Security of the Suffix Keyed Sponge

Christoph Dobraunig and Bart Mennink

Digital Security Group, Radboud University, Nijmegen, The Netherlands
cdobraunig@cs.ru.nl, b.mennink@cs.ru.nl

Abstract. We formalize and analyze the general suffix keyed sponge construction, a pseudorandom function built on top of a cryptographic permutation. The construction hashes its data using the (keyless) sponge construction, transforms part of the state using the secret key, and generates the tag from the output of a final permutation call. In its simplest form, if the key and tag size are at most the rate of the sponge, one can see the suffix keyed sponge as a simple sponge function evaluation whose input is the plaintext appended with the key. The suffix keyed sponge is, however, much more general: the key and tag size may exceed the rate without any need to make extra permutation calls. We prove that the suffix keyed sponge construction achieves birthday-bound PRF security in the capacity, even if key and tag size exceed the rate. Furthermore, we prove that if the absorption of the key into the state happens in a leakage resilient manner, the suffix keyed sponge itself is leakage resilient as well. Our findings show that the suffix keyed sponge compares favorably with the hash-then-MAC construction. For instance, to reach a security level of k bits, the side-channel protected component in the suffix keyed sponge just needs to process k bits of input besides the key, whereas schemes following the hash-then-MAC construction need a side-channel protected MAC function that processes $2k$ bits of input besides the key. Moreover, even if we just consider black-box attacks, the MAC function in a hash-then-MAC scheme needs to be cryptographically strong whereas in the suffix keyed sponge the key may be absorbed by a simple XOR. The security proofs are performed using the H-coefficient technique, and make effective use of the multicollision limit function results of Daemen et al. (ASIACRYPT 2017), both for arguing that state manipulation larger than the rate is tolerated after key processing and for upper bounding the amount of leakage an attacker may gain about the secret key.

Keywords: suffix MAC · sponge · SuKS · PRF · leakage resilience · proof

1 Introduction

Whenever a device operates in a hostile environment and side-channel attacks [34] are a threat, protection against them becomes a necessity. Hence, a lot of effort has been put in the design of countermeasures against side-channel attacks like masking [17, 31], threshold implementations [41, 42], or special primitives that limit the available data complexity for an attacker [36, 37, 47]. However, the application of such countermeasures is not for free and requires additional resources like chip area, energy, power, computation time, randomness, and so on. In the context of message authentication codes (MACs), this has led to an increased popularity of functions that process the bulk of the input in a keyless manner and use the secret key for finalization only. This way, the first and largest part is not prone to side-channel attacks; only the second part must be protected, but as it is reasonably small this is an easier task. This particularly applies to the case where messages may be arbitrarily large (as in our use case).

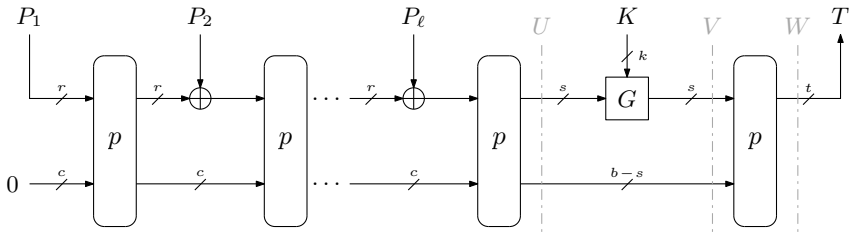

Figure 1: The suffix keyed sponge. The plaintext P is first injectively padded into r-bit blocks $P_1 \dots P_\ell$.

This approach comes to life in the hash-then-MAC construction, used in [7,9,10,32], among others. In the hash-then-MAC construction, one first hashes the plaintext P using a hash function H, and subsequently authenticates the outcome using a MAC function F with key K to obtain a tag T:

$$T = F(K, H(P)) .$$

The advantage of hash-then-MAC over traditional MAC functions like PMAC [15] and CMAC [27] in the context of side-channel protection is evident: as such constructions apply a cryptographic primitive with secret key at least once per plaintext block, each of these invocations must have countermeasures against side-channel attacks in place. In contrast, for hash-then-MAC, the amount of exposure of the secret key is independent of the number of processed input blocks.

However, in order for the hash-then-MAC construction to be k-bit secure, where k is the security level, one must necessarily take a hash function with digest size $2k$ [7,9,10,32]. This also means that the function $F(K, \cdot)$, which is the critical part in the construction from a side-channel perspective, must be able to process $2k$-bit inputs. This negatively impacts the resources needs to implement countermeasures; in general, it is cheaper to protect a smaller primitive against side-channel attacks.

An alternative to the hash-then-MAC approach is the suffix keyed sponge approach used in ISAP [22]. A generalized depiction of the suffix keyed sponge is given in Figure 1, and it is described in detail in Section 3. The function operates as a sponge on top of a permutation p with a state of $b = c + r$ bits, split into an outer part of size r (the rate) and an inner part of size c (the capacity). The k-bit key K is absorbed using an s-bit to s-bit function G and one output block T of size t is squeezed. For a specific case where the state size s and the tag size t are at most the rate r and in addition the key is absorbed by a simple XOR, the construction matches the original description of Bertoni et al. [12, Section 5.11.2]. However, the suffix keyed sponge construction that we consider is more general.

For the suffix keyed sponge, there is no reason to believe that the function G needs to be $2k$ bits large, like it is required for F in hash-then-MAC. In the case of G, it seems to suffice to use a k-bit function only. Indeed, in the suffix keyed sponge we can resort to the secrecy of the state, and collisions in the input to G are not necessarily harmful. Because for the suffix keyed sponge the function G is the focal point of protection against side-channel attacks, the construction compares favorably over the hash-then-MAC mode.

Unfortunately, no analysis of the suffix keyed sponge has appeared so far, neither in the black-box setting nor in the leakage resilient setting. The best we can do is to fall back to the indifferentiability of the keyless sponge [11]. This reduction is valid as long as we consider security in a black-box setting, where the key is absorbed using an XOR, key and tag are of size at most the rate, and the capacity is at least twice the security level. In other words, this is how security of the original and more restrictive version of the suffix keyed sponge of Bertoni et al. [12, Section 5.11.2] was argued.

1.1 Our Contribution

We present a complete and general analysis of the suffix keyed sponge (SuKS), both in the black-box model and in the leakage resilience model. Noting that G is pivotal in the analysis of leakage resilience (the key comes into play only at the occurrence of G), we demonstrate how the security bound depends on the choice of G.

1.1.1 Suffix Keyed Sponge with Restricted Parameters

As a starter, we consider the restricted version of the suffix keyed sponge, where the state size s of G and the tag size t are at most the rate r of the sponge. This restricted version corresponds to the original suffix keyed sponge description of Bertoni et al. [12, Section 5.11.2], though with arbitrary key absorbing function G. In this case, as mentioned above, we can resort to the indifferentiability of the keyless sponge [11] and argue that the resulting construction behaves as a pseudorandom function. This indifferentiability result states that if there is no inner collision on the c-bit inner part, the sponge function (with random permutation) is indifferentiable from a random oracle, and we can analyze security of an idealized version of the suffix keyed sponge. That construction, in turn, can only be broken by an adversary that ever guesses the output of the function G. The result and analysis are given in Section 4.

1.1.2 Suffix Keyed Sponge with Unrestricted Parameters

We present the main result in Section 5: security of the suffix keyed sponge in an unrestricted setting. In this case, both the state size s of the function G as well as the tag size t may be larger than the rate r. We prove that the construction still achieves $c/2$-bit security under the assumption that the outputs of G are uniform (hard to guess) and universal (collide with small probability).

Note that one can consider the suffix keyed sponge state right before the evaluation of G as a state with s-bit rate and $(b-s)$-bit capacity, but $b-s$ may be smaller than c, and hence, it may be that collisions occur on the $(b-s)$-bit inner part even though the keyless part of the suffix keyed sponge does not have capacity collisions. The trick in the proof consists of observing that, even if there are multicollisions on this $(b-s)$-bit inner part, these multicollisions do not harm security of the construction as long as the input to G is different. A symmetric reasoning is applied on the state from which the tag is derived: it can be split into a t-bit outer part and a $(b-t)$-bit inner part. Overall, the bound shows that even partial (multi-)collisions on the input to or output of the keyed permutation call in the suffix keyed sponge provide only limited help to the adversary. Hence, the security level usually stays in the area of $2^{c/2}$.

The result holds for an arbitrary G that is uniform and universal, but this does not mean that it needs to be cryptographically strong: a simple XOR suffices. This observation demonstrates another advantage of the suffix keyed sponge over the hash-then-MAC approach, where F must necessarily be cryptographically strong.

1.1.3 Leakage Resilience of Suffix Keyed Sponge

In practice, however, one might want select a stronger function G, for the plain reason that for an XOR even a small leakage usually allows to draw conclusions about the absorbed secret key. We therefore transform the main result to the setting of leakage resilience PRF design in Section 6. We consider non-adaptive leakage, akin to [26, 28, 44, 46, 49], and restrict our focus to leakage coming from p. In other words, we assume that G is a protected function (it could be instantiated as a leakage resilient keyed duplex, cf., Dobraunig and Mennink [25]), and consider that the adversary may obtain leakage from the last invocation of p in the suffix keyed sponge. It is easily observed from Figure 1 that

only this invocation of p may leak information; all previous ones process no secret data in the first place. Keeping this in mind, the leakage resilience proof is quite clean. First, we note that for repeated inputs X to G for different $(b-s)$-bit inner parts, the permutation p may leak different pieces of information about $G(K, X)$. By upper bounding the maximum size of a multicollision in the input to G, we can henceforth upper bound the amount of leakage that an adversary may learn for any particular $G(K, X)$, and then upper bound its success probability of guessing this value.

In Section 7, we map our results to ISAP [22] and demonstrate that the multicollision events only contribute to the bound negligibly, both in the black-box setting and in the leakage resilience setting. In the conclusion in Section 8, we elaborate on what our leakage resilient PRF result means for leakage resilient MAC design.

1.2 Related Work

A significant amount of research has been put in the analysis of the keyed sponge, where the state of the sponge is keyed prior to the absorption of data [1,13,14,16,19,29,33,38,39,39,40]. These results have not only led to better and more fine grained bounds, but in the end also demonstrated that the full bit-width of the state can be used to absorb data in the case of such "prefix" keyed duplex or sponge [19]. The reason for this is that, as the key is used to initialize the state, all subsequent states are secret to an adversary.

This secrecy of the states also clearly shows why these constructions are more expensive from a side-channel protection point of view. Constructions following the prefix keyed duplex or prefix keyed sponge usually allow for the evaluation of the same secret state for varying inputs. This is for instance the case in authenticated encryption schemes, which are initialized using a static secret key and a nonce. Hence, DPA attacks can be mounted if the protection against side-channel attacks is insufficient, as shown by Samwel and Daemen [45]. Even stronger: not only the initialization has to be protected, but every stage where known changing inputs are mixed with a static secret. In the case of an attacker that can control all the inputs, e.g, for a MAC based on a prefix keyed sponge during verification, this means that the whole construction has to be protected against side-channel attacks. In this light, it makes sense to limit the exposure of the secret key and rely on a suffix keyed sponge as done in ISAP [22].

It is worth noting that there exist other ways besides hash-then-MAC and the suffix keyed sponge to end up with leakage resilient MAC constructions. One way to do so is to base the scheme on ideas from asymmetric cryptography, an approach endeavored in a proposal of Martin et al. [35] that is based on the Barreto-Naehrig [2] family of pairing-friendly elliptic curves.

Alternatively, one can design a leakage resilient PRF from a (smaller) weak PRF in a sequential fashion, where the key input to the underlying weak PRF depends on the processed data so far. This approach is adopted in [26, 28, 46, 49], among others. Such a PRF can, naturally, also be used as instantiation of G in the suffix keyed sponge.

