Elephant v2

Submitters:

Tim Beyne	KU Leuven and imec-COSIC, Belgium
Yu Long Chen	KU Leuven and imec-COSIC, Belgium
Christoph Dobraunig	Lamarr Security Research, Austria
Bart Mennink	Radboud University, The Netherlands

Corresponding submitter:

Bart Mennink
elephant@cs.ru.nl
+31 24 3652672
Radboud University, The Netherlands
Toernooiveld 212
6525 EC Nijmegen
The Netherlands

May 17, 2021

1 Introduction

We introduce the Elephant authenticated encryption scheme. The mode of Elephant is a nonce-based encrypt-then-MAC construction, where encryption is performed using counter mode and message authentication using a variant of the protected counter sum [10, 77] MAC function. Both modes internally use a cryptographic permutation masked using LFSRs, akin to the masked Even-Mansour construction of Granger et al. [53].

The mode is permutation-based and only evaluates this permutation in the forward direction. As such, there is no need to implement multiple primitives or the inverse of the primitive, unlike in OCB-based [64, 92, 93] authenticated encryption schemes. Furthermore, this allows us to rely and build on the extensive literature of permutations used for sponge-based lightweight hashing [6, 23, 56]. That said, Elephant itself is not sponge-based: on the contrary, it departs from the conventional approach of serial permutation-based authenticated encryption. Elephant is parallelizable by design, easy to implement due to the use of LFSRs for masking (no need for finite field multiplication), and finally, it is efficient due to elegant decisions on how the masking should be performed exactly. A security analysis in the ideal permutation model demonstrates that the mode of Elephant is structurally sound, even in the multi-user setting.

Due to the parallelizability of Elephant, there is no need for instances with a large permutation: we can go as small as 160-bit permutations while still matching the security goals recommended by the NIST lightweight call [85]. In detail, the Elephant scheme consists of three instances:

1. Dumbo: Elephant-Spongent-π[160]. This instance meets the minimum permutation size as dictated by the security analysis: it achieves 112-bit security provided that the online complexity is at most around 2^{46} blocks. This instance is particularly well-suited for hardware, as Spongent [23] itself is;

2. Jumbo: Elephant-Spongent-π[176]. This is a slightly more conservative instance of Elephant: it is based on the same permutation family, yet achieves 127-bit security under the same conditions on the online complexity. We note, in particular, that Spongent-π[176] is ISO/IEC standardized [23, 59];

3. Delirium: Elephant-Keccak-f[200]. This variant is developed more towards software use, although it still performs reasonably well in hardware. Elephant instantiated with Keccak-f[200] also achieves 127-bit security, with a higher bound of around 2^{70} blocks on the online complexity. The permutation is the smallest instance that is specified in the NIST SHA-3 standard [15, 51] that fits our needs.

Dumbo is the primary member of the submission. Dumbo and Jumbo are named after two famous elephants; Delirium is named after a Belgian beer, whose logo is a pink elephant. As each of the permutations is relatively small, all versions of Elephant have a small state size, despite its support for parallelism. The

LFSRs used for masking are tailored to the specific instance, one for each, and are developed to operate well with the specific cryptographic permutation. For example, the LFSRs paired with the Spongent instances have been chosen to minimize the number of XOR operations that have to be performed for a state-update, while the Keccak-based instance has been selected to perform well on software platforms.

We note that the three cryptographic permutations in Elephant can also be used for cryptographic hashing – in fact, Spongent [23] and Keccak [15] themselves are sponges – but due to our quest for small permutations, these cryptographic hash functions cannot meet the 112-, or 127-bit security level guaranteed by our authenticated encryption schemes. In contrast, in order to perform sponge-based hashing with at least 112-bit security, a cryptographic permutation of size at least 225 bits must be used.

1.1 Change Log

Elephant v1.1 only had superficial changes compared to v1.

Elephant v2 replaces the variant of the Wegman-Carter-Shoup [11, 96, 107] MAC function in v1 and v1.1 by a variant of the protected counter sum [10, 77] MAC function. The new version v2 achieves confidentiality and authenticity in the nonce-respecting setting, as v1 and v1.1 did, but *in addition* it achieves authenticity under nonce-reuse. Furthermore, a minor change in the positioning in the masks has been made to make v2 slightly more efficient. The roles of the masks are now $(\cdot, 0)$ for associated data authentication (used to be encryption), $(\cdot, 1)$ for encryption (used to be ciphertext authentication), and $(\cdot, 2)$ for ciphertext authentication (used to be associated data authentication). The security analysis (Appendix B, and in particular Theorem B.3) has been adapted to the new version, and has furthermore been generalized to cover multi-user security. Also the implementation has been updated to the new mode.

2 Algorithmic Specification

The generic Elephant mode is presented in Section 2.2, and the three primitives used within the mode are presented in Sections 2.3-2.5. Before going to the mode, we briefly describe the notation used in 2.1.

2.1 Notation

For $n \in \mathbb{N}$, we let $\{0,1\}^n$ denote the set of n-bit strings and $\{0,1\}^*$ the set of arbitrarily length strings. For $X \in \{0,1\}^*$, we define

$$X_1 \dots X_\ell \xleftarrow{n} X \tag{1}$$

to be the function that partitions X into $\ell = \lceil |X|/n \rceil$ blocks of size n bits, where the last block is appended with 0s. The expression "A ? B : C" equals B if A is true, and equals C if A is false. For $x \in \{0,1\}^n$ and $i \leq n$, we denote

Algorithm 1 Elephant encryption algorithm enc

Input: $(K, N, A, M) \in \{0,1\}^k \times \{0,1\}^m \times \{0,1\}^* \times \{0,1\}^*$
Output: $(C, T) \in \{0,1\}^{|M|} \times \{0,1\}^t$

$M_1 \ldots M_{\ell_M} \xleftarrow{n} M$
for $i = 1, \ldots, \ell_M$ **do**
  $C_i \leftarrow M_i \oplus \mathsf{P}(N \| 0^{n-m} \oplus \mathsf{mask}_K^{i-1,1}) \oplus \mathsf{mask}_K^{i-1,1}$
$C \leftarrow \lfloor C_1 \ldots C_{\ell_M} \rfloor_{|M|}$
$A_1 \ldots A_{\ell_A} \xleftarrow{n} N \| A \| 1$
$C_1 \ldots C_{\ell_C} \xleftarrow{n} C \| 1$
$T \leftarrow A_1$
for $i = 2, \ldots, \ell_A$ **do**
  $T \leftarrow T \oplus \mathsf{P}(A_i \oplus \mathsf{mask}_K^{i-1,0}) \oplus \mathsf{mask}_K^{i-1,0}$
for $i = 1, \ldots, \ell_C$ **do**
  $T \leftarrow T \oplus \mathsf{P}(C_i \oplus \mathsf{mask}_K^{i-1,2}) \oplus \mathsf{mask}_K^{i-1,2}$
$T \leftarrow \mathsf{P}(T \oplus \mathsf{mask}_K^{0,0}) \oplus \mathsf{mask}_K^{0,0}$
return $(C, \lfloor T \rfloor_t)$

by $x \ll i$ (resp., $x \gg i$) a shift of x to the left (resp., right) over i positions. We likewise denote by $x \lll i$ (resp., $x \ggg i$) a rotation of x to the left (resp., right) over i positions. We denote by $\lfloor x \rfloor_i$ the i left-most bits of x.

2.2 Elephant Authenticated Encryption Mode

Let $k, m, n, t \in \mathbb{N}$ with $k, m, t \leq n$. Let $\mathsf{P} : \{0,1\}^n \to \{0,1\}^n$ be an n-bit permutation, and $\varphi_1 : \{0,1\}^n \to \{0,1\}^n$ be an LFSR. Define $\varphi_2 = \varphi_1 \oplus \mathsf{id}$, where id is the identity function. Define the function $\mathsf{mask} : \{0,1\}^k \times \mathbb{N}^2 \to \{0,1\}^n$ as follows:

$$\mathsf{mask}_K^{a,b} = \mathsf{mask}(K, a, b) = \varphi_2^b \circ \varphi_1^a \circ \mathsf{P}(K \| 0^{n-k}) \,. \tag{2}$$

We will describe the generic authenticated encryption mode of Elephant. It consists of two algorithms: encryption enc and decryption dec.

2.2.1 Encryption

Encryption enc gets as input a key $K \in \{0,1\}^k$, a nonce $N \in \{0,1\}^m$, associated data $A \in \{0,1\}^*$, and a message $M \in \{0,1\}^*$, and it outputs a ciphertext $C \in \{0,1\}^{|M|}$ and a tag $T \in \{0,1\}^t$. The description of enc is given in Algorithm 1, and it is depicted in Figure 1.

2.2.2 Decryption

Decryption dec gets as input a key $K \in \{0,1\}^k$, a nonce $N \in \{0,1\}^m$, associated data $A \in \{0,1\}^*$, a ciphertext $C \in \{0,1\}^*$, and a tag $T \in \{0,1\}^t$, and it outputs

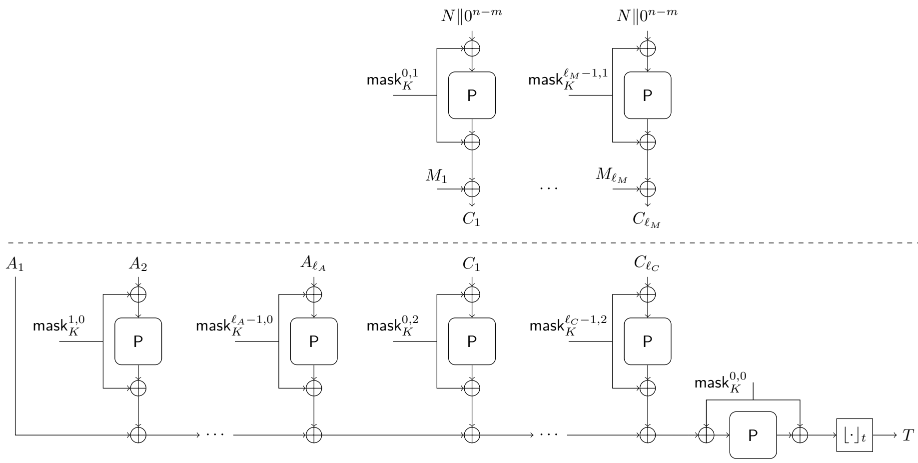

Figure 1: Depiction of Elephant. For the encryption part (top): message is padded as $M_1 \dots M_{\ell_M} \xleftarrow{n} M$, and ciphertext equals $C = \lfloor C_1 \dots C_{\ell_M} \rfloor_{|M|}$. For the authentication part (bottom): nonce and associated data are padded as $A_1 \dots A_{\ell_A} \xleftarrow{n} N\|A\|1$, and ciphertext is padded as $C_1 \dots C_{\ell_C} \xleftarrow{n} C\|1$.

a message $M \in \{0,1\}^{|M|}$ if the tag is correct, or a dedicated $\perp$-sign otherwise. The description of dec is given in Algorithm 2.

2.3 160-Bit Permutation and LFSR

Section 2.3.1 defines the Spongent-π[160] permutation. The 160-bit masking LFSR φ_1 is defined in Section 2.3.2. These components are used in Dumbo.

2.3.1 Spongent Permutation

We denote by Spongent-$\pi[160]\colon \{0,1\}^{160} \to \{0,1\}^{160}$ the 80-round Spongent permutation of Bogdanov et al. [23]. It operates on a 160-bit input X as follows:

for $i = 1, \dots, 80$ **do**
  $X \leftarrow X \oplus 0^{153}\|\mathsf{lCounter}_{160}(i) \oplus \mathsf{rev}\big(0^{153}\|\mathsf{lCounter}_{160}(i)\big)$
  $X \leftarrow \mathsf{sBoxLayer}_{160}(X)$
  $X \leftarrow \mathsf{pLayer}_{160}(X)$

where the function rev reverses the order of the bits of its input, and where the functions $\mathsf{lCounter}_{160}$, $\mathsf{sBoxLayer}_{160}$, and pLayer_{160} are defined as follows:

- $\mathsf{lCounter}_{160}$: this function is a 7-bit LFSR defined by the primitive polynomial $p(x) = x^7 + x^6 + 1$ and initialized with “1110101”;

- $\mathsf{sBoxLayer}_{160}$: this function consists of an S-box $S\colon \{0,1\}^4 \to \{0,1\}^4$ applied 40 times in parallel. In hexadecimal notation, this S-box is defined as

Algorithm 2 Elephant decryption algorithm dec

Input: $(K, N, A, C, T) \in \{0,1\}^k \times \{0,1\}^m \times \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^t$
Output: $M \in \{0,1\}^{|C|}$ or $\perp$

$C_1 \ldots C_{\ell_M} \xleftarrow{n} C$
for $i = 1, \ldots, \ell_M$ **do**
  $M_i \leftarrow C_i \oplus \mathsf{P}(N \| 0^{n-m} \oplus \mathsf{mask}_K^{i-1,1}) \oplus \mathsf{mask}_K^{i-1,1}$
$M \leftarrow \lfloor M_1 \ldots M_{\ell_M} \rfloor_{|C|}$
$A_1 \ldots A_{\ell_A} \xleftarrow{n} N \| A \| 1$
$C_1 \ldots C_{\ell_C} \xleftarrow{n} C \| 1$
$\bar{T} \leftarrow A_1$
for $i = 2, \ldots, \ell_A$ **do**
  $\bar{T} \leftarrow \bar{T} \oplus \mathsf{P}(A_i \oplus \mathsf{mask}_K^{i-1,0}) \oplus \mathsf{mask}_K^{i-1,0}$
for $i = 1, \ldots, \ell_C$ **do**
  $\bar{T} \leftarrow \bar{T} \oplus \mathsf{P}(C_i \oplus \mathsf{mask}_K^{i-1,2}) \oplus \mathsf{mask}_K^{i-1,2}$
$\bar{T} \leftarrow \mathsf{P}(\bar{T} \oplus \mathsf{mask}_K^{0,0}) \oplus \mathsf{mask}_K^{0,0}$
return $\lfloor \bar{T} \rfloor_t = T \ ? \ M \ : \ \perp$

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S(X)$	E	D	B	0	2	1	4	F	7	A	8	5	9	C	3	6

- pLayer_{160}: this function moves the j-th bit of its input to bit position $P_{160}(j)$, where

$$P_{160}(j) = \begin{cases} 40 \cdot j \bmod 159\,, & \text{if } j \in \{0, \ldots, 158\}\,, \\ 159\,, & \text{if } j = 159\,. \end{cases}$$

2.3.2 LFSR

For generating the masks of our scheme, we use the approach of Granger et al. [53]. We define φ_1 as the following $\mathbb{F}_2$-linear map, where the x_i's correspond to 8-bit words:

$$(x_0, \ldots, x_{19}) \mapsto (x_1, \ldots, x_{19}, x_0 \lll 3 \oplus x_3 \ll 7 \oplus x_{13} \gg 7)\,. \tag{3}$$

2.4 176-Bit Permutation and LFSR

Section 2.4.1 defines the Spongent-π[176] permutation. The 176-bit masking LFSR φ_1 is defined in Section 2.4.2. These components are used in Jumbo.

2.4.1 Spongent Permutation

We denote by Spongent-π[176]: $\{0,1\}^{176} \to \{0,1\}^{176}$ the 90-round Spongent permutation of Bogdanov et al. [23]. It operates on a 176-bit input X as follows:

for $i = 1, \ldots, 90$ **do**
  $X \leftarrow X \oplus 0^{169} \| \mathsf{lCounter}_{176}(i) \oplus \mathsf{rev}\big(0^{169} \| \mathsf{lCounter}_{176}(i)\big)$
  $X \leftarrow \mathsf{sBoxLayer}_{176}(X)$
  $X \leftarrow \mathsf{pLayer}_{176}(X)$

where, as before, the function rev reverses the order of the bits of its input. The function $\mathsf{lCounter}_{176}$ is the same as $\mathsf{lCounter}_{160}$ of Section 2.3 but initialized with "1000101", the function $\mathsf{sBoxLayer}_{176}$ consists of the function S of Section 2.3 applied 44 times in parallel, and pLayer_{176} is now defined as the function that moves the j-th bit of its input to bit position $P_{176}(j)$, where

$$P_{176}(j) = \begin{cases} 44 \cdot j \bmod 175\,, & \text{if } j \in \{0, \ldots, 174\}\,, \\ 175\,, & \text{if } j = 175\,. \end{cases}$$

2.4.2 LFSR

For generating the masks of our scheme, we use the approach of Granger et al. [53]. The LFSR φ_1 is defined as the following $\mathbb{F}_2$-linear map, where the x_i's correspond to 8-bit words:

$$(x_0, \ldots, x_{21}) \mapsto (x_1, \ldots, x_{21}, x_0 \lll 1 \oplus x_3 \ll 7 \oplus x_{19} \gg 7)\,. \tag{4}$$

2.5 200-Bit Permutation and LFSR

Section 2.5.1 defines the Keccak-f[200] permutation. The 200-bit masking LFSR φ_1 is defined in Section 2.5.2. These components are used in Delirium.

2.5.1 Keccak Permutation

We denote by Keccak-$f[200]\colon \{0,1\}^{200} \to \{0,1\}^{200}$ the 18-round Keccak permutation of Bertoni et al. [15, 51]. The state $X \in \{0,1\}^{200}$ is represented as a 5-by-5-by-8 array $a \in \{0,1\}^{5\times5\times8}$, where for $(x, y, z) \in \mathbb{Z}_5 \times \mathbb{Z}_5 \times \mathbb{Z}_8$ the bit at position (x, y, z) is set as

$$a[x, y, z] = X[8(5y + x) + z]\,.$$

Keccak-f[200] operates on a 200-bit input X as follows:

for $i = 1, \ldots, 18$ **do**
  $X \leftarrow \iota \circ \chi \circ \pi \circ \rho \circ \theta(X)$

where the functions θ, ρ, π, χ, and ι are defined as follows:

$$\begin{aligned}
\theta\colon a[x, y, z] &\leftarrow a[x, y, z] \oplus \bigoplus_{y'=0}^{4} a[x-1, y', z] \oplus \bigoplus_{y'=0}^{4} a[x+1, y', z-1]\,, \\
\rho\colon a[x, y, z] &\leftarrow a[x, y, z + t[x, y]]\,, \\
\pi\colon a[x, y, z] &\leftarrow a[x + 3y, x, z]\,, \\
\chi\colon a[x, y, z] &\leftarrow a[x, y, z] \oplus (a[x+1, y, z] \oplus 1)a[x+2, y, z]\,, \\
\iota\colon a[x, y, z] &\leftarrow a[x, y, z] \oplus RC[i, x, y, z]\,.
\end{aligned}$$

For ρ, the function $t[x, y]$ is defined as

t	$x = 3$	$x = 4$	$x = 0$	$x = 1$	$x = 2$
$y = 2$	153	231	3	10	171
$y = 1$	55	276	36	300	6
$y = 0$	28	91	0	1	190
$y = 4$	120	78	210	66	253
$y = 3$	21	136	105	45	15

and for ι, the round constants are given by

$$RC[i, x, y, z] = \begin{cases} rc[j + 7i]\,, & \text{if } (x, y, z) = (0, 0, 2^j - 1)\,, \\ 0\,, & \text{otherwise}\,, \end{cases}$$

where rc is computed from a binary LFSR defined by the primitive polynomial $p(x) = x^8 + x^6 + x^5 + x^4 + 1$.

