divided by $2^c/\mu$. In other words, our bounds imply security beyond the birthday bound on the capacity for all existing keyed sponge based modes.

We remark that a recent work of Jovanovic et al. [20] proved bounds on the distinguishing advantage for keyed sponge based authenticated encryption. Their results are specific for authentication encryption modes applying a keyed sponge construction and explicitly require nonce uniqueness. Moreover, unlike the bounds in this paper, their bound contains a term involving the permutation width making it tight only for large rates. Additionally, our results yield a tight bound whatever the rate, exploiting the multiplicity, which is typically small in the case of unique nonce scenarios (see also Sect. 6). Finally, a concurrent work by Gaži et al. [18] proves tight bounds for the specific case of MACs produced by a keyed sponge, but without generalizing to other applications that require longer output lengths.

1.2 Version History

Gaži, Pietrzak, and Tessaro pointed out that the pre-proceedings version contains an oversight in the analysis of the outer-keyed sponge. Informally, the probability that a distinguisher guesses the key was bounded incorrectly. We have fixed the issue, using a result from Gaži et al. [18]. We refer to the proof of Theorem 6 and the subsequent discussion for more details.

1.3 Outline

The remainder of this paper is organized as follows. In Sect. 2, we provide the definitions of the constructions we use. This is followed by an introduction to the security model of indistinguishability in Sect. 3. In Sect. 4, we prove our bounds for the inner-keyed sponge and in Sect. 5 those for the outer-keyed sponge. Finally, we discuss the implications of our bounds in Sect. 6.

2 Definitions of Constructions

In this section we specify the constructions we address in this paper.

2.1 The Sponge Construction

The sponge construction operates on a state s of b bits, and calls a b-bit permutation f. It takes as input a message $m \in \{0,1\}^*$ and natural number ℓ and outputs a potentially infinite string truncated to the chosen length $\ell \in \mathbb{N}$, denoted $z \in \{0,1\}^\ell$.

We express an evaluation of the sponge function as

$$\textsc{Sponge}^f(m, \ell) = z. \tag{1}$$

The sponge function operates as follows. First we apply an injective padding function to the input message m, denoted by $m||\text{pad}[r](|m|)$, with r the *rate*. This padding rule is required to be injective and the last block is required to be different from 0. Then, we initialize the b bits of the sponge state s to zero. We refer to the first r bits of the state s as the *outer part* $\bar{s}$ and to the last $c = b - r$ bits (c is called the *capacity*) as its *inner part* $\hat{s}$ (think back on $\hat{}$ as a roof). The padded message is processed by an *absorbing phase* followed by a *squeezing phase*:

Absorbing phase: the r-bit input message blocks are sequentially XORed into the outer part of the state, interleaved with applications of the function f;

Squeezing phase: the bits of the outer part of the state are returned as output blocks, interleaved with applications of the function f, until enough bits are produced.

An illustration is given in Fig. 1 and a formal description is provided in Algorithm 1. The notation $\lfloor x \rfloor_n$ means that the string x is truncated after its first n bits.

2.2 The Even-Mansour Construction

The (single-key) Even-Mansour construction builds a b-bit block cipher from a b-bit permutation and takes a b-bit key [16,17]. It is defined as $f(x \oplus K) \oplus K$. We consider a variant with the first r bits of the key are zero, reducing its effective length to c bits:

$$E_K^f(x) = f(x \oplus (0^r||K)) \oplus (0^r||K). \tag{2}$$

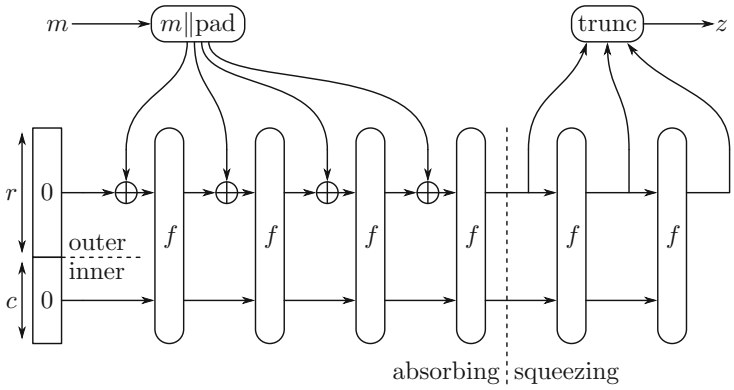

Fig. 1. The sponge construction

Algorithm 1. The sponge construction SPONGE^f

Input: $m \in \{0,1\}^*$, $\ell \in \mathbb{N}$
Output: $z \in \{0,1\}^\ell$
$P = m||\text{pad}[r](|m|)$
Let $P = m_0||m_1||\ldots||m_w$ with $|m_i| = r$
$s = 0^b$
for $i = 0$ to w **do**
  $s = s \oplus (m_i||0^c)$
  $s = f(s)$
$z = \lfloor s \rfloor_r$
while $|z| < \ell$ **do**
  $s = f(s)$
  $z = z||\lfloor s \rfloor_r$
return $\lfloor z \rfloor_\ell$

2.3 The Root-Keyed Sponge

As a way to highlight the similarities between the inner- and outer-keyed sponges, which we will define in the next sections, we define a common construction called the *root-keyed sponge*. Basically, it is a variant of the sponge construction where the state is initialized to a key $K \in \{0,1\}^b$ instead of 0^b. The root-keyed sponge RKS^f_K is defined in Algorithm 2.

The root-keyed sponge can be rewritten using the Even-Mansour block cipher. Indeed, we have

$$\text{RKS}^f_K(m,\ell) = \text{RKS}^{E^f_{\hat{K}}}_{\bar{K}||0^c}(m,\ell), \tag{3}$$

as key additions between subsequent applications of $E^f_{\hat{K}}$ cancel out.

Algorithm 2. The root-keyed sponge construction RKS^f_K

Require: $|K| = b$
Input: $m \in \{0,1\}^*$, $\ell \in \mathbb{N}$
Output: $z \in \{0,1\}^\ell$
$P = m||\mathrm{pad}[r](|m|)$
Let $P = m_0||m_1||\ldots||m_w$ with $|m_i| = r$
$s = K$
for $i = 0$ to w **do**
  $s = s \oplus (m_i||0^c)$
  $s = f(s)$
$z = \lfloor s \rfloor_r$
while $|z| < \ell$ **do**
  $s = f(s)$
  $z = z||\lfloor s \rfloor_r$
return $\lfloor z \rfloor_\ell$

2.4 The Inner-Keyed Sponge

The inner-keyed sponge IKS^f is a pseudorandom function (PRF) that was first introduced by Chang et al. [13] (their case EMKSC3). An inner-keyed sponge instance is defined by a permutation f and a key $K \in \{0,1\}^c$ and simply consists of the sponge construction with E^f_K as permutation:

$$\mathrm{IKS}^f_K(m,\ell) = \textsc{Sponge}^{E^f_K}(m,\ell). \tag{4}$$

Owing to (3), an equivalent definition of $\mathrm{IKS}^f_K(m,\ell)$ is given by

$$\mathrm{IKS}^f_K(m,\ell) = \mathrm{RKS}^{E^f_K}_{0^b}(m,\ell) = \mathrm{RKS}^f_{0^r||K}(m,\ell). \tag{5}$$

