Practical Attacks on the Full-round FRIET

Senpeng Wang[1,2(✉)], Dengguo Feng[1], Bin Hu[2], Jie Guan[2] and Tairong Shi[2]

[1] State Key Laboratory of Cryptology, Beijing, China, wsp2110@126.com
[2] PLA SSF Information Engineering University, Zhengzhou, China

Abstract. FRIET is a duplex-based authenticated encryption scheme proposed at EUROCRYPT 2020. It follows a novel design approach for built-in countermeasures against fault attacks. By a judicious choice of components, the designers propose the permutation FRIET-PC that can be used to build an authenticated encryption cipher denoted as FRIET-AE. And FRIET-AE provides a 128-bit security claim for integrity and confidentiality. In this paper, we research the propagation of differences and liner masks through the round function of FRIET-PC. For full-round FRIET-PC, we can construct a differential distinguisher whose probability is 1 and a linear distinguisher whose absolute value of correlation is 1. For the authenticated encryption cipher PRIET-AE, we use the differential distinguisher with probability 1 to construct a set consisting of valid tags and ciphertexts which are not created by legal users. This breaks FRIET-AE's security claim for integrity and confidentiality. As far as we know, this is the first practical attack that threatens the security of FRIET-AE.

Keywords: FRIET · Authenticated Encryption · Differential Attack · Linear Attack · Fault Injection

1 Introduction

Permutation-based cryptographic components are widely used in the design of ciphers. Firstly, permutations can be used in Sponge [BDPA08] mode to obtain hash functions. For example, KECCAK [BDPA11b] designed based on the permutation Keccak-f won the U.S. National Institute of Standards and Technology (NIST) Secure Hash Algorithm-3 (SHA3) competition in 2012. Secondly, permutations can be used in EVEN-Mansour [EM91] mode to get block ciphers, such as Simpira-EM [GM16]. Thirdly, permutations can be used in Duplex [BDPA11a] construction to design authenticated encryption (AE) ciphers. For example, ASCON [DEMSb] designed in this strategy is one of the final candidates in Competition for Authenticated Encryption: Security, Applicability and Robustness (CAESAR). Under this background, many cryptographic permutations are proposed, such as Alzette [BBdS+20], Gimli [BKL+17], Xoodoo [DHAK18], Frit [SBD+18], FRIET [SBD+20] and *et al.*

For their good security and implementation advantages, permutation-based cryptographic components are also widely used in the design of lightweight ciphers. In March 2021, NIST Lightweight Cryptography Project (LWC) announced the ten finalists. It should be noted that 6 of 10 are permutation based. They are ASCON [DEMSa], Elephant [BCDM], ISAP [DEM+], Photo-Beetle [BCD+], SPARKLE [BBdS+] and Xoodyak [DHP+]. Because lightweight ciphers are often used in constrained environments (constraints on energy, area and memory size). These lightweight ciphers may be exposed to side channel attacks. In order to mitigate such attacks, at EUROCRYPT 2020, Simon *et al.* proposed a novel design method for ciphers with efficient fault-detecting implementations and concrete authenticated encryption scheme called FRIET [SBD+20]. And they design new cryptographic permutations called FRIET-PC and FRIET-P for the implementation of FRIET.

An earlier version of FRIET-PC is called Frit [SBD+18] proposed by the same authors. It wasn't long before Dobraunig *et al.* [DEMS19] studied the algebraic properties of Frit and gave a key recovery attack against the full-round Friet-EM (the block cipher constructed by Frit in Even-Mansour mode). Then, Qin *et al.* [QDJZ19] gave some key-recovery attacks on the round-reduced Frit used in duplex authenticated encryption mode. By taking these attacks into account, a new permutation called FRIET-PC is designed. The designers evaluate the security of FRIET-PC against algebraic attack, slide attack, invariant subspace attack, non-linear invariant attack, differential attack, linear attack and *et al.* For example, by researching the properties of trail with low-weight input differences and linear masks, they obtain a 6-round differential trail with probability 2^{-59} and an 8-round linear trail with correlation 2^{-80}. At EUROCRYPT 2021, Liu *et al.* [LSL21] constructed a 12-round rotational differential-linear distinguisher with correlation $2^{-117.81}$. Then, Ito *et al.* [ISS+21] evaluated the security of FRIET-PC against bit-wise cryptanalysis including rotational attack, bit-wise differential attack and integral attack. It should be noted that the above attacks do not threaten the security of FRIET-PC.

1.1 Our Contributions

FRIET-PC adopts the AND-Rotation-XOR construction. And the only nonlinear operation in FRIET-PC is bitwise AND. By fixing the differential probability and linear correlation of AND operation, we research the propagation of differences and linear masks through the round function of FRIET-PC. For any-round FRIET-PC, we construct a differential distinguisher whose probability is 1 and a linear distinguisher whose absolute value of correlation is 1. The comparison with the previous results is shown in Table 1.

Table 1: The comparison of the distinguishers for FRIET-PC

*Type	Round	†Probability/Correlation/Data	Reference
LC	7	2^{-29}	[SBD+20]
	8	2^{-40}	
	$^*R$	1 or -1	Sect. 3.1
R-DL	8	$2^{-17.81}$	[LSL21]
	9	$2^{-29.81}$	
	13	$2^{-117.81}$	
IC	13	2^{-31}	[ISS+21]
	15	2^{-63}	
	17	2^{-127}	
	30	2^{-383}	
DC	6	2^{-59}	[SBD+20]
	9	$2^{-20.04}$	[ISS+21]
	$^*R$	1	Sect. 3.2

* R-DL denotes rotational differential-linear distinguisher. LC denotes linear distinguisher. DC denotes differential distinguisher. IC denotes integral distinguisher.
† The DC is showed with probability. LC/DL/R-DL are showed with correlation. IC is showed with data.
* R means that the differential or linear distinguisher is valid for any-round FRIET-PC.

What's more, when FRIET-PC is used in FRIET, we get an authenticated encryption cipher denoted as FRIET-AE. And FRIET-AE provides a 128-bit security claim for integrity and confidentiality. Using the above differential distinguisher with probability 1, we can practically construct a set consisting of valid tags and ciphertexts which are not created by legal users. This breaks the claims for integrity and confidentiality of FRIET-AE. Therefore, the design of permutation FRIET-PC has defects.

