# Efficient Length Doubling From Tweakable Block Ciphers

Yu Long Chen[1], Atul Luykx[1,2], Bart Mennink[3,4] and Bart Preneel[1]

[1] imec-COSIC, KU Leuven, Belgium
yulong.chen@student.kuleuven.be, atul.luykx@esat.kuleuven.be
[2] Department of Computer Science, University of California, Davis
One Shields Ave, Davis, California 95616 USA
[3] Digital Security Group, Radboud University, Nijmegen, The Netherlands
b.mennink@cs.ru.nl
[4] CWI, Amsterdam, The Netherlands

**Abstract.** We present a length doubler, LDT, that turns an $n$-bit tweakable block cipher into an efficient and secure cipher that can encrypt any bit string of length $[n..2n-1]$. The LDT mode is simple, uses only two cryptographic primitive calls (while prior work needs at least four), and is a strong length-preserving pseudorandom permutation if the underlying tweakable block ciphers are strong tweakable pseudorandom permutations. We demonstrate that LDT can be used to neatly turn an authenticated encryption scheme for integral data into a mode for arbitrary-length data.

**Keywords:** length doubler · LDT · tweakable block ciphers · authenticated encryption

## 1 Introduction

One of the most important building blocks in cryptography are block ciphers—deterministic functions that encrypt bit strings of length $n$ into bit strings of the same length. Many applications, however, deal with arbitrary-length messages, hence block ciphers on their own are not sufficient. Variable-input-length (VIL) encryption is achieved by evaluating a block cipher iteratively in a mode of operation. Basic solutions such as CBC mode can only encrypt messages of size a multiple of $n$. To handle messages whose size is *not* a multiple of $n$ bits, one can, and often does [ABL+13, ABD+16, MV04, KR11], pad the data to an integral number of $n$-bit blocks.

Padding is, in many cases, an undesirable solution: message length is typically not preserved, which means the resulting ciphertext will always be larger or equal to the original plaintext. This makes the solution unsuitable for disk encryption (where the size of ciphertext and plaintext must remain the same as the sector size of the disk) and low-bandwidth network protocols (as an increase in ciphertext length results in more data to be transmitted).

Many solutions for turning a block cipher into a VIL cipher have appeared over the last years, such as EME [Hal04], TET [Hal07], HEH [Sar07], HCTR [WFW05], HCH [CS06], and XCB [MF07], but none of the above methods is generic, as these algorithms do not specify methods of extending existing encryption schemes to handle fractional message data. A more general method of using a block cipher mode of operation without ciphertext expansion is through ciphertext stealing [Dae95]. However, it only works on modes of operation where each ciphertext block can be decrypted independently of each other. For instance, applying ciphertext stealing to the tweakable online cipher TC3 [RZ11] is not possible.


IACR Transactions on Symmetric Cryptology ISSN 2519-173X, Vol. 2017, No. 3, pp. 253–270
DOI:10.13154/tosc.v2017.i3.253-270

An elegant way of achieving arbitrary length encryption generically is by using length doublers, as introduced by Ristenpart and Rogaway [RR07]. At a high level, a length doubler is a deterministic length-preserving bijection $\mathcal{E} : \mathcal{K} \times \mathcal{M} \to \mathcal{M}$ where $\mathcal{M} = \{0,1\}^{[n..2n-1]}$ and $n$ is the block size of the underlying primitive. Length-preserving VIL encryption can then be achieved by gluing a VIL encryption scheme for integral data blocks with the doubler. Length doublers are suitable solutions for various authenticated encryption schemes that treat integral and fractional data separately. The CAESAR submission AES-COPA [ABL+13, ABL+15] used the XLS length doubler [RR07] to process arbitrary length messages, but a later attack by Nandi on XLS [Nan14] rendered the solution in the COPA mode insecure [Nan15]. Besides broken XLS, there are other constructions of length doublers such as DE by Nandi [Nan09] and HEM by Zhang [Zha12]. However, both constructions make use of four cryptographic primitive calls.

## 1.1 Our Construction

We introduce the length doubler LDT, short for length-doubling with tweakable block ciphers. It makes two calls to a tweakable block cipher with block size $n$ bits, and can handle messages of size $n + s$ bits, with $s \in [0..n-1]$. The mode consists of three layers: in the first layer, the tweakable block cipher is evaluated on the first $n$ bits of the message, with the remaining $s$ bits functioning as tweak. In the second layer, the updated state is mixed using a mixing function. Then, the tweakable block cipher is evaluated for the second time on the updated state. LDT uses ideas of XLS and HEM, in particular the usage of a mix function. As with XLS and HEM, the construction is described generically, allowing any choice of primitives. LDT is depicted in Figure 1 and a formal description is given in Section 3. Note that, by construction, the decryption function of LDT is very similar to the encryption function, and can be easily implemented using the encryption circuit.

In Section 4, we prove that LDT is a secure length-preserving pseudorandom permutation. More specifically, if the underlying tweakable block cipher is secure, we prove security of LDT up to the birthday bound $q^2/2^n$. The proof is performed using Patarin's H-coefficient technique [Pat91, Pat08]. As we demonstrate in Section 5, the bound is tight. In more detail, a distinguisher that makes queries with large $s$ can distinguish the scheme from random in approximately $2^{n-s/2}$ queries, giving birthday bound security for $s = n-1$. It may be possible that our scheme is beyond birthday bound secure if a tighter upper bound (rather than $n-1$) on $s$ is imposed.

In Section 6 we present a generic construction that uses LDT to turn an authenticated encryption scheme that can only handle integral data into a scheme that can handle data of arbitrary size greater than $n$ bits. We prove that the combined construction inherits the security of the original construction, up to the security of LDT and up to the probability of accidental collisions at the point where LDT is glued to the original scheme. The generic construction is comparable to how AES-COPA [ABL+13, ABL+15] used XLS, but it is more general and it is provided with a formal proof.

## 1.2 Comparison

The first length doubler in literature was XLS by Ristenpart and Rogaway [RR07], but it did not achieve the claimed security level as pointed out by Nandi [Nan14]. Other constructions are DE by Nandi [Nan09] and HEM by Zhang [Zha12], both of which are proven secure up to the birthday bound. Cook, Yung, and Keromytis [CYK04b, CYK04a] introduced the "elastic block cipher", which solves essentially the same length-doubling problem. However, in contrast to DE, HEM, and LDT, Cook et al.'s construction does not allow for a reductionist security argument to an underlying, well-analyzed, primitive.

LDT is closely related to HEM, but differs in several aspects. Most importantly, HEM consists of four "rounds" (and minor precomputations), two block cipher calls sandwiched by two universal hash function calls, whereas LDT consists only of two tweakable block cipher calls. A tweakable block cipher can be constructed from a (conventional) block cipher and a universal hash function (see Liskov et al. [LRW02]), but other solutions—including dedicated designs—exist, making LDT more generic and easier to interpret.

Another improvement over HEM is that HEM uses an *$\epsilon$-good mixing function* (just like XLS), a mathematical function where $\epsilon$ is an upper bound on the probability that the mixing function is bad (see Appendix A for details). For LDT's security, we do not need such a strongly restricted function, and we use what we call a *pure mixing function* (see Section 2.4): a simpler mathematical primitive of which the quality is not bound by some probability measure. In particular, whereas XLS and HEM required $\epsilon$ to be small, any 1-good mixing function suffices for LDT.

A more detailed comparison of LDT with DE and HEM (and XLS for completeness) is given in Table 1. It shows that LDT compares favorably, most importantly in the key size and the number of cryptographic primitive calls. This, concretely, means that LDT is the most efficient solution if one uses a dedicated tweakable block cipher such as SKINNY [BJK+16], as long as two tweakable block cipher evaluations are cheaper than two universal hash function evaluations plus either two block cipher evaluations (in HEM) or one block cipher evaluation and one weak pseudorandom function evaluation (in DE). Minor efficiency gains are achieved in the mixing function. In particular, the mixing function that we suggest for LDT, $mix(A, B) = (B, A)$, suffices for our construction but not for HEM, as it is only 1-good. Note that for this swap function, the structure of LDT for larger $s$ is similar to SmallBlock [MI11] with the top and bottom universal hash functions omitted.

Table 1: Comparison of LDT with existing length doublers. Below we equate universal hash function calls with block cipher and tweakable block cipher calls as a heuristic for efficiency, however the relative efficiency of each of these primitive calls depends on the implementation.

| length doubler | security ($\log_2$) | key length | cryptographic primitive calls | mixing function | note |
|---|---|---|---|---|---|
| XLS | $n/2$ | $2n$ | 3 | $\epsilon$-good | [RR07], broken in [Nan14] |
| DE | $n/2$ | $5n$ | 4 | - | [Nan09] |
| HEM | $n/2$ | $3n$ | 4 | $\epsilon$-good | [Zha12] |
| LDT | $n/2$ | $2n$ | 2 | pure | Section 3 |

## 2 Preliminaries

We denote by $(\{0,1\}^n)^+$ the set of strings whose length is a positive multiple of $n$ bits. For two bit strings $X, Y \in \{0,1\}^*$, we let $X\|Y$ or $XY$ be their concatenation and $X \oplus Y$ their bitwise exclusive or. We denote by $|X|$ the length of the string $X$. By $\lfloor X \rfloor_s$ we denote the $s$ most significant bits of $X$. For a natural number $n$, we denote by $\{0,1\}^n$ the set of bit strings of size $n$. For natural numbers $m \leq n$ we define $\{0,1\}^{[m..n]} = \bigcup_{m \leq i \leq n} \{0,1\}^i$. For some value $Z$, we denote by $z \leftarrow Z$ the assignment of $Z$ to the variable $z$. For some finite set $S$, we denote by $s \xleftarrow{\$} S$ the uniformly random selection of $s$ from $S$. Given a function $\pi : \{0,1\}^n \to \{0,1\}^{n'}$, let $\lfloor \pi \rfloor_m : \{0,1\}^n \to \{0,1\}^m$ be the function which removes the leftmost $n' - m$ bits from the output of $\pi$. We denote by $\mathrm{Func}(n, m)$ the set of all functions from $\{0,1\}^n$ to $\{0,1\}^m$.