2.5 The Outer-Keyed Sponge

The outer-keyed sponge OKS^f is a PRF construction that was originally introduced by Bertoni et al. [9] as the *keyed sponge*. An outer-keyed sponge instance is defined by a permutation f and a key $K \in \{0,1\}^k$ and simply consists of an evaluation of the sponge construction where the secret key and the message are concatenated:

$$\mathrm{OKS}^f_K(m,\ell) = \textsc{Sponge}^f(K||m,\ell). \tag{6}$$

While K may be of any size, we limit our analysis to the case where k is a multiple of the rate r, or $\{0,1\}^k = (\{0,1\}^r)^+$. The outer-keyed sponge can be equivalently described as a function that derives the *root key* $L \in \{0,1\}^b$ from the cipher key $K \in \{0,1\}^k$, followed by the root-keyed sponge with root key L. The root key derivation function $\textsc{kd}^f(K)$ is defined in Algorithm 3. We obtain:

$$\mathrm{OKS}^f_K(m,\ell) = \mathrm{RKS}^f_{\textsc{kd}^f(K)}(m,\ell) \overset{(3)}{=} \mathrm{RKS}^{E^f_L}_{L||0^c}(m,\ell), \tag{7}$$

Algorithm 3. The root key derivation function $\text{KD}^f(K)$

Input: $K \in (\{0,1\}^r)^+$
Output: $s \in \{0,1\}^b$
Let $K = K_0||K_1||\ldots||K_w$ with $|K_i| = r$
$s = 0^b$
for $i = 0$ to w **do**
    $s = s \oplus (K_i||0^c)$
    $s = f(s)$
return s

with $L = \text{KD}^f(K)$. This alternative description highlights a similarity with the inner-keyed sponge: the only effective difference lies in the presence of the root key derivation function.

3 Security Model

The security analyses in this work are done in the *indistinguishability* framework where one bounds the advantage of an adversary $\mathcal{A}$ in distinguishing a real system from an ideal system. The real system contains one or more specified constructions, while the ideal one consists of ideal functions with the same interface. We explain the high-level idea for the case where $\mathcal{A}$ attacks one instance of a keyed sponge construction.

Suppose $f : \{0,1\}^b \to \{0,1\}^b$ is a permutation and consider a keyed sponge construction $\mathcal{H}_K^f$ based on f and some key $K \in \{0,1\}^k$. Let $\mathcal{RO}$ be a random oracle [4] with the same interface as $\mathcal{H}_K^f$. Adversary $\mathcal{A}$ is given query access to either $\mathcal{H}_K^f$ or $\mathcal{RO}$ and tries to tell both apart. It is also given access to the underlying permutation f, which is modeled by query access. The random oracle is required to output infinitely long strings truncated to a certain length. The function can be defined as $\mathcal{RO} : \{0,1\}^* \times \mathbb{N} \to \{0,1\}^{\mathbb{N}}$ that on input (m, ℓ) outputs $\mathcal{RO}(m, \ell) = \lfloor \mathcal{RO}^\infty(m) \rfloor_\ell$, where $\mathcal{RO}^\infty : \{0,1\}^* \to \{0,1\}^\infty$ takes inputs of arbitrary but finite length and returns random infinite strings where each output bit is selected uniformly and independently, for every m.

We similarly consider the PRP security of the Even-Mansour construction E_K^f, where $\mathcal{A}$ is given query access to either this construction or a random permutation $\pi \xleftarrow{\$} \mathsf{Perm}(b)$ with domain and range $\{0,1\}^b$, along with query access to f. We also consider a slightly more advanced notion of kdPRP security, where the root key derivation function KD is applied to K first.

The security proofs of IKS and OKS consist of two steps: the first one reduces the security of the construction to the PRP security (for IKS) or kdPRP security (for OKS) of the Even-Mansour construction. This step does not depend on f and is in fact a standard-model reduction. Next, we investigate the PRP/kdPRP security of Even-Mansour under the assumption that f is a random permutation.

3.1 Counting

We express our bound in terms of the query complexities that model the effort by the adversary. Here we distinguish between keyed sponge *construction* or random oracle queries and *primitive* queries to $f^{\pm}$:

Data or online complexity *M:* the amount of access to the construction $\mathcal{H}_K^f$ or $\mathcal{RO}$, that in many practical use cases is limited;

Time or offline complexity *N:* computations requiring no access to the construction, in practical use cases only limited by the computing power and time available to the adversary.

Both M and N are expressed in terms of the number of primitive calls. We include in M only *fresh calls*: a call from $\mathcal{H}_K^f$ to f is not fresh if it has already been made due to a prior query to the construction. In the ideal world a random oracle naturally does not make calls to f, but the data complexity is counted as if it would and as such, it is fully determined by the queries. For N, we assume without loss of generality that the adversary makes no repeated queries.

In our proofs, we use an additional characteristic of the queries called the *total maximum multiplicity* and denote it by μ. Let $\{(s_i, t_i)\}_{i=1}^{M}$ be the set of M input/output pairs for f made in construction evaluations.

Definition 1 (Multiplicity). *The maximum forward and backward multiplicities are given by*

$$\mu_{\text{fw}} = \max_a \#\{i \in \{1, \ldots, M\} \mid \bar{s}_i = a\} \text{ and}$$
$$\mu_{\text{bw}} = \max_a \#\{i \in \{1, \ldots, M\} \mid \bar{t}_i = a\}.$$

The total maximum multiplicity is given by $\mu = \mu_{\text{fw}} + \mu_{\text{bw}}$.

Note that the total maximum multiplicity μ is the sum of the maximum forward and backward multiplicities while [7] uses the maximum over the forward and backward multiplicities.

3.2 Distinguishing Advantage for Keyed Sponges

We are now ready to give the indistinguishability definition for keyed sponges (the PRP and kdPRP security definitions will be discussed in Sects. 4 and 5). Our definition is broad in the sense that it considers also security against a multi-target attack, where an attacker has access to an array of $n \geq 1$ instances of the keyed sponge or random oracles. We refer to this notion as *joint indistinguishability*. Naturally, joint indistinguishability reduces to plain or regular *indistinguishability* for $n = 1$. The model is illustrated in Fig. 2.

Definition 2 (Joint Distinguishing Advantage). *Let* $\mathcal{H}$ *be a PRF function based on a permutation* $f \in \mathsf{Perm}(b)$. *Let* $K_1, \ldots, K_n \xleftarrow{\$} \{0,1\}^k$ *be* $n \geq 1$ *keys*

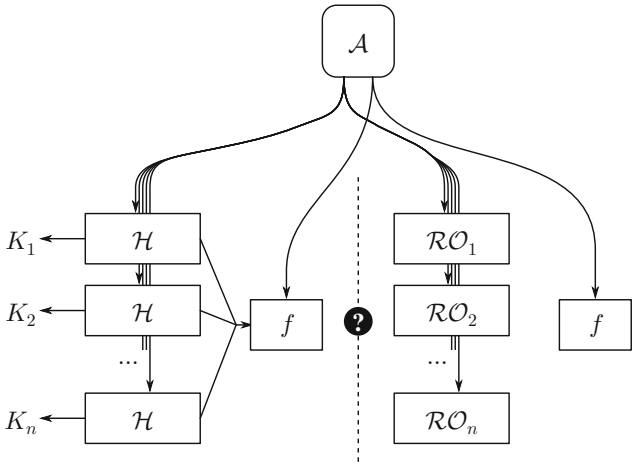

Fig. 2. The keyed sponge distinguishing setup

and $\mathcal{RO}_1, \ldots, \mathcal{RO}_n$ *be* n *independent random oracles with the same interface as* $\mathcal{H}_{K_1}, \ldots, \mathcal{H}_{K_n}$*. The joint distinguishing advantage of* $\mathcal{A}$ *is defined as*

$$\mathbf{Adv}^{\mathrm{ind[n]}}_{\mathcal{H}}(\mathcal{A}) = \left|\Pr\left(\mathcal{A}^{\mathcal{H}^f_{K_1},\ldots,\mathcal{H}^f_{K_n},f} \Rightarrow 1\right) - \Pr\left(\mathcal{A}^{\mathcal{RO}_1,\ldots,\mathcal{RO}_n,f} \Rightarrow 1\right)\right|.$$