1.2 Outline

This paper is organized as follows: Sect. 2 introduces the differential and linear cryptanalysis and briefly describes the specification of FRIET permutation. In Sect. 3, we propose the differential and linear distinguishers for the full-round FRIET-PC. In Sect. 4, we give the practical attacks on the full-round FRIET-AE. Sect. 5 concludes the paper.

2 Preliminaries

2.1 Notations

Notations used in this paper are defined in Table 2.

Table 2: Notations used in this paper

$\mathbb{F}_2$	The finite field $\{0, 1\}$
$x \in \mathbb{F}_2^n$	An n-bit vector
$x \oplus y$	Bitwise XOR of x and y
$\overline{x}$	Bitwise NOT of x
$x \vee y$	Bitwise OR of x and y
$x \wedge y$	Bitwise AND of x and y
$x \cdot y$	The inner product of x and y
$x \|\| y$	The concatenation of x and y
$x \ll r$	Shift x to the left by r bits
$x \lll r$	Rotation of x to the left by r bits
$x \ggg r$	Rotation of x to the right by r bits
$wt\left(x\right)$	The hamming weight of x
$\delta_i\left(c\right)$	The i-th bit of integer c under binary
$\lceil c \rceil$	The nearest integer greater than or equal to c
$\lfloor c \rfloor$	The nearest integer smaller than or equal to c
$\mathbf{0}_n$	An n-bit vector with all entries equal 0
$\mathbf{1}_n$	An n-bit vector with all entries equal 1

2.2 Differential and Linear Cryptanalysis

Differential cryptanalysis [BS90] and linear cryptanalysis [Mat93] are two powerful methods which have been widely used in the security analysis of many symmetric ciphers. The core idea of these methods is to identify the differences (linear masks) with high probabilities (correlations).

Definition 1. (Differential probability [BS90]). *For a vectorial boolean function $f : \mathbb{F}_2^n \to \mathbb{F}_2^m$, let $\alpha \in \mathbb{F}_2^n$ and $\beta \in \mathbb{F}_2^m$ be the input and output differences of f. Then, the differential probability of f is defined as:*

$$Pr[\alpha \to \beta] = 2^{-n}\#\{x \in \mathbb{F}_2^n : f\left(x\right) \oplus f\left(x \oplus \alpha\right) = \beta\},$$

where $\#\{x \in \mathbb{F}_2^n : f\left(x\right) \oplus f\left(x \oplus \alpha\right) = \beta\}$ is the number of x satisfying $f\left(x\right) \oplus f\left(x \oplus \alpha\right) = \beta$.

Definition 2. (Linear correlation [Mat93]). *For a vectorial boolean function $f : \mathbb{F}_2^n \to \mathbb{F}_2^m$, let $\alpha \in \mathbb{F}_2^n$ and $\beta \in \mathbb{F}_2^m$ be the input and output linear masks of f. Then, the correlation of the linear approximation for f is defined as*

$$Cor\left(\alpha, \beta\right) = 2^{-n}\#\{x \in \mathbb{F}_2^n : \alpha \cdot x \oplus \beta \cdot f\left(x\right) = 0\} - 1,$$

where $\#\{x \in \mathbb{F}_2^n : \alpha \cdot x \oplus \beta \cdot f\left(x\right) = 0\}$ is the number of x satisfying $\alpha \cdot x \oplus \beta \cdot f\left(x\right) = 0$.

Based on the above definitions, we introduce the following properties characterizing the behaviours of the differences and linear masks through basic operations.

Differential Property 1 (Branching) [BS90]. *Let $(y, z) = f(x)$ be a branching function, where $x \in \mathbb{F}_2^n$ is the input variable and the output variables y and z are calculated as $y = z = x$. Then,*

$$Pr[\alpha \to \beta||\gamma] = \begin{cases} 1, & \text{if } \alpha = \beta = \gamma, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha \in \mathbb{F}_2^n$ and $\beta||\gamma \in \mathbb{F}_2^{2n}$ are the differences of x and $y||z$, respectively.

Differential Property 2 (XOR) [BS90]. *Let $z = f(x, y)$ be an XOR function, where $x \in \mathbb{F}_2^n$ and $y \in \mathbb{F}_2^n$ are the input variables and the output variable z is calculated as $z = x \oplus y$. Then,*

$$Pr[\alpha||\beta \to \gamma] = \begin{cases} 1, & \text{if } \alpha \oplus \beta = \gamma, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha||\beta \in \mathbb{F}_2^{2n}$ and $\gamma \in \mathbb{F}_2^n$ are the differences of $x||y$ and z, respectively.

Differential Property 3 (XOR-Constant) [BS90]. *Let $z = f(x, c)$ be an XOR-Constant function, where $x \in \mathbb{F}_2^n$ is the input variable, $c \in \mathbb{F}_2^n$ is a constant and the output variable z is calculated as $z = x \oplus c$. Then,*

$$Pr[\alpha \to \beta] = \begin{cases} 1, & \text{if } \alpha = \beta, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha \in \mathbb{F}_2^n$ and $\beta \in \mathbb{F}_2^n$ are the differences of x and z, respectively.

Differential Property 4 (AND) [SBD+20]. *Let $z = f(x, y)$ be an AND function, where $x \in \mathbb{F}_2^n$ and $y \in \mathbb{F}_2^n$ are the input variables, and the output variable z is calculated as $z = x \wedge y$. Then,*

$$Pr[\alpha||\beta \to \gamma] = \begin{cases} 2^{-wt(\alpha \vee \beta)}, & \text{if } \overline{\alpha} \wedge \overline{\beta} \wedge \gamma = \mathbf{0}_n, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha||\beta \in \mathbb{F}_2^{2n}$ and $\gamma \in \mathbb{F}_2^n$ are the differences of $x||y$ and z, respectively.