For a natural number $n$ and $X \in \{0,1\}^{[0..n-1]}$, we define a padding function

$$pad(X) = X \| 10^{n-|X|-1} .$$

As the function is injective, we can consider its inverse *unpad* that on input of a string of length $n$ removes the rightmost string $10^*$ and outputs the remainder. We denote by $\langle t \rangle_n$ the encoding of a value $t \in \{0, \ldots, 2^n - 1\}$ as an $n$-bit string.

## 2.1 Tweakable Block Ciphers

For $k, t, n \in \mathbb{N}$, a tweakable block cipher is a function $\tilde{E} : \{0,1\}^k \times \{0,1\}^t \times \{0,1\}^n \to \{0,1\}^n$ such that for fixed key $K \in \{0,1\}^k$ and tweak $T \in \{0,1\}^t$, $\tilde{E}_K(T, \cdot) = \tilde{E}(K, T, \cdot)$ is a permutation on $\{0,1\}^n$. We denote its inverse (for fixed key and tweak) by $\tilde{E}_K^{-1}(T, \cdot) = \tilde{E}^{-1}(K, T, \cdot)$. The key is usually a secret parameter; the tweak is a public parameter, and $\tilde{E}_K^{-1}$ should behave independently for different tweaks.

Denote by $\mathrm{Perm}(n)$ the set of all permutations on $\{0,1\}^n$. Denote by $\widetilde{\mathrm{Perm}}(t, n)$ the set of all functions $\tilde{\pi} : \{0,1\}^t \times \{0,1\}^n \to \{0,1\}^n$ such that $\tilde{\pi}(T, \cdot)$ is in $\mathrm{Perm}(n)$ for all $T \in \{0,1\}^t$. The security of a tweakable block cipher $\tilde{E}$ is measured by considering a distinguisher $\mathcal{D}$ that is given two-sided access to either $\tilde{E}_K$ for secret key $K \xleftarrow{\$} \{0,1\}^k$, or a random tweakable permutation $\tilde{\pi} \xleftarrow{\$} \widetilde{\mathrm{Perm}}(t, n)$, and its goal is to determine which oracle it is given access to:

$$\mathbf{Adv}^{\widetilde{\mathrm{sprp}}}_{\tilde{E}}(\mathcal{D}) = \left| \Pr\left[K \xleftarrow{\$} \{0,1\}^k : \mathcal{D}^{\tilde{E}_K, \tilde{E}_K^{-1}} = 1\right] - \Pr\left[\tilde{\pi} \xleftarrow{\$} \widetilde{\mathrm{Perm}}(t, n) : \mathcal{D}^{\tilde{\pi}, \tilde{\pi}^{-1}} = 1\right] \right| . \quad (1)$$

## 2.2 Length Doublers

For $k, n \in \mathbb{N}$, a length doubler is a function $\mathcal{E} : \{0,1\}^k \times \{0,1\}^{[n..2n-1]} \to \{0,1\}^{[n..2n-1]}$ such that for fixed key $K \in \{0,1\}^k$, $\mathcal{E}_K(\cdot) = \mathcal{E}(K, \cdot)$ is a length preserving invertible function on $\{0,1\}^{[n..2n-1]}$. We denote its inverse (for fixed key) by $\mathcal{E}_K^{-1}(\cdot) = \mathcal{E}^{-1}(K, \cdot)$.

Note that $\mathcal{E}$ should behave like a random permutation for every length input $m \in [n..2n-1]$. Formally, denote by $\mathrm{VPerm}([n..2n-1])$ the set of all functions $\rho$ that are length-preserving and invertible. Note that a randomly drawn function $\rho \xleftarrow{\$} \mathrm{Vperm}([n..2n-1])$ is equivalent to $n$ random permutations $\rho_i \xleftarrow{\$} \mathrm{Perm}(i)$ for $i = n, \ldots, 2n-1$ as

$$\rho(M) = \rho_{|M|}(M) . \quad (2)$$

The security of a length doubler $\mathcal{E}$ is measured by considering a distinguisher $\mathcal{D}$ that is given two-sided access to either $\mathcal{E}_K$ for secret key $K \xleftarrow{\$} \{0,1\}^k$, or a random length-preserving permutation $\rho \xleftarrow{\$} \mathrm{VPerm}([n..2n-1])$, and its goal is to determine which oracle it is given access to:

$$\mathbf{Adv}^{\mathrm{vsprp}}_{\mathcal{E}}(\mathcal{D}) = \left| \Pr\left[K \xleftarrow{\$} \{0,1\}^k : \mathcal{D}^{\mathcal{E}_K, \mathcal{E}_K^{-1}} = 1\right] - \Pr\left[\rho \xleftarrow{\$} \mathrm{VPerm}([n..2n-1]) : \mathcal{D}^{\rho, \rho^{-1}} = 1\right] \right| . \quad (3)$$

## 2.3 H-Coefficient Technique

Our proof will rely on the H-coefficient technique by Patarin [Pat91, Pat08], but we will follow the modernization of Chen and Steinberger [CS14].

Consider two oracles $\mathcal{O}$ and $\mathcal{P}$, and any distinguisher $\mathcal{D}$ that has query access to either of these oracles. The distinguisher's goal is to distinguish both worlds and we denote by

$$\mathbf{Adv}(\mathcal{D}) = \left|\Pr\left[\mathcal{D}^{\mathcal{O}} = 1\right] - \Pr\left[\mathcal{D}^{\mathcal{P}} = 1\right]\right|$$

its advantage. If we denote the maximum amount of queries by $q$, we can define a transcript $\tau$ which summarizes all query-response tuples seen by the distinguisher during its interaction with its oracle $\mathcal{O}$ or $\mathcal{P}$. Denote by $X_{\mathcal{O}}$ (resp. $X_{\mathcal{P}}$) the probability distribution of transcripts when interacting with $\mathcal{O}$ (resp. $\mathcal{P}$). We call a transcript $\tau \in \mathcal{T}$ attainable if $Pr[X_{\mathcal{P}} = \tau] > 0$, or in other words if the transcript $\tau$ can be obtained from an interaction with $\mathcal{P}$.

**Lemma 1** (H-coefficient Technique)**.** *Consider a fixed distinguisher $\mathcal{D}$. Define a partition $\mathcal{T} = \mathcal{T}_{\text{good}} \cup \mathcal{T}_{\text{bad}}$, where $\mathcal{T}_{\text{good}}$ is the subset of $\mathcal{T}$ which contains all the "good" transcripts and $\mathcal{T}_{\text{bad}}$ is the subset with all the "bad" transcripts. Let $0 \leq \epsilon \leq 1$ be such that for all $\tau \in \mathcal{T}_{\text{good}}$:*

$$\frac{Pr(X_{\mathcal{O}} = \tau)}{Pr(X_{\mathcal{P}} = \tau)} \geq 1 - \epsilon\,. \tag{4}$$

*Then, we have $\mathbf{Adv}(\mathcal{D}) \leq \epsilon + Pr(X_{\mathcal{P}} \in \mathcal{T}_{\text{bad}})$.*

Conventionally, $\mathcal{O}$ corresponds to the real world and $\mathcal{P}$ to the ideal world, but one can (and in our proof we will) swap their roles.

## 2.4 Mixing Functions

Ristenpart and Rogaway [RR07] introduced $\epsilon$-good mixing functions for their length doubler, and they were later used by Zhang [Zha12] as well. In this work, we will also use mixing functions, but these do not necessarily need to be $\epsilon$-good. We refer to these mixing functions as *pure* mixing functions.

**Definition 1.** Let $m, n \in \mathbb{N}$ such that $m \leq n$. Let $mix : \cup_{s=m}^{n}(\{0,1\}^s)^2 \to \cup_{s=m}^{n}(\{0,1\}^s)^2$ be a length-preserving permutation, define by $mix_L$ the left half of its evaluation and by $mix_R$ its right half. The mixing function is called *pure* if for all $s \in [m..n]$ we have:

- $mix_L(A, \cdot)$ is a permutation for all $A \in \{0,1\}^s$,
- $mix_R(\cdot, B)$ is a permutation for all $B \in \{0,1\}^s$.

A pure mixing function resembles ideas of, but is a much weaker concept than a multipermutation [SV93]. A simple example of a pure mixing function is $mix(A, B) = (B, A)$. For completeness, we describe $\epsilon$-good mixing functions as defined by Ristenpart and Rogaway [RR07] in Appendix A. We remark that above-mentioned pure mixing function is 1-good, which essentially means that it is a bad mixing function that cannot be used to make HEM by Zhang [Zha12] secure.