We use $\mathbf{Adv}^{\mathrm{ind[n]}}_{\mathcal{H}}(M_1, \ldots, M_n, \mu, N)$ *to denote the maximum advantage over any adversary with data complexity* M_h *to the* h*-th construction oracle (*$\mathcal{H}_{K_h}$ *or* $\mathcal{RO}_h$*), time complexity* N*, and total maximum multiplicity* μ*.*

Note that, as we consider $n \geq 1$ instances of the construction, we have similarly split the online complexity M into $M_1 + \cdots + M_n$. In other words, M gives the online complexity over all n instances.

3.3 Patarin's H-Coefficient Technique

Our proofs partly rely on Patarin's H-coefficient technique [24]. We briefly summarize this technique, and refer to Chen and Steinberger [14] for further discussion.

Consider an information-theoretic adversary $\mathcal{A}$ whose goal is to distinguish two systems X and Y, denoted as

$$\mathbf{Adv}(\mathcal{A}) = \Delta(X;Y),$$

where $\Delta(X;Y)$ denotes the statistical distance between X and Y. Without loss of generality, we can assume $\mathcal{A}$ is a deterministic adversary and will always do so in the following. Indeed, if $\mathcal{A}$ were a randomized adversary, there exists a deterministic adversary $\mathcal{A}'$ with at least the same advantage (namely the one defined by $\mathcal{A}$ and the fixed random tape). We refer to [14] for details. Its interaction with the system X or Y is summarized in a transcript τ. For $Z \in \{X, Y\}$, denote

by D_Z the probability distribution of transcripts when interacting with Z. Say that a transcript τ is attainable if it can be obtained from interacting with Y, hence if $\Pr(D_Y = \tau) > 0$, and let $\mathcal{T}$ be the set of all attainable transcripts. The H-coefficient technique states the following [14].

Lemma 1 (H-coefficient Technique). *Consider a fixed deterministic adversary $\mathcal{A}$. Let $\mathcal{T} = \mathcal{T}_{\text{good}} \cup \mathcal{T}_{\text{bad}}$ be a partition of the set of attainable transcripts into "good" and "bad" transcripts. Let ε be such that for all $\tau \in \mathcal{T}_{\text{good}}$:*

$$\frac{\Pr(D_X = \tau)}{\Pr(D_Y = \tau)} \geq 1 - \varepsilon.$$

Then, $\mathbf{Adv}(\mathcal{A}) \leq \varepsilon + \Pr(D_Y \in \mathcal{T}_{\text{bad}})$.

Proofs using Patarin's technique consist of first carefully defining a set of "bad" transcripts $\mathcal{T}_{\text{bad}}$, and then showing that both ε and $\Pr(D_Y \in \mathcal{T}_{\text{bad}})$ are small for this set of bad transcripts.

4 Distinguishing Advantage of the Inner-Keyed Sponge

We bound the distinguishing advantage of the inner-keyed sponge construction in the ideal permutation model. A bound for the case of $n = 1$ is given in Sect. 4.1, and it is generalized to arbitrary n in Sect. 4.2. Both proofs consist of two steps that are both of independent interest. Note that we assume equal key size and capacity in our proofs. If $k < c$, the denominator 2^c in the bounds of Theorems 2 and 4 must be replaced by 2^k.

Before proceeding, we define the notion of PRP security that we will use in the security proof of the inner-keyed sponge to replace $E^f_{K_1}, \ldots, E^f_{K_n}$ with random permutations $\pi_1, \ldots, \pi_n$, in analogy with (4). As multiple instances of E for n different keys are considered, we call this notion *joint PRP security*.

Definition 3 (Joint PRP Advantage). *We define the joint PRP advantage of $\mathcal{A}$ for a given block cipher $E : \{0,1\}^c \times \{0,1\}^b \to \{0,1\}^b$ based on a permutation $f \in \mathsf{Perm}(b)$ as*

$$\mathbf{Adv}^{\text{prp}[n]}_E(\mathcal{A}) = \left| \Pr\left(\mathcal{A}^{E^f_{K_1}, \ldots, E^f_{K_n}, f} \Rightarrow 1\right) - \Pr\left(\mathcal{A}^{\pi_1, \ldots, \pi_n, f} \Rightarrow 1\right) \right|.$$

The adversary can make both forward and inverse primitive queries f and f^{-1}, but is restricted to forward construction queries. We use $\mathbf{Adv}^{\text{prp}[n]}_E(M_1, \ldots, M_n, \mu, N)$ to denote the maximum advantage over any adversary with data complexity M_h to the h-th construction oracle ($E^f_{K_h}$ or π_h), time complexity N, and total maximum multiplicity μ.

4.1 Single Target

Theorem 1. *For $\text{IKS}^f_K : \{0,1\}^* \times \mathbb{N} \to \{0,1\}^{\mathbb{N}}$ with $K \xleftarrow{\$} \{0,1\}^c$ and permutation $f \in \mathsf{Perm}(b)$:*

$$\mathbf{Adv}^{\text{ind}[1]}_{\text{IKS}}(M, \mu, N) \leq \frac{M^2}{2^c} + \mathbf{Adv}^{\text{prp}[1]}_E(M, \mu, N).$$

Proof. Using the triangle inequality, we find that for any adversary $\mathcal{A}$:

$$\begin{aligned}\mathbf{Adv}^{\mathrm{ind}[1]}_{\mathrm{IKS}}(\mathcal{A}) &\stackrel{\mathrm{def}}{=} \Delta_{\mathcal{A}}(\mathrm{IKS}^f_K, f; \mathcal{RO}, f)\\ &\stackrel{(4)}{=} \Delta_{\mathcal{A}}(\textsc{Sponge}^{E^f_K}, f; \mathcal{RO}, f)\\ &\leq \Delta_{\mathcal{A}}(\textsc{Sponge}^{\pi}, f; \mathcal{RO}, f) + \Delta_{\mathcal{B}}(E^f_K, f; \pi, f)\\ &= \Delta_{\mathcal{C}}(\textsc{Sponge}^{\pi}, \mathcal{RO}) + \Delta_{\mathcal{B}}(E^f_K, f; \pi, f).\end{aligned}$$

Here, $\mathcal{B}$ and $\mathcal{C}$ are adversaries whose joint cost is not above that of $\mathcal{A}$. Concretely, for our cost functions M, N and total maximum multiplicity μ, this means that $\mathcal{B}$ cannot make more than M construction queries with total maximum multiplicity at most μ and at most N primitive queries. Distinguisher $\mathcal{C}$ can make at most M construction queries, and its advantage is covered by the indifferentiability bound proven in [6]. □

We now bound the PRP security of the Even-Mansour construction in the ideal permutation model. The proof is a generalization of the security analysis of the Even-Mansour block cipher [16,17].