Linear Property 1 (Branching) [Mat93]. *Let $(y, z) = f(x)$ be a branching function, where $x \in \mathbb{F}_2^n$ is the input variable and the output variables y and z are calculated as $y = z = x$. Then,*

$$Cor(\alpha, \beta||\gamma) = \begin{cases} 1, & \text{if } \alpha = \beta \oplus \gamma, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha \in \mathbb{F}_2^n$ and $\beta||\gamma \in \mathbb{F}_2^{2n}$ are the linear masks of x and $y||z$, respectively.

Linear Property 2 (XOR) [Mat93]. *Let $z = f(x, y)$ be an XOR function, where $x \in \mathbb{F}_2^n$ and $y \in \mathbb{F}_2^n$ are the input variables, and the output variable z is calculated as $z = x \oplus y$. Then,*

$$Cor(\alpha||\beta, \gamma) = \begin{cases} 1, & \text{if } \alpha = \beta = \gamma, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha||\beta \in \mathbb{F}_2^{2n}$ and $\gamma \in \mathbb{F}_2^n$ are the linear masks of $x||y$ and z, respectively.

Linear Property 3 (XOR-Constant) [BS90]. *Let $y = f(x, c)$ be an XOR-Constant function, where $x \in \mathbb{F}_2^n$ is the input variable, $c \in \mathbb{F}_2^n$ is a constant, and the output variable*

y is calculated as $y = x \oplus c$. Then,

$$Cor\,(\alpha, \beta) = \begin{cases} (-1)^{\beta \cdot c}, & \text{if } \alpha = \beta, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha \in \mathbb{F}_2^n$ and $\beta \in \mathbb{F}_2^n$ are the linear masks of x and y, respectively.

Linear Property 4 (AND) [SBD+20]. *Let $z = f(x, y)$ be an AND function, where $x \in \mathbb{F}_2^n$ and $y \in \mathbb{F}_2^n$ are the input variables, and the output variable z is calculated as $z = x \wedge y$. Then,*

$$Cor\,(\alpha||\beta, \gamma) = \begin{cases} 2^{-wt(\gamma)}, & \text{if } \gamma \vee \left(\overline{\alpha} \wedge \overline{\beta}\right) = \mathbf{1}_n \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha||\beta \in \mathbb{F}_2^{2n}$ and $\gamma \in \mathbb{F}_2^n$ are the linear masks of $x||y$ and z, respectively.

In order to apply differential (linear) cryptanalysis, cryptanalysts have to build difference (linear approximate) for each round of a cipher, such that the output difference (linear mask) of a round matches the input difference (linear mask) of the next round. The differential probability (linear correlation) of the full-round cipher is computed by multiplying the differential probabilities (linear correlations) of each round. And we call a difference (linear mask) is valid when its differential probability (linear correlation) is nonzero. If a cipher behaves differently from a random cipher for differential (linear) cryptanalysis, this can be used to build a distinguishing or even a key-recovery attack.

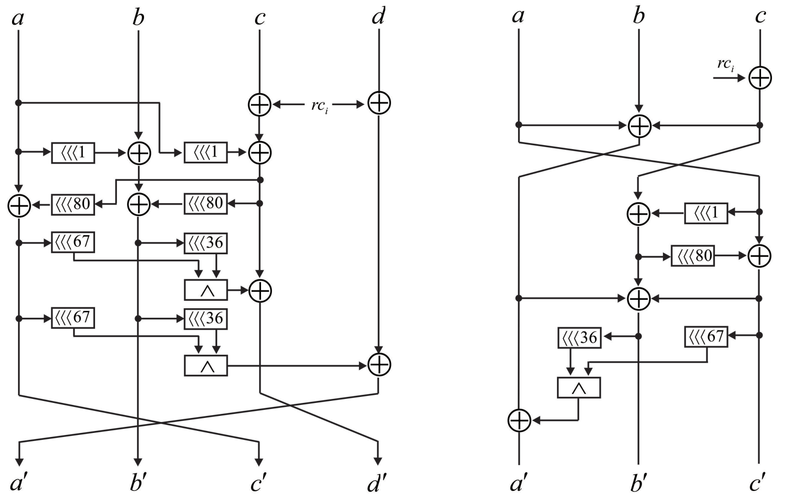
2.3 Description of the Round Function of FRIET

FRIET [SBD+20] is an authenticated encryption scheme with built-in fault detection mechanisms proposed by Simon *et al.* at EUROCRYPT 2020. Its fault detection ability comes from its underlying permutation, which is designed based on the so-called code embedding approach. The core permutation FRIET-P employed in FRIET operates on 4 limbs $(a, b, c, d) \in \mathbb{F}_2^{4\times 128}$. The permutation FRIET-P is an iterative design with its round function $f_{rc_i}(a, b, c, d)$ visualized in the left part of Figure 1, where rc_i is the round constant for the i-th round listed in Table 3.

Table 3: Round constants rc_i in hexadecimal notation

i	0	1	2	3	4	5
rc_i	0x1111	0x11100000	0x1101	0x10100000	0x101	0x10110000
i	6	7	8	9	10	11
rc_i	0x110	0x11000000	0x1001	0x100000	0x100	0x10000000
i	12	13	14	15	16	17
rc_i	0x1	0x110000	0x111	0x11110000	0x1110	0x11010000
i	18	19	20	21	22	23
rc_i	0x1010	0x1010000	0x1011	0x1100000	0x1100	0x10010000

By design, the round function $(a', b', c', d') = f(a, b, c, d)$ has slice-wise code-abiding property. Mathematically, it means that $a \oplus b \oplus c = d$ implies $a' \oplus b' \oplus c' = d'$. Thus, the permutation FREIT-P $= f_{rc_{23}} \circ f_{rc_{22}} \circ \cdots \circ f_{rc_0}$ also has this property. Consequently, faults will be detected if output does not have code-abiding property when the input state has code-abiding property. If ignoring the limb d of FRIET-P, we will obtain a new permutation FRIET-PC visualized in the right part of Figure 1. Since a distinguisher for the permutation FRIET-PC directly translates to a distinguisher for FRIET-P, we focus on the permutation FRIET-PC. And we describe the procedure of FRIET-PC permutation as shown in Algorithm 1.