2.5.2 LFSR

For generating the masks of our scheme, we use the approach of Granger et al. [53]. The LFSR φ_1 is now defined as the following $\mathbb{F}_2$-linear map, where the x_i's correspond to 8-bit words:

$$(x_0, \ldots, x_{24}) \mapsto (x_1, \ldots, x_{24}, x_0 \lll 1 \oplus x_2 \lll 1 \oplus x_{13} \ll 1)\,. \tag{5}$$

3 Parameterization of Elephant

Elephant consists of three instances, namely those built from instantiating the mode using the permutation and LFSR of Sections 2.3, 2.4, and 2.5, respectively. In more detail, we restrict our focus to $n \in \{160, 176, 200\}$. We also set $m = 96$, i.e., we restrict to nonces of size 96 bits. Parameters $k, t \in \mathbb{N}$ are still tunable. We propose the following three instances of Elephant (with Dumbo being the primary member):

instance	k	m	n	t	P	φ_1	expected security strength	limit on online complexity
Dumbo	128	96	160	64	Spongent-π[160]	(3)	2^{112}	$2^{50}/(n/8)$
Jumbo	128	96	176	64	Spongent-π[176]	(4)	2^{127}	$2^{50}/(n/8)$
Delirium	128	96	200	128	Keccak-f[200]	(5)	2^{127}	$2^{74}/(n/8)$

Here, the online complexity is in terms of the number of n-bit blocks (hence all instances support an online complexity of 2^{50} bytes), and the strength is measured in the offline complexity, i.e., the number of primitive evaluations that the adversary can make.

In Appendix B, we give a formal multi-user security analysis of the Elephant authenticated encryption mode in the ideal permutation model. Focusing on the single-user setting, we prove that the advantage of a nonce-based adversary in breaking security of either of the schemes is at most

$$\begin{aligned}\mathbf{Adv}^{\mathrm{ae}}_{\mathsf{Elephant}}(\mathcal{A}) \leq \ell\binom{q_e}{2}/2^n + 3\binom{q_e+q_d}{2}/2^n + q_d/2^t \\ + \frac{4\sigma^2+4\sigma p+4\sigma+p}{2^n} + \frac{p}{2^k}\,,\end{aligned}$$

where q_e expresses an upper bound on the number of evaluations of the encryption function, q_d the number of decryption queries, ℓ the maximum length of a single query in blocks, σ the total online complexity in blocks, and p the number of evaluations of the random primitive P. Note that the dominating term in the bound is $4\sigma p/2^n$. By capping $\sigma \leq 2^{n-114}$, this term is less than 1 as long as $p \leq 2^{112}$. Likewise, by capping $\sigma \leq 2^{n-130}$, this term is less than 1 as long as $p \leq 2^{128}$. However, one also needs to take the other terms of the bound into account. Most of the terms are negligible compared to $4\sigma p/2^n$, and are covered by taking a slightly stricter condition on σ (note that $2^{50}/(n/8) < 2^{46}$ and $2^{74}/(n/8) < 2^{70}$ for each of the instances). There is one exception to these negligible terms, namely the factor $p/2^k$ for Jumbo and Delirium: it equals 1 for $p = 2^{128}$. This term thus accounts for a factor 2 loss in the security strength of Jumbo and Delirium, and we must restrict the offline complexity for these variants by a factor 2, as indicated in above table.

If we consider the multi-user setting, where the adversary has access to multiple independently keyed instantiations of the construction, the security of the Elephant mode only decreases negligibly. To be precise, if the adversary has access to μ instances of the Elephant mode, its advantage is at most

$$\begin{aligned}\mathbf{Adv}^{\mu\text{-ae}}_{\mathsf{Elephant}}(\mathcal{A}) \leq \ell\binom{q_e}{2}/2^n + 3\binom{q_e+q_d}{2}/2^n + q_d/2^t \\ + \frac{4\sigma^2+4\sigma p+(\mu-1)2\sigma+\mu\cdot(4\sigma+p+\frac{\mu-1}{2})}{2^n} + \frac{\mu\cdot(p+\frac{\mu-1}{2})}{2^k}\,.\end{aligned}$$

We can observe that, compared to the single-user bound, μ only appears in the minor terms and thus only plays a small role.

We stress that the authenticated encryption security claims only hold in the nonce-respecting setting: the adversary may not evaluate the encryption function twice under the same nonce (it may make decryption queries for a reused nonce, though). *If the nonce is reused for two different evaluations of* enc*, security is void.* In particular, if the nonce uniqueness condition is released, a trivial confidentiality attack can be mounted. On the upside, nonce uniqueness is only effectively used in the proof of confidentiality: authenticity still holds under nonce-reuse. We also do *not* claim security in case unverified plaintext is released [5]; we note, however, that in practice decryption of the ciphertext C into the message M takes place *only after the tag (in turn, computed from the nonce, associated data, and ciphertext) has been verified.*

4 Design Rationale

The Elephant mode is an encrypt-then-MAC mode, where encryption is performed by counter mode and message authentication by a variant of protected counter sum [10, 77], both implicitly instantiated using a simplification of the masked Even-Mansour (MEM) tweakable block cipher of Granger et al. [53]. This tweakable block cipher, in turn, is based on a Spongent [23] or Keccak [15] permutation. We explain the design rationale of Elephant at the following three levels of granularity: the generic mode in Section 4.1, how the mode uses the permutation, i.e., the masking scheme, in Section 4.2, and the choice of particular primitives in Section 4.3. Finally, Section 4.4 briefly discusses implementation aspects.

4.1 Mode

Generically, encrypt-then-MAC is the most secure approach [9, 84]: unlike its alternatives encrypt-and-MAC and MAC-then-encrypt, this approach yields integrity of ciphertexts. Stated differently, malformed ciphertexts yield failure upon MAC verification, and for these no decryption is needed. This prevents unintended leakage from verification failures. The approach also makes it possible to easily prevent leakage due to release of unverified plaintext (RUP) [5]: simply do not start decrypting before the tag is verified. Note that for the generic alternatives encrypt-and-MAC and MAC-then-encrypt, such a simple countermeasure is impossible. This makes the encrypt-then-MAC mode of Elephant preferable over its alternatives, not only in the lightweight setting but also for general purpose.

The counter encryption mode and protected counter sum style MAC mode within Elephant, in turn, are both fully parallelizable and only evaluate the underlying permutation P in forward direction. The fact that Elephant evaluates its primitive in forward direction is important in the lightweight setting: it allows for smaller implementations, since there is no need to implement the inverse of P. Note, in particular, that due to the rise of the sponge, various cryptographic permutations, including Ascon [46], Gimli [13], Keccak [15], and XOODOO [37], are developed to be particularly efficient in forward direction.

By being parallelizable, Elephant distinguishes itself from a wide range of authenticated encryption schemes that employ a serial permutation-based mode of operation, such as APE [3], Beetle [31], or the Duplex construction [14,38,79]. To support parallelism, we need to store the internal state value, but on the upside, it turns out to give various elegant implementation advantages (see Section 4.2 and Section 4.4) and it means that there is no strict need to employ larger permutations.

We briefly elaborate on existing generic authenticated encryption schemes that are both parallel and permutation-based (but not necessarily inverse-free). Granger et al. [53] introduced OPP, a parallel and permutation-based scheme derived from ΘCB [64], but it is not inverse-free. Minalpher [95], likewise, is parallel and permutation-based but not inverse-free. Finally, a permutation-based version of OTR [80] exists in the embodiment of Prøst-OTR [61]. This construction is parallel, permutation-based, and inverse free, just like Elephant. However, because it processes pairs of message blocks using a two-round Feistel structure, the encryption process differs depending on the parity of the number of message blocks. This stands in contrast to the conceptual simplicity of Elephant. In addition, for short messages, less parallelism is available in Prøst than for Elephant. If the implementation maximally exploits parallelism, Elephant would compare favorably for short messages in terms of latency.

The mode is nonce-based: each of the members of Elephant uses a 96-bit nonce. The nonce is prepended to the associated data, which is then padded and split into n-bit blocks $A_1 \dots A_{\ell_A}$ (see line 5 of Algorithm 1). This way, the scheme is optimized for the parameters specified in the NIST call [85]: the nonce is 96 bits, and in order to avoid a waste of $n - 96$ bits due to padding (where $n \in \{160, 176, 200\}$), the nonce is appended with the first $n - 96$ bits of the associated data. Caution must be paid here, namely that the *nonce is always of fixed length of* 96 *bits*. If variable-length nonces were allowed, the scheme would be vulnerable to trivial padding attacks. We remark that it is theoretically possible to adjust the Elephant mode to allow longer nonces or flexible-length nonces, but we discourage this as it might lead to error-prone designs. Furthermore, we clarify that the nonce is used *both* for encryption and for authentication: the former is needed for confidentiality and the latter is needed in case of authenticated encryption of an empty message. Also, as the mode is nonce-based, authenticated encryption security is guaranteed *only if* the adversary does not repeat nonces for encryption queries. Authenticity still stands under nonce-reuse.

4.2 Masking

As specified in Section 2.2, the inputs to and outputs of the permutation P are masked using $\mathsf{mask}_K^{a,b}$ of (2). The masking function is defined using two LFSRs $\varphi_1, \varphi_2 : \{0,1\}^n \to \{0,1\}^n$ that satisfy $\varphi_2 = \varphi_1 \oplus \mathsf{id}$, and it is parameterized by (a, b) which are used in a manner so as to assure that every occurrence of the masking in the Elephant mode gets different parameters. We have heuristically chosen our LFSRs to give a good match when used in combination with the particular permutations. For the LFSR's matching Spongent, we selected versions that have a small gate count in hardware. In the case of the 200-bit Keccak permutation, we chose an LFSR that can be implemented with a small number of instructions. Hence, we selected an LFSR that allows for implementations with shift/rotation by one. The number of gates needed for a hardware implementation was a secondary consideration in this case.

The LFSR-based masking technique is taken from Granger et al. [53], and so is the security analysis (although different state sizes, discrete logarithm computations, LFSRs, and tweak domains are considered). Granger et al. have argued in favor of this technique over its alternatives for various reasons: (i) the approach is simpler to implement, as the masking is purely linear and does not use finite field multiplication, (ii) it is more efficient (depending on the primitive used), and (iii) the masking is constant time.

The latter point is important in the lightweight setting where resistance against timing attacks comes at a cost. In this respect, the LFSR-based masking approach compares favorably with another, and very popular, masking technique, namely powering-up-based masking (simplified to allow for fair comparison with (2)):

$$3^b 2^a \mathsf{P}(K \| 0^{n-k}) \,,$$

where 2 and 3 are coordinates in the monomial basis in the finite field $\mathbb{F}_{2^n}$. The technique was introduced by Rogaway [92] in the context of OCB2, and it has seen many applications, including CAESAR submissions AES-OTR [80], AEZ [57], COLM [4], Minalpher [95], POET [2], and SHELL [106]. These multiplications can be implemented as an LFSR on one-bit words, but the masking functions φ_1 and φ_2 are constant time by design and allow for more flexibility in the word size.

A related masking approach is that of OCB3 [64] and OMD [35], which use masking based on Gray coding. In detail, Gray coding-based masks can be updated as $G(i) = G(i-1) \oplus 2^{\mathsf{ntz}(i)}$, were $\mathsf{ntz}(i)$ is the number of trailing zeros in the binary representation of i. The masking, unlike powering-up, does not need a conditional XOR, but it requires $\log_2(i)$ field doublings (which may be precomputed). As the LFSR-based masking used in $\mathsf{Elephant}$ does not incur such a cost, it also compares favorably with this technique.

The particular choice of masking, namely $(a, b) = (i, 1)$ in the encryption layer, $(a, b) = (i, 2)$ for ciphertext authentication, and $(a, b) = (i, 0)$ for associated data authentication, allows maskings to cancel out nicely in the implementation. To see this, consider the authentication of ciphertext C_i (for $i < \ell_M \leq \ell_C$), and more detailed the contribution T_i it makes to tag T. This value is computed as

$$T_i = \mathsf{P}\left(M_i \oplus \mathsf{P}(N \| 0^{n-m} \oplus \mathsf{mask}_K^{i-1,1}) \oplus \mathsf{mask}_K^{i-1,1} \oplus \mathsf{mask}_K^{i-1,2}\right) \oplus \mathsf{mask}_K^{i-1,2} \,.$$

By definition of $\mathsf{mask}_K^{a,b}$, and as $\varphi_2 = \varphi_1 \oplus \mathsf{id}$, we have

$$\begin{aligned}
&\mathsf{mask}_K^{i-1,1} \oplus \mathsf{mask}_K^{i-1,2} \\
&= (\varphi_1 \oplus \mathsf{id}) \circ \varphi_1^{i-1} \circ \mathsf{P}(K \| 0^{n-k}) \oplus (\varphi_1 \oplus \mathsf{id})^2 \circ \varphi_1^{i-1} \circ \mathsf{P}(K \| 0^{n-k}) \\
&= (\varphi_1 \oplus \mathsf{id}) \circ \varphi_1^{i} \circ \mathsf{P}(K \| 0^{n-k}) \,.
\end{aligned}$$

This, not surprisingly, is the mask used for the encryption of the next message block M_{i+1}. We note that exploiting this requires extra state.

Another optimization in mask management is in the masks that contribute to the tag, i.e., the sum of all masks that appear in the final tag T. The contribution coming from the ciphertext authentication equals

$$\begin{aligned}
\left(\bigoplus_{i=1}^{\ell_C} \mathsf{mask}_K^{i-1,2}\right) &= \left(\bigoplus_{i=1}^{\ell_C} (\varphi_1 \oplus \mathsf{id})^2 \circ \varphi_1^{i-1} \circ \mathsf{P}(K \| 0^{n-k})\right) \\
&= (\varphi_1^{\ell_C+1} \oplus \varphi_1^{\ell_C} \oplus \varphi_1 \oplus \mathsf{id}) \circ \mathsf{P}(K \| 0^{n-k}) \,,
\end{aligned} \tag{6}$$

and that coming from the associated data likewise equals

$$\left(\bigoplus_{i=1}^{\ell_A} \mathsf{mask}_K^{i-1,0}\right) = (\varphi_1^{\ell_A-1} \oplus \varphi_1^{\ell_A-2} \oplus \cdots \oplus \varphi_1 \oplus \mathsf{id}) \circ \mathsf{P}(K \| 0^{n-k})\,. \qquad (7)$$

This feature of the masking may be useful if Elephant is used for fixed-length data, in which case the (6) and (7) could be precomputed.

4.3 Primitives

4.3.1 Dumbo and Jumbo

Both the 160-bit and 176-bit instance of Elephant are based on a Spongent permutation [23]: the 160-bit instance is based on the Spongent-π[160] permutation, and the 176-bit instance is based on the Spongent-π[176] permutation. The choice for Spongent is natural: it is particularly well-suited for hardware, and the existing third-party analysis (see Section 5.3) does not indicate any weakness of the Spongent family relevant for our use-case. We have used the 160-bit version of Spongent as this is the smallest possible permutation that can be used to efficiently[1] meet the NIST call for proposals. The 176-bit Spongent permutation offers a slightly more comfortable 127-bit security margin. In addition, this particular Spongent permutation is part of the ISO/IEC standard on lightweight hash functions [59].

Bogdanov et al. [23] do not explicitly specify the number of rounds of the 160-bit version of the Spongent permutation; we opt for 80 rounds since this ensures that at least 160 S-boxes are differentially active. This is in accordance with the Spongent design strategy. Note further that this implies that the 7-bit LFSR specified in [23] should be used (with initial value `0x75`) to generate the round constants for the permutation.

The LFSRs of both instances aim to minimize the area required when implemented in hardware. In particular, in addition to the shift register, only two 2-bit XOR gates are needed. Hence, these choices of LFSRs are in line with the strength of the Spongent permutations, making a perfect match for small area hardware implementations. Despite the particular suitability of both LFSRs for small area hardware implementations, it is still possible to implement them rather efficiently on 8-bit platforms.

4.3.2 Delirium

The 200-bit instance of Elephant is based on the Keccak-f[200] permutation [15]. The 200-bit instance is the smallest of the instances that is specified in the NIST standard [51] that fits our need; it is still reasonable in hardware, and particularly good in software on 8-bit platforms, considering that it is naturally defined using 8-bit lanes [17,62]. As such, it is complementary to the Spongent-based instantiation of Elephant.

[1]Beyond birthday bound solutions may use even smaller permutations, but only at an efficiency penalty.

This LFSR shows its full potential when implemented on 8-bit platforms. A state update within the LFSR just updates one byte, while the content of the other 24 bytes is not changed and basically just relabeled. The single updated byte is computed as the XOR sum of 3 other state bytes that are just rotated or shifted by one bit position. Hence, the essential operations that have to be performed on 8-bit platforms are 3 XOR operations, two rotations by one bit to the left plus one shift by one bit to the left.

4.4 Implementation

As discussed in Section 4.1, the Elephant mode allows for a high degree of parallelism. For the hardware-oriented variants of Elephant (Dumbo and Jumbo), this makes it easy to trade-off area for additional throughput. Hardware implementations of the 176-bit Spongent permutation are given by Bogdanov et al. [23], e.g., just needing 1329 GE to implement the Spongent-160 hash function, which is based on the 176-bit Spongent permutation. The 200-bit variant of Elephant primarily targets (embedded) software, but the same remarks concerning hardware implementations apply as, e.g., demonstrated by an implementation of a hash function based on the 200-bit Keccak permutation needing just 2520 GE by Kavun and Yalçin [62].

Software implementations of 200-bit Elephant (Delirium) can also exploit parallelism. If multiple cores are available, several blocks can be processed concurrently – but this is only useful for long messages. More importantly, on processors with a word size above 16 bits, the available parallelism makes it possible to increase the efficiency of the implementation by combining two or more calls to the Keccak permutation. For mid- and high-end processors with SIMD instructions, the same technique can be used to obtain even greater speed-ups.

An increasingly common requirement is the ability to protect implementations against side-channel attacks. As discussed in Section 4.2, the masking scheme is constant time by design. The same applies to the Spongent and Keccak permutations. In addition, all variants of Elephant are well-suited for Boolean masking techniques such as threshold implementations [88].

Finally, it is worth mentioning that a few specific use-cases of Elephant allow for additional optimizations. As discussed in Section 4.2, the contribution of the mask values to the tag can be precomputed for fixed-length messages. In addition, if one or more blocks of associated data are static, it is possible to precompute their contribution to the tag.

A reference implementation of Dumbo, Jumbo, and Delirium written in C99 can be found at `https://github.com/TimBeyne/Elephant`.

5 Summary of Known Cryptanalytic Attacks

After briefly reviewing security aspects of the generic Elephant mode in Section 5.1, and dedicated analysis of Elephant variants in Section 5.2, we discuss the main cryptanalytic results on Spongent in Section 5.3, and on Keccak in Section 5.4.

5.1 Generic Mode

In Appendix B, we prove that the generic mode of Elephant, based on a tweakable block cipher, is secure. The security proof is standard, and it builds among others on ideas of Bellare and Namprempre [9] and Namprempre et al. [84] (for insights in the encrypt-then-MAC approach), and Bernstein [10] and Luykx et al. [77] (for insights in the protected counter sum MAC mode). The analysis of the underlying tweakable block cipher, in turn, builds on Granger et al. [53]. A security proof of Elephant v1.1 was published in [20].

Bonnetain and Jaques [26] considered quantum security of the Elephant mode (both v1.1 and v2), in the setting where the adversary has classical access to the construction but can make quantum evaluations of the primitive. Although the attack is very interesting, it is not a threat for Elephant: the quantum key recovery attack requires more quantum operations than a direct key search.