Theorem 2. *For E^f_K with $K \xleftarrow{\$} \{0,1\}^c$ and ideal permutation $f \xleftarrow{\$} \mathsf{Perm}(b)$ we have*

$$\mathbf{Adv}^{\mathrm{prp}[1]}_{E}(M, \mu, N) \leq \frac{\mu N}{2^c}.$$

Proof. The proof uses Lemma 1. We consider an adversary $\mathcal{A}$ that has access to $X = (E^f_K, f)$ in the real world or $Y = (\pi, f)$ in the ideal world. It can only make forward queries to its oracle $\mathcal{O}_1$, although it can make both forward and backward queries to f. It makes M construction queries with total maximum multiplicity at most μ and at most N primitive queries. The interaction with $\mathcal{O}_1$ is denoted $\tau_1 = \{(s_i, t_i)\}_{i=1}^{M}$ and the interaction with f is denoted $\tau_f = \{(x_j, y_j)\}_{j=1}^{N}$. To ease the analysis, we will disclose K at the end of the experiment (in the ideal world, K will simply be a dummy key). The transcripts are thus of the form $\tau = (K, \tau_1, \tau_f)$. We recall that the total maximum multiplicity is μ, which means that

$$\begin{aligned}\max_a \#\{(s_i, t_i) \in \tau_1 \mid \bar{s}_i = a\} &\leq \mu_{\mathrm{fw}} \text{ and}\\ \max_a \#\{(s_i, t_i) \in \tau_1 \mid \bar{t}_i = a\} &\leq \mu_{\mathrm{bw}},\end{aligned}$$

for some $\mu_{\mathrm{fw}}, \mu_{\mathrm{bw}}$ with $\mu_{\mathrm{fw}} + \mu_{\mathrm{bw}} \leq \mu$.

Definition of good and bad transcripts. We define a transcript τ as *bad* if

$$\exists (s,t) \in \tau_1, (x,y) \in \tau_f \text{ such that } s \oplus x = 0^r || K \vee t \oplus y = 0^r || K. \tag{8}$$

In the real world a bad transcript implies two calls to f with the same input: one directly from querying the primitive oracle and another one indirectly from querying the construction oracle. In a *good* transcript in the real world, all tuples

in (τ_1, τ_f) uniquely define an input-output pair of f. Note also that in the real world the two conditions in (8) are equivalent while in the ideal world they are not.

Bounding the probability of bad transcripts in the ideal world. In the ideal world, (τ_1, τ_f) is a transcript generated independently of the dummy key $K \xleftarrow{\$} \{0,1\}^c$. First consider the first condition of (8). Fix any tuple (x, y) (N choices). By construction, τ_1 contains at most μ_{fw} tuples (s, t) such that $\bar{s} = \bar{x}$. This gives a total of $\mu_{\text{fw}} N$ values $\hat{s} \oplus \hat{x}$, and any could be hit by the randomly generated K. A similar reasoning holds for the second part of (8), resulting in $\mu_{\text{bw}} N$ values. Concluding, $\Pr(D_Y \in \mathcal{T}_{\text{bad}}) \leq \frac{\mu N}{2^c}$, where we use that $\mu = \mu_{\text{fw}} + \mu_{\text{bw}}$.

Bounding the ratio $\Pr(D_X = \tau) / \Pr(D_Y = \tau)$ for good transcripts. Consider a good transcript $\tau \in \mathcal{T}_{\text{good}}$. Denote by Ω_X the set of all possible oracles in the real world and by $\mathsf{comp}_X(\tau) \subseteq \Omega_X$ the set of oracles in Ω_X compatible with transcript τ. Note that $|\Omega_X| = 2^c \cdot 2^b!$. Define Ω_Y and $\mathsf{comp}_Y(\tau)$ similarly, where $|\Omega_Y| = 2^c \cdot (2^b!)^2$. The probabilities appearing in Lemma 1 can be computed as follows:

$$\Pr(D_X = \tau) = \frac{|\mathsf{comp}_X(\tau)|}{|\Omega_X|} \quad \text{and} \quad \Pr(D_Y = \tau) = \frac{|\mathsf{comp}_Y(\tau)|}{|\Omega_Y|}.$$

Starting with $|\mathsf{comp}_X(\tau)|$, the condition $\tau \in \mathcal{T}_{\text{good}}$ imposes uniqueness of the query tuples in τ, or in other words that any tuple defines exactly one input-output pair of f. As $\tau \cup \tau_f$ consists of $M + N$ tuples, the number of possible functions f compliant with τ is $|\mathsf{comp}_X(\tau)| = (2^b - M - N)!$. For the ideal world, the number of compliant functions π equals $(2^b - M)!$ and the number of compliant oracles f equals $(2^b - N)!$. Therefore,

$$|\mathsf{comp}_Y(\tau)| = (2^b - M)!(2^b - N)! \leq (2^b - M - N)!2^b!.$$

We consequently obtain

$$\begin{aligned} \Pr(D_X = \tau) &= \frac{(2^b - M - N)!}{2^c \cdot 2^b!} = \frac{(2^b - M - N)!2^b!}{2^c \cdot (2^b!)^2} \\ &\geq \frac{|\mathsf{comp}_Y(\tau)|}{|\Omega_Y|} = \Pr(D_Y = \tau), \end{aligned}$$

and thus $\Pr(D_X = \tau) / \Pr(D_Y = \tau) \geq 1$. □

In the ideal permutation model, the expressions in Theorems 1 and 2 simplify into

$$\mathbf{Adv}^{\text{ind}[1]}_{\text{IKS}}(M, \mu, N) \leq \frac{M^2 + \mu N}{2^c}.$$

4.2 Multiple Targets

Theorem 3. *For* $\mathrm{IKS}_K^f : \{0,1\}^* \times \mathbb{N} \to \{0,1\}^{\mathbb{N}}$ *with* $K_1, \ldots, K_n \xleftarrow{\$} \{0,1\}^c$ *and permutation* $f \in \mathsf{Perm}(b)$*:*

$$\mathbf{Adv}^{\mathrm{ind[n]}}_{\mathrm{IKS}}(M_1, \ldots, M_n, \mu, N) \leq \frac{\sum_h M_h^2}{2^c} + \mathbf{Adv}^{\mathrm{prp[n]}}_{E}(M_1, \ldots, M_n, \mu, N).$$

Proof. A similar reasoning as for Theorems 1, but now using the notion of joint PRP security to replace $E^f_{K_1}, \ldots, E^f_{K_n}$ with n independent random permutations $\pi_1, \ldots, \pi_n$, results in

$$\begin{aligned}\mathbf{Adv}^{\mathrm{ind[n]}}_{\mathrm{IKS}}(\mathcal{A}) \leq\ & \Delta_{\mathcal{C}}(\textsc{Sponge}^{\pi_1}, \ldots, \textsc{Sponge}^{\pi_n}; \mathcal{RO}_1, \ldots, \mathcal{RO}_n) \\ & + \Delta_{\mathcal{B}}(E^f_{K_1}, \ldots, E^f_{K_n}, f; \pi_1, \ldots, \pi_n, f).\end{aligned}$$

Here, $\mathcal{B}$ and $\mathcal{C}$ are adversaries whose joint cost is not above that of $\mathcal{A}$, and particularly both make at most M_h h-th construction queries for $h = 1, \ldots, n$. The advantage of $\mathcal{C}$ is in fact the distinguishing bound of n sponges with independent permutations. □

We now bound the joint PRP security of the Even-Mansour construction in the ideal permutation model.