(a) Round function of FRIET-P

(b) Round function of FRIET-PC

Figure 1: The round function of FRIET [SBD+20]

Algorithm 1. FRIET-PC [SBD+20]

Input: The three limbs $a, b, c \in \mathbb{F}_2^{128}$ and the round constants $rc_i, 0 \leq i \leq 23$
Output: $(a', b', c') \leftarrow$ FRIET-PC (a, b, c)

for $(i = 0; i \leq 23; i++)$ **do**
  $c \leftarrow c \oplus rc_i$
  $(a, b, c) \leftarrow (a \oplus b \oplus c, c, a)$
  $b \leftarrow b \oplus (c \lll 1)$
  $c \leftarrow c \oplus (b \lll 80)$
  $(a, b, c) \leftarrow (a, a \oplus b \oplus c, c)$
  $a \leftarrow a \oplus ((b \lll 36) \wedge (c \lll 67))$
end for
return (a, b, c)

---

3 Differential and Linear Distinguishers for the Full-Round FRIET-PC

FRIET-PC only has four operations: Rotation, XOR, XOR-Constant and AND. Rotation can be seen as a special form of branching operation. Bitwise AND is the only nonlinear operation in FRIET-PC. If we can effectively control the propagations of differences and linear masks through bitwise AND operation, we can obtain differences with high probabilities and linear masks with high correlations.

3.1 A Differential Distinguisher for the Full-Round FRIET-PC

According to **Differential Property 1 (Branching)**, **Differential Property 2 (XOR)** and **Differential Property 3 (XOR-Constant)** in Sect. 2.2, the differential probability of a valid difference for these three operations is 1. By **Differential Property 4 (AND)**, the differential probability of a valid difference for bitwise AND operation is determined by $wt(\alpha \vee \beta)$, where α and β are the input differences of the bitwise AND operation.

If controlling the value of $wt(\alpha \vee \beta)$ effectively, we may obtain differences with high probabilities.

Lemma 1. *The differential probability of* $(\alpha, \beta, \gamma) \to (\alpha', \beta', \gamma')$ *for the* i*-th round function* $(a', b', c') = \textit{FRIET-PC}_i(a, b, c)$ *of FRIET-PC is 1 if and only if*

$$\begin{cases} \alpha' = \alpha \oplus \beta \oplus \gamma, \\ \alpha \oplus (\alpha \lll 1) \oplus \beta = \mathbf{0}_{128}, \\ \alpha \oplus (\alpha \lll 81) \oplus (\gamma \lll 80) = \mathbf{0}_{128}, \\ \beta' = \mathbf{0}_{128}, \\ \gamma' = \mathbf{0}_{128}. \end{cases} \tag{1}$$

Proof. On one hand, if the differential probability of $(\alpha, \beta, \gamma) \to (\alpha', \beta', \gamma')$ is 1. According to **Differential Property 4 (AND)**, both the two input differences of AND operation should be $\mathbf{0}_n$. Because the two input variables vector of AND are $b' \lll 36$ and $c' \lll 67$, we have $\beta' = \mathbf{0}_{128}$, $\gamma' = \mathbf{0}_{128}$ and the output difference of $(b' \lll 36) \wedge (c' \lll 67)$ should also be $\mathbf{0}_{128}$. And due to

$$\begin{cases} a' = a \oplus b \oplus c \oplus rc_i \oplus ((b' \lll 36) \wedge (c' \lll 67)), \\ b' = a \oplus (a \lll 1) \oplus b \oplus c', \\ c' = a \oplus (a \lll 81) \oplus ((c \oplus rc_i) \lll 80), \end{cases}$$

we have

$$\begin{cases} \alpha' = \alpha \oplus \beta \oplus \gamma, \\ \beta' = \alpha \oplus (\alpha \lll 1) \oplus \beta = \mathbf{0}_{128}, \\ \gamma' = \alpha \oplus (\alpha \lll 81) \oplus (\gamma \lll 80) = \mathbf{0}_{128}. \end{cases}$$

The necessity is proved.

On the other hand, if a difference $(\alpha, \beta, \gamma) \to (\alpha', \beta', \gamma')$ satisfies the Eq. (1). Its differential probabilities through all the basic operations (Rotation, XOR, XOR-Constant, AND) in the round function of FRIET-PC is 1. Thus, the differential probability of $(\alpha, \beta, \gamma) \to (\alpha', \beta', \gamma')$ is 1. The sufficiency is proved. □

Next, we will research the differential property of the 2-round function of FRIET-PC.

Lemma 2. *The differential probability of a nonzero difference* $(\alpha, \beta, \gamma) \to (\alpha', \beta', \gamma') \to (\alpha'', \beta'', \gamma'')$ *for the 2-round FRIET-PC is 1 if and only if*

$$\begin{cases} \alpha = \alpha' = \alpha'' = \mathbf{1}_{128}, \\ \beta = \beta' = \beta'' = \mathbf{0}_{128}, \\ \gamma = \gamma' = \gamma'' = \mathbf{0}_{128}. \end{cases} \tag{2}$$

Proof. According to Lemma 1, the differential probability of $(\alpha, \beta, \gamma) \to (\alpha', \beta', \gamma') \to (\alpha'', \beta'', \gamma'')$ is 1 if and only if

$$\alpha' = \alpha \oplus \beta \oplus \gamma, \tag{3}$$
$$\alpha \oplus (\alpha \lll 1) \oplus \beta = \mathbf{0}_{128}, \tag{4}$$
$$\alpha \oplus (\alpha \lll 81) \oplus (\gamma \lll 80) = \mathbf{0}_{128}, \tag{5}$$
$$\beta' = \mathbf{0}_{128}, \tag{6}$$
$$\gamma' = \mathbf{0}_{128}, \tag{7}$$
$$\alpha'' = \alpha' \oplus \beta' \oplus \gamma', \tag{8}$$
$$\alpha' \oplus (\alpha' \lll 1) \oplus \beta' = \mathbf{0}_{128}, \tag{9}$$
$$\alpha' \oplus (\alpha' \lll 81) \oplus (\gamma' \lll 80) = \mathbf{0}_{128}, \tag{10}$$
$$\beta'' = \mathbf{0}_{128}, \tag{11}$$
$$\gamma'' = \mathbf{0}_{128}. \tag{12}$$

On one hand, from Eq. (6) and Eq. (9), we have $\alpha' = \alpha' \lll 1$. The only two values of α' satisfying $\alpha' = \alpha' \lll 1$ are $\alpha' = \mathbf{0}_{128}$ and $\alpha' = \mathbf{1}_{128}$.