5.2 Dedicated Analysis

Zhou et al. [111] derived a round-reduced interpolation attack against the Keccak permutation, and applied it to Delirium. The analysis targets a round-reduced version of the encryption of Delirium, where the number of rounds of the used variant of the Keccak permutation is reduced from the specified 18 rounds to 8 rounds. The time complexity of the attack is $2^{98.3}$ XOR operations, having a memory complexity of 2^{70}, needing 2^{70} blocks of data to work. The attack makes use of the fact that an affine space of dimension 65 sums to zero after 6 rounds, which are then followed by 2 rounds for key recovery. Since the encryption of Elephant v1.1 and Elephant v2 are the same (the sole change is in the authentication), the analysis is applicable in a similar manner.

5.3 Spongent Permutation

We discuss the main known cryptanalytic results in detail, and refer to Appendix A.2 for a complete list.

Differential Cryptanalysis. The following result of Bogdanov et al. [24] provides a lower bound on the number of active S-boxes in any differential characteristic of Spongent-$\pi[b]$ with $b \geq 64$. The result and its proof are similar to those for the block cipher PRESENT [25].

Theorem 5.1 (Theorem 1 of Bogdanov et al. [24])**.** *Any 5-round differential characteristic of* Spongent-$\pi[b]$ *with* $b \geq 64$ *involves at least 10 differentially active S-boxes.*

Theorem 5.1 implies that after r rounds of Spongent-$\pi[b]$ with $b \geq 64$, at least $2r$ S-boxes are differentially active. Since the S-box is differentially 4-uniform, it follows that the probability of any r-round characteristic is at most 2^{-4r}.

Note that the number of rounds of Spongent-$\pi[b]$ is determined such that at least b S-boxes are differentially active [24]. Equivalently, Spongent-$\pi[b]$ should have at least $b/2$ rounds.

More rounds can be attacked by relying on truncated differentials. For example, for $b = 176$, Zhang and Liu [109] presented a 46-round truncated differential with (marginally) significant probability. These properties are derived from multidimensional linear approximations, following Blondeau and Nyberg [22]. In the next section, linear approximations are discussed in more detail.

In conclusion, (truncated) differential cryptanalysis does not threaten full-round Spongent-$\pi[b]$, for neither $b = 160$ nor $b = 176$. In addition, one should keep in mind that many of the best reduced-round distinguishers require more data than is allowed to be processed by the Elephant mode (i.e., no more than 2^{47} chosen plaintexts).

Linear Cryptanalysis. In order to assess the security of the permutation Spongent-$\pi[b]$ against linear cryptanalysis, we follow the approach used by Bogdanov et al. [24]: rather than computing only the correlation of individual trails, the correlation of linear approximations will be estimated. Previous work has shown that 1-bit (per round) trails are dominant in PRESENT-like designs [34, 69], meaning that one can estimate the correlation of all 1-bit linear approximations over r rounds by computing the product of r sparse matrices of size $b \times b$. Table 1 shows the resulting estimates, where c_r denotes the maximum absolute correlation after r rounds.

Table 1: Estimated maximum correlation of linear approximations of Spongent-$\pi[b]$ with $b \in \{160, 176\}$. The total number of rounds is denoted by R (that is, $R = 80$ for $b = 160$ and $R = 90$ for $b = 176$).

	$b = 160$	$b = 176$
c_{40}	2^{-80}	2^{-80}
c_{44}	2^{-88}	2^{-88}
c_R	2^{-160}	2^{-180}

The estimates in Table 1 could be improved by taking into account additional trails. For example, Abdelraheem [1] gives improved estimates by taking into account all trails with at most four linearly active S-boxes per round. This yields slightly improved distinguishers in some cases, but still covering at most one or two additional rounds.

The results above imply that full-round Spongent-$\pi[b]$ is not threatened by linear attacks, statistical saturation attacks, or multidimensional linear attacks [34, 36]. As for differential cryptanalysis, it should be remarked that the security margin remains large, especially because even the reduced-round distinguishers typically require more data than the Elephant mode can securely process.

Integral Cryptanalysis. Division properties of Spongent-$\pi[b]$ have been analyzed to some extent, in particular for $b = 88$ [50,102,103]. Eskandari et al. [50] built a SAT-solver based tool to find, or show the absence of, division properties. They use this tool to show that Spongent-$\pi[176]$ does not have a bit-based division property covering 12 rounds or more. It was verified that the same holds for Spongent-$\pi[160]$.

It is often possible to setup a distinguisher that covers more rounds, by starting from the middle of the permutation and extending the division property in the forward and backward direction. For example, Sun et al. [103] presented a zero-sum distinguisher for 21 rounds of Spongent-$\pi[160]$ requiring 2^{159} data. Remark that even this reduced-round distinguisher far exceeds the data limits imposed for Elephant.

We now discuss the ramifications of the above results in the context of impossible differentials and zero-correlation linear approximations, by relying on a result of Sun et al. [101]. Sun et al. demonstrated that a nontrivial zero-correlation linear approximation of a permutation constructively implies the existence of an integral distinguisher. They furthermore demonstrated that, as Spongent-$\pi[b]$ has a bitwise (hence self-dual) linear layer, one can conclude that for (round-reduced) Spongent-$\pi[b]$, any nontrivial impossible differential that does not depend on the choice of the S-box constructively implies the existence of an integral distinguisher.

It can be concluded that Spongent has a very large margin against integral-type distinguishers. The same applies to zero correlation linear approximations and impossible differentials (not relying on the S-box structure), due to their links with integral properties.

5.4 Keccak Permutation

We discuss the main known cryptanalytic results in detail, and refer to Appendix A.3 for a complete list.

Differential Cryptanalysis. The differential properties of the permutation Keccak-$f[200]$ have been extensively analyzed and no significant differential distinguishers are expected to exist [15, 39, 78]. Due to Keccak's weak alignment [16], there are no known analytic upper bounds on the probability of differential characteristics. Instead, computer assistance is required to determine bounds.

The analysis in the Keccak reference [15] leads to lower bounds on the weight of symmetric characteristics in Keccak-f – remark that Keccak-$f[200]$ characteristics are symmetric by definition. The results are summarized in the first three rows of Table 2. Improved bounds are presented by Mella, Daemen, and

Van Assche [78] based on a dedicated search algorithm. For the characteristics corresponding to the lower bounds in Table 2, the reader is referred to Table 3 of [78].

Table 2: Lower and upper bounds on the minimum weight of differential characteristics in Keccak-f[200] [15, 78].

Rounds	Lower bound	Upper bound
2	8	8
3	20	20
4	46	46
5	50	89
6	92	142
18	276	—

Of course, the lack of high probability differential characteristics need not imply that all differentials have low probability. Bertoni et al. [16] argue that clustering of 2-round characteristics is prevented by weak alignment. This means that the propagation of differentials does not respect cell-boundaries in Keccak. Weak alignment leads the authors of Keccak to believe that it is unlikely that truncated differentials can be successfully exploited [16].

Linear Cryptanalysis. The Keccak reference [15] provides lower bounds on the weight of linear trails, where the weight of a linear trail equals minus the logarithm of the square of its correlation. These bounds are listed in Table 3. The lower bound for full-round Keccak-f[200] is 204, corresponding to a correlation which is only slightly smaller than the variance of the correlation of linear approximations in a random permutation. It should be emphasized that 204 is a rather rough lower bound, and the true minimum weight is expected to be much larger.

As in the case of differential cryptanalysis, Bertoni et al. [16] provide arguments against clustering of linear trails based on Keccak's weak alignment.

Table 3: Lower and upper bounds on the minimum weight of linear trails in Keccak-f[200] [15].

Rounds	Lower bound	Upper bound
2	8	8
3	20	20
4	46	46
18	204	—

Attacks Exploiting Algebraic Degree. For keyed instances that use variants of Keccak-f, such as Ketje [19] and Keyak [18], the attacks covering the highest number of rounds typically exploit the algebraic degree, e.g., cube [45], cube-like [44], or conditional cube attacks [58]. In the case of Ketje Jr., that builds on a round-reduced version of Keccak-f[200], those attacks can cover up to 6 rounds [97]. If we take a broader look at constructions that use bigger variants of Keccak-f, and also allow the attacker more degrees of freedom in placing the cube variables, those attacks usually lie in the region of 8 rounds [21,44,47,58,99] considering a targeted security level of 128-bits. Since Keccak-f[200] used in Delirium has 18 rounds, we have a huge security margin against this type of attacks.

Acknowledgments. This work was supported in part by the Research Council KU Leuven: GOA TENSE (C16/15/058). Yu Long Chen is supported by a Ph.D. Fellowship from the Research Foundation - Flanders (FWO). 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] Abdelraheem, M.A.: Estimating the Probabilities of Low-Weight Differential and Linear Approximations on PRESENT-Like Ciphers. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. LNCS, vol. 7839, pp. 368–382. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_26

[2] Abed, F., Fluhrer, S., Foley, J., Forler, C., List, E., Lucks, S., McGrew, D., Wenzel, J.: The POET Family of On-Line Authenticated Encryption Schemes v2.0. Submission to the CAESAR competition (2015)

[3] Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mennink, B., Mouha, N., Yasuda, K.: APE: Authenticated Permutation-Based Encryption for Lightweight Cryptography. In: Cid, C., Rechberger, C. (eds.) Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. LNCS, vol. 8540, pp. 168–186. Springer (2014), https://doi.org/10.1007/978-3-662-46706-0_9

[4] Andreeva, E., Bogdanov, A., Datta, N., Luykx, A., Mennink, B., Nandi, M., Tischhauser, E., Yasuda, K.: COLM v1. Submission to the CAESAR competition (2016)

[5] Andreeva, E., Bogdanov, A., Luykx, A., Mennink, B., Mouha, N., Yasuda, K.: How to Securely Release Unverified Plaintext in Authenticated Encryption. In: Sarkar, P., Iwata, T. (eds.) Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and

Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I. LNCS, vol. 8873, pp. 105–125. Springer (2014), https://doi.org/10.1007/978-3-662-45611-8_6

[6] Aumasson, J., Henzen, L., Meier, W., Naya-Plasencia, M.: Quark: A Lightweight Hash. In: Mangard, S., Standaert, F. (eds.) Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings. LNCS, vol. 6225, pp. 1–15. Springer (2010), https://doi.org/10.1007/978-3-642-15031-9_1

[7] Aumasson, J.P., Khovratovich, D.: First Analysis of Keccak. NIST hash forum: https://131002.net/data/papers/AK09.pdf (2009)

[8] Aumasson, J.P., Khovratovich, D.: Zero-sum distinguishers for reduced Keccak-f for the core functions of Luffa and Hamsi. Rump session of CHES: https://131002.net/data/papers/AM09.pdf (2009)

[9] Bellare, M., Namprempre, C.: Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. In: Okamoto, T. (ed.) Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings. LNCS, vol. 1976, pp. 531–545. Springer (2000), https://doi.org/10.1007/3-540-44448-3_41

[10] Bernstein, D.J.: How to Stretch Random Functions: The Security of Protected Counter Sums. J. Cryptology 12(3), 185–192 (1999), https://doi.org/10.1007/s001459900051

[11] Bernstein, D.J.: Stronger Security Bounds for Wegman-Carter-Shoup Authenticators. In: Cramer, R. (ed.) Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings. LNCS, vol. 3494, pp. 164–180. Springer (2005), https://doi.org/10.1007/11426639_10

[12] Bernstein, D.J.: Second preimages for 6 (7? (8??)) rounds of Keccak? NIST hash forum: http://cr.yp.to/hash/keccak-20101127.txt (2009)

[13] Bernstein, D.J., Kölbl, S., Lucks, S., Massolino, P.M.C., Mendel, F., Nawaz, K., Schneider, T., Schwabe, P., Standaert, F., Todo, Y., Viguier, B.: Gimli : A Cross-Platform Permutation. In: Fischer, W., Homma, N. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. LNCS, vol. 10529, pp. 299–320. Springer (2017), https://doi.org/10.1007/978-3-319-66787-4_15

[14] 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.) Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers. LNCS, vol. 7118, pp. 320–337. Springer (2011), https://doi.org/10.1007/978-3-642-28496-0_19

[15] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: The Keccak reference (January 2011)

[16] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On alignment in Keccak. In: ECRYPT II Hash Workshop. vol. 51, p. 122 (2011)

[17] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G., Van Keer, R.: Keccak implementation overview (Version 3.2) (May 2012)

[18] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G., Van Keer, R.: Keyak. Submission to the CAESAR competition: http://competitions.cr.yp.to (2014)

[19] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G., Van Keer, R.: Ketje v2. Submission to the CAESAR competition: http://competitions.cr.yp.to (2016)

[20] Beyne, T., Chen, Y.L., Dobraunig, C., Mennink, B.: Dumbo, Jumbo, and Delirium: Parallel Authenticated Encryption for the Lightweight Circus. IACR Trans. Symmetric Cryptol. 2020(S1), 5–30 (2020), https://doi.org/10.13154/tosc.v2020.iS1.5-30

[21] Bi, W., Dong, X., Li, Z., Zong, R., Wang, X.: MILP-aided cube-attack-like cryptanalysis on Keccak Keyed modes. Des. Codes Cryptogr. 87(6), 1271–1296 (2019), https://doi.org/10.1007/s10623-018-0526-x

[22] Blondeau, C., Nyberg, K.: Links between Truncated Differential and Multidimensional Linear Properties of Block Ciphers and Underlying Attack Complexities. In: Nguyen and Oswald [87], pp. 165–182, https://doi.org/10.1007/978-3-642-55220-5_10

[23] Bogdanov, A., Knezevic, M., Leander, G., Toz, D., Varici, K., Verbauwhede, I.: Spongent: A Lightweight Hash Function. In: Preneel, B., Takagi, T. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings. LNCS, vol. 6917, pp. 312–325. Springer (2011), https://doi.org/10.1007/978-3-642-23951-9_21

[24] Bogdanov, A., Knezevic, M., Leander, G., Toz, D., Varici, K., Verbauwhede, I.: SPONGENT: The Design Space of Lightweight Cryptographic Hashing. IEEE Trans. Computers 62(10), 2041–2053 (2013), https://doi.org/10.1109/TC.2012.196

[25] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings. LNCS, vol. 4727, pp. 450–466. Springer (2007), https://doi.org/10.1007/978-3-540-74735-2_31

[26] Bonnetain, X., Jaques, S.: Quantum Period Finding against Symmetric Primitives in Practice. Cryptology ePrint Archive, Report 2020/1418 (2020)

[27] Boura, C., Canteaut, A.: A zero-sum property for the KECCAK-f permutation with 18 rounds. In: IEEE International Symposium on Information Theory, ISIT 2010, June 13-18, 2010, Austin, Texas, USA, Proceedings. pp. 2488–2492. IEEE (2010), https://doi.org/10.1109/ISIT.2010.5513442

[28] Boura, C., Canteaut, A.: Zero-Sum Distinguishers for Iterated Permutations and Application to Keccak-f and Hamsi-256. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. LNCS, vol. 6544, pp. 1–17. Springer (2010), https://doi.org/10.1007/978-3-642-19574-7_1

[29] Boura, C., Canteaut, A., De Cannière, C.: Higher-Order Differential Properties of Keccak and *Luffa*. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. LNCS, vol. 6733, pp. 252–269. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_15

[30] Chaigneau, C., Fuhr, T., Gilbert, H., Guo, J., Jean, J., Reinhard, J., Song, L.: Key-Recovery Attacks on Full Kravatte. IACR Transactions on Symmetric Cryptology 2018(1), 5–28 (2018), https://doi.org/10.13154/tosc.v2018.i1.5-28

[31] Chakraborti, A., Datta, N., Nandi, M., Yasuda, K.: Beetle Family of Lightweight and Secure Authenticated Encryption Ciphers. IACR Transactions on Cryptographic Hardware and Embedded Systems 2018(2), 218–241 (2018), https://doi.org/10.13154/tches.v2018.i2.218-241

[32] Chen, S., Steinberger, J.P.: Tight Security Bounds for Key-Alternating Ciphers. In: Nguyen and Oswald [87], pp. 327–350, https://doi.org/10.1007/978-3-642-55220-5_19

[33] Chen, Y., Gao, X.: Quantum Algorithms for Boolean Equation Solving and Quantum Algebraic Attack on Cryptosystems. Cryptology ePrint Archive, Report 2018/008 (2018)

[34] Cho, J.Y.: Linear Cryptanalysis of Reduced-Round PRESENT. In: Pieprzyk, J. (ed.) Topics in Cryptology - CT-RSA 2010, The Cryptographers' Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings. Lecture Notes in Computer Science, vol. 5985, pp. 302–317. Springer (2010), https://doi.org/10.1007/978-3-642-11925-5_21

[35] Cogliani, S., Maimut, D., Naccache, D., do Canto, R.P., Reyhanitabar, R., Vaudenay, S., Vizár, D.: Offset Merkle-Damgård (OMD) version 2.0. Submission to the CAESAR competition (2015)

[36] Collard, B., Standaert, F.: A Statistical Saturation Attack against the Block Cipher PRESENT. In: Fischlin, M. (ed.) Topics in Cryptology - CT-RSA 2009, The Cryptographers' Track at the RSA Conference 2009, San Francisco, CA, USA, April 20-24, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5473, pp. 195–210. Springer (2009), https://doi.org/10.1007/978-3-642-00862-7_13

[37] Daemen, J., Hoffert, S., Van Assche, G., Van Keer, R.: The design of Xoodoo and Xoofff. IACR Transactions on Symmetric Cryptology 2018(4), 1–38 (2018), https://doi.org/10.13154/tosc.v2018.i4.1-38

[38] Daemen, J., Mennink, B., Van Assche, G.: Full-State Keyed Duplex with Built-In Multi-user Support. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II. LNCS, vol. 10625, pp. 606–637. Springer (2017), https://doi.org/10.1007/978-3-319-70697-9_21

[39] Daemen, J., Van Assche, G.: Differential Propagation Analysis of Keccak. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. LNCS, vol. 7549, pp. 422–441. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_24

[40] Das, S., Meier, W.: Differential Biases in Reduced-Round Keccak. In: Pointcheval, D., Vergnaud, D. (eds.) Progress in Cryptology - AFRICACRYPT 2014 - 7th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 28-30, 2014. Proceedings. LNCS, vol. 8469, pp. 69–87. Springer (2014), https://doi.org/10.1007/978-3-319-06734-6_5

[41] Dinur, I., Dunkelman, O., Shamir, A.: New Attacks on Keccak-224 and Keccak-256. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. LNCS, vol. 7549, pp. 442–461. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_25

[42] Dinur, I., Dunkelman, O., Shamir, A.: Collision Attacks on Up to 5 Rounds of SHA-3 Using Generalized Internal Differentials. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. LNCS, vol. 8424, pp. 219–240. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_12

[43] Dinur, I., Morawiecki, P., Pieprzyk, J., Srebrny, M., Straus, M.: Practical Complexity Cube Attacks on Round-Reduced Keccak Sponge Function. Cryptology ePrint Archive, Report 2014/259 (2014)

[44] Dinur, I., Morawiecki, P., Pieprzyk, J., Srebrny, M., Straus, M.: Cube Attacks and Cube-Attack-Like Cryptanalysis on the Round-Reduced Keccak Sponge Function. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I. LNCS, vol. 9056, pp. 733–761. Springer (2015), https://doi.org/10.1007/978-3-662-46800-5_28

[45] Dinur, I., Shamir, A.: Cube Attacks on Tweakable Black Box Polynomials. In: Joux, A. (ed.) Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5479, pp. 278–299. Springer (2009), https://doi.org/10.1007/978-3-642-01001-9_16

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

[47] Dong, X., Li, Z., Wang, X., Qin, L.: Cube-like Attack on Round-Reduced Initialization of Ketje Sr. IACR Transactions on Symmetric Cryptology 2017(1), 259–280 (2017), https://doi.org/10.13154/tosc.v2017.i1.259-280

[48] Duan, M., Lai, X.: Improved zero-sum distinguisher for full round Keccak-f permutation. Cryptology ePrint Archive, Report 2011/023 (2011)