Theorem 4. *For* E^f_K *with* $K_1, \ldots, K_n \xleftarrow{\$} \{0,1\}^c$ *and ideal permutation* $f \xleftarrow{\$} \mathsf{Perm}(b)$ *we have*

$$\mathbf{Adv}^{\mathrm{prp[n]}}_{E}(M_1, \ldots, M_n, \mu, N) \leq \frac{\mu N}{2^c} + \frac{2\sum_{h \neq h'} M_h M_{h'}}{2^c}.$$

Proof. The proof follows the one of Theorem 2, with the difference that multiple keys are involved. Adversary $\mathcal{A}$ has access to $X = (E^f_{K_1}, \ldots, E^f_{K_n}, f)$ in the real world or $Y = (\pi_1, \ldots, \pi_n, f)$ in the ideal world. The n construction oracles are also denoted $(\mathcal{O}_1, \ldots, \mathcal{O}_n)$. It makes M_h construction queries to $\mathcal{O}_h$ with total maximum multiplicity at most μ (over all $M = M_1 + \cdots + M_n$ construction queries) and at most N primitive queries. The interaction with $\mathcal{O}_h$ (for $h = 1, \ldots, n$) is denoted $\tau_h = \{(s_i, t_i)\}_{i=1}^{M_h}$ and the interaction with f is denoted $\tau_f = \{(x_j, y_j)\}_{j=1}^{N}$. As before, we will disclose the keys $K_1, \ldots K_n$ at the end of the experiment. The transcripts are thus of the form $\tau = (K_1, \ldots, K_n, \tau_1, \ldots, \tau_n, \tau_f)$.

Definition of good and bad transcripts. We extend the definition of *bad* transcripts from Theorem 2 to multiple keys. Formally, we define a transcript τ as *bad* if one of the following is satisfied:

$$\exists h, (s,t) \in \tau_h, (x,y) \in \tau_f \text{ such that } s \oplus x = 0^r || K_h \vee t \oplus y = 0^r || K_h, \tag{9}$$

$$\exists h \neq h', (s,t) \in \tau_h, (s',t') \in \tau_{h'} \text{ such that } s \oplus s' = 0^r || (K_h \oplus K_{h'}) \vee t \oplus t' = 0^r || (K_h \oplus K_{h'}). \tag{10}$$

The second condition corresponds to colliding calls to f coming from two construction queries with different keys. In the real world, all tuples in a *good* transcript $(\tau_1, \ldots, \tau_n, \tau_f)$ consistently define an input-output pair of f. Note also that in the real world the two conditions in (9) are equivalent, and similarly for the two conditions in (10).

Bounding the probability of bad transcripts in the ideal world. In the ideal world, $(\tau_1, \ldots, \tau_n, \tau_f)$ is a transcript generated independently of the dummy keys $K_1, \ldots, K_n \xleftarrow{\$} \{0,1\}^c$. The proof of Theorem 2 straightforwardly generalizes to show that (9) is set with probability at most $\frac{\mu N}{2^c}$. Here, we use that for any tuple $(x, y) \in \tau_f$, the set $(\tau_1, \ldots, \tau_n)$ of M queries *in total* contains at most μ_{fw} tuples (s, t) such that $\bar{s} = \bar{x}$. A similar exercise is done for (10): for $h \neq h'$, there are at most $2M_hM_{h'}$ values $s \oplus s'$ and $t \oplus t'$ with $(s, t) \in \tau_h$ and $(s', t') \in \tau_{h'}$, and the value $K_h \oplus K_{h'}$ has probability $1/2^c$. Concluding, $\Pr(D_Y \in \mathcal{T}_{\text{bad}}) \leq \frac{\mu N}{2^c} + \frac{2\sum_{h \neq h'} M_h M_{h'}}{2^c}$.

Bounding the ratio $\Pr(D_X = \tau) / \Pr(D_Y = \tau)$ for good transcripts. As in the proof of Theorem 2, we have $|\Omega_X| = (2^c)^n \cdot 2^b!$ and $|\Omega_Y| = (2^c)^n \cdot (2^b!)^{n+1}$. Also, $|\mathsf{comp}_X(\tau)| = (2^b - M - N)!$ by construction. For the ideal world, the number of compliant functions $\pi_1, \ldots, \pi_n$ equals $\prod_h (2^b - M_h)!$ and the number of compliant oracles f equals $(2^b - N)!$. Therefore,

$$|\mathsf{comp}_Y(\tau)| = \left(\prod_h (2^b - M_h)!\right)(2^b - N)! \leq (2^b - M - N)!(2^b!)^n,$$

and the remainder of the proof follows Theorem 2. □

In the ideal permutation model, the expressions in Theorems 3 and 4 simplify into

$$\mathbf{Adv}^{\text{ind[n]}}_{\text{IKS}}(M_1, \ldots, M_n, \mu, N) \leq \frac{M^2 + \mu N}{2^c}.$$

We remark that the bound is independent of n, and particularly matches the bound of Sect. 4.1. This is because M is the sum of the complexities $M_1, \ldots, M_n$, and additionally the multiplicity μ is taken over *all* construction queries.

5 Distinguishing Advantage of the Outer-Keyed Sponge

We bound the distinguishing advantage of the outer-keyed sponge construction in the ideal permutation model. A bound for the case of $n = 1$ is given in Sect. 5.1, and it is generalized to arbitrary n in Sect. 5.2. The high-level ideas of the proofs are the same as the ones of Sect. 4. The outer-keyed sponge differs from the inner-keyed sponge by the presence of a key derivation function using f. Therefore, a more involved version of PRP security is needed, where the key derivation L from K is taken into account. We call this notion *joint kdPRP (key derivated PRP) security*. For simplicity, we assume that all keys have equal length, with $v = k/r$ their block length.

Definition 4 (Joint kdPRP Advantage). *We define the joint kdPRP advantage of $\mathcal{A}$ for a given block cipher $E : \{0,1\}^c \times \{0,1\}^b \to \{0,1\}^b$ based on a permutation $f \in \mathsf{Perm}(b)$ as*

$$\mathbf{Adv}^{\mathrm{kdprp[n]}}_{E,\mathrm{KD}}(\mathcal{A}) = \Big| \Pr\left(L_1 \leftarrow \mathrm{KD}^f_{K_1}, \ldots, L_n \leftarrow \mathrm{KD}^f_{K_n};\ \mathcal{A}^{E^f_{L_1},\ldots,E^f_{L_n},f} \Rightarrow 1\right) - \Pr\left(\mathcal{A}^{\pi_1,\ldots,\pi_n,f} \Rightarrow 1\right)\Big|.$$

The adversary can make both forward and inverse primitive queries f and f^{-1}, but is restricted to forward construction queries. We use $\mathbf{Adv}^{\mathrm{kdprp[n]}}_{E,\mathrm{KD}}(M_1,\ldots, M_n,\mu,N)$ to denote the maximum advantage over any adversary with data complexity M_h to the h-th construction oracle ($E^f_{K_h}$ or π_h), time complexity N, and total maximum multiplicity μ.