2 Preliminaries

Let $m, n \in \mathbb{N}$. The set of n-bit strings is denoted $\{0,1\}^n$, the set of arbitrarily long strings is denoted $\{0,1\}^*$, the set of n-bit functions is denoted $\mathsf{func}(n)$, and the set of n-bit permutations is denoted $\mathsf{perm}(n)$. We let pad_n be any injective padding function that transforms an arbitrarily long string $X \in \{0,1\}^*$ into n-bit blocks $X_1, \ldots, X_\ell$. For $X \in \{0,1\}^n$ and if $m \leq n$, we denote by $\mathrm{left}_m(X)$ (resp., $\mathrm{right}_m(X)$) the m leftmost (resp., rightmost) bits of X. If $m \geq n$, we write $(m)_n = m(m-1)\cdots(m-n+1)$ as the falling factorial. For a finite set $\mathcal{X}$, $X \xleftarrow{\$} \mathcal{X}$ denotes the uniformly random drawing of an element X from $\mathcal{X}$.

2.1 Distinguishing Advantage

An adversary $\mathcal{A}$ is an algorithm. It is given access to one or more oracles $\mathcal{O}$, and after interaction with $\mathcal{O}$ it outputs a decision bit b: $b \leftarrow \mathcal{A}^{\mathcal{O}}$. For two oracles $\mathcal{O}$ and $\mathcal{P}$, the adversarial distinguishing advantage is defined as

$$\Delta_{\mathcal{A}}\left(\mathcal{O}\,;\,\mathcal{P}\right) = \mathbf{Pr}\left(1 \leftarrow \mathcal{A}^{\mathcal{O}}\right) - \mathbf{Pr}\left(1 \leftarrow \mathcal{A}^{\mathcal{P}}\right)\,. \tag{1}$$

2.2 PRF Security

Let $b, k, t \in \mathbb{N}$ and $m \in \mathbb{N} \cup \{*\}$. Consider a function $F : \{0,1\}^k \times \{0,1\}^m \to \{0,1\}^t$ that internally uses a permutation $p \in \mathsf{perm}(b)$. We write F instantiated with permutation p and key $K \in \{0,1\}^k$ as F_K^p. The pseudorandom function (PRF) security of F against an adversary $\mathcal{A}$ is defined as

$$\mathbf{Adv}_F^{\mathrm{prf}}(\mathcal{A}) = \Delta_{\mathcal{A}}\left(F_K^p, p\,;\; R_{m,t}, p\right)\,, \tag{2}$$

where the randomness is taken over the random drawing of $K \xleftarrow{\$} \{0,1\}^k$, the random drawing of $p \xleftarrow{\$} \mathsf{perm}(b)$, the definition of a function $R_{m,t} : \{0,1\}^m \to \{0,1\}^t$ that generates a random t-bit response for each new input, and the random coins of $\mathcal{A}$.

The adversary is typically bounded by three types of complexities: q denotes the construction complexity, the number of queries the adversary may make to F_K^p or $R_{m,t}$; N denotes the total primitive complexity, which accounts for the total number of evaluations of p (either direct forwardly, direct inversely, or via the real-world construction oracle F_K^p); τ denotes the time complexity.

It is a well-known result that a PRF is a secure message authentication code in the black-box model [4, 5, 30]. More detailed, if $\mathbf{Adv}_F^{\mathrm{mac}}(\mathcal{A})$ denotes the advantage of an adversary $\mathcal{A}$ with query access to (F_K^p, p) to output a non-trivial forgery for F_K^p, then

$$\mathbf{Adv}_F^{\mathrm{mac}}(\mathcal{A}) \leq \mathbf{Adv}_F^{\mathrm{prf}}(\mathcal{A}) + \frac{q}{2^t}\,,$$

where q is the number of forgery attempts that $\mathcal{A}$ makes. We note that this reduction does not necessarily apply in the leakage resilience setting, cf., Section 8.

2.3 Uniform and Universal Functions

Let $k, s \in \mathbb{N}$ and $\delta, \varepsilon \in [0, \infty)$. A function $G : \{0,1\}^k \times \{0,1\}^s \to \{0,1\}^s$ is $2^{-\delta}$-uniform if for any $X, Y \in \{0,1\}^s$,

$$\mathbf{Pr}\left(G(K, X) = Y\right) \leq 2^{-\delta}\,,$$

where the randomness is taken over the random drawing of $K \xleftarrow{\$} \{0,1\}^k$. It is $2^{-\varepsilon}$-universal if for any distinct $X, X' \in \{0,1\}^s$,

$$\mathbf{Pr}\left(G(K, X) = G(K, X')\right) \leq 2^{-\varepsilon}\,,$$

where the randomness is taken over the random drawing of $K \xleftarrow{\$} \{0,1\}^k$. Here, we say that a function is 0-uniform (resp., 0-universal) if above definition applies for $\delta = \infty$ (resp., $\varepsilon = \infty$).

Note that for $k \leq s$, the function $G(K, X) = K \| 0^{s-k} \oplus X$ that simply XORs the two input values is 2^{-k}-uniform and 0-universal. Alternatively, we can define G as a random function: $G : |\mathsf{func}(s)| \times \{0,1\}^s \to \{0,1\}^s$ is defined as $G(f, X) = f(X)$. This function G is 2^{-s}-uniform and 2^{-s}-universal.

2.4 Multicollision Limit Function

Daemen et al. [19] introduced the multicollision limit function in the context of keyed sponge proofs. We will use the same notion in our proofs.

Let $q, b, s \in \mathbb{N}$ such that $s \leq b$. Consider the experiment of throwing q balls uniformly at random in 2^{b-s} bins, and denote by μ the maximum number of balls in any single bin. The multicollision limit function $\mu^q_{b-s,s}$ is defined as the smallest natural number x that satisfies

$$\mathbf{Pr}\left(\mu > x\right) \leq \frac{x}{2^s}\,.$$

Daemen et al. also proved that if one does not consider the bins to be uniformly randomly selected, but rather according to a distribution D where the i-th ball ends up in a certain bin with probability

$$\frac{2^s - (i-1)}{2^b - (i-1)} \leq p \leq \frac{2^s}{2^b - (i-1)}\,, \tag{3}$$

the corresponding multicollision function, defined as $\mu^{D,q}_{b-s,s}$, satisfies $\mu^{D,q}_{b-s,s} \leq \mu^{2q}_{b-s,s}$ [19, Lemma 6].

Furthermore, Daemen et al. [19] gave an in-depth analysis of the term $\mu^q_{b-s,s}$. The analysis is tedious, but the conclusion is that the term behaves as follows:

$$\mu^q_{b-s,s} \lesssim \begin{cases} b/\log_2\left(\dfrac{2^{b-s}}{q}\right)\,, & \text{for } q \lesssim 2^{b-s}\,, \\ b \cdot \dfrac{q}{2^{b-s}}\,, & \text{for } q \gtrsim 2^{b-s}\,. \end{cases}$$

3 Suffix Keyed Sponge

Let $b, c, r, k, s, t \in \mathbb{N}$ such that $c + r = b$ and $k, s, t \leq b$. Let $p : \{0,1\}^b \to \{0,1\}^b$ be a permutation and $G : \{0,1\}^k \times \{0,1\}^s \to \{0,1\}^s$ be a function. The suffix keyed sponge (SuKS) $F : \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^t$ is defined in Algorithm 1 and depicted in Figure 1. The function first injectively pads its input P, and hashes it the usual sponge way. Only at this point, the key will be used: the outer k bits of the resulting state are transformed through $G(K, \cdot)$. The resulting state is processed by the permutation p once more, and the tag equals the outer t bits of the state.

Algorithm 1 Suffix keyed sponge construction F

Input: $(K, P) \in \{0,1\}^k \times \{0,1\}^*$
Output: $T \in \{0,1\}^t$
1: $P_1 \ldots P_\ell \leftarrow \mathrm{pad}_r(P)$
2: $S \leftarrow 0^b$
3: **for** $i = 1, \ldots, \ell$ **do**
4: $\quad S \leftarrow S \oplus P_i \| 0^c$
5: $\quad S \leftarrow p(S)$
6: $S \leftarrow G(K, \mathrm{left}_s(S)) \,\|\, \mathrm{right}_{b-s}(S)$
7: $S \leftarrow p(S)$
8: **return** $\mathrm{left}_t(S)$

4 Security of Suffix Keyed Sponge with Restricted Parameters

As a starter, we prove security of the suffix keyed sponge for a restricted case where $s, t \leq r$. This version of the suffix keyed sponge corresponds to the original version suggested by Bertoni et al. [12, Section 5.11.2], though with arbitrary key absorbing function G. The result is mainly included to show how, in this case, we can rely on the indifferentiability of the sponge [11].

Theorem 1. *Let $b, c, r, k, s, t \in \mathbb{N}$ such that $c + r = b$, $k \leq b$, and $s, t \leq r$. Consider the suffix keyed sponge of Section 3 based on random permutation $p \xleftarrow{\$} \mathsf{perm}(b)$ and a function $G : \{0,1\}^k \times \{0,1\}^s \to \{0,1\}^s$. Assume that G is $2^{-\delta}$-uniform. For any adversary $\mathcal{A}$ with construction complexity q and primitive complexity $N \leq 2^{b-1}$,*

$$\mathbf{Adv}_F^{\mathrm{prf}}(\mathcal{A}) \leq \frac{N^2 + N}{2^c} + \frac{N}{2^\delta} .$$

Proof. Let $K \xleftarrow{\$} \{0,1\}^k$ and $p \xleftarrow{\$} \mathsf{perm}(b)$. Let $R_{*,t} : \{0,1\}^* \to \{0,1\}^t$ be a function that generates a random t-bit response for each new input. Consider any adversary $\mathcal{A}$, whose goal is to maximize

$$\Delta_{\mathcal{A}} \left(F_K^p, p \; ; \; R_{*,t}, p \right) , \tag{4}$$

As we will argue security based on the uniformity and universality of G (and not on the computational security of it), we assume that $\mathcal{A}$ is computationally unbounded.

Let sponge^p be the sponge construction based on permutation p with rate r, that gets input strings *of size a multiple of r (so it does not pad itself)* and a natural number $z \in \mathbb{N}$, and it outputs a string of z bits. One can equivalently write F_K^p as a function $F_K^{\mathrm{sponge}^p}$ as follows:

Input: $(K, P) \in \{0,1\}^k \times \{0,1\}^*$
Output: $T \in \{0,1\}^t$
1: $P_1 \dots P_\ell \leftarrow \mathrm{pad}_r(P)$
2: $Z \leftarrow \mathrm{sponge}^p(P_1 \dots P_\ell, s)$
3: $P' \leftarrow \big(G_K(Z) \oplus Z\big) \parallel 0^{r-s}$
4: $Z \leftarrow \mathrm{sponge}^p(P_1 \dots P_\ell \parallel P', t)$
5: **return** Z

We will resort to the indifferentiability of the sponge [11], which states that there exists a simulator S such that $\Delta_{\mathcal{A}'} \left(\mathrm{sponge}^p, p \; ; \; RO, S^{RO} \right) \leq \binom{N+1}{2} / 2^c$ for any adversary $\mathcal{A}'$ with total complexity N, where RO is a random oracle. We write F_K^{RO} as the suffix keyed sponge construction where the two evaluations of sponge^p are replaced by RO, and write $F_K^{RO,RO'}$ as that construction where the first evaluation of sponge^p is replaced by RO and the second one by RO'. By a game hopping argument,

$$\begin{aligned}
&\Delta_{\mathcal{A}} \left(F_K^{\mathrm{sponge}^p}, p \; ; \; R_{*,t}, p \right) \\
&\leq \Delta_{\mathcal{A}} \left(F_K^{\mathrm{sponge}^p}, p \; ; \; F_K^{RO}, S^{RO} \right) \\
&+ \Delta_{\mathcal{A}} \left(F_K^{RO}, S^{RO} \; ; \; F_K^{RO,RO'}, S^{RO} \right) + \Delta_{\mathcal{A}} \left(F_K^{RO,RO'}, S^{RO} \; ; \; R_{*,t}, p \right) \\
&\leq \Delta_{\mathcal{A}'} \left(\mathrm{sponge}^p, p \; ; \; RO, S^{RO} \right) \\
&+ \Delta_{\mathcal{A}} \left(F_K^{RO}, S^{RO} \; ; \; F_K^{RO,RO'}, S^{RO} \right) + \Delta_{\mathcal{A}''} \left(S^{RO} \; ; \; p \right) ,
\end{aligned} \tag{5}$$

where $\mathcal{A}'$ and $\mathcal{A}''$ are some adversaries with the same primitive complexity N as $\mathcal{A}$ (noting that any evaluation of $F_K^{\mathrm{sponge}^p}$ indeed consists of two sponge calls, but they jointly consist

of $\ell + 1$ unique primitive evaluations). The first and last distance in (5) are at most the indifferentiability of the sponge, $\binom{N+1}{2}/2^c$. For the middle term of (5), we define the following event bad:

- bad: S^{RO} calls its oracle for input $P_1 \dots P_\ell \parallel \big(G_K(Z) \oplus Z\big) \parallel 0^{r-s}$, where $Z = RO(P_1 \dots P_\ell, s)$.