[49] Duc, A., Guo, J., Peyrin, T., Wei, L.: Unaligned Rebound Attack: Application to Keccak. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. LNCS, vol. 7549, pp. 402–421. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_23

[50] Eskandari, Z., Kidmose, A.B., Kölbl, S., Tiessen, T.: Finding Integral Distinguishers with Ease. In: Cid, C., Jr., M.J.J. (eds.) Selected Areas in Cryptography - SAC 2018 - 25th International Conference, Calgary, AB, Canada, August 15-17, 2018, Revised Selected Papers. Lecture Notes in Computer Science, vol. 11349, pp. 115–138. Springer (2018), https://doi.org/10.1007/978-3-030-10970-7_6

[51] FIPS 202: SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions (August 2015)

[52] Fuhr, T., Naya-Plasencia, M., Rotella, Y.: State-Recovery Attacks on Modified Ketje Jr. IACR Transactions on Symmetric Cryptology 2018(1), 29–56 (2018), https://doi.org/10.13154/tosc.v2018.i1.29-56

[53] Granger, R., Jovanovic, P., Mennink, B., Neves, S.: Improved Masking for Tweakable Blockciphers with Applications to Authenticated Encryption. In: Fischlin, M., Coron, J. (eds.) Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I. LNCS, vol. 9665, pp. 263–293. Springer (2016), https://doi.org/10.1007/978-3-662-49890-3_11

[54] Guo, J., Liao, G., Liu, G., Liu, M., Qiao, K., Song, L.: Practical Collision Attacks against Round-Reduced SHA-3. J. Cryptology 33(1), 228–270 (2020), https://doi.org/10.1007/s00145-019-09313-3

[55] Guo, J., Liu, M., Song, L.: Linear Structures: Applications to Cryptanalysis of Round-Reduced Keccak. In: Cheon, J.H., Takagi, T. (eds.) Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I. LNCS, vol. 10031, pp. 249–274 (2016), https://doi.org/10.1007/978-3-662-53887-6_9

[56] Guo, J., Peyrin, T., Poschmann, A.: The PHOTON Family of Lightweight Hash Functions. In: Rogaway, P. (ed.) Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings. LNCS, vol. 6841, pp. 222–239. Springer (2011), https://doi.org/10.1007/978-3-642-22792-9_13

[57] Hoang, V.T., Krovetz, T., Rogaway, P.: AEZ v5: Authenticated Encryption by Enciphering. Submission to the CAESAR competition (2017)

[58] Huang, S., Wang, X., Xu, G., Wang, M., Zhao, J.: Conditional Cube Attack on Reduced-Round Keccak Sponge Function. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part II. LNCS, vol. 10211, pp. 259–288 (2017), https://doi.org/10.1007/978-3-319-56614-6_9

[59] ISO/IEC 29192-5:2016. Information technology – Security techniques – Lightweight cryptography – Part 5: Hash-functions (2016)

[60] Jean, J., Nikolic, I.: Internal Differential Boomerangs: Practical Analysis of the Round-Reduced Keccak- f f Permutation. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul,

Turkey, March 8-11, 2015, Revised Selected Papers. LNCS, vol. 9054, pp. 537–556. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_26

[61] Kavun, E., Lauridsen, M., Leander, G., Rechberger, C., Schwabe, P., Yalçın, T.: Prøst v1.1 (2014), submission to CAESAR competition

[62] Kavun, E.B., Yalçin, T.: A Lightweight Implementation of Keccak Hash Function for Radio-Frequency Identification Applications. In: Yalcin, S.B.O. (ed.) Radio Frequency Identification: Security and Privacy Issues - 6th International Workshop, RFIDSec 2010, Istanbul, Turkey, June 8-9, 2010, Revised Selected Papers. LNCS, vol. 6370, pp. 258–269. Springer (2010), https://doi.org/10.1007/978-3-642-16822-2_20

[63] Kölbl, S., Mendel, F., Nad, T., Schläffer, M.: Differential Cryptanalysis of Keccak Variants. In: Stam, M. (ed.) Cryptography and Coding - 14th IMA International Conference, IMACC 2013, Oxford, UK, December 17-19, 2013. Proceedings. LNCS, vol. 8308, pp. 141–157. Springer (2013), https://doi.org/10.1007/978-3-642-45239-0_9

[64] Krovetz, T., Rogaway, P.: The Software Performance of Authenticated-Encryption Modes. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. LNCS, vol. 6733, pp. 306–327. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_18

[65] Kuila, S., Saha, D., Pal, M., Chowdhury, D.R.: Practical Distinguishers against 6-Round Keccak-f Exploiting Self-Symmetry. In: Pointcheval, D., Vergnaud, D. (eds.) Progress in Cryptology - AFRICACRYPT 2014 - 7th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 28-30, 2014. Proceedings. LNCS, vol. 8469, pp. 88–108. Springer (2014), https://doi.org/10.1007/978-3-319-06734-6_6

[66] Kumar, R., Mittal, N., Singh, S.: Cryptanalysis of 2 Round Keccak-384. In: Chakraborty, D., Iwata, T. (eds.) Progress in Cryptology - INDOCRYPT 2018 - 19th International Conference on Cryptology in India, New Delhi, India, December 9-12, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11356, pp. 120–133. Springer (2018), https://doi.org/10.1007/978-3-030-05378-9_7

[67] Kumar, R., Rajasree, M.S., AlKhzaimi, H.: Cryptanalysis of 1-Round KECCAK. In: Joux, A., Nitaj, A., Rachidi, T. (eds.) Progress in Cryptology - AFRICACRYPT 2018 - 10th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 7-9, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10831, pp. 124–137. Springer (2018), https://doi.org/10.1007/978-3-319-89339-6_8

[68] Lathrop, J.: Cube attacks on cryptographic hash functions. Thesis: https://scholarworks.rit.edu/theses/650/ (2009)

[69] Leander, G.: On Linear Hulls, Statistical Saturation Attacks, PRESENT and a Cryptanalysis of PUFFIN. In: Paterson, K.G. (ed.) Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6632, pp. 303–322. Springer (2011), https://doi.org/10.1007/978-3-642-20465-4_18

[70] Li, M., Cheng, L.: Distinguishing Property for Full Round KECCAK-f Permutation. In: Barolli, L., Terzo, O. (eds.) Complex, Intelligent, and Software Intensive Systems - Proceedings of the 11th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2017), Torino, Italy, July 10-12, 2017. Advances in Intelligent Systems and Computing, vol. 611, pp. 639–646. Springer (2017), https://doi.org/10.1007/978-3-319-61566-0_59

[71] Li, T., Sun, Y.: Preimage Attacks on Round-Reduced Keccak-224/256 via an Allocating Approach. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11478, pp. 556–584. Springer (2019), https://doi.org/10.1007/978-3-030-17659-4_19

[72] Li, T., Sun, Y., Liao, M., Wang, D.: Preimage Attacks on the Round-reduced Keccak with Cross-linear Structures. IACR Transactions on Symmetric Cryptology 2017(4), 39–57 (2017), https://doi.org/10.13154/tosc.v2017.i4.39-57

[73] Li, Z., Bi, W., Dong, X., Wang, X.: Improved Conditional Cube Attacks on Keccak Keyed Modes with MILP Method. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I. LNCS, vol. 10624, pp. 99–127. Springer (2017), https://doi.org/10.1007/978-3-319-70694-8_4

[74] Li, Z., Dong, X., Bi, W., Jia, K., Wang, X., Meier, W.: New Conditional Cube Attack on Keccak Keyed Modes. IACR Trans. Symmetric Cryptol. 2019(2), 94–124 (2019), https://doi.org/10.13154/tosc.v2019.i2.94-124

[75] Liu, F., Cao, Z., Wang, G.: Finding Ordinary Cube Variables for Keccak-MAC with Greedy Algorithm. In: Attrapadung, N., Yagi, T. (eds.) Advances in Information and Computer Security - 14th International Workshop on Security, IWSEC 2019, Tokyo, Japan, August 28-30, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11689, pp. 287–305. Springer (2019), https://doi.org/10.1007/978-3-030-26834-3_17

[76] Liu, F., Isobe, T., Meier, W., Yang, Z.: Algebraic Attacks on Round-Reduced Keccak/Xoodoo. Cryptology ePrint Archive, Report 2020/346 (2020)

[77] Luykx, A., Preneel, B., Tischhauser, E., Yasuda, K.: A MAC Mode for Lightweight Block Ciphers. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 43–59. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_3

[78] Mella, S., Daemen, J., Van Assche, G.: New techniques for trail bounds and application to differential trails in Keccak. IACR Transactions on Symmetric Cryptology 2017(1), 329–357 (2017), https://doi.org/10.13154/tosc.v2017.i1.329-357

[79] 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.) Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II. LNCS, vol. 9453, pp. 465–489. Springer (2015), https://doi.org/10.1007/978-3-662-48800-3_19

[80] Minematsu, K.: AES-OTR v3.1. Submission to the CAESAR competition (2016)

[81] Morawiecki, P., Pieprzyk, J., Srebrny, M.: Rotational Cryptanalysis of Round-Reduced Keccak. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. LNCS, vol. 8424, pp. 241–262. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_13

[82] Morawiecki, P., Pieprzyk, J., Srebrny, M., Straus, M.: Preimage attacks on the round-reduced Keccak with the aid of differential cryptanalysis. Cryptology ePrint Archive, Report 2013/561 (2013)

[83] Morawiecki, P., Srebrny, M.: A SAT-based preimage analysis of reduced Keccak hash functions. Inf. Process. Lett. 113(10-11), 392–397 (2013), https://doi.org/10.1016/j.ipl.2013.03.004

[84] Namprempre, C., Rogaway, P., Shrimpton, T.: Reconsidering Generic Composition. In: Nguyen and Oswald [87], pp. 257–274, https://doi.org/10.1007/978-3-642-55220-5_15

[85] National Institute of Standards and Technology (NIST): Submission requirements and evaluation criteria for the lightweight cryptography standardization process (Aug 2018)

[86] Naya-Plasencia, M., Röck, A., Meier, W.: Practical Analysis of Reduced-Round Keccak. In: Bernstein, D.J., Chatterjee, S. (eds.) Progress in Cryptology - INDOCRYPT 2011 - 12th International Conference on Cryptology in India, Chennai, India, December 11-14, 2011. Proceedings. LNCS, vol. 7107, pp. 236–254. Springer (2011), https://doi.org/10.1007/978-3-642-25578-6_18

[87] Nguyen, P.Q., Oswald, E. (eds.): Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings, LNCS, vol. 8441. Springer (2014), https://doi.org/10.1007/978-3-642-55220-5

[88] Nikova, S., Rechberger, C., Rijmen, V.: Threshold Implementations Against Side-Channel Attacks and Glitches. In: Ning, P., Qing, S., Li, N. (eds.) Information and Communications Security, 8th International Conference, ICICS 2006, Raleigh, NC, USA, December 4-7, 2006, Proceedings. LNCS, vol. 4307, pp. 529–545. Springer (2006), https://doi.org/10.1007/11935308_38

[89] Patarin, J.: The "Coefficients H" Technique. In: Avanzi, R.M., Keliher, L., Sica, F. (eds.) Selected Areas in Cryptography, 15th International Workshop, SAC 2008, Sackville, New Brunswick, Canada, August 14-15, Revised Selected Papers. LNCS, vol. 5381, pp. 328–345. Springer (2008), https://doi.org/10.1007/978-3-642-04159-4_21

[90] Qiao, K., Song, L., Liu, M., Guo, J.: New Collision Attacks on Round-Reduced Keccak. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part III. LNCS, vol. 10212, pp. 216–243 (2017), https://doi.org/10.1007/978-3-319-56617-7_8

[91] Rajasree, M.: Cryptanalysis of Round-Reduced KECCAK using Non-Linear Structures. Cryptology ePrint Archive, Report 2019/884 (2019)

[92] Rogaway, P.: Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In: Lee, P.J. (ed.) Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5-9, 2004, Proceedings. LNCS, vol. 3329, pp. 16–31. Springer (2004), https://doi.org/10.1007/978-3-540-30539-2_2

[93] Rogaway, P., Bellare, M., Black, J., Krovetz, T.: OCB: a block-cipher mode of operation for efficient authenticated encryption. In: Reiter, M.K., Samarati, P. (eds.) CCS 2001, Proceedings of the 8th ACM Conference on Computer and Communications Security, Philadelphia, Pennsylvania, USA, November 6-8, 2001. pp. 196–205. ACM (2001), https://doi.org/10.1145/501983.502011

[94] Saha, D., Kuila, S., Chowdhury, D.R.: SymSum: Symmetric-Sum Distinguishers Against Round Reduced SHA3. IACR Transactions on Symmetric Cryptology 2017(1), 240–258 (2017), https://doi.org/10.13154/tosc.v2017.i1.240-258

[95] Sasaki, Y., Todo, Y., Aoki, K., Naito, Y., Sugawara, T., Murakami, Y., Matsui, M., Hirose, S.: Minalpher v1.1. Submission to the CAESAR competition (2015)

[96] Shoup, V.: On Fast and Provably Secure Message Authentication Based on Universal Hashing. In: Koblitz, N. (ed.) Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings. LNCS, vol. 1109, pp. 313–328. Springer (1996), https://doi.org/10.1007/3-540-68697-5_24

[97] Song, L., Guo, J.: Cube-Attack-Like Cryptanalysis of Round-Reduced Keccak Using MILP. IACR Trans. Symmetric Cryptol. 2018(3), 182–214 (2018), https://doi.org/10.13154/tosc.v2018.i3.182-214

[98] Song, L., Guo, J., Shi, D.: New MILP Modeling: Improved Conditional Cube Attacks to Keccak-based Constructions. Cryptology ePrint Archive, Report 2017/1030 (2017)

[99] Song, L., Guo, J., Shi, D., Ling, S.: New MILP Modeling: Improved Conditional Cube Attacks on Keccak-Based Constructions. In: Peyrin, T., Galbraith, S.D. (eds.) Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11273, pp. 65–95. Springer (2018), https://doi.org/10.1007/978-3-030-03329-3_3

[100] Song, L., Liao, G., Guo, J.: Non-full Sbox Linearization: Applications to Collision Attacks on Round-Reduced Keccak. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part II. LNCS, vol. 10402, pp. 428–451. Springer (2017), https://doi.org/10.1007/978-3-319-63715-0_15

[101] Sun, B., Liu, Z., Rijmen, V., Li, R., Cheng, L., Wang, Q., AlKhzaimi, H., Li, C.: Links Among Impossible Differential, Integral and Zero Correlation Linear Cryptanalysis. In: Gennaro, R., Robshaw, M. (eds.) Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9215, pp. 95–115. Springer (2015), https://doi.org/10.1007/978-3-662-47989-6_5

[102] Sun, L., Wang, M.: Towards a Further Understanding of Bit-Based Division Property. Cryptology ePrint Archive, Report 2016/392 (withdrawn) (2016)

[103] Sun, L., Wang, W., Wang, M.: MILP-Aided Bit-Based Division Property for Primitives with Non-Bit-Permutation Linear Layers. Cryptology ePrint Archive, Report 2016/811 (2016)

[104] Sun, L., Wang, W., Wang, M.: MILP-aided bit-based division property for primitives with non-bit-permutation linear layers. IET Inf. Secur. 14(1), 12–20 (2020), https://doi.org/10.1049/iet-ifs.2018.5283

[105] The Sage Developers: SageMath, the Sage Mathematics Software System (Version 8.1) (2017), https://www.sagemath.org

[106] Wang, L.: SHELL v2.0. Submission to the CAESAR competition (2015)

[107] Wegman, M.N., Carter, L.: New Hash Functions and Their Use in Authentication and Set Equality. J. Comput. Syst. Sci. 22(3), 265–279 (1981), https://doi.org/10.1016/0022-0000(81)90033-7

[108] Ye, C., Tian, T.: New Insights into Divide-and-Conquer Attacks on the Round-Reduced Keccak-MAC. Cryptology ePrint Archive, Report 2018/059 (2018)

[109] Zhang, G., Liu, M.: A distinguisher on PRESENT-like permutations with application to SPONGENT. SCIENCE CHINA Information Sciences 60(7), 72101 (2017), https://doi.org/10.1007/s11432-016-0165-6

[110] Zhou, H., Li, Z., Dong, X., Jia, K., Meier, W., Ashur, T.: Practical Key-Recovery Attacks On Round-Reduced Ketje Jr, Xoodoo-AE And Xoodyak. Comput. J. 63(8), 1231–1246 (2020), https://doi.org/10.1093/comjnl/bxz152

[111] Zhou, H., Zong, R., Dong, X., Jia, K., Meier, W.: Interpolation Attacks on Round-Reduced Elephant, Kravatte and Xoofff. Cryptology ePrint Archive, Report 2020/781 (2020)

A List of Cryptanalysis

A.1 Dedicated Analysis

- Bonnetain, X., Jaques, S.: Quantum Period Finding against Symmetric Primitives in Practice. Cryptology ePrint Archive, Report 2020/1418 (2020)
- Zhou, H., Zong, R., Dong, X., Jia, K., Meier, W.: Interpolation Attacks on Round-Reduced Elephant, Kravatte and Xoofff. Cryptology ePrint Archive, Report 2020/781 (2020)
- Beyne, T., Chen, Y.L., Dobraunig, C., Mennink, B.: Dumbo, Jumbo, and Delirium: Parallel Authenticated Encryption for the Lightweight Circus. IACR Trans. Symmetric Cryptol. 2020(S1), 5–30 (2020), https://doi.org/10.13154/tosc.v2020.iS1.5-30

A.2 Spongent Permutation

- Sun, L., Wang, W., Wang, M.: MILP-aided bit-based division property for primitives with non-bit-permutation linear layers. IET Inf. Secur. 14(1), 12–20 (2020), https://doi.org/10.1049/iet-ifs.2018.5283
- Eskandari, Z., Kidmose, A.B., Kölbl, S., Tiessen, T.: Finding Integral Distinguishers with Ease. In: Cid, C., Jr., M.J.J. (eds.) Selected Areas in Cryptography - SAC 2018 - 25th International Conference, Calgary, AB, Canada, August 15-17, 2018, Revised Selected Papers. Lecture Notes in Computer Science, vol. 11349, pp. 115–138. Springer (2018), https://doi.org/10.1007/978-3-030-10970-7_6
- Zhang, G., Liu, M.: A distinguisher on PRESENT-like permutations with application to SPONGENT. SCIENCE CHINA Information Sciences 60(7), 72101 (2017), https://doi.org/10.1007/s11432-016-0165-6
- Sun, L., Wang, W., Wang, M.: MILP-Aided Bit-Based Division Property for Primitives with Non-Bit-Permutation Linear Layers. Cryptology ePrint Archive, Report 2016/811 (2016)
- Sun, L., Wang, M.: Towards a Further Understanding of Bit-Based Division Property. Cryptology ePrint Archive, Report 2016/392 (withdrawn) (2016)
- Bogdanov, A., Knezevic, M., Leander, G., Toz, D., Varici, K., Verbauwhede, I.: SPONGENT: The Design Space of Lightweight Cryptographic Hashing. IEEE Trans. Computers 62(10), 2041–2053 (2013), https://doi.org/10.1109/TC.2012.196
- Abdelraheem, M.A.: Estimating the Probabilities of Low-Weight Differential and Linear Approximations on PRESENT-Like Ciphers. In: Kwon,

T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. LNCS, vol. 7839, pp. 368–382. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_26

- Bogdanov, A., Knezevic, M., Leander, G., Toz, D., Varici, K., Verbauwhede, I.: Spongent: A Lightweight Hash Function. In: Preneel, B., Takagi, T. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings. LNCS, vol. 6917, pp. 312–325. Springer (2011), https://doi.org/10.1007/978-3-642-23951-9_21