Intuitively, permutation f results in a kdPRP secure block cipher if (i) it renders sufficiently secure evaluations of KD and (ii) $E^f_{\hat{L}_1}, \ldots, E^f_{\hat{L}_n}$ are secure Even-Mansour block ciphers. Note that, indeed, Definition 4 generalizes Definition 3 in the same way the OKS^f_K of (7) generalizes over IKS^f_K of (5).

5.1 Single Target

Theorem 5. *For $\mathrm{OKS}^f_K : \{0,1\}^* \times \mathbb{N} \to \{0,1\}^{\mathbb{N}}$ with $K \xleftarrow{\$} \{0,1\}^k$ and permutation $f \in \mathsf{Perm}(b)$:*

$$\mathbf{Adv}^{\mathrm{ind[1]}}_{\mathrm{OKS}}(M,\mu,N) \le \frac{M^2}{2^c} + \mathbf{Adv}^{\mathrm{kdprp[1]}}_{E,\mathrm{KD}}(M,\mu,N).$$

Proof. The proof follows the one of Theorem 1 with the difference that now we have $L = \mathrm{KD}^f(K)$, and therefore we bound $\Delta_{\mathcal{B}}(E^f_{\hat{L}}, f; \pi, f) \le \mathbf{Adv}^{\mathrm{kdprp[1]}}_{E,\mathrm{KD}}(M,\mu,N)$. We note that the initial state $\bar{L}||0^c$ (cf. (7)) has no influence on the proof, and we can assume it to be disclosed to the adversary. □

We now bound the kdPRP security of the Even-Mansour construction in the ideal permutation model.

Theorem 6. *For $E^f_{\hat{L}}$ with $L = \mathrm{KD}^f(K)$ for $K \xleftarrow{\$} \{0,1\}^k$ and ideal permutation $f \xleftarrow{\$} \mathsf{Perm}(b)$ we have*

$$\mathbf{Adv}^{\mathrm{kdprp[1]}}_{E,\mathrm{KD}}(M,\mu,N) \le \frac{2\mu N}{2^c} + \lambda(N) + \frac{2\binom{k}{r}N}{2^b},$$

where $\lambda(N)$ is a term bounded in Lemma 2.

Proof. The proof follows the one of Theorem 2, where now the key generation function KD^f_K needs to be taken into account. Adversary $\mathcal{A}$ has access to $X = (E^f_{\hat{L}}, f)$ in the real world or $Y = (\pi, f)$ in the ideal world. To ease the

analysis, we will disclose K at the end of the experiment, *as well as the evaluations to f corresponding to the evaluation of* KD_K^f. These evaluations are written as $\kappa = \{(k_j, l_j)\}_{j=1}^{v}$. In the ideal world, the key K will simply be a dummy key, and κ corresponds to the evaluation of KD_K^f for this dummy key. The transcripts are thus of the form $\tau = (K, \kappa, \tau_1, \tau_f)$. We denote by α the number of distinct elements in κ that are *not* in τ_f:

$$\alpha = \left|\kappa \backslash \tau_f\right|.$$

Definition of good and bad transcripts. We extend the definition of *bad* transcripts from Theorem 2 to additionally cover the case κ shows no surprises to $\mathcal{A}$. Formally, we define a transcript τ as *bad* if one of the following is satisfied:

$$(k_v, l_v) \in \tau_f, \tag{11}$$

$$\exists (s,t) \in \tau_1, (x,y) \in \tau_f \text{ such that } s \oplus x = 0^r || \hat{L} \vee t \oplus y = 0^r || \hat{L}. \tag{12}$$

In the real world, all tuples in a *good* transcript (κ, τ_1, τ_f) consistently define an input-output pair of f. Furthermore, all of the pairs defined by (τ_1, τ_f) are unique, but there may be duplicates without contradictions in (κ, τ_f). In fact, this set contains $|\tau_f| + \alpha$ unique tuples.

Bounding the probability of bad transcripts in the ideal world. In the ideal world, (κ, τ_1, τ_f) is a transcript generated independently of the dummy key $K \xleftarrow{\$} \{0,1\}^c$. By basic probability theory,

$$\begin{aligned}
\Pr\left(D_Y \in \mathcal{T}_{\text{bad}}\right) \leq\ & \Pr\left(D_Y \text{ satisfies (11)}\right) + \\
& \Pr\left(D_Y \text{ satisfies (12)} \mid D_Y \text{ does not satisfy (11)}\right) \\
=\ & \Pr\left(D_Y \text{ satisfies (11)} \wedge \alpha = 0\right) + \\
& \Pr\left(D_Y \text{ satisfies (11)} \wedge \alpha \neq 0\right) + \\
& \Pr\left(D_Y \text{ satisfies (12)} \mid D_Y \text{ does not satisfy (11)}\right).
\end{aligned}$$

We define $\lambda(N) = \Pr\left(D_Y \text{ satisfies (11)} \wedge \alpha = 0\right)$, a term which is bounded in Lemma 2. For the second probability, "D_Y satisfies (11) and $\alpha \neq 0$" implies the existence of a maximal index $j \in \{1, \ldots, v-1\}$ such that $(k_j, l_j) \notin \tau_f$ but $(k_{j+1}, l_{j+1}) \in \tau_f$. As the evaluation of f corresponding to (k_j, l_j) is randomly drawn from a set of size at least $2^b - N - \alpha$, the next evaluation $k_{j+1} = l_j \oplus (K_{j+1} || 0^c)$ happens to be in τ_f with probability at most $\frac{N}{2^b - N - \alpha} \leq \frac{2N}{2^b}$, using that $N + \alpha \leq 2^{b-1}$ without loss of generality. Quantification over j gives bound $\frac{2vN}{2^b}$.

Finally, consider the probability that D_Y satisfies (12). Conditioned on the fact that (11) is not satisfied, L is randomly generated from a set of size at least $2^b - N - \alpha$. This particularly means that a given value for $\hat{L}_i$ has probability at most $1/(2^c - (N+\alpha)2^{-r})$. A straightforward generalization of the proof of Theorem 2 shows that the second probability is bound by $\frac{\mu N}{2^c - (N+\alpha)2^{-r}} \leq \frac{2\mu N}{2^c}$, again using that $N + \alpha \leq 2^{b-1}$.

Bounding the ratio $\mathbf{Pr}(\boldsymbol{D_X} = \boldsymbol{\tau}) / \mathbf{Pr}(\boldsymbol{D_Y} = \boldsymbol{\tau})$ for good transcripts. We have $|\Omega_X| = 2^k \cdot 2^b!$ and $|\Omega_Y| = 2^k \cdot (2^b!)^2$ as before. Also, $|\mathsf{comp}_X(\tau)| = (2^b - M - N - \alpha)!$ by construction. Similarly, we find

$$|\mathsf{comp}_Y(\tau)| = (2^b - M)!(2^b - N - \alpha)! \leq (2^b - M - N - \alpha)!2^b!,$$

and the remainder of the proof follows Theorem 2. □

In the pre-proceedings version, $\lambda(N)$ was inadvertently bounded by $N/2^k$. A similar event was considered by Gaži et al. [18], and we can use their result. We restate it in Lemma 2.