We will rely on the following observation.

**Lemma 2.** *Let mix be a pure mixing function as in Definition 1. Given $B, D \in \{0,1\}^s$, there exists a unique value $A \in \{0,1\}^s$ such that $mix_R(A, B) = D$ and a unique value $C \in \{0,1\}^s$ such that $mix_L^{-1}(C) = (A, B)$.*

*Proof.* The former follows from the fact that $mix_R(\cdot, B)$ is a permutation for all $B \in \{0,1\}^s$ (the second condition of Definition 1). Given the existence of $A$, uniqueness of the value $C$ follows naturally by the fact that $C = mix_L(A, B)$ and $mix_L(A, \cdot)$ is a permutation for all $A \in \{0,1\}^s$. □

**Algorithm 1** $\mathcal{E} = \mathrm{LDT}[\tilde{E}, mix]$ encryption

**Input:** $(K_1, K_2) \in \{0,1\}^{2k}$, $M \in \{0,1\}^{[n..2n-1]}$
**Output:** $C \in \{0,1\}^{|M|}$
1: $s \leftarrow |M| - n$
2: $M_1 \| M_2 \leftarrow M$ with $|M_1| = n$ and $|M_2| = s$
3: $Z \| M_3 \leftarrow \tilde{E}_{K_1}(pad(M_2), M_1)$ with $|Z| = n - s$ and $|M_3| = s$
4: $(C_3, C_2) \leftarrow mix(M_3, M_2)$
5: $C_1 \leftarrow \tilde{E}_{K_2}(pad(C_2), Z \| C_3)$
6: **return** $C_1 \| C_2$

Therefore, related to a function $mix$ we can describe two functions $mix_L^{\mathrm{in}}$ and $mix_L^{\mathrm{out}}$ that, given the right parts of the input and output determine the (unique) left input and output values. Formally:

$$mix_L^{\mathrm{in}}(B, D) = A \iff mix_R(A, B) = D\,, \tag{5}$$

$$mix_L^{\mathrm{out}}(B, D) = C \iff mix_R^{-1}(C, D) = B\,. \tag{6}$$

## 3 LDT Doubler

We introduce our length doubler LDT, designed upon a tweakable block cipher and a mixing function.

Let $k, n \in \mathbb{N}$. Let $\tilde{E} : \{0,1\}^k \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a tweakable block cipher, and $mix : S^2 \to S^2$ for $S = \{0,1\}^{[0..n-1]}$ a pure mixing function. Our length doubler $\mathcal{E} = \mathrm{LDT}[\tilde{E}, mix]$ with key space $\{0,1\}^{2k}$ and state $\{0,1\}^{[n..2n-1]}$ is described in Algorithm 1 and given in Figure 1. Note that the decryption function is very similar to the encryption function and can be defined as

$$\mathrm{LDT}[\tilde{E}, mix]^{-1}_{K_1,K_2} = \mathrm{LDT}[\tilde{E}^{-1}, mix^{-1}]_{K_2,K_1}\,. \tag{7}$$

## 4 Security Lower Bound

We prove security of the LDT length doubler of Section 3.

**Theorem 1.** *Let $k, n \in \mathbb{N}$. Let $\widetilde{E} : \{0,1\}^k \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a tweakable block cipher, $mix : S^2 \to S^2$ for $S = \{0,1\}^{[0\ldots n-1]}$ a pure mixing function, and consider $\mathcal{E} := \mathrm{LDT}[\widetilde{E}, mix]$ of Algorithm 1. For any distinguisher $\mathcal{D}$ making at most $q$ queries, there exist distinguishers $\mathcal{D}'_1$ and $\mathcal{D}'_2$ with the same query complexity such that*

$$\mathbf{Adv}^{\mathrm{vsprp}}_{\mathcal{E}}(\mathcal{D}) \le \mathbf{Adv}^{\widetilde{\mathrm{sprp}}}_{\widetilde{E}}(\mathcal{D}'_1) + \mathbf{Adv}^{\widetilde{\mathrm{sprp}}}_{\widetilde{E}}(\mathcal{D}'_2) + \frac{q(q-1)}{2^n}\,. \tag{8}$$

*Proof.* Consider any distinguisher $\mathcal{D}$ making at most $q$ queries. It has access to either $\mathcal{E}_K$ for $K = (K_1, K_2) \xleftarrow{\$} \{0,1\}^{2k}$ or a random length-preserving invertible permutation $\rho \xleftarrow{\$} \mathrm{VPerm}([n \ldots 2n-1])$. As $\mathcal{E}_K$ evaluates its underlying tweakable block ciphers for the two independently generated random keys $K_1, K_2$, we use an extended notation for $\mathcal{E}_K$:

$$\mathcal{E}_K = \mathcal{E}[\widetilde{E}_{K_1}, mix, \widetilde{E}_{K_2}]\,.$$

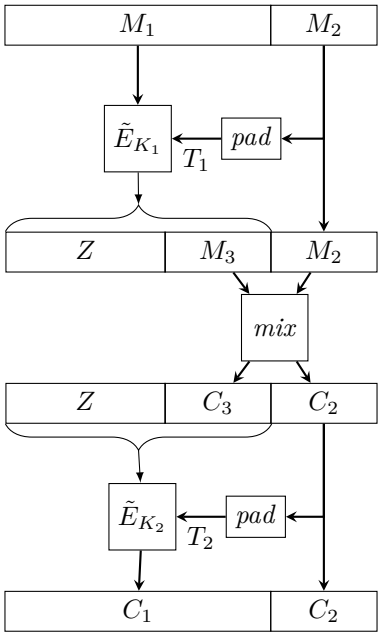


Figure 1: Encryption of length doubler LDT, with $\tilde{E}$ a tweakable block cipher and *mix* a mix function.

Let $\widetilde{\pi}_1, \widetilde{\pi}_2 \xleftarrow{\$} \widetilde{\mathrm{Perm}}(n,n)$. We have

$$\begin{aligned}\mathbf{Adv}^{\mathrm{vsprp}}_{\mathcal{E}}(\mathcal{D}) &= \Delta_{\mathcal{D}}\Big(\mathcal{E}[\widetilde{E}_{K_1}, mix, \widetilde{E}_{K_2}]^{\pm}\ ;\ \rho^{\pm}\Big) \\ &\leq \Delta_{\mathcal{D}}\Big(\mathcal{E}[\widetilde{\pi}_1, mix, \widetilde{\pi}_2]^{\pm}\ ;\ \rho^{\pm}\Big) + \Delta_{\mathcal{D}'_1}\Big(\widetilde{E}_{K_1}\ ;\ \widetilde{\pi}_1\Big) + \Delta_{\mathcal{D}'_2}\Big(\widetilde{E}_{K_2}\ ;\ \widetilde{\pi}_2\Big) \\ &= \Delta_{\mathcal{D}}\Big(\mathcal{E}[\widetilde{\pi}_1, mix, \widetilde{\pi}_2]^{\pm}\ ;\ \rho^{\pm}\Big) + \mathbf{Adv}^{\widetilde{\mathrm{sprp}}}_{\widetilde{E}}(\mathcal{D}'_1) + \mathbf{Adv}^{\widetilde{\mathrm{sprp}}}_{\widetilde{E}}(\mathcal{D}'_2)\,, \end{aligned} \tag{9}$$

for some distinguishers $\mathcal{D}'_1$ and $\mathcal{D}'_2$ with the same complexity as $\mathcal{D}$.

We focus on the remaining distance in (9). We will allow $\mathcal{D}$ to be information-theoretic. In other words, it has unbounded computational power and its advantage is solely measured by its query complexity. We can assume that $\mathcal{D}$ is deterministic and we will apply the H-coefficient technique of Lemma 1.

**Transcripts.** Oracle $\mathcal{O}$ will represent the ideal world $\rho^{\pm}$ and oracle $\mathcal{P}$ will represent the real world $\mathcal{E}[\widetilde{\pi}_1, mix, \widetilde{\pi}_2]^{\pm}$. Distinguisher $\mathcal{D}$ makes $q$ queries to its oracle ($\mathcal{O}$ or $\mathcal{P}$) and these queries are summarized in a transcript of the form

$$\tau = \{(M_1^{(i)}, M_2^{(i)}, C_1^{(i)}, C_2^{(i)}) \mid i = 1, \ldots, q\}\,.$$

We can assume without loss of generality that the distinguisher $\mathcal{D}$ does not repeat any query since both $\mathcal{O}$ and $\mathcal{P}$ give the same output with repeated input, which means that $\tau$ does not contain duplicate elements.

After $\mathcal{D}$'s interaction, but before it outputs 0/1, we disclose the values $Z^{(i)}$ for $i = 1, \ldots, q$. In the real world $\mathcal{P}$, these are the first $n - s$ bits of the output of $\widetilde{\pi}_1$ or for an inverse query $\widetilde{\pi}_2^{-1}$ (see also Figure 1). In the ideal world $\mathcal{O}$, the $Z$'s are generated as follows:

1: **for** $i = 1, \ldots, q$ **do**

2: $\mathcal{Z} \leftarrow \{Z^{(j)} \mid j < i \wedge (M_2^{(i)}, C_2^{(i)}) = (M_2^{(j)}, C_2^{(j)})\}$
3: $Z^{(i)} \xleftarrow{\$} \{0,1\}^{n-s} \backslash \mathcal{Z}$

Informally, the values $Z^{(i)}$ are generated uniformly at random with the constraint that $Z^{(i)} \neq Z^{(j)}$ whenever two distinct queries satisfy $(M_2^{(i)}, C_2^{(i)}) = (M_2^{(j)}, C_2^{(j)})$.