A.3 Keccak Permutation

- Liu, F., Isobe, T., Meier, W., Yang, Z.: Algebraic Attacks on Round-Reduced Keccak/Xoodoo. Cryptology ePrint Archive, Report 2020/346 (2020)
- Zhou, H., Li, Z., Dong, X., Jia, K., Meier, W., Ashur, T.: Practical Key-Recovery Attacks On Round-Reduced Ketje Jr, Xoodoo-AE And Xoodyak. Comput. J. 63(8), 1231–1246 (2020), https://doi.org/10.1093/comjnl/bxz152
- Guo, J., Liao, G., Liu, G., Liu, M., Qiao, K., Song, L.: Practical Collision Attacks against Round-Reduced SHA-3. J. Cryptology 33(1), 228–270 (2020), https://doi.org/10.1007/s00145-019-09313-3
- Liu, F., Cao, Z., Wang, G.: Finding Ordinary Cube Variables for Keccak-MAC with Greedy Algorithm. In: Attrapadung, N., Yagi, T. (eds.) Advances in Information and Computer Security - 14th International Workshop on Security, IWSEC 2019, Tokyo, Japan, August 28-30, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11689, pp. 287–305. Springer (2019), https://doi.org/10.1007/978-3-030-26834-3_17
- Li, Z., Dong, X., Bi, W., Jia, K., Wang, X., Meier, W.: New Conditional Cube Attack on Keccak Keyed Modes. IACR Trans. Symmetric Cryptol. 2019(2), 94–124 (2019), https://doi.org/10.13154/tosc.v2019.i2.94-124
- Rajasree, M.: Cryptanalysis of Round-Reduced KECCAK using Non-Linear Structures. Cryptology ePrint Archive, Report 2019/884 (2019)
- Li, T., Sun, Y.: Preimage Attacks on Round-Reduced Keccak-224/256 via an Allocating Approach. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11478, pp. 556–584. Springer (2019), https://doi.org/10.1007/978-3-030-17659-4_19

- Bi, W., Dong, X., Li, Z., Zong, R., Wang, X.: MILP-aided cube-attack-like cryptanalysis on Keccak Keyed modes. Des. Codes Cryptogr. 87(6), 1271–1296 (2019), https://doi.org/10.1007/s10623-018-0526-x
- Song, L., Guo, J.: Cube-Attack-Like Cryptanalysis of Round-Reduced Keccak Using MILP. IACR Trans. Symmetric Cryptol. 2018(3), 182–214 (2018), https://doi.org/10.13154/tosc.v2018.i3.182-214
- Kumar, R., Mittal, N., Singh, S.: Cryptanalysis of 2 Round Keccak-384. In: Chakraborty, D., Iwata, T. (eds.) Progress in Cryptology - INDOCRYPT 2018 - 19th International Conference on Cryptology in India, New Delhi, India, December 9-12, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11356, pp. 120–133. Springer (2018), https://doi.org/10.1007/978-3-030-05378-9_7
- Chen, Y., Gao, X.: Quantum Algorithms for Boolean Equation Solving and Quantum Algebraic Attack on Cryptosystems. Cryptology ePrint Archive, Report 2018/008 (2018)
- Kumar, R., Rajasree, M.S., AlKhzaimi, H.: Cryptanalysis of 1-Round KECCAK. In: Joux, A., Nitaj, A., Rachidi, T. (eds.) Progress in Cryptology - AFRICACRYPT 2018 - 10th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 7-9, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10831, pp. 124–137. Springer (2018), https://doi.org/10.1007/978-3-319-89339-6_8
- Chaigneau, C., Fuhr, T., Gilbert, H., Guo, J., Jean, J., Reinhard, J., Song, L.: Key-Recovery Attacks on Full Kravatte. IACR Transactions on Symmetric Cryptology 2018(1), 5–28 (2018), https://doi.org/10.13154/tosc.v2018.i1.5-28
- Fuhr, T., Naya-Plasencia, M., Rotella, Y.: State-Recovery Attacks on Modified Ketje Jr. IACR Transactions on Symmetric Cryptology 2018(1), 29–56 (2018), https://doi.org/10.13154/tosc.v2018.i1.29-56
- Ye, C., Tian, T.: New Insights into Divide-and-Conquer Attacks on the Round-Reduced Keccak-MAC. Cryptology ePrint Archive, Report 2018/059 (2018)
- Song, L., Guo, J., Shi, D.: New MILP Modeling: Improved Conditional Cube Attacks to Keccak-based Constructions. Cryptology ePrint Archive, Report 2017/1030 (2017)
- Li, Z., Bi, W., Dong, X., Wang, X.: Improved Conditional Cube Attacks on Keccak Keyed Modes with MILP Method. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I. LNCS, vol. 10624, pp. 99–127. Springer (2017), https://doi.org/10.1007/978-3-319-70694-8_4

- Li, T., Sun, Y., Liao, M., Wang, D.: Preimage Attacks on the Round-reduced Keccak with Cross-linear Structures. IACR Transactions on Symmetric Cryptology 2017(4), 39–57 (2017), https://doi.org/10.13154/tosc.v2017.i4.39-57

- Dong, X., Li, Z., Wang, X., Qin, L.: Cube-like Attack on Round-Reduced Initialization of Ketje Sr. IACR Transactions on Symmetric Cryptology 2017(1), 259–280 (2017), https://doi.org/10.13154/tosc.v2017.i1.259-280

- Mella, S., Daemen, J., Van Assche, G.: New techniques for trail bounds and application to differential trails in Keccak. IACR Transactions on Symmetric Cryptology 2017(1), 329–357 (2017), https://doi.org/10.13154/tosc.v2017.i1.329-357

- Li, M., Cheng, L.: Distinguishing Property for Full Round KECCAK-f Permutation. In: Barolli, L., Terzo, O. (eds.) Complex, Intelligent, and Software Intensive Systems - Proceedings of the 11th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2017), Torino, Italy, July 10-12, 2017. Advances in Intelligent Systems and Computing, vol. 611, pp. 639–646. Springer (2017), https://doi.org/10.1007/978-3-319-61566-0_59

- Song, L., Liao, G., Guo, J.: Non-full Sbox Linearization: Applications to Collision Attacks on Round-Reduced Keccak. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part II. LNCS, vol. 10402, pp. 428–451. Springer (2017), https://doi.org/10.1007/978-3-319-63715-0_15

- Huang, S., Wang, X., Xu, G., Wang, M., Zhao, J.: Conditional Cube Attack on Reduced-Round Keccak Sponge Function. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part II. LNCS, vol. 10211, pp. 259–288 (2017), https://doi.org/10.1007/978-3-319-56614-6_9

- Qiao, K., Song, L., Liu, M., Guo, J.: New Collision Attacks on Round-Reduced Keccak. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part III. LNCS, vol. 10212, pp. 216–243 (2017), https://doi.org/10.1007/978-3-319-56617-7_8

- Saha, D., Kuila, S., Chowdhury, D.R.: SymSum: Symmetric-Sum Distinguishers Against Round Reduced SHA3. IACR Transactions on Symmetric Cryptology 2017(1), 240–258 (2017), https://doi.org/10.13154/tosc.v2017.i1.240-258

- Guo, J., Liu, M., Song, L.: Linear Structures: Applications to Cryptanalysis of Round-Reduced Keccak. In: Cheon, J.H., Takagi, T. (eds.) Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I. LNCS, vol. 10031, pp. 249–274 (2016), https://doi.org/10.1007/978-3-662-53887-6_9

- Dinur, I., Morawiecki, P., Pieprzyk, J., Srebrny, M., Straus, M.: Cube Attacks and Cube-Attack-Like Cryptanalysis on the Round-Reduced Keccak Sponge Function. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I. LNCS, vol. 9056, pp. 733–761. Springer (2015), https://doi.org/10.1007/978-3-662-46800-5_28

- Jean, J., Nikolic, I.: Internal Differential Boomerangs: Practical Analysis of the Round-Reduced Keccak- f f Permutation. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. LNCS, vol. 9054, pp. 537–556. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_26

- Kuila, S., Saha, D., Pal, M., Chowdhury, D.R.: Practical Distinguishers against 6-Round Keccak-f Exploiting Self-Symmetry. In: Pointcheval, D., Vergnaud, D. (eds.) Progress in Cryptology - AFRICACRYPT 2014 - 7th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 28-30, 2014. Proceedings. LNCS, vol. 8469, pp. 88–108. Springer (2014), https://doi.org/10.1007/978-3-319-06734-6_6

- Das, S., Meier, W.: Differential Biases in Reduced-Round Keccak. In: Pointcheval, D., Vergnaud, D. (eds.) Progress in Cryptology - AFRICACRYPT 2014 - 7th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 28-30, 2014. Proceedings. LNCS, vol. 8469, pp. 69–87. Springer (2014), https://doi.org/10.1007/978-3-319-06734-6_5

- Dinur, I., Morawiecki, P., Pieprzyk, J., Srebrny, M., Straus, M.: Practical Complexity Cube Attacks on Round-Reduced Keccak Sponge Function. Cryptology ePrint Archive, Report 2014/259 (2014)

- Morawiecki, P., Pieprzyk, J., Srebrny, M., Straus, M.: Preimage attacks on the round-reduced Keccak with the aid of differential cryptanalysis. Cryptology ePrint Archive, Report 2013/561 (2013)

- Kölbl, S., Mendel, F., Nad, T., Schläffer, M.: Differential Cryptanalysis of Keccak Variants. In: Stam, M. (ed.) Cryptography and Coding - 14th IMA International Conference, IMACC 2013, Oxford, UK, December 17-19, 2013. Proceedings. LNCS, vol. 8308, pp. 141–157. Springer (2013), https://doi.org/10.1007/978-3-642-45239-0_9

- Dinur, I., Dunkelman, O., Shamir, A.: Collision Attacks on Up to 5 Rounds of SHA-3 Using Generalized Internal Differentials. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. LNCS, vol. 8424, pp. 219–240. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_12

- Morawiecki, P., Pieprzyk, J., Srebrny, M.: Rotational Cryptanalysis of Round-Reduced Keccak. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. LNCS, vol. 8424, pp. 241–262. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_13

- Morawiecki, P., Srebrny, M.: A SAT-based preimage analysis of reduced Keccak hash functions. Inf. Process. Lett. 113(10-11), 392–397 (2013), https://doi.org/10.1016/j.ipl.2013.03.004

- Dinur, I., Dunkelman, O., Shamir, A.: New Attacks on Keccak-224 and Keccak-256. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. LNCS, vol. 7549, pp. 442–461. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_25

- Duc, A., Guo, J., Peyrin, T., Wei, L.: Unaligned Rebound Attack: Application to Keccak. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. LNCS, vol. 7549, pp. 402–421. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_23

- Daemen, J., Van Assche, G.: Differential Propagation Analysis of Keccak. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. LNCS, vol. 7549, pp. 422–441. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_24

- Naya-Plasencia, M., Röck, A., Meier, W.: Practical Analysis of Reduced-Round Keccak. In: Bernstein, D.J., Chatterjee, S. (eds.) Progress in Cryptology - INDOCRYPT 2011 - 12th International Conference on Cryptology in India, Chennai, India, December 11-14, 2011. Proceedings. LNCS, vol. 7107, pp. 236–254. Springer (2011), https://doi.org/10.1007/978-3-642-25578-6_18

- Duan, M., Lai, X.: Improved zero-sum distinguisher for full round Keccak-f permutation. Cryptology ePrint Archive, Report 2011/023 (2011)

- Boura, C., Canteaut, A., De Cannière, C.: Higher-Order Differential Properties of Keccak and *Luffa*. In: Joux, A. (ed.) Fast Software Encryption -

18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. LNCS, vol. 6733, pp. 252–269. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_15

- Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: The Keccak reference (January 2011)
- Boura, C., Canteaut, A.: Zero-Sum Distinguishers for Iterated Permutations and Application to Keccak-f and Hamsi-256. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. LNCS, vol. 6544, pp. 1–17. Springer (2010), https://doi.org/10.1007/978-3-642-19574-7_1
- Boura, C., Canteaut, A.: A zero-sum property for the KECCAK-f permutation with 18 rounds. In: IEEE International Symposium on Information Theory, ISIT 2010, June 13-18, 2010, Austin, Texas, USA, Proceedings. pp. 2488–2492. IEEE (2010), https://doi.org/10.1109/ISIT.2010.5513442
- Bernstein, D.J.: Second preimages for 6 (7? (8??)) rounds of Keccak? NIST hash forum: http://cr.yp.to/hash/keccak-20101127.txt (2009)
- Lathrop, J.: Cube attacks on cryptographic hash functions. Thesis: https://scholarworks.rit.edu/theses/650/ (2009)
- Aumasson, J.P., Khovratovich, D.: Zero-sum distinguishers for reduced Keccak-f for the core functions of Luffa and Hamsi. Rump session of CHES: https://131002.net/data/papers/AM09.pdf (2009)
- Aumasson, J.P., Khovratovich, D.: First Analysis of Keccak. NIST hash forum: https://131002.net/data/papers/AK09.pdf (2009)

B Security of Elephant Mode

We describe the security model in Section B.1, introduce a simplified version of masked Even-Mansour in Section B.2, and state the formal security result on Elephant in Section B.3. We discuss the implication of this result for the three instances Dumbo, Jumbo, and Delirium in Section B.4.

B.1 Security Model

For a finite set $\mathcal{T}$, we denote by $\mathrm{perm}(n)$ the set of all n-bit permutations and by $\mathrm{perm}(\mathcal{T}, n)$ the set of all families of permutations indexed by $T \in \mathcal{T}$. We denote by $\mathrm{func}(n)$ the set of all n-bit functions. For a finite set $\mathcal{S}$, we denote by $s \xleftarrow{\$} \mathcal{S}$ the uniform random sampling of an element s from $\mathcal{S}$.

An adversary $\mathcal{A}$ is an algorithm that is given access to one or more oracles $\mathcal{O}$, and after interaction with $\mathcal{O}$ outputs a bit $b \in \{0,1\}$. This event is denoted as $\mathcal{A}^{\mathcal{O}} \to b$. In our work, we will be concerned with computationally unbounded adversaries $\mathcal{A}$; their complexities are only measured by the number of oracle queries. For two randomized oracles $\mathcal{O}, \mathcal{P}$, we denote the advantage of an adversary $\mathcal{A}$ in distinguishing both by

$$\Delta_{\mathcal{A}}\left(\mathcal{O}\ ;\ \mathcal{P}\right) = \mathbf{Pr}\left(\mathcal{A}^{\mathcal{O}} \to 1\right) - \mathbf{Pr}\left(\mathcal{A}^{\mathcal{P}} \to 1\right) . \tag{8}$$

Finally, let $k, m, n, t, \mu \in \mathbb{N}$ with $k, m, t \leq n$ throughout.

B.1.1 Authenticated Encryption

An authenticated encryption scheme AE consists of two algorithms enc and dec. Encryption enc gets as input a key $K \in \{0,1\}^k$, a nonce $N \in \{0,1\}^m$, associated data $A \in \{0,1\}^*$, and a message $M \in \{0,1\}^*$, and it outputs a ciphertext $C \in \{0,1\}^{|M|}$ and a tag $T \in \{0,1\}^t$. Decryption dec gets as input a key $K \in \{0,1\}^k$, a nonce $N \in \{0,1\}^m$, associated data $A \in \{0,1\}^*$, a ciphertext $C \in \{0,1\}^*$, and a tag $T \in \{0,1\}^t$, and it outputs a message $M \in \{0,1\}^{|C|}$ if the tag is correct, or a dedicated $\perp$-sign otherwise. The two functions are required to satisfy

$$\mathsf{dec}(K, N, A, \mathsf{enc}(K, N, A, M)) = M .$$

In our work, the authenticated encryption scheme AE is based on an n-bit permutation P, which is modeled as a random permutation: $\mathsf{P} \xleftarrow{\$} \mathrm{perm}(n)$. We will consider multi-user security of AE, where an adversary can query up to μ versions of the scheme. The multi-user security of AE against an adversary $\mathcal{A}$ is defined as

$$\mathbf{Adv}^{\mu\text{-ae}}_{\mathsf{AE}}(\mathcal{A}) = \Delta_{\mathcal{A}}\left(\left(\mathsf{enc}^{\mathsf{P}}_{K_j}, \mathsf{dec}^{\mathsf{P}}_{K_j}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm}\ ;\ \left(\mathsf{rand}_j, \perp\right)_{j=1}^{\mu}, \mathsf{P}^{\pm}\right) , \tag{9}$$

where the randomness of the oracles is taken over $K_1, \ldots, K_\mu \xleftarrow{\$} \{0,1\}^k$, $\mathsf{P} \xleftarrow{\$} \mathrm{perm}(n)$, and the functions $\mathsf{rand}_1, \ldots, \mathsf{rand}_\mu$ that for each input (N, A, M) return a random string of size $|M|+t$ bits. The superscript $^{\pm}$ indicates two-sided access by $\mathcal{A}$. The function $\perp$ returns the $\perp$-sign for each query.

We only consider *nonce-respecting* adversaries: $\mathcal{A}$ is not allowed to make two encryption queries to the same oracle $j \in \{1, \ldots, \mu\}$ for the same nonce. It is also not allowed to relay the output of the encryption oracle (enc_{K_j} in the real world and rand_j in the ideal world) to the decryption oracle (dec_{K_j} in the real world and $\perp$ in the ideal world).

B.1.2 Tweakable Block Ciphers

A tweakable block cipher $\widetilde{\mathsf{E}}$ is a function that gets as input a key $K \in \{0,1\}^k$, tweak $T \in \mathcal{T}$,[2] and message $M \in \{0,1\}^n$, and it outputs a ciphertext $C \in$

[2] In our application, the tweak space is of a specific form and cannot be conveniently expressed as a set of binary strings.

$\{0,1\}^n$. The tweakable block cipher is required to be bijective for any fixed (K, T).

In our application, we will not make use of the inverse $\widetilde{\mathsf{E}}^{-1}$. More importantly, for our authenticated encryption scheme it suffices to use a tweakable block cipher that is secure against adversaries that only have access to $\widetilde{\mathsf{E}}$, and not to $\widetilde{\mathsf{E}}^{-1}$. The tweakable block cipher considered in this work is based on an n-bit permutation P, which is modeled as a random permutation: $\mathsf{P} \xleftarrow{\$} \text{perm}(n)$. We will consider multi-user security of $\widetilde{\mathsf{E}}$, where an adversary can query up to μ versions of the scheme. The security of $\widetilde{\mathsf{E}}$ against an adversary $\mathcal{A}$ is defined as

$$\mathbf{Adv}_{\widetilde{\mathsf{E}}}^{\mu\text{-tprp}}(\mathcal{A}) = \Delta_{\mathcal{A}}\left(\left(\widetilde{\mathsf{E}}_{K_j}^{\mathsf{P}}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \; ; \; \left(\widetilde{\pi}_j\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \right), \tag{10}$$

where the randomness of the oracles is taken over $K_1, \ldots, K_\mu \xleftarrow{\$} \{0,1\}^k$, $\mathsf{P} \xleftarrow{\$} \text{perm}(n)$, and $\widetilde{\pi}_1, \ldots, \widetilde{\pi}_\mu \xleftarrow{\$} \text{perm}(\mathcal{T}, n)$.