Lemma 2 (Gaži et al. [18], Lemma 12). *If $k = r$, we have $\lambda(N) \leq \dfrac{N}{2^k}$. Otherwise,*

$$\lambda(N) \leq \min\left\{\frac{N^2}{2^{c+1}} + \frac{N}{2^k}, \frac{1}{2^b} + \frac{N}{2^{\left(\frac{1}{2} - \frac{\log_2(3b)}{2r} - \frac{1}{r}\right)k}}\right\}.$$

In the ideal permutation model, the expressions in Theorems 5 and 6 simplify into

$$\mathbf{Adv}^{\mathrm{ind}[1]}_{\mathrm{OKS}}(M, \mu, N) \leq \frac{M^2 + 2\mu N}{2^c} + \lambda(N) + \frac{2\binom{k}{r} N}{2^b}.$$

5.2 Multiple Targets

Theorem 7. *For $\mathrm{OKS}^f_K : \{0,1\}^* \times \mathbb{N} \to \{0,1\}^{\mathbb{N}}$ with $K_1, \ldots, K_n \xleftarrow{\$} \{0,1\}^k$ and permutation $f \in \mathsf{Perm}(b)$:*

$$\mathbf{Adv}^{\mathrm{ind}[n]}_{\mathrm{OKS}}(M_1, \ldots, M_n, \mu, N) \leq \frac{\sum_h M_h^2}{2^c} + \mathbf{Adv}^{\mathrm{kdprp}[n]}_{E,\mathrm{KD}}(M_1, \ldots M_n, \mu, N).$$

Proof. The proof is a combination of the ones of Theorems 3 and 5, and therefore omitted. □

We now bound the joint kdPRP security of the Even-Mansour construction in the ideal permutation model.

Theorem 8. *For $E^f_{L_h}$ with $L_h = \mathrm{KD}^f(K)$ for $K_h \xleftarrow{\$} \{0,1\}^k$ $(h = 1, \ldots, n)$ and ideal permutation $f \xleftarrow{\$} \mathsf{Perm}(b)$ we have*

$$\mathbf{Adv}^{\mathrm{kdprp}[n]}_{E,\mathrm{KD}}(M_1, \ldots, M_n, \mu, N) \leq \frac{2\mu N}{2^c} + \frac{4\sum_{h \neq h'} M_h M_{h'}}{2^c} + n\lambda(N) + \frac{\binom{n}{2}}{2^k} + \frac{2\binom{k}{r}\left(nN + \binom{n}{2}\right)}{2^b},$$

where $\lambda(N)$ is a term bounded in Lemma 2.

Proof. The proof combines the ones of Theorems 4 and 6, with the difference that multiple derivated keys are involved. Adversary $\mathcal{A}$ has access to $X = (E^f_{\hat{L}_1}, \ldots, E^f_{\hat{L}_n}, f)$ in the real world or $Y = (\pi_1, \ldots, \pi_n, f)$ in the ideal world. As before, we will disclose the keys $K_1, \ldots K_n$ at the end of the experiment, *as well as the evaluations to f corresponding to the evaluations of* $\text{KD}^f_{K_1}, \ldots, \text{KD}^f_{K_n}$. These evaluations are written as $\kappa_h = \{(k_j^{(h)}, l_j^{(h)})\}_{j=1}^{v}$. The transcripts are thus of the form $\tau = (K_1, \ldots, K_n, \kappa_1, \ldots, \kappa_n, \tau_1, \ldots, \tau_n, \tau_f)$. We denote by α the number of distinct elements in $\cup_h \kappa_h$ that are *not* in τ_f:

$$\alpha = \left| \cup_h \kappa_h \backslash \tau_f \right|.$$

Definition of good and bad transcripts. We extend the definition of *bad* transcripts from Theorem 6 to multiple keys. Formally, we define a transcript τ as *bad* if one of the following is satisfied:

$$\exists h \text{ such that } (k_v^{(h)}, l_v^{(h)}) \in \tau_f, \tag{13}$$

$$\exists h \neq h' \text{ such that } (k_v^{(h)}, l_v^{(h)}) = (k_v^{(h')}, l_v^{(h')}), \tag{14}$$

$$\exists h, (s,t) \in \tau_h, (x,y) \in \tau_f \text{ such that } s \oplus x = 0^r || \hat{L}_h \vee t \oplus y = 0^r || \hat{L}_h, \tag{15}$$

$$\exists h \neq h', (s,t) \in \tau_h, (s',t') \in \tau_{h'} \text{ such that } s \oplus s' = 0^r || (\hat{L}_h \oplus \hat{L}_{h'}) \vee t \oplus t' = 0^r || (\hat{L}_h \oplus \hat{L}_{h'}). \tag{16}$$

The only condition different from the ones in Theorems 4 and 6 is (14), which assures that there are no two distinct evaluations of KD that produce the same $\hat{L}$. As before, all query pairs defined by $(\tau_1, \ldots, \tau_n, \tau_f)$ are unique, and $(\kappa_1, \ldots, \kappa_n, \tau_f)$. contains $|\tau_f| + \alpha$ unique query tuples.

Bounding the probability of bad transcripts in the ideal world. In the ideal world, $(\tau_1, \ldots, \tau_n, \tau_f)$ is a transcript generated independently of the dummy keys $K_1, \ldots, K_n \xleftarrow{\$} \{0,1\}^c$. By basic probability theory,

$$\begin{aligned} \Pr\left(D_Y \in \mathcal{T}_{\text{bad}}\right) \leq\ & \Pr\left(D_Y \text{ satisfies (13)}\right) + \\ & \Pr\left(D_Y \text{ satisfies (15)} \mid D_Y \text{ does not satisfy (13)}\right) + \\ & \Pr\left(D_Y \text{ satisfies (14)}\right) + \\ & \Pr\left(D_Y \text{ satisfies (16)} \mid D_Y \text{ does not satisfy (14)}\right). \end{aligned}$$

Before reasoning generalized to n targets shows that the first two probabilities are bounded by $n\left(\lambda(N) + \frac{2vN}{2^b}\right)$ and $\frac{2\mu N}{2^c}$, respectively. Using that the keys K_h are all randomly drawn from a set of size at least 2^k, the same reasoning directly shows that the third probability is bounded by $\frac{\binom{n}{2}}{2^k} + \frac{2v\binom{n}{2}}{2^b}$. Finally, the fourth probability, that D_Y satisfies (16), can be analyzed slightly differently from the proof of Theorem 4. More formally, as (14) is not satisfied, $\hat{L}_h \oplus \hat{L}_{h'}$ has probability at most $1/(2^c - (N+\alpha)2^{-r})$ for all $h \neq h'$. This leads to a probability upper bound $\frac{4\sum_{h \neq h'} M_h M_{h'}}{2^c}$.