It might seem that the generation is not well-defined if $q > 2^{n-s}$ and there are sufficiently many queries such that $(M_2^{(i)}, C_2^{(i)}) = (M_2^{(j)}, C_2^{(j)})$. However, since transcript attainability is defined with respect to the real world, $\mathcal{P}$, any transcript for which more than $2^{n-s}$ $Z$ values are generated is unattainable, as established by the following claim.

**Claim.** *If transcript $\tau \in \mathcal{T}$ is attainable ($Pr[X_{\mathcal{P}} = \tau] > 0$), there do not exist $s \in [0, \ldots, n-1]$ and $\xi > 2^{n-s}$ indices $i_1, \ldots, i_\xi$ such that $|M_2^{(i_x)}| = |C_2^{(i_x)}| = s$ for $x = 1, \ldots, \xi$, and*

$$(M_2^{(i_1)}, C_2^{(i_1)}) = \cdots = (M_2^{(i_\xi)}, C_2^{(i_\xi)})\,. \tag{10}$$

*Proof.* Suppose to the contrary that for some $s \in [0, \ldots, n-1]$, there exist $\xi = 2^{n-s} + 1$ indices $i_1, \ldots, i_\xi$ such that $|M_2^{(i_x)}| = |C_2^{(i_x)}| = s$ and (10) holds. By Lemma 2, this particularly means that $M_3^{(i_1)} = \cdots = M_3^{(i_\xi)}$. However, the $M_1^{(i)}$'s are pairwise distinct, and $\widetilde{\pi}_1$ is a permutation for fixed $M_2^{(i_1)} = \cdots = M_2^{(i_\xi)}$. In other words, we have obtained a $(2^{n-s}+1)$-fold collision on the rightmost $s$ bits of an $n$-bits permutation, which is impossible by design. □

We denote the complete transcripts by

$$\tau = \{(M_1^{(i)}, M_2^{(i)}, Z^{(i)}, C_1^{(i)}, C_2^{(i)}) \mid i = 1, \ldots, q\}\,.$$

In our proof, we will not consider bad transcripts, hence $\mathcal{T}_{\text{bad}} = \varnothing$ and $\mathcal{T}_{\text{good}} = \mathcal{T}$.

**$\mathbf{Pr}(X_{\mathcal{O}} = \tau)/\mathbf{Pr}(X_{\mathcal{P}} = \tau)$.** Let $\tau \in \mathcal{T}_{\text{good}}$ be a good transcript. To determine the two probabilities, it suffices to compute the probability, over the drawing of the oracle, that the oracle extends $\tau$. Let $all_{X_{\mathcal{O}}}$ be the set of all possible oracles in the ideal world $\mathcal{O}$, and $comp_{X_{\mathcal{O}}}(\tau)$ the fraction of them compatible with $\tau$. Define $all_{X_{\mathcal{P}}}$ and $comp_{X_{\mathcal{P}}}(\tau)$ analogously. Then, we obtain $\Pr(X_{\mathcal{O}} = \tau) = |comp_{X_{\mathcal{O}}}(\tau)|/|all_{X_{\mathcal{O}}}|$ and $\Pr(X_{\mathcal{P}} = \tau) = |comp_{X_{\mathcal{P}}}(\tau)|/|all_{X_{\mathcal{P}}}|$.

We will introduce some parameters related to $\tau$.

- For $t = 0, \ldots, 2^n - 1$, $\alpha_t$ denotes the number of tuples in $\tau$ such that $pad(M_2^{(i)}) = \langle t \rangle_n$;
- For $t = 0, \ldots, 2^n - 1$, $\beta_t$ denotes the number of tuples in $\tau$ such that $pad(C_2^{(i)}) = \langle t \rangle_n$;
- For $t, t' = 0, \ldots, 2^n - 1$, $\gamma_{t,t'}$ denotes the number of tuples in $\tau$ such that $pad(M_2^{(i)}) = \langle t \rangle_n$ and $pad(C_2^{(i)}) = \langle t' \rangle_n$;
- For $s = 0, \ldots, n-1$, $\delta_s$ denotes the number of tuples in $\tau$ such that $|M_2^{(i)}| = |C_2^{(i)}| = s$.

For the real world $\mathcal{P}$, we have $|all_{X_{\mathcal{P}}}| = (2^n!)^{2^n}(2^n!)^{2^n}$, the number of elements in $\widetilde{\text{Perm}}(n,n) \times \widetilde{\text{Perm}}(n,n)$. The computation of $|comp_{X_{\mathcal{P}}}(\tau)|$ boils down to the number of oracles $\widetilde{\pi}_1, \widetilde{\pi}_2$ that could yield the transcript $\tau$. As *mix* is a pure mixing function, using the functions $mix_L^{\text{in}}$ of (5) and $mix_L^{\text{out}}$ of (6) we can determine for each tuple in the transcript the unique values $M_3^{(i)}$ (the last $s$ bits of the output of $\widetilde{\pi}_1$) and $C_3^{(i)}$ (the last $s$ bits of the input to $\widetilde{\pi}_2$). Therefore, each tuple in the transcript uniquely defines an input-output tuple

$$(pad(M_2^{(i)}), M_1^{(i)}) \mapsto (Z^{(i)} \| M_3^{(i)}) \tag{11}$$

for $\widetilde{\pi}_1$ and an input-output tuple

$$(pad(C_2^{(i)}), Z^{(i)} \| C_3^{(i)}) \mapsto (C_1^{(i)}) \tag{12}$$

for $\widetilde{\pi}_2$. The queries in $\tau$ are unique by assumption, and the entire transcript defines exactly $\alpha_t$ tuples in (11) and exactly $\beta_t$ tuples in (12) for tweak $\langle t \rangle_n$, where $t = 0, \ldots, 2^n - 1$. This leaves $\prod_{t=0}^{2^n-1}(2^n - \alpha_t)!$ tweakable permutations $\widetilde{\pi}_1$ and $\prod_{t=0}^{2^n-1}(2^n - \beta_t)!$ tweakable permutations $\widetilde{\pi}_2$ compliant with $v$, and thus

$$|comp_{X_{\mathcal{P}}}(\tau)| = \prod_{t=0}^{2^n-1}(2^n - \alpha_t)! \prod_{t=0}^{2^n-1}(2^n - \beta_t)!. \tag{13}$$

We obtain for the real world $\mathcal{P}$ that

$$\Pr(X_{\mathcal{P}} = \tau) = \frac{\prod_{t=0}^{2^n-1}(2^n - \alpha_t)! \prod_{t=0}^{2^n-1}(2^n - \beta_t)!}{(2^n!)^{2^n}(2^n!)^{2^n}} = \frac{1}{\prod_{t=0}^{2^n-1}(2^n)_{\alpha_t} \prod_{t=0}^{2^n-1}(2^n)_{\beta_t}}, \tag{14}$$

where $(x)_y = x!/(x-y)!$.

Define $T_s = \{t \mid |unpad(\langle t \rangle_n)| = s\}$. For the ideal world $\mathcal{O}$, we have

$$|all_{X_{\mathcal{O}}}| = \prod_{s=0}^{n-1} 2^{n+s}! \prod_{s=0}^{n-1} \prod_{t \in T_s} \prod_{t' \in T_s} 2^{n-s}!,$$

where the first part counts the number of elements in $\mathrm{VPerm}([n \ldots 2n-1])$ and the latter part counts the total number of solutions to the $Z$-values (which were disclosed to the distinguisher after the queries were made). The computation of $|comp_{X_{\mathcal{O}}}(\tau)|$ likewise boils down to computing the number of oracles $\rho$ that could yield transcript $\tau$, multiplied with the number of possible choices for $Z$-values that comply with $\{Z^{(1)}, \ldots, Z^{(q)}\}$ in the transcript. The queries in $\tau$ are unique by assumption, and for $s = 0, \ldots, n-1$ there are exactly $\delta_s$ queries of size $n + s$. In our disclosure to the distinguisher, the generation of the $Z$-values is in such a way that these are distinct whenever two queries satisfy $(M_2^{(i)}, C_2^{(i)}) = (M_2^{(j)}, C_2^{(j)})$. Therefore, we find

$$|comp_{X_{\mathcal{O}}}(\tau)| = \prod_{s=0}^{n-1}(2^{n+s} - \delta_s)! \prod_{s=0}^{n-1} \prod_{t \in T_s} \prod_{t' \in T_s} (2^{n-s} - \gamma_{t,t'})!,$$

using our definition of $\gamma_{t,t'}$ and $\delta_s$. As before, we obtain for the ideal world $\mathcal{O}$ that

$$Pr(X_{\mathcal{O}} = \tau) = \frac{1}{\prod_{s=0}^{n-1}(2^{n+s})_{\delta_s} \prod_{s=0}^{n-1} \prod_{t \in T_s} \prod_{t' \in T_s} (2^{n-s})_{\gamma_{t,t'}}}. \tag{15}$$