If $\alpha' = \mathbf{0}_{128}$, we have $(\alpha', \beta', \gamma') = \mathbf{0}_{3\times128}$. Because the round function of FRIET-PC is bijective, it contradicts with that $(\alpha, \beta, \gamma) \to (\alpha', \beta', \gamma') \to (\alpha'', \beta'', \gamma'')$ is a nonzero difference.

If $\alpha' = \mathbf{1}_{128}$, from Eq. (6), (7) and (8), we have $\alpha'' = \mathbf{1}_{128}$. According to Eq. (4) and Eq. (5) we have

$$((\alpha \oplus (\alpha \lll 81) \oplus (\gamma \lll 80)) \ggg 80) \oplus \alpha \oplus (\alpha \lll 1) \oplus \beta = (\mathbf{0}_{128} \ggg 80) \oplus \mathbf{0}_{128}$$

Combining with Eq. (3), we have

$$(\alpha \ggg 80) = \alpha' = \mathbf{1}_{128}.$$

Thus, $\alpha = \mathbf{1}_{128}$. Substituting the value $\alpha = \mathbf{1}_{128}$ into Eq. (4) and Eq. (5) , we have $\beta = \mathbf{0}_{128}$ and $\gamma = \mathbf{0}_{128}$. The necessity is proved.

On the other hand, the nonzero difference $(\mathbf{1}_{128}, \mathbf{0}_{128}, \mathbf{0}_{128}) \to (\mathbf{1}_{128}, \mathbf{0}_{128}, \mathbf{0}_{128}) \to (\mathbf{1}_{128}, \mathbf{0}_{128}, \mathbf{0}_{128})$ satisfies all the Eq. (3-12). The sufficiency is proved. □

Based on Lemma 2, we can get the following corollary easily.

Corollary 1. *For n-round FRIET-PC,* $(\mathbf{1}_{128}, \mathbf{0}_{128}, \mathbf{0}_{128}) \to \cdots \to (\mathbf{1}_{128}, \mathbf{0}_{128}, \mathbf{0}_{128})$ *is the only nonzero difference with probability 1, where* $n \geq 2$.

Thus, we obtain a differential distinguisher with probability 1 for the full-round FRIET-PC.

3.2 A Linear Distinguisher for the Full-Round FRIET-PC

According to **Linear Property 1 (Branching)**, **Linear Property 2 (XOR)** and **Linear Property 3 (XOR-Constant)** in Sect. 2.2, the linear correlation of a valid linear mask for these three operations is 1 or -1. By **Linear Property 4 (AND)**, the linear correlation of a valid linear mask for bitwise AND operation is determined by $wt(\gamma)$, where γ is the output linear mask of the bitwise AND operation. If controlling the value of $wt(\gamma)$ effectively, we may obtain linear masks with high correlations.

Lemma 3. *Let* $\Gamma_{in} = (\alpha, \beta, \gamma)$ *and* $\Gamma_{out} = (\alpha', \beta', \gamma')$ *be the input and output linear masks of the i-th round function* $(a', b', c') = FRIET\text{-}PC_i(a, b, c)$. *The absolute value of correlation* $Cor(\Gamma_{in}, \Gamma_{out})$ *is 1 if and only if*

$$\begin{cases} \alpha' = \mathbf{0}_{128}, \\ \alpha \oplus (\beta' \ggg 1) \oplus \gamma' \oplus ((\beta' \oplus \gamma') \ggg 81) = \mathbf{0}_{128}, \\ \beta \oplus \beta' = \mathbf{0}_{128}, \\ \gamma \oplus ((\beta' \oplus \gamma') \ggg 80) = \mathbf{0}_{128}. \end{cases} \tag{13}$$

Proof. By the round function of FRIET-PC, we have

$$\begin{cases} a' = a \oplus b \oplus c \oplus rc_i \oplus ((b' \lll 36) \wedge (c' \lll 67)) \\ b' = (a \lll 1) \oplus b \oplus (a \lll 81) \oplus ((c \oplus rc_i) \lll 80), \\ c' = a \oplus (a \lll 81) \oplus ((c \oplus rc_i) \lll 80). \end{cases}$$

On one hand, if the absolute value of $Cor(\Gamma_{in}, \Gamma_{out})$ is 1. According to **Linear Property 4 (AND)**, the input linear mask and output linear mask of AND operation should be $\mathbf{0}_{128}||\mathbf{0}_{128}$ and $\mathbf{0}_{128}$, respectively. Because $((b' \lll 36) \wedge (c' \lll 67))$ only appear in a', we have $\alpha' = 0$. Then,

$$\begin{aligned}\Gamma_{in} \cdot (a,b,c) \oplus \Gamma_{out} \cdot (a',b',c') &= \alpha \cdot a \oplus \beta \cdot b \oplus \gamma \cdot c \oplus \alpha' \cdot a' \oplus \beta' \cdot b' \oplus \gamma' \cdot c' \\ &= \alpha \cdot a \oplus \beta' \cdot (a \lll 1) \oplus \gamma' \cdot a \oplus (\beta' \oplus \gamma') \cdot (a \lll 81) \\ &\quad \oplus (\beta \oplus \beta') \cdot b \oplus \gamma \cdot c \oplus (\beta' \oplus \gamma') \cdot (c \lll 80) \\ &\quad \oplus (\beta' \oplus \gamma') \cdot (rc_i \lll 80) \\ &= (\alpha \oplus (\beta' \ggg 1) \oplus \gamma' \oplus ((\beta' \oplus \gamma') \ggg 81)) \cdot a \\ &\quad \oplus (\beta \oplus \beta') \cdot b \oplus (\gamma \oplus ((\beta' \oplus \gamma') \ggg 80)) \cdot c \\ &\quad \oplus (\beta' \oplus \gamma') \cdot (rc_i \lll 80)\,.\end{aligned}$$

We know that the above $\Gamma_{in} \cdot (a,b,c) \oplus \Gamma_{out} \cdot (a',b',c')$ is a linear function. Thus, if $|Cor\,(\Gamma_{in}, \Gamma_{out})\,| = 1$, we have

$$\begin{cases} \alpha' = \mathbf{0}_{128}, \\ \alpha \oplus (\beta' \ggg 1) \oplus \gamma' \oplus ((\beta' \oplus \gamma') \ggg 81) = \mathbf{0}_{128}, \\ \beta \oplus \beta' = \mathbf{0}_{128}, \\ \gamma \oplus ((\beta' \oplus \gamma') \ggg 80) = \mathbf{0}_{128}. \end{cases}$$

The necessity is proved.