B.2 Simplified Masked Even-Mansour

The Elephant authenticated encryption family uses its underlying permutation in a "Masked Even-Mansour" (MEM) construction [53]: the input to and output of the permutation P are masked using an LFSR evaluated on the secret key. However, the tweakable block cipher used in our proposal is simpler than the original construction in two ways: (i) the tweak only consists of the exponents of the LFSRs and not the nonce and (ii) in our application, the tweakable block cipher is only evaluated in the forward direction. The changes are not huge, but they do allow for a simpler description, security analysis, and bound. We will refer to this scheme as SiM (Simplified MEM). For generality, we will keep the formalization for an arbitrary amount of LFSRs, even though we will only use it for two LFSRs.

B.2.1 Specification

Let $k, n, z \in \mathbb{N}$. Let $\mathsf{P} \in \text{perm}(n)$ be an n-bit permutation, and let $\varphi_1, \ldots, \varphi_z : \{0,1\}^n \to \{0,1\}^n$ be z LFSRs. Let $\mathcal{T} \subseteq \mathbb{N}^z$ be a finite tweak space. Define the function $\mathsf{mask} : \{0,1\}^k \times \mathcal{T} \to \{0,1\}^n$ as follows:

$$\mathsf{mask}_K^{a_1,\ldots,a_z} = \mathsf{mask}(K, a_1, \ldots, a_z) = \varphi_z^{a_z} \circ \cdots \circ \varphi_1^{a_1} \circ \mathsf{P}(K \| 0^{n-k}). \tag{11}$$

We define the tweakable block cipher $\mathsf{SiM} : \{0,1\}^k \times \mathcal{T} \times \{0,1\}^n \to \{0,1\}^n$ as

$$\mathsf{SiM}(K, (a_1, \ldots, a_z), M) = \mathsf{P}(M \oplus \mathsf{mask}_K^{a_1,\ldots,a_z}) \oplus \mathsf{mask}_K^{a_1,\ldots,a_z}. \tag{12}$$

B.2.2 Security of SiM

We need a restriction on the tweak space $\mathcal{T}$ in order for SiM to be a secure tweakable block cipher. As Granger et al. [53], we say that $\mathcal{T}$ is $2^{-\alpha}$-proper with respect to $(\varphi_1, \ldots, \varphi_z)$ if the function $L \mapsto \varphi_z^{a_z} \circ \cdots \circ \varphi_1^{a_1}(L)$ is $2^{-\alpha}$-uniform and $2^{-\alpha}$-XOR-uniform.

Definition B.1. Let $n, z \in \mathbb{N}$. Let $\varphi_1, \ldots, \varphi_z : \{0,1\}^n \to \{0,1\}^n$ be z LFSRs. The tweak space $\mathcal{T}$ is called $2^{-\alpha}$*-proper* with respect to $(\varphi_1, \ldots, \varphi_z)$ if the following two properties hold:

1. For any $Y \in \{0,1\}^n$ and $(a_1, \ldots, a_z) \in \mathcal{T} \cup \{(0, \ldots, 0)\}$,

$$\mathbf{Pr}\left(L \xleftarrow{\$} \{0,1\}^n \ : \ \varphi_z^{a_z} \circ \cdots \circ \varphi_1^{a_1}(L) = Y\right) \leq 2^{-\alpha}\,;$$

2. For any $Y \in \{0,1\}^n$ and distinct $(a_1, \ldots, a_z), (a'_1, \ldots, a'_z) \in \mathcal{T} \cup \{(0, \ldots, 0)\}$,

$$\mathbf{Pr}\left(L \xleftarrow{\$} \{0,1\}^n \ : \ \varphi_z^{a_z} \circ \cdots \circ \varphi_1^{a_1}(L) \oplus \varphi_z^{a'_z} \circ \cdots \circ \varphi_1^{a'_1}(L) = Y\right) \leq 2^{-\alpha}\,.$$

In Section C, we will prove Theorem B.2, which says that if the tweak space is $2^{-\alpha}$-proper for sufficiently small $2^{-\alpha}$ (note that $2^{-\alpha}$ cannot be smaller than 2^{-n}), then SiM is a secure tweakable block cipher. When restricted to the single-user setting (i.e., $\mu = 1$), the proof is a direct simplification of Granger et al.'s analysis of MEM [53], due to the changes described in the introductory text of Section B.2. These simplifications allow us to derive a slightly improved bound on the advantage, noting for comparison that Granger et al. [53] proved security up to $(4.5q^2 + 3qp)/2^\alpha + p/2^k$, as opposed to our bound of $(q^2 + 2qp)/2^\alpha + (2q + p)/2^n + p/2^k$. On the other hand, our analysis is more general than that of Granger et al. in the fact that we consider the multi-user setting (i.e., $\mu \geq 1$).

Theorem B.2. *Let $k, n, z, \mu \in \mathbb{N}$. Let $\varphi_1, \ldots, \varphi_z : \{0,1\}^n \to \{0,1\}^n$ be z LFSRs, and let $\mathcal{T}$ be a $2^{-\alpha}$-proper tweak space with respect to $(\varphi_1, \ldots, \varphi_z)$. Consider* SiM *of (12) based on random permutation* $\mathsf{P} \xleftarrow{\$} \mathrm{perm}(n)$*. For any multi-user adversary $\mathcal{A}$ making at most $q \leq 2^{n-1}$ construction queries (in total to all μ construction oracles) and p primitive queries,*

$$\mathbf{Adv}_{\mathsf{SiM}}^{\mu\text{-tprp}}(\mathcal{A}) \leq \frac{q^2 + 2qp + (\mu-1)q}{2^\alpha} + \frac{\mu \cdot (2q + p + \frac{\mu-1}{2})}{2^n} + \frac{\mu \cdot (p + \frac{\mu-1}{2})}{2^k}\,.$$

The proof is given in Section C.

B.3 Security of Elephant

We will prove security of Elephant of Section 2 for any $2^{-\alpha}$-proper tweak space. The specific choice of tweak space for the three instances of Elephant will be discussed in Section B.4.

Theorem B.3. *Let $k, m, n, t, \mu \in \mathbb{N}$ with $k, m, t \leq n$. Let $\varphi_1, \varphi_2 : \{0,1\}^n \to \{0,1\}^n$ be LFSRs, and let $\mathcal{T}$ be a $2^{-\alpha}$-proper tweak space with respect to (φ_1, φ_2). Consider* Elephant $=$ (enc, dec) *of Section 2 based on random permutation* $\mathsf{P} \xleftarrow{\$} \mathrm{perm}(n)$*. For any multi-user adversary $\mathcal{A}$ making at most $q_e \leq 2^{n-1}$ construction encryption queries (in total to all μ construction oracles) for unique nonces whenever the same oracle is queried, q_d construction decryption queries (in total* *to all μ construction oracles), each query at most ℓ padded nonce and associated data and message blocks, and in total at most σ padded nonce and associated data and message blocks, and p primitive queries,*

$$\mathbf{Adv}^{\mu\text{-ae}}_{\mathsf{Elephant}}(\mathcal{A}) \leq \ell\binom{q_e}{2}/2^n + 3\binom{q_e+q_d}{2}/2^n + q_d/2^t + \mathbf{Adv}^{\mu\text{-tprp}}_{\mathsf{SiM}}(\mathcal{A}'),$$

for some multi-user adversary $\mathcal{A}'$ that makes 2σ construction queries and p primitive queries.

The proof is given in Section D. As a matter of fact, although the theorem restricts itself to nonce-respecting adversaries, this nonce-respecting behavior is *only* used for the portion of the proof related to confidentiality. The $\mathsf{Elephant}$ mode thus even achieves *authenticity* under nonce-reuse, up to above bound minus $\ell\binom{q_e}{2}/2^n$.

B.4 Implication for Dumbo, Jumbo, and Delirium

B.4.1 Dumbo: 160-Bit Elephant

We will prove that the 160-bit LFSR defined by (3) has maximal length, and that the tweak space used in $\mathsf{Elephant}$ with this LFSR is 2^{-n}-proper with respect to (φ_1, φ_2).

Proposition B.4. *Let $n = 160$. Let $\varphi_1 : \{0,1\}^{160} \mapsto \{0,1\}^{160}$ be the LSFR given in (3), and $\varphi_2 = \varphi_1 \oplus \mathsf{id}$. The tweak space $\mathcal{T} = \mathcal{T}_1 \times \mathcal{T}_2 = \{0, 1, \ldots, 2^{154}\} \times \{0, 1, 2\}$ is 2^{-n}-proper with respect to (φ_1, φ_2).*

Proof. The proof is almost identical to [53, Lemma 4], with the main difference that a different discrete logarithm must be computed. Let V be the 160×160 matrix over $\mathbb{F}_2$ that represents φ_1 of (3). As shown in [53, Lemma 3], $\varphi_1^i(L) = V^i \cdot L$ is 2^{-n}-proper for $i \in \{0, \ldots 2^n - 2\}$ if the minimal polynomial of V is primitive and of degree n. A quick computation using Sage [105] shows that this polynomial

$$p(x) = x^{160} + x^{136} + x^{83} + x^{53} + 1$$

is irreducible and primitive.

Next, let $\ell = \log_x(x+1)$ in the field $\mathbb{F}_2[x]/p(x)$. We have to show that $\varphi_2^b \circ \varphi_1^a(L) = (V+I)^b \cdot V^a \cdot L = V^{\ell \cdot b} \cdot V^a \cdot L$ is unique for any distinct set of tweaks. A simple Sage computation gives the following values for ℓ and 2ℓ:

$$\begin{aligned} \ell &= 742800116542094474882643562714650758474536684889 \approx 2^{159.02}, \\ 2\ell &= 24098595753286031561602292713018497293140826803 \approx 2^{154.08}. \end{aligned}$$

If we consider that $b \in \{0, 1, 2\}$ divides the tweak space into three sets, the smallest difference is between the set with $b = 0$ and the set corresponding to $b = 2$, which is bigger than 2^{154}. Hence, by ensuring that $0 \leq a \leq 2^{154}$, we have that for any two distinct $(a,b),(a',b') \in \{0,1,\ldots,2^{154}\} \times \{0,1,2\}$, $\varphi_2^b \circ \varphi_1^a \neq \varphi_2^{b'} \circ \varphi_1^{a'}$.

Finally, using both of the above observations, one can easily observe that $\mathcal{T}$ is 2^{-n}-proper in light of Definition B.1. □

We directly obtain that Dumbo is secure in the random permutation model.

Corollary B.5. *Let* $(k,m,n,t) = (128,96,160,64)$. *Let* $\mu \geq 1$. *Let* $\mathcal{T} = \{0,1,\ldots,2^{154}\} \times \{0,1,2\}$. *Consider* Dumbo*:* Elephant $= (\mathsf{enc},\mathsf{dec})$ *of Section 2 based on the permutation* Spongent-$\pi[160]$*, modeled as a random* 160*-bit permutation, and on* $\varphi_1 : \{0,1\}^{160} \to \{0,1\}^{160}$ *of (3). For any multi-user adversary* $\mathcal{A}$ *making at most* q_e *construction encryption queries (in total to all* μ *construction oracles) for unique nonces whenever the same oracle is queried,* q_d *construction decryption queries (in total to all* μ *construction oracles), each query at most* ℓ *padded nonce and associated data and message blocks, and in total at most* $\sigma \leq 2^{158}$ *padded nonce and associated data and message blocks, and* p *primitive queries,*

$$\mathbf{Adv}^{\mu\text{-ae}}_{\mathsf{Dumbo}}(\mathcal{A}) \leq \ell\binom{q_e}{2}/2^{160} + 3\binom{q_e+q_d}{2}/2^{160} + q_d/2^{64} + \frac{4\sigma^2 + 4\sigma p + (\mu-1)2\sigma + \mu\cdot(4\sigma + p + \frac{\mu-1}{2})}{2^{160}} + \frac{\mu\cdot(p+\frac{\mu-1}{2})}{2^{128}}\,.$$

Recall that NIST's call for lightweight authenticated encryption schemes [85] requested security up to an online complexity of around 2^{50} bytes. If we focus on the single-user case $\mu = 1$, by limiting the total online complexity σ to $2^{50}/(n/8)$ blocks, the bound of Corollary B.5 is at most 1 for $p \leq 2^{112}$.

B.4.2 Jumbo: 176-Bit Elephant

We will prove that the 176-bit LFSR defined by (4) has maximal length, and that the tweak space used in Elephant with this LFSR is 2^{-n}-proper with respect to (φ_1,φ_2).

Proposition B.6. *Let* $n = 176$. *Let* $\varphi_1 : \{0,1\}^{176} \mapsto \{0,1\}^{176}$ *be the LSFR given in (4), and* $\varphi_2 = \varphi_1 \oplus \mathsf{id}$. *The tweak space* $\mathcal{T} = \mathcal{T}_1 \times \mathcal{T}_2 = \{0,1,\ldots,2^{173}\} \times \{0,1,2\}$ *is* 2^{-n}*-proper with respect to* (φ_1,φ_2).

Proof. The proof is identical to that of Proposition B.4, with the difference that for the 176×176 matrix V that represents φ_1 of (4), the corresponding polynomial

$$p(x) = x^{176} + x^{154} + x^{135} + x^{19} + 1$$

is irreducible and primitive. The discrete logarithm $\ell = \log_x(x+1)$ in the field $\mathbb{F}_2[x]/p(x)$ and its related 2ℓ are computed as

$$\ell = 18881376151403786777481463432029450294100461562220699 \approx 2^{173.66}\,,$$
$$2\ell = 37762752302807573554962926864058900588200923124441398 \approx 2^{174.66}\,.$$

Again, dividing the tweak space into 3 sets according to the value $b \in \{0, 1, 2\}$, the smallest difference is between set $b = 0$ and set $b = 1$, or $b = 1$ and $b = 2$, which is bigger than 2^{173}. Hence, by ensuring that $0 \leq a \leq 2^{173}$, we have that for any two distinct $(a, b), (a', b') \in \{0, 1, \ldots, 2^{173}\} \times \{0, 1, 2\}$, $\varphi_2^b \circ \varphi_1^a \neq \varphi_2^{b'} \circ \varphi_1^{a'}$. $\square$

We directly obtain that Jumbo is secure in the random permutation model.

Corollary B.7. *Let* $(k, m, n, t) = (128, 96, 176, 64)$. *Let* $\mu \geq 1$. *Let* $\mathcal{T} = \{0, 1, \ldots, 2^{173}\} \times \{0, 1, 2\}$. *Consider* Jumbo*:* Elephant = (enc, dec) *of Section 2 based on the permutation* Spongent-$\pi[176]$*, modeled as a random* 176*-bit permutation, and on* $\varphi_1 : \{0,1\}^{176} \to \{0,1\}^{176}$ *of (4). For any multi-user adversary* $\mathcal{A}$ *making at most* q_e *construction encryption queries (in total to all* μ *construction oracles) for unique nonces whenever the same oracle is queried,* q_d *construction decryption queries (in total to all* μ *construction oracles), each query at most* ℓ *padded nonce and associated data and message blocks, and in total at most* $\sigma \leq 2^{174}$ *padded nonce and associated data and message blocks, and* p *primitive queries,*

$$\begin{aligned}\mathbf{Adv}^{\mu\text{-ae}}_{\mathsf{Jumbo}}(\mathcal{A}) \leq \ell\binom{q_e}{2}/2^{176} + 3\binom{q_e + q_d}{2}/2^{176} + q_d/2^{64} \\ + \frac{4\sigma^2 + 4\sigma p + (\mu - 1)2\sigma + \mu \cdot (4\sigma + p + \frac{\mu-1}{2})}{2^{176}} + \frac{\mu \cdot (p + \frac{\mu-1}{2})}{2^{128}}.\end{aligned}$$

As before, focusing on the single-user case $\mu = 1$, and limiting the total online complexity σ to $2^{50}/(n/8)$ blocks, the bound of Corollary B.7 is at most 1 for $p \leq 2^{127}$.

B.4.3 Delirium: 200-Bit Elephant

We will prove that the 200-bit LFSR defined by (5) has maximal length, and that the tweak space used in Elephant with this LFSR is 2^{-n}-proper with respect to (φ_1, φ_2).

Proposition B.8. *Let* $n = 200$. *Let* $\varphi_1 : \{0,1\}^{200} \mapsto \{0,1\}^{200}$ *be the LSFR given in (5), and* $\varphi_2 = \varphi_1 \oplus \mathsf{id}$. *The tweak space* $\mathcal{T} = \mathcal{T}_1 \times \mathcal{T}_2 = \{0, 1, \ldots, 2^{197}\} \times \{0, 1, 2\}$ *is* 2^{-n}*-proper with respect to* (φ_1, φ_2).

Proof. The proof is identical to that of Proposition B.4, with the difference that for the 200×200 matrix V that represents φ_1 of (5), the corresponding polynomial

$$\begin{aligned}p(x) = x^{200} + x^{93} + x^{91} + x^{82} + x^{78} + x^{71} + x^{69} + x^{67} + x^{65} \\ + x^{60} + x^{52} + x^{49} + x^{47} + x^{41} + x^{39} + x^{38} + x^{34} + x^{30} + x^{27} \\ + x^{26} + x^{25} + x^{23} + x^{21} + x^{19} + x^{17} + x^{16} + x^{15} + x^{13} + 1\end{aligned}$$

is irreducible and primitive. The discrete log $\ell = \log_x(x+1)$ in the field $\mathbb{F}_2[x]/p(x)$ and its related 2ℓ are computed as

$$\begin{aligned}\ell = {} & 692180606625676931900534627786122994390018641930530681719698 \\ & \approx 2^{198.78}\,, \\ 2\ell = {} & 1384361213251353863801069255572245988780037283861061363439396 \\ & \approx 2^{199.78}\,.\end{aligned}$$

Again, dividing the tweak space into 3 sets according to the value $b \in \{0,1,2\}$, the smallest difference is between set $b=2$ and set $b=0$, which is bigger than 2^{197}. Hence, by ensuring that $0 \le a \le 2^{197}$, we have that for any two distinct $(a,b),(a',b') \in \{0,1,\ldots,2^{197}\} \times \{0,1,2\}$, $\varphi_2^b \circ \varphi_1^a \neq \varphi_2^{b'} \circ \varphi_1^{a'}$. □

We directly obtain that Delirium is secure in the random permutation model.

Corollary B.9. *Let* $(k,m,n,t) = (128,96,200,128)$. *Let* $\mu \ge 1$. *Let* $\mathcal{T} = \{0,1,\ldots,2^{197}\} \times \{0,1,2\}$. *Consider* Delirium*:* Elephant $=$ (enc, dec) *of Section 2 based on the permutation* Keccak-$f[200]$*, modeled as a random* 200*-bit permutation, and on* $\varphi_1 : \{0,1\}^{200} \to \{0,1\}^{200}$ *of (5). For any multi-user adversary* $\mathcal{A}$ *making at most* q_e *construction encryption queries (in total to all* μ *construction oracles) for unique nonces whenever the same oracle is queried,* q_d *construction decryption queries (in total to all* μ *construction oracles), each query at most* ℓ *padded nonce and associated data and message blocks, and in total at most* $\sigma \le 2^{198}$ *padded nonce and associated data and message blocks, and* p *primitive queries,*

$$\begin{aligned}\mathbf{Adv}^{\mu\text{-ae}}_{\mathsf{Delirium}}(\mathcal{A}) \le {} & \ell\binom{q_e}{2}/2^{200} + 3\binom{q_e+q_d}{2}/2^{200} + q_d/2^{128} \\ & + \frac{4\sigma^2 + 4\sigma p + (\mu-1)2\sigma + \mu\cdot(4\sigma + p + \frac{\mu-1}{2})}{2^{200}} + \frac{\mu\cdot(p+\frac{\mu-1}{2})}{2^{128}}\,.\end{aligned}$$

As before, focusing on the single-user case $\mu = 1$, and limiting the total online complexity σ to $2^{74}/(n/8)$ blocks, the bound of Corollary B.9 is at most 1 for $p \le 2^{127}$.