From (14) and (15) we can compute the fraction:

$$\begin{aligned} \frac{Pr(X_{\mathcal{O}} = \tau)}{Pr(X_{\mathcal{P}} = \tau)} &= \frac{\prod_{t=0}^{2^n-1}(2^n)_{\alpha_t} \prod_{t=0}^{2^n-1}(2^n)_{\beta_t}}{\prod_{s=0}^{n-1}(2^{n+s})_{\delta_s} \prod_{s=0}^{n-1} \prod_{t \in T_s} \prod_{t' \in T_s} (2^{n-s})_{\gamma_{t,t'}}} \\ &= \prod_{s=0}^{n-1} \frac{\prod_{t \in T_s}(2^n)_{\alpha_t} \prod_{t \in T_s}(2^n)_{\beta_t}}{(2^{n+s})_{\delta_s} \prod_{t \in T_s} \prod_{t' \in T_s} (2^{n-s})_{\gamma_{t,t'}}}. \end{aligned} \tag{16}$$

Note that, for all $s$,

$$\prod_{t \in T_s} \prod_{t' \in T_s} (2^{n-s})_{\gamma_{t,t'}} \leq \prod_{t \in T_s} \prod_{t' \in T_s} (2^{n-s})^{\gamma_{t,t'}}.$$

Proceeding from (16), we have

$$\begin{aligned}(16) &\geq \prod_{s=0}^{n-1} \frac{\prod_{t\in T_s}(2^n)_{\alpha_t}\prod_{t\in T_s}(2^n)_{\beta_t}}{(2^{n+s})_{\delta_s}\prod_{t\in T_s}\prod_{t'\in T_s}(2^{n-s})^{\gamma_{t,t'}}} \\ &\geq \prod_{s=0}^{n-1} \frac{(2^n)_{\delta_s}(2^n)_{\delta_s}}{(2^{n+s})_{\delta_s}(2^{n-s})^{\delta_s}}\,, \end{aligned} \tag{17}$$

where we use that for all $s \in \{0,\ldots,n-1\}$,

$$\sum_{t\in T_s}\alpha_t = \sum_{t\in T_s}\beta_t = \sum_{t\in T_s}\sum_{t'\in T_s}\gamma_{t,t'} = \delta_s\,,$$

and, in addition, $(x)_y(x)_z \geq (x)_{y+z}$. Proceeding from (17),

$$\begin{aligned}(17) &= \prod_{s=0}^{n-1}\prod_{i=0}^{\delta_s-1}\frac{(2^n-i)(2^n-i)}{(2^{n+s}-i)2^{n-s}} \\ &= \prod_{s=0}^{n-1}\prod_{i=0}^{\delta_s-1}\left(1-\frac{2i2^n-i2^{n-s}-i^2}{2^{2n}-i2^{n-s}}\right) \\ &\geq \prod_{s=0}^{n-1}\prod_{i=0}^{\delta_s-1}\left(1-\frac{2i}{2^n}\right)\,, \end{aligned} \tag{18}$$

where we use that $\frac{2i2^n-i2^{n-s}-i^2}{2^{2n}-i2^{n-s}} \leq \frac{2i}{2^n}$. Using that $(1-x)(1-y) = 1-x-y+xy \geq 1-x-y$, we can proceed from (18) as follows:

$$\begin{aligned}(18) &\geq 1-\sum_{s=0}^{n-1}\sum_{i=0}^{\delta_s-1}\frac{2i}{2^n} \\ &= 1-\frac{\sum_{s=0}^{n-1}(\delta_s^2-\delta_s)}{2^n} \\ &\geq 1-\frac{q^2-q}{2^n}\,, \end{aligned} \tag{19}$$

where we use that $q = \sum_{s=0}^{n-1}\delta_s$ and $q^2 \geq \sum_{s=0}^{n-1}\delta_s^2$. We conclude from (16,17,18,19) that

$$\frac{Pr(X_{\mathcal{O}}=\tau)}{Pr(X_{\mathcal{P}}=\tau)} \geq 1-\frac{q(q-1)}{2^n} =: 1-\epsilon\,. \qquad \square$$

## 5 Security Upper Bound

We describe an attack against LDT in approximately $2^{n-s/2}$ queries, where the distinguisher makes queries of size $n+s$ for $s \in [0, n-1]$. This attack is based on distinguishing a truncated permutation from a random function. In the attack, we essentially reduce the security of length doubler LDT to the advantage in distinguishing a truncated LDT from a random function.

**Theorem 2.** *Let $k, n \in \mathbb{N}$. Let $\widetilde{E} : \{0,1\}^k \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a tweakable block cipher, $mix : \cup_{s=0}^{n-1}(\{0,1\}^s)^2 \to \cup_{s=0}^{n-1}(\{0,1\}^s)^2$ a pure mixing function, and consider $\mathcal{E} := \mathrm{LDT}[\widetilde{E}, mix]$ of Algorithm 1. Let $s \in [kn, n-1]$ for some constant $k < 1$. There exists a distinguisher $\mathcal{D}$ making $q$ queries such that*

$$\mathbf{Adv}^{\mathrm{vsprp}}_{\mathcal{E}}(\mathcal{D}) \in \Omega\left(\frac{q^2}{2^{2n-s}}\right)\,. \tag{20}$$

*Proof.* Let $K = (K_1, K_2) \xleftarrow{\$} \{0,1\}^{2k}$ be the key to $\mathcal{E}_K$ and let $\rho \xleftarrow{\$} \mathrm{VPerm}([n \dots 2n-1])$ be a random length-preserving invertible permutation. For any distinguisher $\mathcal{D}'$, whose goal it is to distinguish $\lfloor \mathcal{E}_K \rfloor_s$ from $\lfloor \rho \rfloor_s$, there exists a distinguisher $\mathcal{D}$ for $\mathcal{E}_K$ versus $\rho$ with the same complexity and at least the same success probability:

$$\Delta_{\mathcal{D}'}\Big(\lfloor \mathcal{E}_K \rfloor_s \; ; \; \lfloor \rho \rfloor_s\Big) \leq \Delta_{\mathcal{D}}\Big(\mathcal{E}_K \; ; \; \rho\Big) = \mathbf{Adv}^{\mathrm{vsprp}}_{\mathcal{E}}(\mathcal{D}) \,, \tag{21}$$

as the advantage to distinguish a truncated length doubler from a truncated random length-preserving permutation is smaller or equal to the advantage to distinguish the non-truncated version of both.

In addition, let $\pi \xleftarrow{\$} \mathrm{Func}(n+s, s)$. Using the triangle inequality, we have

$$\Delta_{\mathcal{D}'}\Big(\lfloor \mathcal{E}_K \rfloor_s \; ; \; \pi\Big) \leq \Delta_{\mathcal{D}'}\Big(\lfloor \mathcal{E}_K \rfloor_s \; ; \; \lfloor \rho \rfloor_s\Big) + \Delta_{\mathcal{D}'}\Big(\lfloor \rho \rfloor_s \; ; \; \pi\Big) \,. \tag{22}$$

From (21) and (22) we obtain that for any distinguisher $\mathcal{D}'$ there exists a distinguisher $\mathcal{D}$ such that

$$\mathbf{Adv}^{\mathrm{vsprp}}_{\mathcal{E}}(\mathcal{D}) \geq \Delta_{\mathcal{D}'}\Big(\lfloor \mathcal{E}_K \rfloor_s \; ; \; \pi\Big) - \Delta_{\mathcal{D}'}\Big(\lfloor \rho \rfloor_s \; ; \; \pi\Big) \,. \tag{23}$$

Our goal is to describe a distinguisher $\mathcal{D}'$ such that the right hand side of (23) is non-negligible. Regarding $\Delta_{\mathcal{D}'}\Big(\lfloor \mathcal{E}_K \rfloor_s \; ; \; \pi\Big)$, when $M_2$ is fixed, $\lfloor \mathcal{E}_K(\cdot, M_2) \rfloor_s$ is an $n$-bit permutation with the leftmost $n-s$ bits truncated. Hall et al. [HWKS98] show that there exists a distinguisher $\mathcal{D}'$ such that

$$\Delta_{\mathcal{D}'}\Big(\lfloor \mathcal{E}_K \rfloor_s \; ; \; \pi\Big) \in \Omega\left(\frac{q^2}{2^{2n-s}}\right) ,$$

and for simplicity we assume the existence of a constant $c_1$ such that

$$\Delta_{\mathcal{D}'}\Big(\lfloor \mathcal{E}_K \rfloor_s \; ; \; \pi\Big) \geq c_1 \frac{q^2}{2^{2n-s}} \,. \tag{24}$$

On the other hand, regarding $\Delta_{\mathcal{D}'}\Big(\lfloor \rho \rfloor_s \; ; \; \pi\Big)$, Gilboa and Gueron [GG15] proved that for any distinguisher $\mathcal{D}'$,

$$\Delta_{\mathcal{D}'}\Big(\lfloor \rho \rfloor_s \; ; \; \pi\Big) \in O\left(\frac{q^2}{2^{2n+s}}\right) ,$$

i.e., there is a constant $c_2$ such that

$$\Delta_{\mathcal{D}'}\Big(\lfloor \rho \rfloor_s \; ; \; \pi\Big) \leq c_2 \frac{q^2}{2^{2n+s}} \,. \tag{25}$$

Plugging (24) and (25) into (23) yields

$$\mathbf{Adv}^{\mathrm{vsprp}}_{\mathcal{E}}(\mathcal{D}) \geq c_1 \frac{q^2}{2^{2n-s}} - c_2 \frac{q^2}{2^{2n+s}},$$

for some distinguisher $\mathcal{D}$ constructed out of $\mathcal{D}'$. For increasing $s$, the first term becomes larger whereas the second term becomes negligible. For $s = n-1$, the bound is of the form $\Omega\left(q^2/2^n\right)$. □

# 6 Use of LDT in Modes of Operation

We will demonstrate how to use LDT in combination with an online authenticated encryption scheme for integral length messages in order to obtain a scheme for arbitrary length messages greater than $n$ bits. Before doing so, we briefly discuss the security model of online authenticated encryption schemes in Section 6.1. The extension is given in Section 6.2.