As long as bad does not happen, the adversary cannot notice the difference between RO and RO' for the finalization calls in both world. Therefore, (F_K^{RO}, S^{RO}) and $(F_K^{RO,RO'}, S^{RO})$ are indistinguishable, and thus

$$\Delta_{\mathcal{A}}\left(F_K^{RO}, S^{RO}\ ;\ F_K^{RO,RO'}, S^{RO}\right) \le \mathbf{Pr}\left(\mathsf{bad}\right) . \tag{6}$$

The event bad requires an adversary to force S into querying the random oracle for value $Y := G_K(Z) \oplus Z$, where Z is fixed by the specific query that $\mathcal{A}$ makes (by design of the actual simulator of [11]). The adversary makes N attempts, and any of them succeeds with probability at most $2^{-\delta}$. We thus obtain that $\mathbf{Pr}\left(\mathsf{bad}\right) \le N/2^{\delta}$. □

5 Security of Suffix Keyed Sponge with Unrestricted Parameters

In this section, we set aside the restriction $s, t \le r$ of the previous section and present the main result, namely security of the suffix keyed sponge for arbitrary r, s, t.

Theorem 2. *Let $b, c, r, k, s, t \in \mathbb{N}$ such that $c + r = b$ and $k, s, t \le b$. Consider the suffix keyed sponge of Section 3 based on random permutation $p \xleftarrow{\$} \mathsf{perm}(b)$ and a function $G : \{0,1\}^k \times \{0,1\}^s \to \{0,1\}^s$. Assume that G is $2^{-\delta}$-uniform and $2^{-\varepsilon}$-universal. For any adversary $\mathcal{A}$ with construction complexity $q \ge 2$ and primitive complexity $N \le 2^{b-1}$,*

$$\mathbf{Adv}_F^{\mathrm{prf}}(\mathcal{A}) \le \frac{2N^2}{2^c} + \frac{\mu_{b-s,s}^{2(N-q)} \cdot N}{2^{\min\{\delta,\varepsilon\}}} + \frac{\mu_{t,b-t}^{q} \cdot N}{2^{b-t}} .$$

The proof of Theorem 2 is given in Section 5.1.

Note that G itself may be anything, as long as it is $2^{-\delta}$-uniform and $2^{-\varepsilon}$-universal. The simple XOR function, i.e., setting $s = k$ and taking $G(K, X) = K \oplus X$, is 2^{-k}-uniform and 0-universal. A more general instantiation is to have G to be a pseudorandom function independent of the permutation p. In this case, the first step of the proof would be to replace G by a random function at cost $\mathbf{Adv}_G^{\mathrm{prf}}(\mathcal{A}')$, for some adversary $\mathcal{A}'$. The second step would be to rely on the observation that a random s-bit function is 2^{-s}-uniform and 2^{-s}-universal (see Section 2.3). This leads to the following corollary, for which the formal proof is given in Section 5.2.

Corollary 1. *Let $b, c, r, k, s, t \in \mathbb{N}$ such that $c + r = b$ and $k, s, t \le b$. Consider the suffix keyed sponge of Section 3 based on random permutation $p \xleftarrow{\$} \mathsf{perm}(b)$ and a function $G : \{0,1\}^k \times \{0,1\}^s \to \{0,1\}^s$ independent of p. For any adversary $\mathcal{A}$ with construction complexity $q \ge 2$, primitive complexity $N \le 2^{b-1}$, and time complexity τ,*

$$\mathbf{Adv}_F^{\mathrm{prf}}(\mathcal{A}) \le \frac{2N^2}{2^c} + \frac{\mu_{b-s,s}^{2(N-q)} \cdot N}{2^{s}} + \frac{\mu_{t,b-t}^{q} \cdot N}{2^{b-t}} + \mathbf{Adv}_G^{\mathrm{prf}}(\mathcal{A}') ,$$

for some adversary $\mathcal{A}'$ with construction complexity q, primitive complexity 0, and time complexity $\tau' \approx \tau$.

5.1 Proof of Theorem 2

Let $K \xleftarrow{\$} \{0,1\}^k$ and $p \xleftarrow{\$} \mathsf{perm}(b)$. Let $R_{*,t} : \{0,1\}^* \to \{0,1\}^t$ be a function that generates a random t-bit response for each new input. Consider any adversary $\mathcal{A}$, whose goal is to maximize

$$\Delta_{\mathcal{A}}\left(F_K^p, p\ ;\ R_{*,t}, p\right)\ , \tag{7}$$

We will call $\mathcal{O} := (F_K^p, p)$ the real world and $\mathcal{P} := (R_{*,t}, p)$ the ideal world. As we will argue security based on the uniformity and universality of G (and not on the computational security of it), we assume that $\mathcal{A}$ is computationally unbounded. Without loss of generality, it is deterministic: for any probabilistic adversary there is a deterministic one with the same success probability.

5.1.1 Oracle Interaction

The adversary has a total primitive complexity N, meaning that for the *real world* the total number primitive queries (direct or through F_K^p) does not exceed N. Of course, the ideal world $R_{*,t}$ does not query p, but the complexity is counted in terms of the number of evaluations that would be made to p in the real world. Note that in the real world $\mathcal{O} = (F_K^p, p)$, any construction query P of ℓ blocks entails $\ell + 1$ evaluations of p: the first ℓ evaluations are "offline" and the adversary can evaluate them itself using the primitive oracle, and the last evaluation is keyed. We will thus consider $\mathcal{A}$ to be allowed an "offline" complexity $N - q$ and an online complexity q. Without loss of generality, as duplicated queries are not counted doubly, we may assume that before each construction query, the adversary makes *all primitive queries but the last one* itself offline.

The interaction of $\mathcal{A}$ with its oracle ($\mathcal{O}$ or $\mathcal{P}$) is summarized in a view. All q construction queries (to F_K^p in the real world and to $R_{*,t}$ in the ideal world) are summarized in a view

$$v_c = \{(P_1, T_1), \ldots, (P_q, T_q)\}\,.$$

All primitive evaluations (to p, in both worlds) are summarized in a view

$$v_p = \{(\mathsf{dir}_1, X_1, Y_1), \ldots, (\mathsf{dir}_{N-q}, X_{N-q}, Y_{N-q})\}\,,$$

where $\mathsf{dir}_j \in \{+, -\}$ denotes the direction of the j-th query: forward primitive queries are denoted by $\mathsf{dir}_j = +$ and inverse primitive queries by $\mathsf{dir}_j = -$. The variables dir_j are used to distinguish between the direction of the queries in case of capacity collisions.

After the conversation of $\mathcal{A}$ with its oracle, but before it outputs its decision bit, we reveal additional information to $\mathcal{A}$. First of all, we reveal a key K. In the real world, this is the key $K \xleftarrow{\$} \{0,1\}^k$ that is effectively used by G_K; in the ideal world, it is a dummy key. In addition, we reveal a tuple $\{Z_1, \ldots, Z_q\}$. In the real world, these are the values that are truncated at the end of each construction query to F_K^p (i.e., the values $\mathrm{right}_{b-t}(S)$ with S the state of line 7 of Algorithm 1); in the ideal world, these are random values $Z_i \xleftarrow{\$} \{0,1\}^{b-t}$.

The values Z_i are appended to v_c:

$$v_c' = \{(P_1, T_1, Z_1), \ldots, (P_q, T_q, Z_q)\}\,.$$

The aggregate view is defined as $v = (v_c', v_p, K)$. We assume that the adversary never makes any duplicate query, hence $P_i \neq P_{i'}$ for any two distinct queries in v_c' and $(X_j, Y_j) \neq (X_{j'}, Y_{j'})$ for any two distinct queries in v_p.

It is important to note that, as we force $\mathcal{A}$ to make all primitive queries corresponding to the unkeyed part of F_K^p, for each construction query (P_i, T_i), the state value U_i that

is at the end of the unkeyed part of F_K^p (see Figure 1) can be retrieved from v_p. We subsequently define V_i and W_i from U_i, K, and v'_c as follows:

$$V_i = G(K, \mathrm{left}_s(U_i)) \parallel \mathrm{right}_{b-s}(U_i)\,, \tag{8}$$

$$W_i = T_i \parallel Z_i\,. \tag{9}$$

Note that in the real world (F_K^p, p), the values V_i and W_i are the actual values indicated in Figure 1. In the ideal world, they are simply defined as in (8) and (9).

5.1.2 H-Coefficient Technique

We denote by $D_{\mathcal{O}}$ the probability distribution of views in interaction with $\mathcal{O}$, and by $D_{\mathcal{P}}$ the probability distribution of views in interaction with $\mathcal{P}$. Define $\mathcal{V}$ to be the set of "attainable views": views v such that $\mathbf{Pr}\left(D_{\mathcal{P}} = v\right) > 0$. We will use the H-coefficient technique [18, 43].

Lemma 1 (H-coefficient technique)**.** *Consider a partition $\mathcal{V} = \mathcal{V}_{\mathrm{good}} \cup \mathcal{V}_{\mathrm{bad}}$ of the set of views into "good" and "bad" views. Let $\epsilon \in [0,1]$ be such that $\frac{\mathbf{Pr}(D_{\mathcal{O}}=v)}{\mathbf{Pr}(D_{\mathcal{P}}=v)} \geq 1-\epsilon$ for all $v \in \mathcal{V}_{\mathrm{good}}$. Then, $\Delta_{\mathcal{A}}\left(\mathcal{O}\,;\,\mathcal{P}\right) \leq \epsilon + \mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right)$.*

5.1.3 Bad Views

Informally, we consider a view *bad* if for any construction query P_i in the real world $\mathcal{O} = (F_K^p, p)$, the evaluation (V_i, W_i) of p is repeated. This is formally covered by the following two events:

- $\mathsf{coll}_{\mathsf{cc}}$: there exist distinct $i, i' \in \{1, \ldots, q\}$ with

$$V_i = V_{i'} \text{ or } W_i = W_{i'}\,;$$

- $\mathsf{coll}_{\mathsf{cp}}$: there exist $i \in \{1, \ldots, q\}$ and $j \in \{1, \ldots, N-q\}$ with

$$V_i = X_j \text{ or } W_i = Y_j\,.$$

We indeed need these two bad events, because in the non-occurrence of either of those, $v = (v'_c, v_p, K)$ corresponds to *exactly* N input/output tuples of p (and one key).

However, analyzing bad events $\mathsf{coll}_{\mathsf{cc}}$ and $\mathsf{coll}_{\mathsf{cp}}$ is quite involved, and we will define four auxiliary bad events. For this, let $\nu_{\mathsf{right}} = \mu_{b-s,s}^{2(N-q)}$ and $\nu_{\mathsf{tag}} = \mu_{t,b-t}^{q}$ be two thresholds.