C Proof of Theorem B.2 (on SiM)

The proof closely follows Granger et al. [53] and is performed using the H-coefficient technique [32,89]. The main difference is in the fact that we consider multi-user security, where the adversary can query $\mu \ge 1$ construction oracles.

Let $K_1,\ldots,K_\mu \xleftarrow{\$} \{0,1\}^k$, $\mathsf{P} \xleftarrow{\$} \mathrm{perm}(n)$, and $\widetilde{\pi}_1,\ldots,\widetilde{\pi}_\mu \xleftarrow{\$} \mathrm{perm}(\mathcal{T},n)$, where $\mathcal{T}$ is $2^{-\alpha}$-proper with respect to LFSRs $(\varphi_1,\ldots,\varphi_z)$. Consider a computationally unbounded adversary $\mathcal{A}$ that tries to distinguish $\mathcal{O} := ((\widetilde{\mathsf{E}}^{\mathsf{P}}_{K_j})_{j=1}^{\mu}, \mathsf{P}^{\pm})$ from $\mathcal{P} := ((\widetilde{\pi}_j)_{j=1}^{\mu}, \mathsf{P}^{\pm})$. Without loss of generality, we can consider it to be deterministic: for any probabilistic adversary there exists a deterministic one that has at least the same success probability. The interaction of $\mathcal{A}$ with its oracle ($\mathcal{O}$ or $\mathcal{P}$) is gathered in a view ν. Denote by $D_{\mathcal{O}}$ (resp., $D_{\mathcal{P}}$) the probability distribution of views in interaction with $\mathcal{O}$ (resp., $\mathcal{P}$). Denote by $\mathcal{V}$ the set of "attainable views", i.e., views ν such that $\mathbf{Pr}\left(D_{\mathcal{P}} = \nu\right) > 0$.

Lemma C.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 $\varepsilon \in [0,1]$ be such that $\frac{\mathbf{Pr}(D_{\mathcal{O}}=\nu)}{\mathbf{Pr}(D_{\mathcal{P}}=\nu)} \geq 1 - \varepsilon$ for all $\nu \in \mathcal{V}_{\mathrm{good}}$. Then,*

$$\Delta_{\mathcal{A}}\left(\mathcal{O}\ ;\ \mathcal{P}\right) \leq \varepsilon + \mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right) . \tag{13}$$

For view $\nu = \{(x_1, y_1), \ldots, (x_q, y_q)\}$ consisting of q input/output tuples, we denote by $\mathcal{O} \vdash \nu$ the event that oracle $\mathcal{O}$ satisfies that $\mathcal{O}(x_i) = y_i$ for all $i = \{1, \ldots, q\}$.

The remainder of the proof is structured as follows. We specify the views of an adversary in Section C.1 and define the bad views in Section C.2. The probability of bad views is analyzed in Section C.3 and the probability ratio for good views is considered in Section C.4. Section C.5 concludes the proof.

C.1 Views

The adversary can make q construction queries to $(\widetilde{\mathsf{E}}^{\mathsf{P}}_{K_j})_{j=1}^{\mu}$ or $(\widetilde{\pi})_{j=1}^{\mu}$, all *in forward direction only*. Each such query is made for user index $j_i \in \{1, \ldots, \mu\}$, some tweak $\bar{a}_i = (a_1, \ldots, a_z)_i$, and message input M_i, and results in an output C_i. The q queries are summarized in a view

$$\nu_c = \{(j_1, \bar{a}_1, M_1, C_1), \ldots, (j_q, \bar{a}_q, M_q, C_q)\} .$$

The adversary can make p primitive queries to $\mathsf{P}^{\pm}$, and these are likewise summarized in a view

$$\nu_p = \{(X_1, Y_1), \ldots, (X_p, Y_p)\} .$$

After the conversation of $\mathcal{A}$ with its oracle, but before it makes its final decision, we reveal the key material used in the interaction. This can be done without loss of generality; it only improves the adversarial success probability. The first values that are revealed are values $K_1, \ldots, K_\mu$. In the real world, these are the keys $K_1, \ldots, K_\mu \xleftarrow{\$} \{0,1\}^k$ that are actually used by the construction oracle; in the ideal world, these are dummy keys $K_1, \ldots, K_\mu \xleftarrow{\$} \{0,1\}^k$. The second values that are revealed are values $L_1, \ldots, L_\mu \in \{0,1\}^n$. In the real world, these are the values $L_j = \mathsf{P}(K_j \| 0^{n-k})$ for $j = 1, \ldots, \mu$; in the ideal world, these are dummy keys $L_1, \ldots, L_\mu \xleftarrow{\$} \{0,1\}^n$.[3] The revealed data is summarized in a view

$$\nu_k = \{(K_1, L_1), \ldots, (K_\mu, L_\mu)\} .$$

[3]In the original analysis of MEM [53] (that was about single-user security only), the mask involves a computation $\mathsf{P}(K \| N)$ for nonce N. This not only complicates the values that have to be revealed; it also results in a larger view and hence a higher collision probability among tuples in the view.

The complete view is defined as $\nu = (\nu_c, \nu_p, \nu_k)$. We assume that the adversary never makes any duplicate query, hence ν_c and ν_p contain no duplicate elements.

C.2 Definition of Good and Bad Views

In the real world, all tuples in ν_p define exactly one input-output pair for P. Likewise, the tuples in ν_k are input-output pairs for P. Using these tuples, one can observe that any tuple $(j_i, \bar{a}_i, M_i, C_i) \in \nu_c$ also defines an input-output pair for P, namely

$$\left(M_i \oplus \bar{\varphi}^{\bar{a}_i}(L_{j_i}), C_i \oplus \bar{\varphi}^{\bar{a}_i}(L_{j_i})\right) ,$$

see (12), where we define $\bar{\varphi}^{\bar{a}_i} := \varphi_z^{a_{zi}} \circ \cdots \circ \varphi_1^{a_{1i}}$ for brevity. If among all these $q+p+\mu$ input-output pairs defined by ν, there are two that have colliding input or output values, we consider ν to be a bad view. Formally, ν is called "bad" if one of the following conditions is satisfied, where we recall that the user index j in a tuple in ν_c determines which key tuple from ν_k has to be used:

$$\begin{aligned}
\mathrm{bad}_{c,c}:\ & \text{for some distinct } (j, \bar{a}, M, C), (j', \bar{a}', M', C') \in \nu_c\text{:} \\
& \bar{\varphi}^{\bar{a}}(L_j) \oplus \bar{\varphi}^{\bar{a}'}(L_{j'}) \in \{M \oplus M', C \oplus C'\} , \\
\mathrm{bad}_{c,p}:\ & \text{for some } (j, \bar{a}, M, C) \in \nu_c \text{ and } (X, Y) \in \nu_p\text{:} \\
& \bar{\varphi}^{\bar{a}}(L_j) \in \{M \oplus X, C \oplus Y\} , \\
\mathrm{bad}_{c,k}:\ & \text{for some } (j, \bar{a}, M, C) \in \nu_c \text{ and } (K, L) \in \nu_k\text{:} \\
& \bar{\varphi}^{\bar{a}}(L_j) \in \{M \oplus K \| 0^{n-k}, C \oplus L\} , \\
\mathrm{bad}_{p,k}:\ & \text{for some } (X, Y) \in \nu_p \text{ and } (K, L) \in \nu_k\text{:} \\
& X = K \| 0^{n-k} \text{ or } Y = L , \\
\mathrm{bad}_{k,k}:\ & \text{for some distinct } (K, L), (K', L') \in \nu_k\text{:} \\
& K = K' \text{ or } L = L' .
\end{aligned}$$

We write $\mathrm{bad} = \mathrm{bad}_{c,c} \vee \mathrm{bad}_{c,p} \vee \mathrm{bad}_{c,k} \vee \mathrm{bad}_{p,k} \vee \mathrm{bad}_{k,k}$.

C.3 Probability of Bad View in Ideal World

Our goal is to bound $\mathbf{Pr}\left(D_{\mathcal{P}} \in \mathcal{V}_{\mathrm{bad}}\right)$, the probability of a bad view in the ideal world $\mathcal{P} = ((\widetilde{\pi}_j)_{j=1}^{\mu}, \mathsf{P}^{\pm})$. For brevity, denote by $D_{\mathcal{P}} \propto \mathrm{bad}$ the event that $D_{\mathcal{P}}$ satisfies bad. By the union bound,

$$\begin{aligned}
\mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}\right) &= \mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{c,c} \vee \mathrm{bad}_{c,p} \vee \mathrm{bad}_{c,k} \vee \mathrm{bad}_{p,k} \vee \mathrm{bad}_{k,k}\right) \\
&\leq \mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{c,c}\right) + \mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{c,p}\right) + \mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{c,k}\right) \\
&\quad + \mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{p,k}\right) + \mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{k,k}\right) . \qquad (14)
\end{aligned}$$

We will analyze the five probabilities separately, thereby noticing that (i) $K_1, \ldots, K_\mu \xleftarrow{\$} \{0,1\}^k$ and $L_1, \ldots, L_\mu \xleftarrow{\$} \{0,1\}^n$ are random variables in the ideal world, and (ii) as the adversary only makes forward construction queries, each tuple $(j, \bar{a}, M, C) \in \nu_c$ satisfies that C is randomly drawn from a set of size at least $2^n - q$.

Event $\mathrm{bad}_{c,c}$**.** For $\mathrm{bad}_{c,c}$, let $(j, \bar{a}, M, C), (j', \bar{a}', M', C') \in \nu_c$ be any two distinct tuples. If $j = j'$ and $\bar{a} = \bar{a}'$, then necessarily $M \neq M'$ and $C \neq C'$, and $\mathrm{bad}_{c,c}$ holds with probability 0. Otherwise, if $j = j'$ but $\bar{a} \neq \bar{a}'$, we can deduce from $2^{-\alpha}$-properness of $\mathcal{T}$, namely property 2 of Definition B.1, that event $\mathrm{bad}_{c,c}$ holds with probability at most $2/2^\alpha$. Finally, if $j \neq j'$, the subkeys $L_j, L_{j'}$ are independent and we can likewise deduce from $2^{-\alpha}$-properness of $\mathcal{T}$, namely property 1 of Definition B.1, that event $\mathrm{bad}_{c,c}$ holds with probability at most $2/2^\alpha$. Thus, summing over all $\binom{q}{2}$ possible choices of queries,

$$\mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{c,c}\right) \leq \frac{q(q-1)}{2^\alpha}\,.$$

Event $\mathrm{bad}_{c,p}$**.** For $\mathrm{bad}_{c,p}$, let $(j, \bar{a}, M, C) \in \nu_c$ and $(X, Y) \in \nu_p$ be any two tuples. We can deduce from $2^{-\alpha}$-properness of $\mathcal{T}$, namely property 1 of Definition B.1, that event $\mathrm{bad}_{c,p}$ holds with probability at most $2/2^\alpha$. Thus, summing over all qp possible choices of queries,

$$\mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{c,p}\right) \leq \frac{2qp}{2^\alpha}\,.$$

Event $\mathrm{bad}_{c,k}$**.** For $\mathrm{bad}_{c,k}$, let $(j, \bar{a}, M, C) \in \nu_c$ and $(K, L) \in \nu_k$ be any two tuples. We consider the two equations of $\mathrm{bad}_{c,k}$ separately. For the first equation,

$$\bar{\varphi}^{\bar{a}}(L_j) = M \oplus K \| 0^{n-k}\,,$$

we will use that $L_j \xleftarrow{\$} \{0,1\}^n$ is a randomly generated value independent of K. We can deduce from $2^{-\alpha}$-properness of $\mathcal{T}$, namely property 1 of Definition B.1, that this equation holds with probability at most $1/2^\alpha$.

For the second equation,

$$\bar{\varphi}^{\bar{a}}(L_j) = C \oplus L\,,$$

it might be that $L = L_j$, and we cannot rely on Definition B.1. Instead, we will use that all construction queries are made in forward direction, and that C is randomly drawn from a set of size at least $2^n - q$ elements. Above equation thus holds with probability at most $1/(2^n - q)$.

Thus, summing over all μq possible choices of queries,

$$\mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{c,k}\right) \leq \frac{\mu q}{2^\alpha} + \frac{\mu q}{2^n - q}\,.$$

Event $\mathrm{bad}_{p,k}$. For $\mathrm{bad}_{p,k}$, let $(X, Y) \in \nu_p$ and $(K, L) \in \nu_k$ be any two tuples. As $K \xleftarrow{\$} \{0,1\}^k$ and $L \xleftarrow{\$} \{0,1\}^n$, the tuples set $\mathrm{bad}_{p,k}$ with probability at most $1/2^k + 1/2^n$. Thus, summing over all μp possible choices of queries,

$$\mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{p,k}\right) \leq \frac{\mu p}{2^k} + \frac{\mu p}{2^n}\,.$$

Event $\mathrm{bad}_{k,k}$. For $\mathrm{bad}_{k,k}$, let $(K, L), (K', L') \in \nu_k$ be any two distinct tuples. As $K, K' \xleftarrow{\$} \{0,1\}^k$ and $L, L' \xleftarrow{\$} \{0,1\}^n$, the tuples set $\mathrm{bad}_{k,k}$ with probability at most $1/2^k + 1/2^n$. Thus, summing over all $\binom{\mu}{2}$ possible choices of queries,

$$\mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}_{k,k}\right) \leq \frac{\mu(\mu-1)}{2^{k+1}} + \frac{\mu(\mu-1)}{2^{n+1}}\,.$$

Conclusion. Concluding, we obtain for (14):

$$\mathbf{Pr}\left(D_{\mathcal{P}} \propto \mathrm{bad}\right) \leq \frac{q^2 + 2qp + (\mu-1)q}{2^\alpha} + \frac{\mu \cdot (2q + p + \frac{\mu-1}{2})}{2^n} + \frac{\mu \cdot (p + \frac{\mu-1}{2})}{2^k}\,. \tag{15}$$

using that $2^n - q \geq 2^{n-1}$.

C.4 Probability Ratio for Good Views

Consider any good view $\nu \in \mathcal{V}_{\mathrm{good}}$. We will prove the inequality $\mathbf{Pr}\left(D_{\mathcal{O}} = \nu\right) \geq \mathbf{Pr}\left(D_{\mathcal{P}} = \nu\right)$. The proof is a direct generalization of that of Granger et al. [53], noting that in our case, we consider multi-user security. The proof is included for completeness.

Real World. In the real world $\mathcal{O} = ((\widetilde{\mathsf{E}}^{\mathsf{P}}_{K_j})_{j=1}^{\mu}, \mathsf{P}^{\pm})$, goodness of the view means that $\nu = (\nu_c, \nu_p, \nu_k)$ defines exactly $q + p + \mu$ input-output pairs for P, and no two of them collide on the input or output, and ν_k consists of random values $K_1, \ldots, K_\mu \xleftarrow{\$} \{0,1\}^k$. Therefore, we obtain:

$$\begin{aligned}
\mathbf{Pr}\left(D_{\mathcal{O}} = \nu\right) &= \mathbf{Pr}\left(K'_1, \ldots, K'_\mu \xleftarrow{\$} \{0,1\}^k \;:\; K'_1 = K_1 \wedge \cdots \wedge K'_\mu = K_\mu\right) \cdot \\
&\quad\ \mathbf{Pr}\left(\mathsf{P} \xleftarrow{\$} \mathrm{perm}(n) \;:\; (\widetilde{\mathsf{E}}^{\mathsf{P}}_{K_j})_{j=1}^{\mu} \vdash \nu_c \wedge \mathsf{P} \vdash \nu_p \wedge \mathsf{P} \vdash \nu_k\right) \\
&= \frac{1}{2^{k\mu}} \cdot \frac{(2^n - (q + p + \mu))!}{2^n!}\,.
\end{aligned} \tag{16}$$

Ideal World. In the ideal world $\mathcal{P} = ((\widetilde{\pi}_j)_{j=1}^{\mu}, \mathsf{P}^{\pm})$, the view $\nu = (\nu_c, \nu_p, \nu_k)$ consists of three lists of independent tuples: ν_c defines exactly q input-output pairs for $\widetilde{\pi}_j$, ν_p defines exactly p input-output pairs for P, and ν_k consists of μ random tuples $(K_1, L_1), \ldots, (K_\mu, L_\mu) \xleftarrow{\$} \{0,1\}^k \times \{0,1\}^n$. For counting, it is convenient to group the tuples in ν_c depending on the user index j and tweak value $\bar{a}$. For $J \in \{1, \dots, \mu\}$ and $T \in \mathcal{T}$, define

$$q_{J,T} = |\{(j, \bar{a}, M, C) \in \nu_c \mid j = J \wedge \bar{a} = T\}| \ ,$$

where $\sum_{(J,T)\in\{1,\dots,\mu\}\times\mathcal{T}} q_{J,T} = q$. We obtain:

$$\begin{aligned}
\mathbf{Pr}\left(D_{\mathcal{P}} = \nu\right) = \mathbf{Pr}&\left(K'_1, \dots, K'_\mu \xleftarrow{\$} \{0,1\}^k \ : \ K'_1 = K_1 \wedge \dots \wedge K'_\mu = K_\mu\right) \cdot \\
&\mathbf{Pr}\left(L'_1, \dots, L'_\mu \xleftarrow{\$} \{0,1\}^n \ : \ L'_1 = L_1 \wedge \dots \wedge L'_\mu = L_\mu\right) \cdot \\
&\mathbf{Pr}\left(\widetilde{\pi}_1, \dots, \widetilde{\pi}_\mu \xleftarrow{\$} \mathrm{perm}(\mathcal{T}, n) \ : \ (\widetilde{\pi}_j)_{j=1}^{\mu} \vdash \nu_c\right) \cdot \\
&\mathbf{Pr}\left(\mathsf{P} \xleftarrow{\$} \mathrm{perm}(n) \ : \ \mathsf{P} \vdash \nu_p\right) \\
&= \frac{1}{2^{(k+n)\mu}} \cdot \prod_{\substack{J\in\{1,\dots,\mu\}\\ T\in\mathcal{T}}} \frac{(2^n - q_{J,T})!}{2^n!} \cdot \frac{(2^n - p)!}{2^n!} \\
&= \frac{1}{2^{k\mu}} \cdot \left(\frac{(2^n - 1)!}{2^n!}\right)^{\mu} \cdot \prod_{\substack{J\in\{1,\dots,\mu\}\\ T\in\mathcal{T}}} \frac{(2^n - q_{J,T})!}{2^n!} \cdot \frac{(2^n - p)!}{2^n!} \\
&\leq \frac{1}{2^{k\mu}} \cdot \frac{(2^n - (q + p + \mu))!}{2^n!} \ ,
\end{aligned} \tag{17}$$

using that for any $\sigma + \tau \leq 2^n$ we have $\frac{(2^n-\sigma)!}{2^n!} \cdot \frac{(2^n-\tau)!}{2^n!} \leq \frac{(2^n-(\sigma+\tau))!}{2^n!}$.