## 6.1 Online Authenticated Encryption

Authenticated encryption provides both data confidentiality and data authentication, and consists of an encryption function Enc and a decryption function Dec. Enc takes as input a key $K \in \{0,1\}^k$, a nonce $N \in \{0,1\}^n$, associated data $A \in \{0,1\}^*$, and a message $M \in \{0,1\}^*$, and outputs a ciphertext $C \in \{0,1\}^*$, and tag $T \in \{0,1\}^n$: $(C,T) \leftarrow \mathrm{Enc}(K,N,A,M)$. Dec takes as input a key $K \in \{0,1\}^k$, a nonce $N \in \{0,1\}^n$, and associated data $A \in \{0,1\}^*$, a ciphertext $C \in \{0,1\}^*$, and tag $T \in \{0,1\}^n$, and outputs a message $M \in \{0,1\}^*$ if the tag $T$ is correct and $\perp$ otherwise: $M/\perp \leftarrow \mathrm{Dec}(K,N,A,C,T)$. In this work we focus on online authenticated encryption schemes, and we first give an explicit definition of an ideal online function in terms of arbitrary input length URFs: a function $f$ with arbitrary input size and range $\{0,1\}^n$ is a uniform random function (URF) if it outputs a random value from $\{0,1\}^n$ for every new input.

**Definition 2** (Ideal Online Function)**.** Let $f_i : \{0,1\}^n \times \{0,1\}^* \times \{0,1\}^{ni} \to \{0,1\}^n$, for $i = 1,\ldots,l-1$, $f_l : \{0,1\}^n \times \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^n$, and $f^* : \{0,1\}^n \times \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^n$ be URFs. We define $\$ : \{0,1\}^n \times \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^* \times \{0,1\}^n$ as

$$\$(N,A,M) = f_1(N,A,M_1) f_2(N,A,M_1M_2) \cdots f_{l-1}(N,A,M_1 \cdots M_{l-1}) \| \\ \lfloor f_l(N,A,M)) \rfloor_{|M_l^*|} f^*(N,A,M)\,,$$

where $M_1M_2 \cdots M_{l-1}M_l^* \leftarrow M$.

Let $\mathcal{E} = (\mathrm{Enc}, \mathrm{Dec})$ be an online authenticated encryption scheme, and let $\mathcal{P}$ be an idealized underlying primitive of $\mathcal{E}$, if $\mathcal{P}$ exists. Let $K$ be a randomly drawn key. Let $\$$ be a function as defined in Definition 2. Let $\perp$ be a function that always outputs $\perp$. We define the confidentiality insecurity of $\mathcal{E}$ based on $\mathcal{P}$ as

$$\mathbf{Adv}^{\mathrm{cpa}}_{\mathcal{E}}(\mathcal{D}) = \left| \Pr\left[K \xleftarrow{\$} \{0,1\}^k : \mathcal{D}^{\mathrm{Enc}_K} = 1\right] - \Pr\left[\mathcal{D}^{\$} = 1\right] \right|,$$

where $\$$ is an ideal online function as in Definition 2, and the integrity security of $\mathcal{E}$ based on $\mathcal{P}$ as

$$\mathbf{Adv}^{\mathrm{int}}_{\mathcal{E}}(\mathcal{D}) = \Pr\left[K \xleftarrow{\$} \{0,1\}^k : \mathcal{D}^{\mathrm{Enc}_K, \mathrm{Dec}_K} \text{ forges}\right],$$

where "forges" means that the distinguisher returns a tuple $(N,A,C,T)$ such that $\mathrm{Dec}_K(N,A,C,T)$ returns a valid message $M$ and $(N,A,C,T)$ has not been output by $\mathrm{Enc}_K$.

## 6.2 Fractional Data Coverage

Let $k', k, n \in \mathbb{N}$. Let $\mathcal{E} : \{0,1\}^{k'} \times \{0,1\}^n \times \{0,1\}^* \times (\{0,1\}^n)^+ \to (\{0,1\}^n)^+ \times \{0,1\}^n$ be an authenticated encryption scheme for integral length messages as defined in the previous section. Let LDT: $\{0,1\}^{2k} \times \{0,1\}^{[n\ldots 2n-1]} \to \{0,1\}^{[n\ldots 2n-1]}$ be the length doubler defined in Section 3. In Algorithm 2 (and in Figure 2) we describe an authenticated encryption scheme $\mathcal{F}[\mathcal{E}, \mathrm{LDT}]$ for arbitrary length messages greater than $n$ bits. It has key space $\{0,1\}^{k'} \times \{0,1\}^{2k}$ and internally uses $\mathcal{E}$ and LDT.

One can prove that if $\mathcal{E}$ is a secure online cipher over integral data, and LDT is a secure length doubler, then $\mathcal{F}$ is a secure online cipher over data of arbitrary size greater than $n$ bits.

**Theorem 3.** *Let $k', k, n \in \mathbb{N}$. Let $\mathcal{E} : \{0,1\}^{k'} \times \{0,1\}^n \times \{0,1\}^* \times (\{0,1\}^n)^+ \to (\{0,1\}^n)^+ \times \{0,1\}^n$ be an authenticated encryption scheme for integral length messages as defined in previous section. Let* LDT*: $\{0,1\}^{2k} \times \{0,1\}^{[n\ldots 2n-1]} \to \{0,1\}^{[n\ldots 2n-1]}$ be the length doubler*

**Algorithm 2** $\mathcal{F}[\mathcal{E}, \mathrm{LDT}]$ authenticated encryption

**Input:** $K_{\mathcal{E}} \in \{0,1\}^{k'}$,$K_{\mathrm{LDT}} \in \{0,1\}^{2k}$, $N \in \{0,1\}^n$, $A \in \{0,1\}^*$, $M \in \{0,1\}^*$
**Output:** $C \in \{0,1\}^{|M|+n}$
1: $M_1 M_2 \cdots M_{l-1} M_l^* \leftarrow M$ with $|M_i| = n$ $(i = 1, \ldots, l-1)$ and $|M_l^*| \leq n$
2: $C_1 \cdots C_{l-1}, T' \leftarrow \mathcal{E}_{K_{\mathcal{E}}}(N, A, M_1 \cdots M_{l-1})$ with $|C_i| = |T'| = n$ $(i = 1, \ldots, l-1)$
3: $C_l T \leftarrow \mathrm{LDT}_{K_{\mathrm{LDT}}}(T' M_l^*)$ with $|C_l| = |M_l^*|$ and $|T| = n$
4: $C \leftarrow C_1 \cdots C_l$
5: **return** $(C, T)$

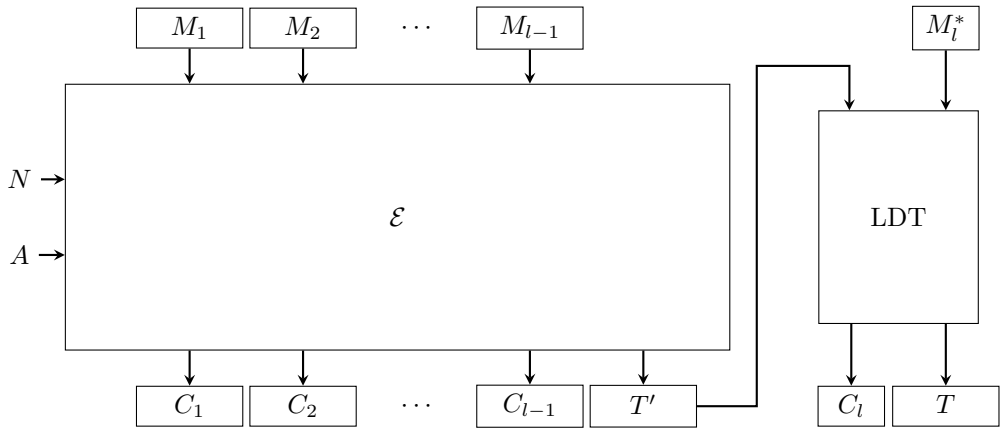


Figure 2: Authenticated encryption $\mathcal{F}$, with $\mathcal{E}$ an authenticated encryption scheme for integral data and LDT our length doubler

*defined in Section 3. For any distinguisher $\mathcal{D}$ making at most $q$ queries, each of length at most $\ell$ and of total size $\sigma$, there exist distinguishers $\mathcal{D}_1'$ with the same query complexity and $\mathcal{D}_2'$ making at most $q$ queries such that*

$$\mathbf{Adv}_{\mathcal{F}}^{\mathrm{cpa/int}}(\mathcal{D}) \leq \mathbf{Adv}_{\mathcal{E}}^{\mathrm{cpa/int}}(\mathcal{D}_1') + \mathbf{Adv}_{\mathrm{LDT}}^{\mathrm{vsprp}}(\mathcal{D}_2') + 2\binom{q}{2}/2^n\,. \tag{26}$$

*Proof.* We treat confidentiality and integrity separately.