- $\mathsf{cap}_{\mathsf{fwd}}$: there exists $j \in \{1, \ldots, N-q\}$ with $\mathsf{dir}_j = +$ and

$$\mathrm{right}_c(Y_j) \in \{\mathrm{right}_c(X_1), \ldots, \mathrm{right}_c(X_j), \mathrm{right}_c(Y_1), \ldots, \mathrm{right}_c(Y_{j-1})\}\,;$$

- $\mathsf{cap}_{\mathsf{inv}}$: there exists $j \in \{1, \ldots, N-q\}$ with $\mathsf{dir}_j = -$ and

$$\mathrm{right}_c(X_j) \in \{\mathrm{right}_c(Y_1), \ldots, \mathrm{right}_c(Y_{j-1}), 0^c\}\,;$$

- $\mathsf{mc}_{\mathsf{right}}$: there exist distinct $j_1, \ldots, j_{\nu_{\mathsf{right}}+1} \in \{1, \ldots, N-q\}$ with $\mathsf{dir}_j = +$ and

$$\mathrm{right}_{b-s}(Y_{j_1}) = \cdots = \mathrm{right}_{b-s}(Y_{j_{\nu_{\mathsf{right}}+1}})\,;$$

- $\mathsf{mc}_{\mathsf{tag}}$: there exist distinct $i_1, \ldots, i_{\nu_{\mathsf{tag}}+1} \in \{1, \ldots, q\}$ with

$$\mathrm{left}_t(W_{i_1}) = \cdots = \mathrm{left}_t(W_{i_{\nu_{\mathsf{tag}}+1}})\,.$$

Events $\mathsf{cap}_{\mathsf{fwd}}$ and $\mathsf{cap}_{\mathsf{inv}}$ cover the case that there exists a capacity collision, and that the adversary has potentially found two different plaintexts leading to the same inner part. At the cost of readability, the event $\mathsf{cap}_{\mathsf{fwd}}$ can be tightened slightly, noting that it is not a problem if $\mathrm{right}_c(Y_j)$ hits an older inner part if this j-th query is not "rooted", i.e., if there is no path from 0^b to this query. See also Daemen et al. [20].

The events $\mathsf{mc}_{\mathsf{right}}$ and $\mathsf{mc}_{\mathsf{tag}}$ cover multicollisions on part of the state, and are included to tighten the bounding of the occurrence of $\mathsf{coll}_{\mathsf{cc}} \vee \mathsf{coll}_{\mathsf{cp}}$. We note that the choice of thresholds ν_{right} and ν_{tag} (and our reliance on the multicollision limit function of Section 2.4) is for the sake of generality. It may be that for specific parameter choices b, c, s, t, better thresholds may render a better bound. See also Section 8.

We write $\mathsf{cap} = \mathsf{cap}_{\mathsf{fwd}} \vee \mathsf{cap}_{\mathsf{inv}}$ and

$$\mathsf{bad} = \mathsf{cap} \vee \mathsf{mc}_{\mathsf{right}} \vee \mathsf{mc}_{\mathsf{tag}} \vee \mathsf{coll}_{\mathsf{cc}} \vee \mathsf{coll}_{\mathsf{cp}} \,. \tag{10}$$

5.1.4 Probability of Bad View

Our goal is to bound the probability that a bad transcript is generated in the ideal world, $\mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right)$. This probability equals the probability that a view v generated by $D_{\mathcal{P}}$ satisfies bad. By basic probability theory,

$$\begin{aligned}\mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right) = \mathbf{Pr}\left(\mathsf{bad}\right) \leq\ & \mathbf{Pr}\left(\mathsf{cap}\right) + \mathbf{Pr}\left(\mathsf{mc}_{\mathsf{right}}\right) + \mathbf{Pr}\left(\mathsf{mc}_{\mathsf{tag}}\right) \\ & + \mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cc}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{right}})\right) \\ & + \mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cp}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{right}} \vee \mathsf{mc}_{\mathsf{tag}})\right) \,. \end{aligned} \tag{11}$$

Here, we recall that in the ideal world, $K \xleftarrow{\$} \{0,1\}^k$, $W_1, \ldots, W_q \xleftarrow{\$} \{0,1\}^b$, and each tuple $(\mathsf{dir}_j, X_j, Y_j) \in v_p$ has either Y_j random bar repetition (if $\mathsf{dir}_j = +$) or X_j random bar repetition (if $\mathsf{dir}_j = -$).

cap. Consider the j-th query $(\mathsf{dir}_j, X_j, Y_j)$. If it is a forward query it can only set $\mathsf{cap}_{\mathsf{fwd}}$ and if it is an inverse query it can only set $\mathsf{cap}_{\mathsf{inv}}$. In either case, the response is uniformly randomly generated using a random permutation p from a set of size at least $2^b - (j-1)$ elements. The query sets $\mathsf{cap}_{\mathsf{fwd}}$ with probability at most $(2j-1)2^r/(2^b-(j-1))$ and $\mathsf{cap}_{\mathsf{inv}}$ with probability at most $j2^r/(2^b-(j-1))$. In either case, as any query is either forward or inverse, the success probability is at most $2(2j-1)/2^c$, using that $j - 1 \leq 2^{b-1}$. Summing over all queries, we obtain

$$\mathbf{Pr}\left(\mathsf{cap}\right) = \mathbf{Pr}\left(\mathsf{cap}_{\mathsf{fwd}} \vee \mathsf{cap}_{\mathsf{inv}}\right) \leq \sum_{j=1}^{N-q} \frac{2(2j-1)}{2^c} = \frac{2(N-q)^2}{2^c} \,.$$

$\mathsf{mc}_{\mathsf{right}}$. The values Y_j are randomly generated from $\{0,1\}^b \backslash \{Y_1, \ldots, Y_{j-1}\}$, and at most $N - q$ draws are done. The event $\mathsf{mc}_{\mathsf{right}}$ is thus a balls-and-bins experiment with $N - q$ balls uniformly randomly bar repetition thrown into 2^{b-s} bins, in such a way that any of the bins contains more than ν_{right} balls. Note that the distribution of balls satisfies the condition of (3). By definition of $\nu_{\mathsf{right}} = \mu_{b-s,s}^{2(N-q)}$, we can resort to the multicollision limit function of Section 2.4 and obtain

$$\mathbf{Pr}\left(\mathsf{mc}_{\mathsf{right}}\right) \leq \frac{\mu_{b-s,s}^{2(N-q)}}{2^s} \,.$$

$\mathsf{mc}_{\mathsf{tag}}$. The values W_i are randomly generated from $\{0,1\}^b$, and q draws are done. The event $\mathsf{mc}_{\mathsf{tag}}$ is thus a balls-and-bins experiment with q balls uniformly randomly thrown

into 2^t bins, in such a way that any of the bins contains more than ν_{tag} balls. By definition of $\nu_{\mathsf{tag}} = \mu^q_{t,b-t}$, we can resort to the multicollision limit function of Section 2.4 and obtain

$$\mathbf{Pr}\left(\mathsf{mc}_{\mathsf{tag}}\right) \leq \frac{\mu^q_{t,b-t}}{2^{b-t}}\,.$$

$\mathsf{coll}_{\mathsf{cc}}$. The event $V_i = V_{i'}$ is equivalent to stating that

$$G(K, \mathrm{left}_s(U_i)) \parallel \mathrm{right}_{b-s}(U_i) = G(K, \mathrm{left}_s(U_{i'})) \parallel \mathrm{right}_{b-s}(U_{i'})\,. \tag{12}$$

Consider any $i, i' \in \{1, \ldots, q\}$. If $s \leq r$, then by $\neg\mathsf{cap}$ we necessarily have $\mathrm{right}_{b-s}(U_i) \neq \mathrm{right}_{b-s}(U_{i'})$. Consider the case that $s > r$ and that $\mathrm{right}_{b-s}(U_i) = \mathrm{right}_{b-s}(U_{i'})$ holds. By the non-occurrence of cap, we necessarily have $\mathrm{left}_s(U_i) \neq \mathrm{left}_s(U_{i'})$. By $2^{-\varepsilon}$-universality of G, the two queries satisfy above equation with probability at most $2^{-\varepsilon}$.

We thus have to count the number of possible choices $i, i' \in \{1, \ldots, q\}$ such that $\mathrm{right}_{b-s}(U_i) = \mathrm{right}_{b-s}(U_{i'})$. Consider any i' (q possibilities). Recall that we assumed that the keyless part of F^p_K is evaluated by the adversary. By $\neg\mathsf{cap}$, any construction query must necessarily be formed by making forward evaluations of p starting from 0^b, and none of these collide with an earlier primitive query. This, particularly, means that the last evaluation of p in the keyless part of each construction query is a forward query. By $\neg\mathsf{mc}_{\mathsf{right}}$, there are at most $\nu_{\mathsf{right}} = \mu^{2(N-q)}_{b-s,s}$ possible paths from 0^b.

By eliminating symmetric cases, the number of possible (i, i')'s with $\mathrm{right}_{b-s}(U_i) = \mathrm{right}_{b-s}(U_{i'})$ is at most $\mu^{2(N-q)}_{b-s,s} \cdot q/2$, and we obtain that there exist i, i' such that $V_i = V_{i'}$ with probability at most

$$\frac{\mu^{2(N-q)}_{b-s,s} \cdot q/2}{2^{\varepsilon}}\,.$$

For the second event of $\mathsf{coll}_{\mathsf{cc}}$, as the values W_i are randomly generated from $\{0,1\}^b$, there exist i, i' with $W_i = W_{i'}$ with probability at most $\binom{q}{2}/2^b \leq q^2/2^b$.

We thus obtain

$$\mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cc}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{right}})\right) \leq \frac{\mu^{2(N-q)}_{b-s,s} \cdot q/2}{2^{\varepsilon}} + \frac{q^2}{2^b}\,.$$

$\mathsf{coll}_{\mathsf{cp}}$. Fix any primitive query $(\mathsf{dir}_j, X_j, Y_j)$ ($N - q$ possibilities). We consider both equations of the event separately. First consider equation $V_i = X_j$:

- If $\mathsf{dir}_j = +$, the equation $V_i = X_j$ is equivalent to stating that

  $$G(K, \mathrm{left}_s(U_i)) \parallel \mathrm{right}_{b-s}(U_i) = \mathrm{left}_s(X_j) \parallel \mathrm{right}_{b-s}(X_j)\,.$$

  As for the analysis $\mathsf{coll}_{\mathsf{cc}}$, by $\neg\mathsf{cap}$ and $\neg\mathsf{mc}_{\mathsf{right}}$, there are at most $\nu_{\mathsf{right}} = \mu^{2(N-q)}_{b-s,s}$ construction queries with $\mathrm{right}_{b-s}(U_i) = \mathrm{right}_{b-s}(X_j)$. By $2^{-\delta}$-uniformity of G, any of these satisfies above equation with probability at most $2^{-\delta}$.

- If $\mathsf{dir}_j = -$, the equation happens with probability at most $\mu^{2(N-q)}_{b-s,s}/2^{\delta}$ as it is symmetric to the case of $V_i = X_j$ in forward queries.

Next consider equation $W_i = Y_j$:

- If $\mathsf{dir}_j = -$, the equation $W_i = Y_j$ is equivalent to stating that

  $$T_i \parallel Z_i = \mathrm{left}_s(Y_j) \parallel \mathrm{right}_{b-s}(Y_j)\,.$$

  By $\neg\mathsf{mc}_{\mathsf{tag}}$, there are at most $\nu_{\mathsf{tag}} = \mu^q_{t,b-t}$ construction queries with $T_i = \mathrm{left}_s(Y_j)$. As $Z_i \xleftarrow{\$} \{0,1\}^{b-t}$, any of these satisfies above equation with probability at most $1/2^{b-t}$.

- If $\mathsf{dir}_j = +$, the equation $W_i = Y_j$ happens with probability at most $q/(2^b-(j-1)) \leq 2q/2^b$, using that $j-1 \leq 2^{b-1}$.