Conclusion. Combining (16) and (17), we obtain that for any good view $\nu \in \mathcal{V}_{\mathrm{good}}$:

$$\frac{\mathbf{Pr}\left(D_{\mathcal{O}} = \nu\right)}{\mathbf{Pr}\left(D_{\mathcal{P}} = \nu\right)} \geq 1 \, . \tag{18}$$

C.5 Conclusion

By the H-coefficient technique (Lemma C.1), we directly obtain from (15) and (18):

$$\mathbf{Adv}_{\widetilde{\mathsf{E}}}^{\mu\text{-tprp}}(\mathcal{A}) \leq 0 + \frac{q^2 + 2qp + (\mu - 1)q}{2^\alpha} + \frac{\mu \cdot (2q + p + \frac{\mu-1}{2})}{2^n} + \frac{\mu \cdot (p + \frac{\mu-1}{2})}{2^k} \, .$$

D Proof of Theorem B.3 (on Elephant)

Let $K_1, \dots, K_\mu \xleftarrow{\$} \{0,1\}^k$, $\mathsf{P} \xleftarrow{\$} \mathrm{perm}(n)$, and $\mathsf{rand}_1, \dots, \mathsf{rand}_\mu$ be functions that for each input $(N, A, M) \in \{0,1\}^m \times \{0,1\}^* \times \{0,1\}^*$ return a random string of size $|M| + t$ bits. Consider a deterministic computationally unbounded multi-user adversary $\mathcal{A}$ that tries to distinguish $\mathcal{O} := ((\mathsf{enc}^{\mathsf{P}}_{K_j}, \mathsf{dec}^{\mathsf{P}}_{K_j})_{j=1}^{\mu}, \mathsf{P}^{\pm})$ from

$\mathcal{P} := ((\mathsf{rand}_j, \perp)_{j=1}^{\mu}, \mathsf{P}^{\pm})$:

$$\mathbf{Adv}^{\mu\text{-ae}}_{\mathsf{Elephant}}(\mathcal{A}) = \Delta_{\mathcal{A}}\left(\left(\mathsf{enc}^{\mathsf{P}}_{K_j}, \mathsf{dec}^{\mathsf{P}}_{K_j}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \ ; \ \left(\mathsf{rand}_j, \perp\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \right) . \tag{19}$$

First Step: Isolating SiM

As a first step, we will describe an alternative authenticated encryption scheme AE' based on tweakable permutations $\widetilde{\pi}_1, \ldots, \widetilde{\pi}_\mu \xleftarrow{\$} \mathrm{perm}(\mathcal{T}, n)$, where $\mathcal{T}$ is $2^{-\alpha}$-proper with respect to LFSRs (φ_1, φ_2). Its encryption function $\overline{\mathsf{enc}}$ and decryption function $\overline{\mathsf{dec}}$ are given in Algorithms 3 and 4, respectively, for any function $\widetilde{\pi} \in \{\widetilde{\pi}_1, \ldots, \widetilde{\pi}_\mu\}$. Unlike the original functions enc and dec of Algorithms 1 and 2, the functions $\overline{\mathsf{enc}}$ and $\overline{\mathsf{dec}}$ are not explicitly keyed by $K_1, \ldots, K_\mu$, but are instead implicitly keyed by the use of random secret tweakable permutations $\widetilde{\pi}_1, \ldots, \widetilde{\pi}_\mu$.

Algorithm 3 encryption $\overline{\mathsf{enc}}$

Input: (N, A, M)
Output: (C, T)

$M_1 \ldots M_{\ell_M} \xleftarrow{n} M$
for $i = 1, \ldots, \ell_M$ **do**
  $C_i \leftarrow M_i \oplus \widetilde{\pi}((i-1, 1), N \| 0^{n-m})$
$C \leftarrow \lfloor C_1 \ldots C_{\ell_M} \rfloor_{|M|}$
$A_1 \ldots A_{\ell_A} \xleftarrow{n} N \| A \| 1$
$C_1 \ldots C_{\ell_C} \xleftarrow{n} C \| 1$
$T \leftarrow A_1$
for $i = 2, \ldots, \ell_A$ **do**
  $T \leftarrow T \oplus \widetilde{\pi}((i-1, 0), A_i)$
for $i = 1, \ldots, \ell_C$ **do**
  $T \leftarrow T \oplus \widetilde{\pi}((i-1, 2), C_i)$
$T \leftarrow \widetilde{\pi}((0, 0), T)$
return $(C, \lfloor T \rfloor_t)$

Algorithm 4 decryption $\overline{\mathsf{dec}}$

Input: (N, A, C, T)
Output: M or $\perp$

$C_1 \ldots C_{\ell_M} \xleftarrow{n} C$
for $i = 1, \ldots, \ell_M$ **do**
  $M_i \leftarrow C_i \oplus \widetilde{\pi}((i-1, 1), N \| 0^{n-m})$
$M \leftarrow \lfloor M_1 \ldots M_{\ell_M} \rfloor_{|C|}$
$A_1 \ldots A_{\ell_A} \xleftarrow{n} N \| A \| 1$
$C_1 \ldots C_{\ell_C} \xleftarrow{n} C \| 1$
$\bar{T} \leftarrow A_1$
for $i = 2, \ldots, \ell_A$ **do**
  $\bar{T} \leftarrow \bar{T} \oplus \widetilde{\pi}((i-1, 0), A_i)$
for $i = 1, \ldots, \ell_C$ **do**
  $\bar{T} \leftarrow \bar{T} \oplus \widetilde{\pi}((i-1, 2), C_i)$
$\bar{T} \leftarrow \widetilde{\pi}((0, 0), \bar{T})$
return $\lfloor \bar{T} \rfloor_t = T \ ? \ M \ : \ \perp$

By a simple hybrid argument, we obtain for the distance of (19):

$$\begin{aligned}(19) \leq\ & \Delta_{\mathcal{A}}\left(\left(\mathsf{enc}^{\mathsf{P}}_{K_j}, \mathsf{dec}^{\mathsf{P}}_{K_j}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \ ; \ \left(\overline{\mathsf{enc}}^{\mathsf{SiM}^{\mathsf{P}}_{K_j}}, \overline{\mathsf{dec}}^{\mathsf{SiM}^{\mathsf{P}}_{K_j}}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \right) \\ &+ \Delta_{\mathcal{A}}\left(\left(\overline{\mathsf{enc}}^{\mathsf{SiM}^{\mathsf{P}}_{K_j}}, \overline{\mathsf{dec}}^{\mathsf{SiM}^{\mathsf{P}}_{K_j}}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \ ; \ \left(\overline{\mathsf{enc}}^{\widetilde{\pi}_j}, \overline{\mathsf{dec}}^{\widetilde{\pi}_j}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \right) \\ &+ \Delta_{\mathcal{A}}\left(\left(\overline{\mathsf{enc}}^{\widetilde{\pi}_j}, \overline{\mathsf{dec}}^{\widetilde{\pi}_j}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \ ; \ \left(\mathsf{rand}_j, \perp\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \right) . \end{aligned} \tag{20}$$

The first distance of (20) equals 0 by design of AE'. The second distance of (20) is at most $\Delta_{\mathcal{A}'}\left(\left(\mathsf{SiM}^{\mathsf{P}}_{K_j}\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \ ; \ \left(\widetilde{\pi}_j\right)_{j=1}^{\mu}, \mathsf{P}^{\pm} \right) = \mathbf{Adv}^{\mu\text{-tprp}}_{\mathsf{SiM}}(\mathcal{A}')$, where $\mathcal{A}'$ is a multi-user adversary that makes 2σ construction queries (in total to all μ construction oracles) and p primitive queries in order to simulate $\mathcal{A}$'s oracles.

For the third distance of (20), access to P does not help the adversary, and the oracle can be dropped. We obtain from (20):

$$(19) \leq \mathbf{Adv}_{\mathsf{SiM}}^{\mu\text{-tprp}}(\mathcal{A}') + \Delta_{\mathcal{A}}\left(\left(\overline{\mathsf{enc}}^{\widetilde{\pi}_j}, \overline{\mathsf{dec}}^{\widetilde{\pi}_j}\right)_{j=1}^{\mu} \; ; \; \left(\mathsf{rand}_j, \perp\right)_{j=1}^{\mu}\right) . \tag{21}$$

Due to the independence of the oracles for $j = 1, \ldots, \mu$, the remaining distance in (21) can be upper bounded by the sum of the distances for the j oracles, maximized over any choice of adversaries $\mathcal{A}_1, \ldots, \mathcal{A}_\mu$ whose accumulated query complexity is at most that of $\mathcal{A}$:

$$(19) \leq \mathbf{Adv}_{\mathsf{SiM}}^{\mu\text{-tprp}}(\mathcal{A}') + \max_{\mathcal{A}_1,\ldots,\mathcal{A}_\mu} \sum_{j=1}^{\mu} \Delta_{\mathcal{A}_j}\left(\overline{\mathsf{enc}}^{\widetilde{\pi}_j}, \overline{\mathsf{dec}}^{\widetilde{\pi}_j} \; ; \; \mathsf{rand}_j, \perp\right) , \tag{22}$$

where the query complexity of adversary $\mathcal{A}_j$ (for $j = 1, \ldots, \mu$) is parametrized by $q_{e,j}$, $q_{d,j}$, and σ_j, and where $\sum_{j=1}^{\mu} q_{e,j} = q_e$, $\sum_{j=1}^{\mu} q_{d,j} = q_d$, and $\sum_{j=1}^{\mu} \sigma_j = \sigma$. Here, we recall that complexity parameter ℓ denotes the maximum length of a single query, and it stays the same for each adversary.

The authenticated encryption scheme AE' can be seen as a generic encrypt-then-MAC construction, and more detailed the N2 construction of Namprempre et al. [84], where encryption is done in counter mode and message authentication using a variant of the protected counter sum [10,77] MAC function. We describe a dedicated security proof that is more compact and gives a better bound.

Second Step: Simplifying Authentication

The simplification of (22) allows us to focus on a single-user instance, and in this step, we drop the subscript j of the oracles for readability. Nevertheless, we keep the subscript j for $\mathcal{A}$ and for its complexities for clarity. In other words, this second step is about bounding

$$\Delta_{\mathcal{A}_j}\left(\overline{\mathsf{enc}}^{\widetilde{\pi}}, \overline{\mathsf{dec}}^{\widetilde{\pi}} \; ; \; \mathsf{rand}, \perp\right) \tag{23}$$

for $\widetilde{\pi} \xleftarrow{\$} \mathrm{perm}(\mathcal{T}, n)$ and rand a function that for each input $(N, A, M) \in \{0,1\}^m \times \{0,1\}^* \times \{0,1\}^*$ returns a random string of size $|M| + t$ bits. The query complexity of the adversary is measured by parameters $q_{e,j}$, $q_{d,j}$, ℓ, and σ_j.

Let ρ be a random function that for each input $(N, A, C) \in \{0,1\}^m \times \{0,1\}^* \times \{0,1\}^*$ returns a random string of size n bits. We describe an alternative authenticated encryption scheme AE'' based on tweakable permutation $\widetilde{\pi}$ and on ρ. Its encryption function $\overline{\overline{\mathsf{enc}}}$ and decryption function $\overline{\overline{\mathsf{dec}}}$ are given in Algorithms 5 and 6, respectively.

Algorithm 5 encryption $\overline{\overline{\mathsf{enc}}}$

Input: (N, A, M)
Output: (C, T)

$M_1 \ldots M_{\ell_M} \xleftarrow{n} M$
for $i = 1, \ldots, \ell_M$ **do**
  $C_i \leftarrow M_i \oplus \widetilde{\pi}((i-1, 1), N \| 0^{n-m})$
$C \leftarrow \lfloor C_1 \ldots C_{\ell_M} \rfloor_{|M|}$
$A_1 \ldots A_{\ell_A} \xleftarrow{n} N \| A \| 1$
$C_1 \ldots C_{\ell_C} \xleftarrow{n} C \| 1$
$T \leftarrow \rho(N, A, C)$
return $(C, \lfloor T \rfloor_t)$

Algorithm 6 decryption $\overline{\overline{\mathsf{dec}}}$

Input: (N, A, C, T)
Output: M or $\perp$

$C_1 \ldots C_{\ell_M} \xleftarrow{n} C$
for $i = 1, \ldots, \ell_M$ **do**
  $M_i \leftarrow C_i \oplus \widetilde{\pi}((i-1, 1), N \| 0^{n-m})$
$M \leftarrow \lfloor M_1 \ldots M_{\ell_M} \rfloor_{|C|}$
$A_1 \ldots A_{\ell_A} \xleftarrow{n} N \| A \| 1$
$C_1 \ldots C_{\ell_C} \xleftarrow{n} C \| 1$
$\bar{T} \leftarrow \rho(N, A, C)$
return $\lfloor \bar{T} \rfloor_t = T \ ? \ M : \perp$

Proceeding from (23):

$$\begin{aligned}
(23) \leq\ & \Delta_{\mathcal{A}_j}\left(\overline{\mathsf{enc}}^{\widetilde{\pi}}, \overline{\mathsf{dec}}^{\widetilde{\pi}} \ ; \ \overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}, \overline{\overline{\mathsf{dec}}}^{\widetilde{\pi},\rho}\right) \\
&+ \Delta_{\mathcal{A}_j}\left(\overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}, \overline{\overline{\mathsf{dec}}}^{\widetilde{\pi},\rho} \ ; \ \overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}, \perp\right) \\
&+ \Delta_{\mathcal{A}_j}\left(\overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}, \perp \ ; \ \mathsf{rand}, \perp\right) . \qquad (24)
\end{aligned}$$

We will analyze the three distances of (24) separately.

First Distance of (24)

Define the function $h : \{0,1\}^m \times \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^n$ as

$$h^{\widetilde{\pi}}(N, A, C) = A_1 \oplus \left(\bigoplus_{i=2}^{\ell_A} \widetilde{\pi}((i-1, 0), A_i)\right) \oplus \left(\bigoplus_{i=1}^{\ell_C} \widetilde{\pi}((i-1, 2), C_i)\right) ,$$

where $A_1 \ldots A_{\ell_A} \xleftarrow{n} N \| A \| 1$ and $C_1 \ldots C_{\ell_C} \xleftarrow{n} C \| 1$. This function is $(2^n - 1)^{-1}$-uniform: for any distinct $(N, A, C) \neq (N', A', C')$, the probability over the drawing of $\widetilde{\pi} \xleftarrow{\$} \mathrm{perm}(\mathcal{T}, n)$ that $h^{\widetilde{\pi}}(N, A, C) = h^{\widetilde{\pi}}(N', A', C')$ is at most $(2^n - 1)^{-1}$.

Next, define $f : \{0,1\}^m \times \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^n$ as

$$f^{\widetilde{\pi}}(N, A, C) = \widetilde{\pi}((0, 0), h^{\widetilde{\pi}}(N, A, C)) .$$

In $\overline{\mathsf{enc}}^{\widetilde{\pi}}$ and $\overline{\mathsf{dec}}^{\widetilde{\pi}}$ the tag (before truncation) is computed as $f^{\widetilde{\pi}}(N, A, C)$, whereas in $\overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}$ and $\overline{\overline{\mathsf{dec}}}^{\widetilde{\pi},\rho}$ it is computed as $\rho(N, A, C)$. Therefore, the first distance of (24) is at most $\Delta_{\mathcal{A}'_j}\left(f^{\widetilde{\pi}} \ ; \ \rho\right)$, where $\mathcal{A}'_j$ is an adversary that makes $q_{e,j} + q_{d,j}$ construction queries, each query at most ℓ padded nonce and associated data and ciphertext blocks. Here, we make use of the fact that $f^{\widetilde{\pi}}$ only evaluates its tweakable permutation for tweaks $(\cdot, 0)$ and $(\cdot, 2)$, these tweaks do not occur elsewhere in the encryption and decryption function, and thus $\mathcal{A}'_j$ can properly simulate $\mathcal{A}_j$'s oracles.

Looking inside $f^{\widetilde{\pi}}$, it consists of an independent composition of $h^{\widetilde{\pi}}$, that never evaluates its primitive for tweak $(0,0)$, and $\widetilde{\pi}((0,0),\cdot)$. We replace $\widetilde{\pi}((0,0),\cdot)$ with a random function $\tau \xleftarrow{\$} \text{func}(n)$, which by the PRP-PRF switch comes at a cost of $\binom{q_{e,j}+q_{d,j}}{2}/2^n$:

$$\Delta_{\mathcal{A}'_j}\left(f^{\widetilde{\pi}} \ ; \ \rho\right) \le \Delta_{\mathcal{A}'_j}\left(\tau \circ h^{\widetilde{\pi}} \ ; \ \rho\right) + \binom{q_{e,j}+q_{d,j}}{2}/2^n \,.$$

The function $\tau \circ h^{\widetilde{\pi}}$ is perfectly indistinguishable from ρ as long as no two inputs to $h^{\widetilde{\pi}}$ collide. As $h^{\widetilde{\pi}}$ is $(2^n-1)^{-1}$-uniform, we in turn have

$$\Delta_{\mathcal{A}'_j}\left(\tau \circ h^{\widetilde{\pi}} \ ; \ \rho\right) \le \binom{q_{e,j}+q_{d,j}}{2}(2^n-1)^{-1} \,.$$

Concluding, we obtain for the first distance of (24):

$$\Delta_{\mathcal{A}_j}\left(\overline{\mathsf{enc}}^{\widetilde{\pi}}, \overline{\mathsf{dec}}^{\widetilde{\pi}} \ ; \ \overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}, \overline{\overline{\mathsf{dec}}}^{\widetilde{\pi},\rho}\right) \le 3\binom{q_{e,j}+q_{d,j}}{2}/2^n \,. \tag{25}$$

Second Distance of (24)

The adversary $\mathcal{A}_j$ can only distinguish both worlds if it ever makes a non-trivial evaluation to $\overline{\overline{\mathsf{dec}}}^{\widetilde{\pi},\rho}$ that succeeds. Consider any forgery attempt (N,A,C,T). If the tuple (N,A,C) occurred in an earlier encryption query, then necessarily the tag T must differ and the forgery will not succeed. Otherwise, ρ has never been evaluated on (N,A,C) before, and $\lfloor \rho(N,A,C) \rfloor_t = T$ with probability $1/2^t$. By summing over all $q_{d,j}$ forgery attempts, we obtain for the second distance of (24):

$$\Delta_{\mathcal{A}_j}\left(\overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}, \overline{\overline{\mathsf{dec}}}^{\widetilde{\pi},\rho} \ ; \ \overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}, \perp\right) \le q_{d,j}/2^t \,. \tag{26}$$

Third Distance of (24)

We remark that every query is made for a unique nonce, and in more detail:

- The i-th block of ciphertext equals $\widetilde{\pi}((i-1,1),N) \oplus M_i$, where M_i is the i-th block of plaintext;
- The tag equals $\rho(N,A,C)$.

The tweakable permutation $\widetilde{\pi}$ is independent for different tweaks, but two different inputs for the same tweak never collide. Therefore, this third distance of (24) satisfies

$$\Delta_{\mathcal{A}_j}\left(\overline{\overline{\mathsf{enc}}}^{\widetilde{\pi},\rho}, \perp \ ; \ \mathsf{rand}, \perp\right) \le \ell\binom{q_{e,j}}{2}/2^n \,. \tag{27}$$

Conclusion

Proceeding from (24) and the individual bounds of (25), (26), and (27), we obtain:

$$(23) \leq 3\binom{q_{e,j} + q_{d,j}}{2}/2^n + q_{d,j}/2^t + \ell\binom{q_{e,j}}{2}/2^n\,. \tag{28}$$

Third Step: Conclusion

Recall that in the second step, we dropped the subscripts and focused on a single-user case, as inspired by (22). We have to sum the bound of (28) over $j = 1, \ldots, \mu$, and maximize over the choice of $q_{e,j}$ and $q_{d,j}$ such that $\sum_{j=1}^{\mu} q_{e,j} = q_e$ and $\sum_{j=1}^{\mu} q_{d,j} = q_d$. As (28) is convex in $q_{e,j}$ and $q_{d,j}$, we obtain from (22):

$$(19) \leq \mathbf{Adv}^{\mu\text{-tprp}}_{\mathsf{SiM}}(\mathcal{A}') + 3\binom{q_e + q_d}{2}/2^n + q_d/2^t + \ell\binom{q_e}{2}/2^n\,,$$

and this completes the proof of Theorem B.3.