On the other hand, if the input linear mask (α, β, γ) and output linear mask $(\alpha', \beta', \gamma')$ satisfy Eq. (13). Its linear correlations through all the basic operations (Rotation, XOR, XOR-Constant, AND) in the round function of FRIET-PC is 1 or -1. Thus, the absolute value of the linear correlation is 1. The sufficiency is proved. □

According to Lemma 3, we obtain the following corollary.

Corollary 2. *For the input linear mask $\Gamma_{in} = (\mathbf{0}_{128}, \mathbf{0}_{128}, \mathbf{1}_{128})$ and output linear mask $\Gamma_{out} = (\mathbf{0}_{128}, \mathbf{0}_{128}, \mathbf{1}_{128})$, the absolute value of correlation $Cor\,(\Gamma_{in}, \Gamma_{out})$ for n-round FRIET-PC is 1, where $n \geq 1$.*

Proof. Because $(\alpha, \beta, \gamma) = \Gamma_{in}$ and $(\alpha', \beta', \gamma') = \Gamma_{out}$ satisfy Eq. (13). The absolute value of correlation $Cor\,(\Gamma_{in}, \Gamma_{out})$ for 1-round FRIET-PC is 1. By applying the propagation of linear mask $(\mathbf{0}_{128}, \mathbf{0}_{128}, \mathbf{1}_{128}) \rightarrow (\mathbf{0}_{128}, \mathbf{0}_{128}, \mathbf{1}_{128})$ iteratively, the absolute value of correlation $Cor\,(\Gamma_{in}, \Gamma_{out})$ for n-round FRIET-PC is 1. □

Thus, we obtain a linear distinguisher whose absolute value of correlation is 1 for full-round FRIET-PC.

4 Practical Attacks on the Full-Round FRIET-AE

When FRIET-P is used in FRIET authenticated encryption scheme, FRIET-AE is obtained. It is based on duplex construction and its mode SpongeWrap [BDPA11a], but some modifications are made. FRIET-AE limits the key length to $k \leq 160$ bits and takes tag length as 128 bits. The encryption and decryption operations are illustrated in Figure 2 and Figure 3.

In this paper, we do not study its fault-resistance ability. Without affecting the correctness, F denotes the permutation FRIET-PC whose input and output are 384 bits (3 limbs). And $\mathbf{0}_*$ means adding a bit vector whose binary entries are all 0 until the length of the entire vector reaches 384 bits. In FRIET-AE, the block length is 128 bits, that is all the input data (Key, Nonce, Associate data, Plaintext, Ciphertext) are split into 128-bit blocks and the last block may be shorter. It should be noted that the authors in the paper