Aggregating both cases, we obtain

$$\mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cp}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{right}} \vee \mathsf{mc}_{\mathsf{tag}})\right) \leq \frac{\mu_{b-s,s}^{2(N-q)} \cdot (N-q)}{2^\delta} + \frac{\mu_{t,b-t}^{q} \cdot (N-q)}{2^{b-t}} + \frac{2q(N-q)}{2^b}\,.$$

Conclusion. Summing the individual terms, we obtain for (11) that

$$\begin{aligned}\mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right) &\leq \frac{2(N-q)^2}{2^c} + \frac{q^2+2q(N-q)}{2^b} \\ &\quad + \frac{\mu_{b-s,s}^{2(N-q)} \cdot (N-q/2+1)}{2^{\min\{\delta,\varepsilon\}}} + \frac{\mu_{t,b-t}^{q} \cdot (N-q+1)}{2^{b-t}} \\ &\leq \frac{2N^2}{2^c} + \frac{\mu_{b-s,s}^{2(N-q)} \cdot N}{2^{\min\{\delta,\varepsilon\}}} + \frac{\mu_{t,b-t}^{q} \cdot N}{2^{b-t}}\,, \end{aligned} \tag{13}$$

where we used that $\min\{\delta,\varepsilon\} \leq s$ and $q \geq 2$.

5.1.5 Ratio for Good Views

Consider any good view $v \in \mathcal{V}_{\mathrm{good}}$. We will prove that $\mathbf{Pr}\left(D_{\mathcal{O}} = v\right) \geq \mathbf{Pr}\left(D_{\mathcal{P}} = v\right)$.

Real World. We start with the real world $\mathcal{O} = (F_K^p, p)$. The view v_p corresponds to exactly $N-q$ distinct input/output tuples of p. For any construction query P_i in the extended view v_c', the unkeyed part of the evaluation of $F_K^p(M)$ is included in v_p. The keyed evaluation of p is the tuple (V_i, W_i) defined in (8)-(9). By the non-occurrence of $\mathsf{coll}_{\mathsf{cp}}$, it is different from any tuple in v_p, and by the non-occurrence of $\mathsf{coll}_{\mathsf{cc}}$ it is different from any earlier tuple $(V_{i'}, W_{i'})$ with $i' < i$. Therefore, the good view $v = (v_c', v_p, K)$ corresponds to *exactly* N input/output tuples of p and one random key K. Therefore, we obtain:

$$\mathbf{Pr}\left(D_{\mathcal{O}} = v\right) = \frac{1}{(2^b)_N} \cdot \frac{1}{2^k}\,. \tag{14}$$

Ideal World. We next consider the ideal world $\mathcal{P} = (R_{*,t}, p)$. The view v_p corresponds to exactly $N-q$ distinct input/output tuples of p. The extended view v_c' consists of q outputs $T_1, \ldots, T_q$ of $R_{*,t}$ and q dummy values $Z_1, \ldots, Z_q \xleftarrow{\$} \{0,1\}^{b-t}$. Finally, the key K is random as before. Therefore, we obtain:

$$\mathbf{Pr}\left(D_{\mathcal{P}} = v\right) = \frac{1}{(2^b)_{N-q}} \cdot \frac{1}{(2^t)^q} \cdot \frac{1}{(2^{b-t})^q} \cdot \frac{1}{2^k} = \frac{1}{(2^b)_{N-q}} \cdot \frac{1}{2^{bq}} \cdot \frac{1}{2^k}\,. \tag{15}$$

Bounding the Ratio. Combining (14) and (15), we obtain the following for any good view $v \in \mathcal{V}_{\mathrm{good}}$:

$$\frac{\mathbf{Pr}\left(D_{\mathcal{O}} = v\right)}{\mathbf{Pr}\left(D_{\mathcal{P}} = v\right)} = \frac{2^{bq}}{(2^b-(N-q))_q} \geq 1\,. \tag{16}$$

5.1.6 Conclusion

The H-coefficient technique of Lemma 1 gives, using (13) and (16):

$$\Delta_{\mathcal{A}}\left(F_K^p, p\,;\, R_{*,t}, p\right) \leq \frac{2N^2}{2^c} + \frac{\mu_{b-s,s}^{2(N-q)} \cdot N}{2^{\min\{\delta,\varepsilon\}}} + \frac{\mu_{t,b-t}^{q} \cdot N}{2^{b-t}}\,.$$

5.2 Proof of Corollary 1

Let $K \xleftarrow{\$} \{0,1\}^k$ and $p \xleftarrow{\$} \mathsf{perm}(b)$. Let $R_{*,t} : \{0,1\}^* \to \{0,1\}^t$ be a function that generates a random t-bit response for each new input. Consider any adversary $\mathcal{A}$, whose goal is to maximize

$$\Delta_{\mathcal{A}}\left(F_K^p, p\ ;\ R_{*,t}, p\right)\ , \tag{17}$$

As a first step, we replace G_K by a function $R_s : \{0,1\}^s \to \{0,1\}^s$ that generates a random s-bit response for each new input. Denote the resulting suffix keyed sponge construction by F^{p,R_s}. By a straightforward reduction,

$$\begin{aligned}\Delta_{\mathcal{A}}\left(F_K^p, p\ ;\ R_{*,t}, p\right) &\le \Delta_{\mathcal{A}}\left(F^{p,R_s}, p\ ;\ R_{*,t}, p\right) + \Delta_{\mathcal{A}'}\left(G_K\ ;\ R_s\right) \\ &= \Delta_{\mathcal{A}}\left(F^{p,R_s}, p\ ;\ R_{*,t}, p\right) + \mathbf{Adv}_G^{\mathrm{prf}}(\mathcal{A}')\,, \end{aligned} \tag{18}$$

where $\mathcal{A}'$ is some adversary with construction complexity q, primitive complexity 0 (as G does not depend on p), and time complexity $\tau' \approx \tau$.

The remaining distance $\Delta_{\mathcal{A}}\left(F^{p,R_s}, p\ ;\ R_{*,t}, p\right)$ of (18) is bounded using Theorem 2, noting that R_s is an instance of a random function family, which is 2^{-s}-uniform and 2^{-s}-universal (see Section 2.3).

6 Leakage Resilience of Suffix Keyed Sponge

We will demonstrate how Theorem 2 carries over to leakage resilience. We first describe the model of leakage resilient PRFs in Section 6.1. The leakage resilience of the suffix keyed sponge is stated in Section 6.2.

6.1 Leakage Resilient PRF Security

We transform the definition of PRF security of Section 2.2 to security in case of leakage resilience. We will restrict our focus to non-adaptive $\mathcal{L}$-resilience of Dodis and Pietrzak [26], where the adversary receives leakage under any leakage $L \in \mathcal{L}$ of the scheme under investigation. We adopt the well-established approach [3, 26, 28, 44, 46, 49] where the adversary has access to a leak-free version of the construction which it has to distinguish from random, and a leaky version, which it may use to obtain leakage.

Let $b, k, t, \lambda \in \mathbb{N}$ and $m \in \mathbb{N} \cup \{*\}$. Consider a function $F : \{0,1\}^k \times \{0,1\}^m \to \{0,1\}^t$ that internally uses a permutation $p \in \mathsf{perm}(b)$. Let $\mathcal{L} = \{L \colon \{0,1\}^s \times \{0,1\}^{b-t} \to \{0,1\}^\lambda\}$ be a class of leakage functions independent of p. For any leakage function L, define by $[F_K^p]_L$ an evaluation of F that leaks $L(\mathrm{left}_s(V), \mathrm{right}_{b-t}(W))$, where V and W are the state before and after the last evaluation of p in F_K^p (see Figure 1). Note that, indeed, $\mathrm{left}_s(V)$ and $\mathrm{right}_{b-t}(W)$ are the only pieces of information of the states of F_K^p that are a priori unknown to an adversary. The function gives the same leakage whenever the inputs are the same, but possibly different leakages for different inputs. The non-adaptive leakage resilient pseudorandom function (NALR-PRF) security of F against an adversary $\mathcal{A}$ is defined as

$$\mathbf{Adv}_F^{\text{nalr-prf}}(\mathcal{A}) = \max_{L \in \mathcal{L}} \Delta_{\mathcal{A}}\left([F_K^p]_L, F_K^p, p\ ;\ [F_K^p]_L, R_{m,t}, p\right)\ , \tag{19}$$

where the randomness is taken over the random drawing of $K \xleftarrow{\$} \{0,1\}^k$, the random drawing of $p \xleftarrow{\$} \mathsf{perm}(b)$, the definition of a function $R_{m,t} : \{0,1\}^m \to \{0,1\}^t$ that generates a random t-bit response for each new input, and the random coins of $\mathcal{A}$.

The adversary never repeats construction queries (both to the leaky and leak-free oracle). The adversary is typically bounded by three types of complexities: q denotes the construction complexity, the number of queries it may make to $([F_K^p]_L, F_K^p)$ or $([F_K^p]_L, R_{m,t})$; N

denotes the total primitive complexity, which accounts for the total number of evaluations of p (either direct or via the real-world construction oracle); τ denotes the time complexity.

6.2 Security of Suffix Keyed Sponge under Leakage

We will prove that the suffix keyed sponge is a leakage resilient PRF under the assumption that p is a random permutation and G is a $2^{-\delta}$-uniform and $2^{-\varepsilon}$-universal function that is strongly protected against side-channel attacks. Stated differently, we inherit the assumptions of Theorem 2, and *in addition* require that G is strongly protected. In particular, G is assumed to be secure against key recovery attacks through side-channel attacks.

The assumption that G is strongly protected is well-established; see [8, 10, 32] among others. It allows us to abstract the leakage incurred by G and to reason solely on the leakage that an adversary learns from the evaluations of p within the suffix keyed sponge. For instance, G could be an "adjusted ideal extendable output function (AIXIF)" in light of the formalism of Dobraunig and Mennink [25]. Simply said, an AIXIF is the ideal equivalent of a leakage resilient keyed duplex construction, and Dobraunig and Mennink proved that the keyed duplex is indistinguishable from an AIXIF (under certain conditions) even if the construction leaks upon each duplexing call. Such a function would, indeed, do the job in the instantiation of G, as an AIXIF leaks no useful information. Clearly, upon instantiation of the AIXIF as a leakage resilient keyed duplex [25], one has to take into account the distinguishing advantage with respect to the AIXIF.

We are ready to state the theorem on the leakage resilience of the suffix keyed sponge.

Theorem 3. *Let $b, c, r, k, s, t, \lambda \in \mathbb{N}$ such that $c + r = b$ and $k, s, t \leq b$. Consider the suffix keyed sponge of Section 3 based on random permutation $p \xleftarrow{\$} \mathsf{perm}(b)$ and a function $G : \{0,1\}^k \times \{0,1\}^s \to \{0,1\}^s$. Assume that G is strongly protected, $2^{-\delta}$-uniform, and $2^{-\varepsilon}$-universal. For any adversary $\mathcal{A}$ with construction complexity $q \geq 2$ and primitive complexity $N \leq 2^{b-1}$,*

$$\mathbf{Adv}_F^{\text{nalr-prf}}(\mathcal{A}) \leq \frac{2N^2}{2^c} + \frac{\mu_{s,b-s}^{2(N-q)}}{2^{b-s}} + \frac{\mu_{b-s,s}^{2(N-q)} \cdot N}{2^{\min\{\delta,\varepsilon\} - \mu_{s,b-s}^{2(N-q)}\lambda}} + \frac{\mu_{t,b-t}^{2q} \cdot N}{2^{b-t-\lambda}} .$$

The proof is given in Section 6.3.

6.3 Proof of Theorem 3

The proof is a mere extension of that of Section 5.1, but as the adversary is given access to an additional oracle, and gets leakage for that oracle, care must be taken in the formal application of the H-coefficient technique. In current proof, we highlight the changes.