**Confidentiality.** Consider any distinguisher $\mathcal{D}$ making at most $q$ queries, each of length at most $\ell$ and of total size $\sigma$. It has access to either $\mathcal{F}_{K_{\mathcal{E}},K_{\mathrm{LDT}}}$ for $K_{\mathcal{E}} \xleftarrow{\$} \{0,1\}^{k'}$ and $K_{\mathrm{LDT}} = (K_1, K_2) \xleftarrow{\$} \{0,1\}^{2k}$, or an ideal online function \$ as defined in Definition 2. As $\mathcal{F}_{K_{\mathcal{E}},K_{\mathrm{LDT}}}$ evaluates its underlying authenticated encryption scheme $\mathcal{E}$ for a randomly generated key $K_{\mathcal{E}}$ and its underlying length doubler LDT for random key $K_{\mathrm{LDT}}$, we use an extended notation for $\mathcal{F}_{K_{\mathcal{E}},K_{\mathrm{LDT}}}$:

$$\mathcal{F}_{K_{\mathcal{E}},K_{\mathrm{LDT}}} = \mathcal{F}[\mathcal{E}_{K_{\mathcal{E}}}, \mathrm{LDT}_{K_{\mathrm{LDT}}}]\,.$$

Let \$′ be another ideal online function and $\rho \xleftarrow{\$} \mathrm{VPerm}([n \ldots 2n-1])$. We have

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{F}}^{\mathrm{cpa}}(\mathcal{D}) &= \Delta_{\mathcal{D}}\Big(\mathcal{F}[\mathcal{E}_{K_{\mathcal{E}}}, \mathrm{LDT}_{K_{\mathrm{LDT}}}]\ ;\ \$\Big) \\
&\leq \Delta_{\mathcal{D}}\Big(\mathcal{F}[\$', \rho]\ ;\ \$\Big) + \Delta_{\mathcal{D}_1'}\Big(\mathcal{E}_{K_{\mathcal{E}}}\ ;\ \$'\Big) + \Delta_{\mathcal{D}_2'}\Big(\mathrm{LDT}_{K_{\mathrm{LDT}}}\ ;\ \rho\Big) \\
&= \Delta_{\mathcal{D}}\Big(\mathcal{F}[\$', \rho]\ ;\ \$\Big) + \mathbf{Adv}_{\mathcal{E}}^{\mathrm{cpa}}(\mathcal{D}_1') + \mathbf{Adv}_{\mathrm{LDT}}^{\mathrm{vsprp}}(\mathcal{D}_2')\,,
\end{aligned} \tag{27}$$

for some distinguishers $\mathcal{D}'_1$ with the same query complexity as $\mathcal{D}$ and $\mathcal{D}'_2$ making at most $q$ queries.

We focus on the remaining distance in (27). As both oracles are completely randomized, it remains to evaluate the statistical distance. Note that both oracles behave identically on the first ciphertext blocks $C_1 \cdots C_{l-1}$. The function $\mathcal{F}[\$', \rho]$ then outputs $C_l T = \rho(T' M_l^*)$ where $T' = f^*_{\$'}(N, A, M)$, whereas the ideal online function outputs

$$C_l = \lfloor f_{\$,l}(N, A, M) \rfloor_{|M_l^*|} ,$$
$$T = f^*_{\$}(N, A, M) .$$

As $\rho$ is a random length preserving invertible function (see Section 2.2) whose input is partially randomized, both responses are distributed identically up to collisions on the input $T'$ or on the output $C_l T$, and we have $\Delta_{\mathcal{D}}\Big(\mathcal{F}[\$', \rho] \; ; \; \$\Big) \leq 2\binom{q}{2}/2^n$, as in the worst case $|M_l^*| = 0$ and $\rho$ outputs $n$-bit strings.

**Integrity.** As before, we consider any distinguisher $\mathcal{D}$ with the same complexity, but now it has access to

$$\mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \mathrm{LDT}_{K_{\mathrm{LDT}}}] \text{ and } \mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \mathrm{LDT}_{K_{\mathrm{LDT}}}]^{-1} ,$$

for $K_\mathcal{E} \xleftarrow{\$} \{0,1\}^{k'}$ and $K_{\mathrm{LDT}} = (K_1, K_2) \xleftarrow{\$} \{0,1\}^{2k}$. Let $\rho \xleftarrow{\$} \mathrm{VPerm}([n \ldots 2n-1])$. We have

$$\begin{aligned}
\mathbf{Adv}^{\mathrm{int}}_{\mathcal{F}}(\mathcal{D}) &= \Pr\left[\mathcal{D}^{\mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \mathrm{LDT}_{K_{\mathrm{LDT}}}], \mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \mathrm{LDT}_{K_{\mathrm{LDT}}}]^{-1}} \text{ forges}\right] \\
&\leq \Pr\left[\mathcal{D}^{\mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \rho], \mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \rho]^{-1}} \text{ forges}\right] + \Delta_{\mathcal{D}'_2}\Big(\mathrm{LDT}_{K_{\mathrm{LDT}}}, \mathrm{LDT}^{-1}_{K_{\mathrm{LDT}}} \; ; \; \rho, \rho^{-1}\Big) \\
&= \Pr\left[\mathcal{D}^{\mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \rho], \mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \rho]^{-1}} \text{ forges}\right] + \mathbf{Adv}^{\mathrm{vsprp}}_{\mathrm{LDT}}(\mathcal{D}'_2) , \qquad (28)
\end{aligned}$$

for some distinguisher $\mathcal{D}'_2$ making at most $q$ queries.

We focus on the remaining distance in (28), and we will demonstrate that any forgery attempt either reduces to a forgery attempt on $\mathcal{E}$, or corresponds to an accidental collision in $\rho^{-1}$. In more detail, consider any new forgery attempt $(N, A, C_1 \cdots C_l, T)$. Let $T' M_l^* = \rho^{-1}(C_l T)$. We make the following case distinction:

- There exists an earlier query with identical $(C_l, T)$. As, w.l.o.g., the distinguisher never repeats any query, the tuple $(N, A, C_1 \cdots C_{l-1}, T')$ is new, and if the forgery attempt against $\mathcal{F}$ is successful, then $(N, A, C_1 \cdots C_{l-1}, T')$ is a forgery on $\mathcal{E}$;
- There does not exist any earlier query with identical $(C_l, T)$. We make a further case distinction:
  - There does not exist any earlier query with identical $(N, A, C_1 \cdots C_{l-1}, T')$. This means that we have likewise found a forgery on $\mathcal{E}$;
  - There exists an earlier query with identical $(N, A, C_1 \cdots C_{l-1}, T')$. This means that the new and older query have two different input values of $\rho^{-1}$ but the same output value $T'$, which happens with probability at most $\binom{q}{2}/2^n$.

We have thus obtained that

$$\Pr\left[\mathcal{D}^{\mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \rho], \mathcal{F}[\mathcal{E}_{K_\mathcal{E}}, \rho]^{-1}} \text{ forges}\right] \leq \mathbf{Adv}^{\mathrm{int}}_{\mathcal{E}}(\mathcal{D}'_1) + \binom{q}{2}/2^n ,$$

for some distinguisher $\mathcal{D}'_1$ with the same query complexity as $\mathcal{D}$. □

Note that $\mathcal{F}$ reminds of ciphertext stealing [Dae95], but it differs in the fact that ciphertext stealing only works for plaintexts which are at least two blocks long. $\mathcal{F}$ works on arbitrary sized messages greater than $n$ bits.

## Acknowledgments

This work was supported in part by the Research Council KU Leuven: GOA TENSE (GOA/11/007). Atul Luykx is supported by a Fellowship from the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen) and in part by NSF grants CNS-1314885 and CNS-1717542. Bart Mennink is supported by a postdoctoral fellowship from the Netherlands Organisation for Scientific Research (NWO) under Veni grant 016.Veni.173.017. The authors would like to thank the anonymous reviewers of ToSC for their comments and suggestions.

## References

[ABD+16] Elena Andreeva, Andrey Bogdanov, Nilanjan Datta, Atul Luykx, Bart Mennink, Mridul Nandi, Elmar Tischhauser, and Kan Yasuda. COLM v1, 2016. Submission to CAESAR competition.

[ABL+13] Elena Andreeva, Andrey Bogdanov, Atul Luykx, Bart Mennink, Elmar Tischhauser, and Kan Yasuda. Parallelizable and authenticated online ciphers. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I*, volume 8269 of *Lecture Notes in Computer Science*, pages 424–443. Springer, 2013.

[ABL+15] Elena Andreeva, Andrey Bogdanov, Atul Luykx, Bart Mennink, Elmar Tischhauser, and Kan Yasuda. AES-COPA v.1, 2015. Submission to CAESAR competition.

[BJK+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY family of block ciphers and its low-latency variant MANTIS. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815 of *Lecture Notes in Computer Science*, pages 123–153. Springer, 2016.