Bounding the ratio $\Pr(\boldsymbol{D_X} = \tau) / \Pr(\boldsymbol{D_Y} = \tau)$ for good transcripts. The analysis is a direct combination of the proofs of Theorems 4 and 6. □

In the ideal permutation model, the expressions in Theorems 7 and 8 simplify into

$$\mathbf{Adv}^{\mathrm{ind[n]}}_{\mathrm{OKS}}(M_1, \ldots, M_n, \mu, N) \leq \frac{2M^2 + 2\mu N}{2^c} + n\lambda(N) + \frac{\binom{n}{2}}{2^k} + \frac{2\left(\frac{k}{r}\right)\left(nN + \binom{n}{2}\right)}{2^b}.$$

6 Discussion and Conclusions

Our theorems have implications on *all* keyed-sponge based modes as they impose upper bounds to the success probability for both single-target and multi-target attacks, generic in f. In general, a designer of a cryptographic system has a certain security level in mind, where a security level of s bits implies that it should resist against adversaries with resources for performing an amount of computation equivalent to 2^s executions of f. This security level s determines a lower bound for the choice of the sponge capacity c. The indifferentiability bound of Bertoni et al. [6] gives a bound $\frac{(M+N)^2}{2^{c+1}}$ resulting in the requirement $c \geq 2s - 1$ bits. For attack complexities that are relevant in practice, our success probability bounds are dominated by $\frac{\mu N}{2^c}$, combining the time complexity and the multiplicity. This results in the requirement $c \geq s + \log_2(\mu)$ bits. The designer can use this in its advantage by increasing the rate for higher speed or to take a permutation with smaller width for smaller footprint.

The main advantage of having a dependence on μ in the bound is that it makes its application flexible. The proof in this paper remains generic and independent of any use case scenario by considering an adversary who can perform all kinds of queries. Yet, the way a keyed sponge function is used in a concrete protocol can restrict what the attacker can actually do, and the bound follows depending on how these restrictions affect the multiplicity.

In general, μ depends on the mode of use and on the ability of the adversary, and a designer that cares about efficiency has the challenge to reliably estimate it. In real-world applications, the amount of data that will be available to an adversary can easily be upper bound due to physical, protocol-level or other restrictions, imposing an upper bound to M. As per definition $\mu \leq 2M$ the value of c can be taken $c \geq s + \log_2(M) + 1$.

The bound $\mu \leq 2M$ is actually very pessimistic and virtually never reached. The multiplicity is the sum of two components: the forward multiplicity μ_{fw} and the backward multiplicity μ_{bw}. The latter is determined by the responses of the keyed sponge and even an active attacker has little grip on it. For small rates, it is typically $M2^{-r}$ multiplied by a small constant.

The forward multiplicity, however, can be manipulated in some settings. An example of such a use case is a very liberal mode of use on top of the duplex construction [8]. At each duplexing call, the adversary can choose the input for the next duplexing call to force the outer part to some fixed value and let μ_{fw} approach M. The dominating security term then becomes $\frac{MN}{2^c}$, reducing the requirement to $c \geq s + \log_2(M)$. However, most modes and attack circumstances

do not allow the adversary to increase the forward multiplicity μ_{fw} beyond a small multiple of $M2^{-r}$. This is in general the case if the adversary cannot *choose* the outer values. For instance, for sponge based stream ciphers which output a keystream on input of a nonce: if the total number of output blocks is much smaller than $2^{r/2}$, we have $\mu = 2$ with overwhelming probability, reducing the requirement to $c \geq s + 1$. A similar effect occurs in the case of nonce-respecting authenticated encryption scenarios.

Knowing the mode of use and the relevant adversary model, one can often demonstrate an upper bound to the multiplicity. If no sharp bounds can be demonstrated, it may be possible to prove that the multiplicity is only higher than some value μ_{limit} with a very low probability. This probability should then be included in the bound as an additional term.

Acknowledgments. This work was supported in part by the Research Council KU Leuven: GOA TENSE (GOA/11/007). Elena Andreeva and Bart Mennink are Postdoctoral Fellows of the Research Foundation – Flanders (FWO). We thank Peter Gaži, Krzysztof Pietrzak, and Stefano Tessaro for pointing out a flaw in an earlier version of the proof.

References

1. Alizadeh, J., Aref, M., Bagheri, N.: Artemia v1, submission to CAESAR competition (2014)
2. Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mendel, F., Mennink, B., Mouha, N., Wang, Q., Yasuda, K.: PRIMATEs v1, submission to CAESAR competition (2014)
3. Aumasson, J., Jovanovic, P., Neves, S.: NORX v1, submission to CAESAR competition (2014)
4. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Denning, D.E., Pyle, R., Ganesan, R., Sandhu, R.S., Ashby, V. (eds.) ACM CCS 1993, pp. 62–73. ACM (1993)
5. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Sponge functions. In: Ecrypt Hash Workshop 2007, May 2007
6. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the indifferentiability of the sponge construction. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 181–197. Springer, Heidelberg (2008)
7. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Sponge-based pseudo-random number generators. In: Mangard, S., Standaert, F.-X. (eds.) CHES 2010. LNCS, vol. 6225, pp. 33–47. Springer, Heidelberg (2010)
8. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Duplexing the sponge: single-pass authenticated encryption and other applications. In: Miri, A., Vaudenay, S. (eds.) SAC 2011. LNCS, vol. 7118, pp. 320–337. Springer, Heidelberg (2012)
9. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the security of the keyed sponge construction. In: Symmetric Key Encryption Workshop, February 2011
10. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G., Van Keer, R.: Ketje v1, submission to CAESAR competition (2014)
11. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G., Van Keer, R.: Keyak v1, submission to CAESAR competition (2014)

12. CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, November 2014. http://competitions.cr.yp.to/caesar.html
13. Chang, D., Dworkin, M., Hong, S., Kelsey, J., Nandi, M.: A keyed sponge construction with pseudorandomness in the standard model. In: NIST SHA-3 Workshop, March 2012
14. 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)
15. Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1, submission to CAESAR competition (2014)
16. Even, S., Mansour, Y.: A construction of a cipher from a single pseudorandom permutation. In: Matsumoto, T., Imai, H., Rivest, R.L. (eds.) ASIACRYPT 1991. LNCS, vol. 739, pp. 210–224. Springer, Heidelberg (1993)
17. Even, S., Mansour, Y.: A construction of a cipher from a single pseudorandom permutation. J. Cryptol. **10**(3), 151–162 (1997)
18. Gaži, P., Pietrzak, K., Tessaro, S.: Tight bounds for keyed sponges and truncated CBC. In: Cryptology ePrint Archive, Report 2015/053, 22 January 2015
19. Gligoroski, D., Mihajloska, H., Samardjiska, S., Jacobsen, H., El-Hadedy, M., Jensen, R.: π-Cipher v1, submission to CAESAR competition (2014)
20. Jovanovic, P., Luykx, A., Mennink, B.: Beyond $2^{c/2}$ security in sponge-based authenticated encryption modes. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014. LNCS, vol. 8873, pp. 85–104. Springer, Heidelberg (2014)
21. Kavun, E., Lauridsen, M., Leander, G., Rechberger, C., Schwabe, P., Yalçın, T.: Prøst v1, submission to CAESAR competition (2014)
22. 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)
23. Morawiecki, P., Gaj, K., Homsirikamol, E., Matusiewicz, K., Pieprzyk, J., Rogawski, M., Srebrny, M., Wójcik, M.: ICEPOLE v1, submission to CAESAR competition (2014)
24. 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)
25. Perlner, R.: SHA3-based MACs. In: NIST SHA-3 Workshop, August 2014
26. Ristenpart, T., Shacham, H., Shrimpton, T.: Careful with composition: limitations of the indifferentiability framework. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 487–506. Springer, Heidelberg (2011)
27. Rivest, R.L., Schuldt, J.C.N.: Spritz - a spongy RC4-like stream cipher and hash function, October 2014
28. Saarinen, M.: STRIBOB r1, submission to CAESAR competition (2014)
29. Turan, M.S.: Special publication on authenticated encryption. In: NIST SHA-3 Workshop, August 2014