Let $K \xleftarrow{\$} \{0,1\}^k$ and $p \xleftarrow{\$} \mathsf{perm}(b)$. Let $R_{*,t} : \{0,1\}^* \to \{0,1\}^t$ be a function that generates a random t-bit response for each new input. Let $L \in \mathcal{L}$ be any leakage function. Consider any adversary $\mathcal{A}$, whose goal is to maximize

$$\Delta_{\mathcal{A}} \left([F_K^p]_L , F_K^p, p \; ; \; [F_K^p]_L , R_{*,t}, p \right) , \tag{20}$$

We will call $\mathcal{O} := ([F_K^p]_L , F_K^p, p)$ the real world and $\mathcal{P} := ([F_K^p]_L , R_{*,t}, p)$ the ideal world. As before, we assume that $\mathcal{A}$ is deterministic.

6.3.1 Oracle Interaction

As before, the adversary has a total primitive complexity N, meaning that for the *real world* the total number primitive queries (direct or through $[F_K^p]_L$ or F_K^p) does not exceed

N. We again require $\mathcal{A}$ to make all unkeyed primitive queries itself offline, and all primitive evaluations (to p, in both worlds) are summarized in a view

$$v_p = \{(\mathsf{dir}_1, X_1, Y_1), \ldots, (\mathsf{dir}_{N-q}, X_{N-q}, Y_{N-q})\}\,.$$

For construction queries, queries to $[F_K^p]_L$ leak the value $L(\mathrm{left}_s(V), \mathrm{right}_{b-t}(W)) \in \{0,1\}^\lambda$. We will be slightly more generous, and assume that for each query the adversary learns λ bits of $\mathrm{left}_s(V)$, called $\ell_V \in \{0,1\}^\lambda$, *and* λ bits of $\mathrm{right}_{b-t}(W)$, called $\ell_W \in \{0,1\}^\lambda$. For consistency, we simply define $\ell_V = \ell_W = \perp$ if the corresponding query is made to the challenge oracle (F_K^p in the real world or $R_{*,t}$ in the ideal world). All q construction queries (to $([F_K^p]_L, F_K^p)$ in the real world and to $([F_K^p]_L, R_{*,t})$ in the ideal world) are summarized in a view

$$v_c = \{(P_1, T_1, \ell_{V1}, \ell_{W1}), \ldots, (P_q, T_q, \ell_{Vq}, \ell_{Wq})\}\,.$$

(One can deduce from (ℓ_{Vi}, ℓ_{Wi}) whether the i-th query was made to the leaky or leak-free oracle.) After the conversation of $\mathcal{A}$ with its oracle, but before it outputs its decision bit, we reveal the key K and the values Z_i as before, but now the definition of these values Z_i is slightly more delicate. For queries to $[F_K^p]_L$ (in either world) or to F_K^p (in the real world only), the values Z_i are the values that are truncated at the end of the construction query; for $R_{*,t}$, these are random values $Z_i \xleftarrow{\$} \{0,1\}^{b-t}$. The values Z_i are appended to v_c:

$$v_c' = \{(P_1, T_1, Z_1, \ell_{V1}, \ell_{W1}), \ldots, (P_q, T_q, Z_q, \ell_{Vq}, \ell_{Wq})\}\,.$$

The aggregate view is defined as $v = (v_c', v_p, K)$.

Note that the definitions of U_i, V_i, and W_i (see also (8)-(9)) carry over. In particular, if the i-th query was made to the leaky oracle $[F_K^p]_L$, the leakages ℓ_{Vi} and ℓ_{Wi} are consistent with the values V_i and W_i.

6.3.2 Bad Views

We inherit the bad events of Section 5.1, but now with thresholds $\nu_{\mathsf{right}} = \mu_{b-s,s}^{2(N-q)}$ and $\nu_{\mathsf{tag}} = \mu_{t,b-t}^{2q}$. We furthermore need to split event $\mathsf{coll}_{\mathsf{cp}}$ into input and output collisions:

- $\mathsf{coll}_{\mathsf{cp\text{-}in}}$: there exist $i \in \{1,\ldots,q\}$ and $j \in \{1,\ldots,N-q\}$ with
$$V_i = X_j\,;$$
- $\mathsf{coll}_{\mathsf{cp\text{-}out}}$: there exist $i \in \{1,\ldots,q\}$ and $j \in \{1,\ldots,N-q\}$ with
$$W_i = Y_j\,.$$

The reason for this separation is that in the proof of Section 5.1, $\mathsf{coll}_{\mathsf{cp\text{-}out}}$ is proven under the non-occurrence of $\mathsf{mc}_{\mathsf{tag}}$, but as the view v_c' also includes queries to $[F_K^p]_L$, one can only upper bound the probability that $\mathsf{mc}_{\mathsf{tag}}$ occurs under the condition that $\mathsf{coll}_{\mathsf{cp\text{-}in}}$ does not occur.

Finally, we define a helping event used to bound the amount of leakage. Let $\nu_{\mathsf{left}} = \mu_{s,b-s}^{2(N-q)}$ be a threshold.

- $\mathsf{mc}_{\mathsf{left}}$: there exist distinct $j_1, \ldots, j_{\nu_{\mathsf{left}}+1} \in \{1,\ldots,N-q\}$ with $\mathsf{dir}_j = +$ and
$$\mathrm{left}_s(Y_{j_1}) = \cdots = \mathrm{left}_s(Y_{j_{\nu_{\mathsf{left}}+1}})\,.$$

The event $\mathsf{mc}_{\mathsf{left}}$ is motivated by the observation that any time G is evaluated for the same input $\mathrm{left}_s(U_i)$ but alongside a different inner part $\mathrm{right}_{b-s}(U_i)$ (see (8)), the adversary may learn different information about $\mathrm{left}_s(V_i) = G(K, \mathrm{left}_s(U_i))$. The bad event $\mathsf{mc}_{\mathsf{left}}$ upper bounds the maximum number of times G is invoked on the same input $\mathrm{left}_s(U_i)$.

We write $\mathsf{cap} = \mathsf{cap}_{\mathsf{fwd}} \vee \mathsf{cap}_{\mathsf{inv}}$ and

$$\mathsf{bad} = \mathsf{cap} \vee \mathsf{mc}_{\mathsf{left}} \vee \mathsf{mc}_{\mathsf{right}} \vee \mathsf{mc}_{\mathsf{tag}} \vee \mathsf{coll}_{\mathsf{cc}} \vee \mathsf{coll}_{\mathsf{cp\text{-}in}} \vee \mathsf{coll}_{\mathsf{cp\text{-}out}}\,. \tag{21}$$

6.3.3 Probability of Bad View

Our goal is to bound the probability that a bad view is generated in the ideal world, $\mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right)$. This probability equals the probability that a view v generated by $D_{\mathcal{P}}$ sets bad. By basic probability theory,

$$\begin{aligned}\mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right) &= \mathbf{Pr}\left(\mathsf{bad}\right) \\ &\leq \mathbf{Pr}\left(\mathsf{cap}\right) + \mathbf{Pr}\left(\mathsf{mc}_{\mathsf{left}}\right) + \mathbf{Pr}\left(\mathsf{mc}_{\mathsf{right}}\right) + \mathbf{Pr}\left(\mathsf{mc}_{\mathsf{tag}} \mid \neg\mathsf{coll}_{\mathsf{cp\text{-}in}}\right) \\ &+ \mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cc}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{right}})\right) \\ &+ \mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cp\text{-}in}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{left}} \vee \mathsf{mc}_{\mathsf{right}})\right) \\ &+ \mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cp\text{-}out}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{tag}} \vee \mathsf{coll}_{\mathsf{cp\text{-}in}})\right) . \end{aligned} \tag{22}$$

Much of the analysis of the proof of Section 5.1 carries over verbatim, but there are some differences, most notably in the analysis of $\mathsf{mc}_{\mathsf{left}}$ (the new bad event) and $\mathsf{coll}_{\mathsf{cp}}$ (leakages must be taken into account).

cap. The analysis is identical to the one in Section 5.1, and we obtain

$$\mathbf{Pr}\left(\mathsf{cap}\right) = \mathbf{Pr}\left(\mathsf{cap}_{\mathsf{fwd}} \vee \mathsf{cap}_{\mathsf{inv}}\right) \leq \sum_{j=1}^{N-q} \frac{2(2j-1)}{2^c} = \frac{2(N-q)^2}{2^c} .$$

$\mathsf{mc}_{\mathsf{left}}$. The analysis is symmetric to the one of $\mathsf{mc}_{\mathsf{right}}$ in Section 5.1, and we obtain

$$\mathbf{Pr}\left(\mathsf{mc}_{\mathsf{left}}\right) \leq \frac{\mu_{s,b-s}^{2(N-q)}}{2^{b-s}} .$$

$\mathsf{mc}_{\mathsf{right}}$. The analysis is identical to the one in Section 5.1, and we obtain

$$\mathbf{Pr}\left(\mathsf{mc}_{\mathsf{right}}\right) \leq \frac{\mu_{b-s,s}^{2(N-q)}}{2^{s}} .$$

$\mathsf{mc}_{\mathsf{tag}}$. Note that the tuples in v'_c either come from $R_{*,t}$ or $[F_K^p]_L$. In the former case, the values W_i are randomly generated from $\{0,1\}^b$. In the latter case, by $\neg\mathsf{coll}_{\mathsf{cp\text{-}in}}$, they are generated uniformly randomly bar repetition. The event $\mathsf{mc}_{\mathsf{tag}}$ is thus a balls-and-bins experiment with q balls randomly (with some distribution) thrown into 2^t bins, in such a way that any of the bins contains more than ν_{tag} balls. Note that the distribution of balls satisfies the condition of (3). By definition of $\nu_{\mathsf{tag}} = \mu_{t,b-t}^{2q}$, we can resort to the multicollision limit function of Section 2.4 and obtain

$$\mathbf{Pr}\left(\mathsf{mc}_{\mathsf{tag}} \mid \neg\mathsf{coll}_{\mathsf{cp\text{-}in}}\right) \leq \frac{\mu_{t,b-t}^{2q}}{2^{b-t}} .$$

$\mathsf{coll}_{\mathsf{cc}}$. The analysis for the case of $V_i = V_{i'}$ is identical to the one in Section 5.1. Here, we note that, particularly, the probability analysis does not change as it is independent of the leakages.

For the case of $W_i = W_{i'}$, we have to be a bit more careful as v'_c contains queries to $[F_K^p]_L$, where the W_i's are the outputs of a permutation, as well as $R_{*,t}$, where the W_i's are random b-bit values. If both queries i and i' are made to $[F_K^p]_L$, we have $W_i = W_{i'}$ if and only if $V_i = V_{i'}$, but that case was already covered above. If at least one of the two queries is made to $R_{*,t}$, equation $W_i = W_{i'}$ holds with probability at most $1/(2^b - (j-1)) \leq 2/2^b$, using that $j - 1 \leq 2^{b-1}$. There are at most $\binom{q}{2} \leq q^2/2$ choices $i, i' \in \{1, \ldots, q\}$ such that at least one of the two queries is made to $R_{*,t}$.

We obtain

$$\mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cc}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{right}})\right) \leq \frac{\mu_{b-s,s}^{2(N-q)} \cdot q/2}{2^{\varepsilon}} + \frac{q^2}{2^b}\,.$$

$\mathsf{coll}_{\mathsf{cp\text{-}in}}$. Fix any primitive query $(\mathsf{dir}_j, X_j, Y_j)$. We make a distinction depending on the direction of this query.