[CS06] Debrup Chakraborty and Palash Sarkar. HCH: A new tweakable enciphering scheme using the hash-encrypt-hash approach. In Rana Barua and Tanja Lange, editors, *Progress in Cryptology - INDOCRYPT 2006, 7th International Conference on Cryptology in India, Kolkata, India, December 11-13, 2006, Proceedings*, volume 4329 of *Lecture Notes in Computer Science*, pages 287–302. Springer, 2006.

[CS14] Shan Chen and John P. Steinberger. Tight security bounds for key-alternating ciphers. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, volume 8441 of *Lecture Notes in Computer Science*, pages 327–350. Springer, 2014.

[CV04] Anne Canteaut and Kapalee Viswanathan, editors. *Progress in Cryptology - INDOCRYPT 2004, 5th International Conference on Cryptology in India, Chennai, India, December 20-22, 2004, Proceedings*, volume 3348 of *Lecture Notes in Computer Science*. Springer, 2004.

[CYK04a] Debra L. Cook, Moti Yung, and Angelos D. Keromytis. Elastic AES. Cryptology ePrint Archive, Report 2004/141, 2004.

[CYK04b] Debra L. Cook, Moti Yung, and Angelos D. Keromytis. Elastic block ciphers. Cryptology ePrint Archive, Report 2004/128, 2004.

[Dae95] Joan Daemen. *Hash Function and Cipher Design: Strategies Based on Linear and Differential Cryptanalysis*. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 1995.

[GG15] Shoni Gilboa and Shay Gueron. Distinguishing a truncated random permutation from a random function. Cryptology ePrint Archive, Report 2015/773, 2015.

[Hal04] Shai Halevi. EME*: Extending EME to handle arbitrary-length messages with associated data. In Canteaut and Viswanathan [CV04], pages 315–327.

[Hal07] Shai Halevi. Invertible universal hashing and the TET encryption mode. In Alfred Menezes, editor, *Advances in Cryptology - CRYPTO 2007, 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007, Proceedings*, volume 4622 of *Lecture Notes in Computer Science*, pages 412–429. Springer, 2007.

[HWKS98] Chris Hall, David Wagner, John Kelsey, and Bruce Schneier. Building prfs from prps. In Hugo Krawczyk, editor, *Advances in Cryptology - CRYPTO '98, 18th Annual International Cryptology Conference, Santa Barbara, California, USA, August 23-27, 1998, Proceedings*, volume 1462 of *Lecture Notes in Computer Science*, pages 370–389. Springer, 1998.

[KR11] Ted Krovetz and Phillip Rogaway. The software performance of authenticated-encryption modes. In Antoine Joux, editor, *Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers*, volume 6733 of *Lecture Notes in Computer Science*, pages 306–327. Springer, 2011.

[LRW02] Moses Liskov, Ronald L. Rivest, and David Wagner. Tweakable block ciphers. In Moti Yung, editor, *Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings*, volume 2442 of *Lecture Notes in Computer Science*, pages 31–46. Springer, 2002.

[MF07] David A. McGrew and Scott R. Fluhrer. The security of the extended codebook (XCB) mode of operation. In Carlisle M. Adams, Ali Miri, and Michael J. Wiener, editors, *Selected Areas in Cryptography, 14th International Workshop, SAC 2007, Ottawa, Canada, August 16-17, 2007, Revised Selected Papers*, volume 4876 of *Lecture Notes in Computer Science*, pages 311–327. Springer, 2007.

[MI11] Kazuhiko Minematsu and Tetsu Iwata. Building blockcipher from tweakable blockcipher: Extending FSE 2009 proposal. In Liqun Chen, editor, *Cryptography and Coding - 13th IMA International Conference, IMACC 2011, Oxford, UK, December 12-15, 2011. Proceedings*, volume 7089 of *Lecture Notes in Computer Science*, pages 391–412. Springer, 2011.

[MV04] David A. McGrew and John Viega. The security and performance of the Galois/Counter Mode (GCM) of operation. In Canteaut and Viswanathan [CV04], pages 343–355.

[Nan09] Mridul Nandi. A generic method to extend message space of a strong pseudorandom permutation. *Computación y Sistemas*, 12(3), 2009.

[Nan14] Mridul Nandi. XLS is not a strong pseudorandom permutation. In Palash Sarkar and Tetsu Iwata, editors, *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*, volume 8873 of *Lecture Notes in Computer Science*, pages 478–490. Springer, 2014.

[Nan15] Mridul Nandi. Revisiting security claims of XLS and COPA. Cryptology ePrint Archive, Report 2015/444, 2015.

[Pat91] Jacques Patarin. *Étude des Générateurs de Permutations Basés sur le Schéma du D.E.S.* PhD thesis, Université Paris 6, Paris, France, 1991.

[Pat08] Jacques Patarin. The "coefficients H" technique. In Roberto Maria Avanzi, Liam Keliher, and Francesco Sica, editors, *Selected Areas in Cryptography, 15th International Workshop, SAC 2008, Sackville, New Brunswick, Canada, August 14-15, Revised Selected Papers*, volume 5381 of *Lecture Notes in Computer Science*, pages 328–345. Springer, 2008.

[RR07] Thomas Ristenpart and Phillip Rogaway. How to enrich the message space of a cipher. In Alex Biryukov, editor, *Fast Software Encryption, 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers*, volume 4593 of *Lecture Notes in Computer Science*, pages 101–118. Springer, 2007.

[RZ11] Phillip Rogaway and Haibin Zhang. Online ciphers from tweakable blockciphers. In Aggelos Kiayias, editor, *Topics in Cryptology - CT-RSA 2011 - The Cryptographers' Track at the RSA Conference 2011, San Francisco, CA, USA, February 14-18, 2011. Proceedings*, volume 6558 of *Lecture Notes in Computer Science*, pages 237–249. Springer, 2011.

[Sar07] Palash Sarkar. Improving upon the TET mode of operation. In Kil-Hyun Nam and Gwangsoo Rhee, editors, *Information Security and Cryptology - ICISC 2007, 10th International Conference, Seoul, Korea, November 29-30, 2007, Proceedings*, volume 4817 of *Lecture Notes in Computer Science*, pages 180–192. Springer, 2007.

[SV93] Claus-Peter Schnorr and Serge Vaudenay. Parallel fft-hashing. In Ross J. Anderson, editor, *Fast Software Encryption, Cambridge Security Workshop, Cambridge, UK, December 9-11, 1993, Proceedings*, volume 809 of *Lecture Notes in Computer Science*, pages 149–156. Springer, 1993.

[WFW05] Peng Wang, Dengguo Feng, and Wenling Wu. HCTR: A variable-input-length enciphering mode. In Dengguo Feng, Dongdai Lin, and Moti Yung, editors, *Information Security and Cryptology, First SKLOIS Conference, CISC 2005, Beijing, China, December 15-17, 2005, Proceedings*, volume 3822 of *Lecture Notes in Computer Science*, pages 175–188. Springer, 2005.

[Zha12] Haibin Zhang. Length-doubling ciphers and tweakable ciphers. In Feng Bao, Pierangela Samarati, and Jianying Zhou, editors, *Applied Cryptography and Network Security - 10th International Conference, ACNS 2012, Singapore, June 26-29, 2012. Proceedings*, volume 7341 of *Lecture Notes in Computer Science*, pages 100–116. Springer, 2012.

# A Example Mixing Functions

Ristenpart and Rogaway [RR07] defined $\epsilon$-good mixing functions as follows.

**Definition 3.** Let $m, n \in \mathbb{N}$ such that $m \leq n$, and denote $S = \{0,1\}^{[m..n]}$. Let $mix : S^2 \to S^2$ be a length-preserving permutation, define by $mix_L$ the left half of its evaluation and by $mix_R$ its right half. Let $\epsilon : \{m..n\} \to [0..1]$. The mixing function is called $\epsilon$-*good* if for all $s \in [m..n]$ we have:

- $mix_L(A, \cdot)$ is a permutation for all $A \in \{0,1\}^s$,
- $mix_R(\cdot, B)$ is a permutation for all $B \in \{0,1\}^s$,
- $\Pr[R \xleftarrow{\$} \{0,1\}^s : C = mix_L(R, B)] \leq \epsilon(s)$ for all $B, C \in \{0,1\}^s$, and
- $\Pr[R \xleftarrow{\$} \{0,1\}^s : C = mix_R(A, R)] \leq \epsilon(s)$ for all $A, C \in \{0,1\}^s$.

Note that the definition differs from the pure mixing functions of Definition 1 in the presence of the third and fourth condition.

The best one can hope for is a $2^{-s}$-good mixing function. Ristenpart and Rogaway proposed two efficient mixing functions, *mix1* and *mix2* defined below:

$$mix1(A, B) = \big(3A + 2B, 2A + 3B\big) = \big(A + 2(A + B), B + 2(A + B)\big)\,,$$
$$mix2(A, B) = \big(A \oplus rol(A \oplus B), (B \oplus rol(A \oplus B)\big)\,,$$

where $rol(X)$ is the left circular bit-rotation, that means for any string $X$ of length $s$, $rol(X) = X[2]X[3] \cdots X[s]X[1]$. Ristenpart and Rogaway proved that *mix1* is $2^{-s}$-good and *mix2* is $2^{1-s}$-good. Both are also pure in the sense of Definition 1, yet not as efficient as, e.g., $mix(A, B) = (B, A)$.