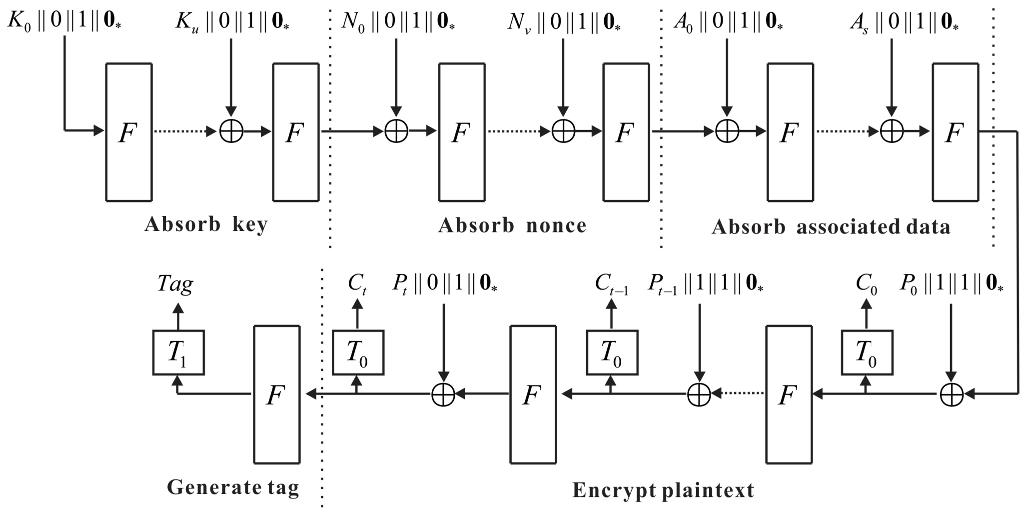


Figure 2: The encryption operation of FRIET-AE

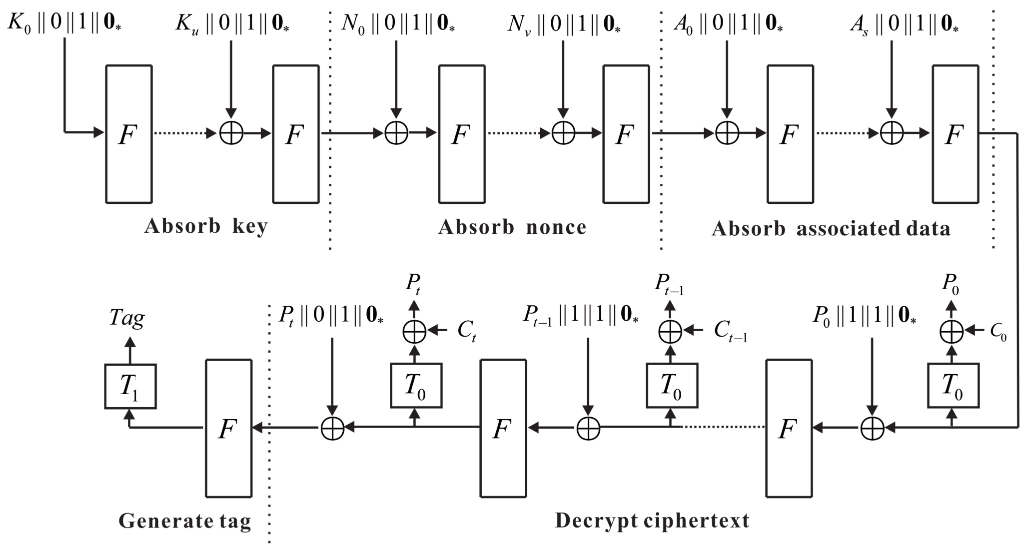


Figure 3: The decryption operation of FRIET-AE

[SBD+20] do not clarify how the external data such as $A_0||0||1||\mathbf{0}_*$ combines with the internal state $(a, b, c) \in \mathbb{F}_2^{384}$. But in the analysis of differential and linear propagations, the authors wrote "Because an adversary can only access the outer state in FRIET, we restricted our analysis to differential trail with input differences in limb a". Therefore, the way of combining the external data $A_0||0||1||\mathbf{0}_*$ with the internal three limbs $(a, b, c) \in \mathbb{F}_2^{384}$ should be $(A_0||0||1||\mathbf{0}_*) \oplus (a||b||c)$. And the corresponding truncated function should be $T_0\,(a, b, c) = a$. The way of extracting the tag from the internal state will not affect our attacks in this paper. We denote the function of generating tag as $T_1\,(a, b, c) = Tag$.

Under the assumption that adversaries respect the nonce requirement for the diversifier and do not get access to deciphered ciphertexts of cryptograms with an invalid tag, FRIET-AE claims a 128-bit security of integrity and confidentiality. If adversaries can construct a new cryptogram which has not ever been created by legal users and the cryptogram can be successfully decrypted by a legal user, the integrity is broken. If keystream, i.e., keyed duplex output, can be predicted or a cryptogram can be decrypted by adversaries, the confidentiality is broken.

For FRIET-AE, if we get a tag $Tag \in \mathbb{F}_2^{128}$ and a ciphertext C which are generated by $\mathit{FRIET\text{-}AE}\,(K, N, AD, P)$, where K is the key, N is the nonce, AD is the associate data, P is the plaintext. Take K for an example, let $|K|$ denote the bit length of K. The number of blocks of K is $\left\lceil \frac{|K|}{128} \right\rceil$, denoted as $K = K_{\left\lceil \frac{|K|}{128} \right\rceil - 1} || \ldots || K_1 || K_0$. And the number of blocks of K whose length is 128 bits is $\left\lfloor \frac{|K|}{128} \right\rfloor$. By using the differential distinguisher with probability 1 in Corollary 1, we design an algorithm to generate a set consisting of valid tags and ciphertexts which are not created by legal users. We illustrate the whole framework in Algorithm 2.

Algorithm 2. $\mathit{Attack}\,(K, N, AD, P, C, Tag)$

Input: K, N, AD, P, C, Tag
Output: A set Ω consisting of valid tags and ciphertexts

Initialize $\Omega = \emptyset$, $flag = 0$, $I_k = \left\lfloor \frac{|K|}{128} \right\rfloor$, $I_n = \left\lfloor \frac{|N|}{128} \right\rfloor$, $I_{ad} = \left\lfloor \frac{|AD|}{128} \right\rfloor$, $I_p = \left\lfloor \frac{|P|}{128} \right\rfloor$,
 $U_k = 2^{I_k}$, $U_n = 2^{I_n}$, $U_{ad} = 2^{I_{ad}}$, $U_p = 2^{I_p}$
for $(u_k = 0; u_k < U_k; u_k{+}{+})$ **do**
 for $(u_n = 0; u_n < U_n; u_n{+}{+})$ **do**
  for $(u_{ad} = 0; u_{ad} < U_{ad}; u_{ad}{+}{+})$ **do**
   for $(u_p = 0; u_p < U_p; u_p{+}{+})$ **do**
    if $u_k = u_n = u_{ad} = u_p = 0$ **do**
     continue
    let $K' = K$, $N' = N$, $AD' = AD$, $P' = P$, $C' = C$, $Tag' = Tag$
    for $(i_k = 0; i_k < I_k; i_k{+}{+})$ **do**
     if $\delta_{i_k}(u_k) == 1$ **do**
      $K' = K' \oplus (\mathbf{1}_{128} \lll (i_k \times 128))$
      $flag = falg \oplus 1$
    for $(i_n = 0; i_n < I_n; i_n{+}{+})$ **do**
     if $\delta_{i_n}(u_n) == 1$ **do**
      $N' = N' \oplus (\mathbf{1}_{128} \lll (i_n \times 128))$
      $flag = falg \oplus 1$
    for $(i_{ad} = 0; i_{ad} < I_{ad}; i_{ad}{+}{+})$ **do**
     if $\delta_{i_{ad}}(u_{ad}) == 1$ **do**
      $AD' = AD' \oplus (\mathbf{1}_{128} \lll (i_{ad} \times 128))$
      $flag = falg \oplus 1$
    for $(i_p = 0; i_p < I_p; i_p{+}{+})$ **do**
     if $\delta_{i_p}(u_p) == 1$ **do**
      $P' = P' \oplus (\mathbf{1}_{128} \lll (i_p \times 128))$
      $flag = falg \oplus 1$
     if $flag == 1$ **do**
      $C' = C' \oplus (\mathbf{1}_{128} \lll (i_p \times 128))$
    if $flag == 1$ **do**
     $Tag' = Tag' \oplus T_1(\mathbf{1}_{128}, \mathbf{0}_{128}, \mathbf{0}_{128})$
    $\Omega = \Omega \cup \{(K', N', AD', P', C', Tag')\}$
return Ω