If $\mathsf{dir}_j = +$, the equation $V_i = X_j$ is equivalent to stating that

$$G(K, \mathrm{left}_s(U_i)) \parallel \mathrm{right}_{b-s}(U_i) = \mathrm{left}_s(X_j) \parallel \mathrm{right}_{b-s}(X_j)\,.$$

As for the analysis $\mathsf{coll}_{\mathsf{cc}}$, by $\neg\mathsf{cap}$ and $\neg\mathsf{mc}_{\mathsf{right}}$, there are at most $\nu_{\mathsf{right}} = \mu_{b-s,s}^{2(N-q)}$ construction queries with $\mathrm{right}_{b-s}(U_i) = \mathrm{right}_{b-s}(X_j)$. By $\neg\mathsf{mc}_{\mathsf{left}}$, the adversary might have seen at most $\nu_{\mathsf{left}} = \mu_{s,b-s}^{2(N-q)}$ leakages for $G(K, \mathrm{left}_s(U_i))$. As G itself is $2^{-\delta}$-uniform, any of the queries satisfies above equation with probability at most $2^{-(\delta - \mu_{s,b-s}^{2(N-q)}\lambda)}$.

If $\mathsf{dir}_j = -$, the case of equation $V_i = X_j$ is symmetric, as in the proof of Section 5.1, but now with probability $2^{-\delta}$ as no leakage can be exploited.

Aggregating both cases, we obtain

$$\mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cp\text{-}in}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{left}} \vee \mathsf{mc}_{\mathsf{right}})\right) \leq \frac{\mu_{b-s,s}^{2(N-q)} \cdot (N-q)}{2^{\delta - \mu_{s,b-s}^{2(N-q)}\lambda}}\,.$$

$\mathsf{coll}_{\mathsf{cp\text{-}out}}$. Fix any primitive query $(\mathsf{dir}_j, X_j, Y_j)$. We make a distinction depending on the direction of this query.

If $\mathsf{dir}_j = -$, the equation $W_i = Y_j$ is equivalent to stating that

$$T_i \parallel Z_i = \mathrm{left}_s(Y_j) \parallel \mathrm{right}_{b-s}(Y_j)\,.$$

By $\neg\mathsf{mc}_{\mathsf{tag}}$, there are at most $\nu_{\mathsf{tag}} = \mu_{t,b-t}^{2q}$ construction queries with $T_i = \mathrm{left}_s(Y_j)$. As $Z_i \xleftarrow{\$} \{0,1\}^{b-t}$, but the adversary may have received λ bits of leakage of Z_i any of these satisfies above equation with probability at most $1/2^{b-t-\lambda}$.

If $\mathsf{dir}_j = +$ and the i-th query is to $[F_K^p]_L$, the equation cannot be set by condition $\neg\mathsf{coll}_{\mathsf{cp\text{-}in}}$. If the i-th query is to $R_{*,t}$, the equation $W_i = Y_j$ happens with probability at most $q/(2^b - (j-1)) \leq 2q/2^b$, using that $j - 1 \leq 2^{b-1}$.

Aggregating both cases, we obtain

$$\mathbf{Pr}\left(\mathsf{coll}_{\mathsf{cp\text{-}out}} \mid \neg(\mathsf{cap} \vee \mathsf{mc}_{\mathsf{tag}} \vee \mathsf{coll}_{\mathsf{cp\text{-}in}})\right) \leq \frac{\mu_{t,b-t}^{2q} \cdot (N-q)}{2^{b-t-\lambda}} + \frac{2q(N-q)}{2^b}\,.$$

Conclusion. Summing the individual terms, we obtain for (22) that

$$\mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right) \leq \frac{2N^2}{2^c} + \frac{\mu_{s,b-s}^{2(N-q)}}{2^{b-s}} + \frac{\mu_{b-s,s}^{2(N-q)} \cdot N}{2^{\min\{\delta,\varepsilon\} - \mu_{s,b-s}^{2(N-q)}\lambda}} + \frac{\mu_{t,b-t}^{2q} \cdot N}{2^{b-t-\lambda}}\,, \tag{23}$$

where we used that $\min\{\delta, \varepsilon\} \leq s$ and $q \geq 2$.

6.3.4 Ratio for Good Views

Consider any good view $v \in \mathcal{V}_{\mathrm{good}}$. We will prove that $\mathbf{Pr}\left(D_{\mathcal{O}} = v\right) \geq \mathbf{Pr}\left(D_{\mathcal{P}} = v\right)$.

Real World. For the real world, the analysis is exactly as that of Section 5.1, using that the leakage values $\ell_{V\,i}, \ell_{W\,i}$ in v'_c – the only difference compared with the analysis in Section 5.1 – contain no additional information after the values Z_i and K have been revealed. Therefore, we obtain:

$$\mathbf{Pr}\left(D_{\mathcal{O}} = v\right) = \frac{1}{(2^b)_N} \cdot \frac{1}{2^k}\,. \tag{24}$$

Ideal World. For the ideal world, the analysis is more involved, as construction queries may be to the real but leaky oracle $[F_K^p]_L$ or to the ideal oracle $R_{*,t}$. Assume that v contains $q' \leq q$ queries to the ideal oracle. By independence of $R_{*,t}$ from $([F_K^p]_L\,, p)$, we can resort to the analysis for the real world for all elements in v except for the q' queries to $R_{*,t}$. For these q' queries, we can observe that the outputs $T_i \xleftarrow{\$} \{0,1\}^t$ and dummy values $Z_i \xleftarrow{\$} \{0,1\}^{b-t}$ are randomly generated, as before. Therefore, we obtain:

$$\mathbf{Pr}\left(D_{\mathcal{P}} = v\right) = \frac{1}{(2^b)_{N-q'}} \cdot \frac{1}{(2^t)^{q'}} \cdot \frac{1}{(2^{b-t})^{q'}} \cdot \frac{1}{2^k} = \frac{1}{(2^b)_{N-q'}} \cdot \frac{1}{2^{bq'}} \cdot \frac{1}{2^k}\,. \tag{25}$$

Bounding the Ratio. Combining (24) and (25), we obtain the following for any good view $v \in \mathcal{V}_{\text{good}}$:

$$\frac{\mathbf{Pr}\left(D_{\mathcal{O}} = v\right)}{\mathbf{Pr}\left(D_{\mathcal{P}} = v\right)} = \frac{2^{bq'}}{(2^b - (N-q'))_{q'}} \geq 1\,. \tag{26}$$

6.3.5 Conclusion

The H-coefficient technique of Lemma 1 gives, using (23) and (26):

$$\Delta_{\mathcal{A}}\left([F_K^p]_L\,, F_K^p, p\,;\; [F_K^p]_L\,, R_{m,t}, p\right) \leq \frac{2N^2}{2^c} + \frac{\mu_{s,b-s}^{2(N-q)}}{2^{b-s}} + \frac{\mu_{b-s,s}^{2(N-q)} \cdot N}{2^{\min\{\delta,\varepsilon\} - \mu_{s,b-s}^{2(N-q)}\lambda}} + \frac{\mu_{t,b-t}^{2q} \cdot N}{2^{b-t-\lambda}}\,.$$

As this holds for any leakage function $L \in \mathcal{L}$, this completes the proof.

7 Application

One clear application of our bound is the message authentication part of ISAP [22]. The MAC of ISAP is based on a suffix keyed sponge construction that uses a 400-bit permutation with rate $r = 144$, capacity $c = 256$, and $s = t = 128$. According to Daemen et al. [19, Equation (32)], we compute $\mu_{t,b-t}^q = \mu_{128,272}^{2^{128}} \leq 80$ and $\mu_{b-s,s}^{2(N-q)} = \mu_{272,128}^{2^{129}} \leq 3$. If we assume that G is 2^{-128}-uniform, and 2^{-128}-universal, then an attacker has advantage

$$\mathbf{Adv}_F^{\text{prf}}(\mathcal{A}) \leq \frac{2N^2}{2^{256}} + \frac{3N}{2^{128}} + \frac{80N}{2^{272}}\,.$$

If we consider a smaller permutation of, e.g., 320 bits like the one used for Ascon [23,24] and aim to achieve a similar security level, we have to change the rate to $r = 64$, but can leave $c = 256$ and $s = t = 128$. Then, we get $\mu_{t,b-t}^q = \mu_{128,192}^{2^{128}} \leq 67$ and $\mu_{b-s,s}^{2(N-q)} = \mu_{192,128}^{2^{129}} \leq 5$. In this case, we get a bound

$$\mathbf{Adv}_F^{\text{prf}}(\mathcal{A}) \leq \frac{2N^2}{2^{256}} + \frac{5N}{2^{128}} + \frac{67N}{2^{272}}\,.$$

Comparing both construction, we see that they achieve a similar security level, namely 128 bits up to small constant factors. The biggest difference in both bounds is the factor $\mu_{b-s,s}^{2(N-q)}$, which changes from 3 to 5. This is a marginal change considering that the inner part of the state for the 320-bit construction is just 192 bits, which is significantly smaller than the 256-bit capacity one would need in order to rely on the indifferentiability result of the sponge [11].

In previous examples, we upper bounded the values of $\mu_{t,b-t}^{q}$ and $\mu_{b-s,s}^{2(N-q)}$ by upper bounding the query complexities q and N to 2^{128}. However, if we consider the leakage resilience bound of Theorem 3, doing the same thing would distort the advantage, since the advantage an attacker gains by exploiting the leakage is tightly coupled with $\mu_{s,b-s}^{2(N-q)}$. Hence, we have to consider $\mu_{s,b-s}^{2(N-q)}$ as a function, which grows for a fixed b and s if $2(N-q)$ increases. To get some insight what this means for practical cases, let us consider an attacker with complexities $N = 2^{65}$ and $q = 2^{64}$. Furthermore, assume that $r = 64$, $c = 256$, and $s = t = 128$ as in the example above. In this case, we get $\mu_{s,b-s}^{2(N-q)} = \mu_{t,b-t}^{2q} = \mu_{128,192}^{2^{65}} \leq 5$ and $\mu_{b-s,s}^{2(N-q)} = \mu_{192,128}^{2^{65}} \leq 3$. If we assume that G is strongly protected, 2^{-128}-uniform, and 2^{-128}-universal, we get

$$\mathbf{Adv}_{F}^{\text{nalr-prf}}(\mathcal{A}) \leq \frac{1}{2^{61}} + \frac{5}{2^{192}} + \frac{3 \cdot 2^{65}}{2^{128-5\lambda}} + \frac{5 \cdot 2^{65}}{2^{192-\lambda}} ,$$

meaning that in this case λ can be up to 12 bits while still having an advantage smaller than one.

8 Conclusion and Discussion

Towards Leakage Resilient Message Authentication. In this paper, we provided the – to the best of our knowledge – first dedicated analysis of the suffix keyed sponge and bounded the advantage of an attacker in distinguishing this construction from a PRF in the black-box security model and in the leakage resilience model. However, we stress that such a leakage resilient PRF not always results in a leakage resilient MAC.

This remark seems counter-intuitive, but recently, Berti et al. [7] and Guo et al. [32] detailed a problem with MACs during the verification. In most cases, this is done by computing a tag T' from the transmitted data M by using the PRF and compare it with the transmitted tag T. If this comparison is not done in a leakage resilient manner, information about T' can leak. Thus, an attacker could learn information about T' by repeated queries of the same M with different T. If all the information about T' is learned, a forgery $M \parallel T'$ can be created.

In their schemes, Berti et al. [7] and Guo et al. [32] overcome this issue in their hash-then-MAC construction by computing the inverse of a perfectly protected tweakable block cipher on the transmitted T to compare directly the outcome of the hash function, so no information about T' can leak. The approach is also used in Spook [6]. Clearly, this can also be done for a MAC based on a suffix keyed sponge if we do not use T directly, but also put T in a perfectly protected block cipher and transmit $E_K(T)$, and also doing the comparison via the inverse. Another way to protect the comparison is, as suggested by Dobraunig et al. [21], via one additional permutation call during the verification. In this case, T and T' are computed and transmitted as normal, but instead of a direct comparison, e.g., $\text{left}_t(p(T'\|0^{b-t}))$ is compared with $\text{left}_t(p(T\|0^{b-t}))$ first. In such a comparison, quite in a similar manner to [7, 32], the comparison only reveals mostly useless information, namely only at most t out of $b-t$ bits of $p(T'\|0^{b-t})$, which gives no advantage in retrieving T'.

Estimation of Multicollision Probabilities. The leakage resilience security bound of Theorem 3 has a term that bounds the probability of a large multicollision in the outer part of the state:

$$\frac{\mu_{s,b-s}^{2(N-q)}}{2^{b-s}}\,.$$

The bound is based on the multicollision limit function bounding from Section 2.4. This bound, however, shows an unexpected behavior if $s = b$, i.e., if one opts to absorb the key over the full state: the bound of Theorem 3 becomes meaningless. However, this is purely due to generality. For specific cases, in particular when $b - s \ll c$, other choices for the threshold value ν_{left} would give a better bound. For example, in case $b = s$ one would simply not even have collisions in the first place (by assumption that $\neg\mathsf{cap}$ holds).

Suffix Keyed Sponge for Short Messages. The description of the suffix keyed sponge of Section 3 is given in a general manner in such a way that it still resembles as much as possible from the sponge. One can improve the efficiency of the suffix keyed sponge slightly by absorbing more data bits in the first round. Let $a \in \mathbb{N}$ satisfy $a \leq c/2$, and consider an adjustment of the suffix keyed sponge where the first block is of size $r + a$ bits, but all remaining blocks are of size r bits. In the security proofs of both the black-box and the leakage resilience case, this *only affects the definition and analysis of* $\mathsf{cap}_{\mathsf{inv}}$, which is set if an inverse query satisfies $\mathrm{right}_{c-a}(X_j) = 0^{c-a}$. This, in turn, happens with probability at most $2(N-q)/2^{c-a}$. Summarizing, one can increase the rate for the first block by $a \leq c/2$ bits, and this only adds $2(N-q)/2^{c-a}$ to the bound of Theorem 2, Corollary 1, or Theorem 3. This term would not dominate the bound as long as $a \leq c/2$.

Acknowledgments. We thank Joan Daemen, Maria Eichlseder, and Florian Mendel for the fruitful discussions. Christoph Dobraunig is supported by the Austrian Science Fund (FWF): J 4277-N38. Bart Mennink is supported by a postdoctoral fellowship from the Netherlands Organisation for Scientific Research (NWO) under Veni grant 016.Veni.173.017.

References

[1] Andreeva, E., Daemen, J., Mennink, B., Van Assche, G.: Security of Keyed Sponge Constructions Using a Modular Proof Approach. In: Leander, G. (ed.) FSE 2015. LNCS, vol. 9054, pp. 364–384. Springer (2015)

[2] Barreto, P.S.L.M., Naehrig, M.: Pairing-Friendly Elliptic Curves of Prime Order. In: Preneel, B., Tavares, S.E. (eds.) SAC 2005. LNCS, vol. 3897, pp. 319–331. Springer (2005)

[3] Barwell, G., Martin, D.P., Oswald, E., Stam, M.: Authenticated Encryption in the Face of Protocol and Side Channel Leakage. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017. LNCS, vol. 10624, pp. 693–723. Springer (2017)

[4] Bellare, M., Goldreich, O., Mityagin, A.: The Power of Verification Queries in Message Authentication and Authenticated Encryption. Cryptology ePrint Archive, Report 2004/309 (2004)

[5] Bellare, M., Kilian, J., Rogaway, P.: The security of cipher block chaining. In: Desmedt, Y. (ed.) CRYPTO '94. LNCS, vol. 839, pp. 341–358. Springer (1994)

[6] Bellizia, D., Berti, F., Bronchain, O., Cassiers, G., Duval, S., Guo, C., Leander, G., Leurent, G., Levi, I., Momin, C., Pereira, O., Peters, T., Standaert, F.X., Wiemer, F.:

Spook: Sponge-Based Leakage-Resilient Authenticated Encryption with a Masked Tweakable Block Cipher. Submission to NIST Lightweight Cryptography (2019)

[7] Berti, F., Guo, C., Pereira, O., Peters, T., Standaert, F.X.: TEDT, a Leakage-Resilient AEAD mode for High (Physical) Security Applications. Cryptology ePrint Archive, Report 2019/137 (2019)

[8] Berti, F., Koeune, F., Pereira, O., Peters, T., Standaert, F.X.: Ciphertext Integrity with Misuse and Leakage: Definition and Efficient Constructions with Symmetric Primitives. In: Kim, J., Ahn, G.J., Kim, S., Kim, Y., López, J., Kim, T. (eds.) AsiaCCS 2018. pp. 37–50. ACM (2018)

[9] Berti, F., Koeune, F., Pereira, O., Peters, T., Standaert, F.X.: Leakage-Resilient and Misuse-Resistant Authenticated Encryption. Cryptology ePrint Archive, Report 2016/996 (2016)

[10] Berti, F., Pereira, O., Peters, T., Standaert, F.X.: On Leakage-Resilient Authenticated Encryption with Decryption Leakages. IACR Trans. Symmetric Cryptol. 2017(3), 271–293 (2017)

[11] 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 (2008)

[12] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Cryptographic sponge functions (January 2011)

[13] 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 (2011)

[14] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the Security of the Keyed Sponge Construction. Symmetric Key Encryption Workshop (February 2011)

[15] Black, J., Rogaway, P.: A Block-Cipher Mode of Operation for Parallelizable Message Authentication. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 384–397. Springer (2002)

[16] Chang, D., Dworkin, M., Hong, S., Kelsey, J., Nandi, M.: A Keyed Sponge Construction with Pseudorandomness in the Standard Model. NIST SHA-3 Workshop (March 2012)

[17] Chari, S., Jutla, C.S., Rao, J.R., Rohatgi, P.: Towards Sound Approaches to Counteract Power-Analysis Attacks. In: Wiener [48], pp. 398–412

[18] Chen, S., Steinberger, J.P.: Tight Security Bounds for Key-Alternating Ciphers. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 327–350. Springer (2014)

[19] Daemen, J., Mennink, B., Van Assche, G.: Full-State Keyed Duplex with Built-In Multi-user Support. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017. LNCS, vol. 10625, pp. 606–637. Springer (2017)

[20] Daemen, J., Mennink, B., Van Assche, G.: Sound Hashing Modes of Arbitrary Functions, Permutations, and Block Ciphers. IACR Trans. Symmetric Cryptol. 2018(4), 197–228 (2018)

[21] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Mennink, B., Primas, R., Unterluggauer, T.: ISAP v2. Submission to NIST Lightweight Cryptography (2019)

[22] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Unterluggauer, T.: ISAP - Towards Side-Channel Secure Authenticated Encryption. IACR Trans. Symmetric Cryptol. 2017(1), 80–105 (2017)

[23] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1.2. Submission to Round 3 of the CAESAR competition (2016)

[24] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1.2. Submission to NIST Lightweight Cryptography (2019)

[25] Dobraunig, C., Mennink, B.: Leakage Resilience of the Duplex Construction. Cryptology ePrint Archive, Report 2019/225 (2019)

[26] Dodis, Y., Pietrzak, K.: Leakage-Resilient Pseudorandom Functions and Side-Channel Attacks on Feistel Networks. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 21–40. Springer (2010)

[27] Dworkin, M.: NIST SP 800-38B: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication (2005)

[28] Faust, S., Pietrzak, K., Schipper, J.: Practical Leakage-Resilient Symmetric Cryptography. In: Prouff, E., Schaumont, P. (eds.) CHES 2012. LNCS, vol. 7428, pp. 213–232. Springer (2012)

[29] Gazi, P., Pietrzak, K., Tessaro, S.: The Exact PRF Security of Truncation: Tight Bounds for Keyed Sponges and Truncated CBC. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015. LNCS, vol. 9215, pp. 368–387. Springer (2015)

[30] Goldreich, O., Goldwasser, S., Micali, S.: On the Cryptographic Applications of Random Functions. In: Blakley, G.R., Chaum, D. (eds.) CRYPTO '84. LNCS, vol. 196, pp. 276–288. Springer (1984)

[31] Goubin, L., Patarin, J.: DES and Differential Power Analysis (The "Duplication" Method). In: Koç, Ç.K., Paar, C. (eds.) CHES'99. LNCS, vol. 1717, pp. 158–172. Springer (1999)

[32] Guo, C., Pereira, O., Peters, T., Standaert, F.X.: Towards Lightweight Side-Channel Security and the Leakage-Resilience of the Duplex Sponge. Cryptology ePrint Archive, Report 2019/193 (2019)

[33] 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 (2014)

[34] Kocher, P.C., Jaffe, J., Jun, B.: Differential Power Analysis. In: Wiener [48], pp. 388–397

[35] Martin, D.P., Oswald, E., Stam, M., Wójcik, M.: A leakage resilient MAC. In: Groth, J. (ed.) IMACC 2015. LNCS, vol. 9496, pp. 295–310. Springer (2015)

[36] Medwed, M., Standaert, F.X., Joux, A.: Towards Super-Exponential Side-Channel Security with Efficient Leakage-Resilient PRFs. In: Prouff, E., Schaumont, P. (eds.) CHES 2012. LNCS, vol. 7428, pp. 193–212. Springer (2012)

[37] Medwed, M., Standaert, F.X., Nikov, V., Feldhofer, M.: Unknown-Input Attacks in the Parallel Setting: Improving the Security of the CHES 2012 Leakage-Resilient PRF. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016. LNCS, vol. 10031, pp. 602–623 (2016)

[38] Mennink, B.: Key Prediction Security of Keyed Sponges. IACR Trans. Symmetric Cryptol. 2018(4), 128–149 (2018)

[39] Mennink, B., Reyhanitabar, R., Vizár, D.: Security of Full-State Keyed Sponge and Duplex: Applications to Authenticated Encryption. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015. LNCS, vol. 9453, pp. 465–489. Springer (2015)

[40] Naito, Y., Yasuda, K.: New Bounds for Keyed Sponges with Extendable Output: Independence Between Capacity and Message Length. In: Peyrin, T. (ed.) FSE 2016. LNCS, vol. 9783, pp. 3–22. Springer (2016)

[41] Nikova, S., Rechberger, C., Rijmen, V.: Threshold Implementations Against Side-Channel Attacks and Glitches. In: Ning, P., Qing, S., Li, N. (eds.) ICICS 2006. LNCS, vol. 4307, pp. 529–545. Springer (2006)

[42] Nikova, S., Rijmen, V., Schläffer, M.: Secure Hardware Implementation of Nonlinear Functions in the Presence of Glitches. J. Cryptology 24(2), 292–321 (2011)

[43] Patarin, J.: The "Coefficients H" Technique. In: Avanzi, R.M., Keliher, L., Sica, F. (eds.) SAC 2008. LNCS, vol. 5381, pp. 328–345. Springer (2008)

[44] Pietrzak, K.: A Leakage-Resilient Mode of Operation. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 462–482. Springer (2009)

[45] Samwel, N., Daemen, J.: DPA on hardware implementations of Ascon and Keyak. In: CF'17. pp. 415–424. ACM (2017)

[46] Standaert, F.X., Pereira, O., Yu, Y., Quisquater, J.J., Yung, M., Oswald, E.: Leakage Resilient Cryptography in Practice. In: Sadeghi, A.R., Naccache, D. (eds.) Towards Hardware-Intrinsic Security - Foundations and Practice, pp. 99–134. Information Security and Cryptography, Springer (2010)

[47] Taha, M.M.I., Schaumont, P.: Side-channel countermeasure for SHA-3 at almost-zero area overhead. In: HOST 2014. pp. 93–96. IEEE Computer Society (2014)

[48] Wiener, M.J. (ed.): CRYPTO '99, LNCS, vol. 1666. Springer (1999)

[49] Yu, Y., Standaert, F.X., Pereira, O., Yung, M.: Practical leakage-resilient pseudorandom generators. In: Al-Shaer, E., Keromytis, A.D., Shmatikov, V. (eds.) CCS 2010. pp. 141–151. ACM (2010)