In Algorithm 2, $flag == 0$ means that the difference of the internal state of FRIET-AE is $\mathbf{0}_{128}||\mathbf{0}_{128}||\mathbf{0}_{128}$ and $flag == 1$ means the difference of the internal state is $\mathbf{1}_{128}||\mathbf{0}_{128}||\mathbf{0}_{128}$. Because the round function of FRIET-AE will not change the differences $\mathbf{0}_{128}||\mathbf{0}_{128}||\mathbf{0}_{128}$ and $\mathbf{1}_{128}||\mathbf{0}_{128}||\mathbf{0}_{128}$. Thus, we can get the corresponding C' and Tag' from the differences of states, C and Tag. Because the condition $u_k = u_n = u_{ad} = u_p = 0$

will not add any element into the set Ω. All the $(K', N', AD', P', C', Tag') \in \Omega$ have valid tags and ciphertexts which are not created by legal users. It should be noted that they belong to different attack conditions. We will have a classified discussion.

Related-Key Attack. According to Algorithm 2, we only introduce difference of the form $\mathbf{1}_{128}||\mathbf{0}_{128}||\mathbf{0}_{128}$ to the internal state. When there is difference in K, the number of elements in the set Ω is $\left(2^{\left\lfloor \frac{|K|}{128} \right\rfloor} - 1\right) \times 2^{\left\lfloor \frac{|N|}{128} \right\rfloor} \times 2^{\left\lfloor \frac{|AD|}{128} \right\rfloor} \times 2^{\left\lfloor \frac{|P|}{128} \right\rfloor}$.

Single-Key Attack. If there is no difference in K, under the condition that nonce cannot be reused, we must introduce differences into N. The number of elements in the set Ω is $\left(2^{\left\lfloor \frac{|N|}{128} \right\rfloor} - 1\right) \times 2^{\left\lfloor \frac{|AD|}{128} \right\rfloor} \times 2^{\left\lfloor \frac{|P|}{128} \right\rfloor}$. If adversaries have the ability of reusing nonce, the number of elements in the set Ω is $2^{\left\lfloor \frac{|N|}{128} \right\rfloor} \times 2^{\left\lfloor \frac{|AD|}{128} \right\rfloor} \times 2^{\left\lfloor \frac{|P|}{128} \right\rfloor} - 1$.

According to the above analysis, we can construct valid tags and ciphertexts which are not created by legal users. And the single-key attack without reusing nonce fully complies with the security assumption of FRIET-AE. This breaks the integrity and confidentiality security claims. And our attack can be conducted in practical time.

5 Conclusions

In this paper, differential and linear distinguishers for the full-round FRIET-PC are proposed. Using the differential distinguisher with probability 1, we proposed an algorithm which can generate a set consisting of valid tags and ciphertexts which are not created by legal users. This breaks the integrity and confidentiality security claims of FRIET-AE. It should be noted that our attack does not recover the secret key of FRIET-AE. How to give a key-recovery attack needs further research.

References

[BBdS+] Christof Beierle, Alex Biryukov, Luan Cardoso dos Santos, Johann Großschädl, Léo Perrin, Aleksei Udovenko, Vesselin Velichkov, Qingju Wang, Amir Moradi, and Aein Rezaei Shahmirzadi. Sparkle. Submission as a Finalist to the NIST Lightweight Crypto Standardization Process 2021. https://csrc.nist.gov/projects/lightweight-cryptography/finalists.

[BBdS+20] Christof Beierle, Alex Biryukov, Luan Cardoso dos Santos, Johann Großschädl, Léo Perrin, Aleksei Udovenko, Vesselin Velichkov, and Qingju Wang. Alzette: A 64-bit arx-box - (feat. CRAX and TRAX). In Daniele Micciancio and Thomas Ristenpart, editors, *Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III*, volume 12172 of *Lecture Notes in Computer Science*, pages 419–448. Springer, 2020.

[BCD+] Zhenzhen Bao, Avik Chakraborti, Nilanjan Datta, Jian Guo, Mridul Nandi, Thomas Peyrin, and Kan Yasuda. Photon-beetle. Submission as a Finalist to the NIST Lightweight Crypto Standardization Process 2021. https://csrc.nist.gov/projects/lightweight-cryptography/finalists.

[BCDM] Tim Beyne, Yu Long Chen, Christoph Dobraunig, and Bart Mennink. Elephant. Submission as a Finalist to the NIST Lightweight

Crypto Standardization Process 2021. https://csrc.nist.gov/projects/lightweight-cryptography/finalists.

[BDPA08] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. On the indifferentiability of the sponge construction. In Nigel P. Smart, editor, *Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings*, volume 4965 of *Lecture Notes in Computer Science*, pages 181–197. Springer, 2008.

[BDPA11a] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Duplexing the sponge: Single-pass authenticated encryption and other applications. In Ali Miri and Serge Vaudenay, editors, *Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*, volume 7118 of *Lecture Notes in Computer Science*, pages 320–337. Springer, 2011.

[BDPA11b] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. The keccak reference. https://keccak.team/keccak.html, 2011.

[BKL+17] Daniel J. Bernstein, Stefan Kölbl, Stefan Lucks, Pedro Maat Costa Massolino, Florian Mendel, Kashif Nawaz, Tobias Schneider, Peter Schwabe, François-Xavier Standaert, Yosuke Todo, and Benoît Viguier. Gimli : A cross-platform permutation. In Wieland Fischer and Naofumi Homma, editors, *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, volume 10529 of *Lecture Notes in Computer Science*, pages 299–320. Springer, 2017.

[BS90] Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. In Alfred Menezes and Scott A. Vanstone, editors, *Advances in Cryptology - CRYPTO '90, 10th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1990, Proceedings*, volume 537 of *Lecture Notes in Computer Science*, pages 2–21. Springer, 1990.

[DEM+] Christoph Dobraunig, Maria Eichlseder, Stefan Mangard, Florian Mendel, Bart Mennink, Robert Primas, and Thomas Unterluggauer. Isap. Submission as a Finalist to the NIST Lightweight Crypto Standardization Process 2021. https://csrc.nist.gov/projects/lightweight-cryptography/finalists.

[DEMSa] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon. Submission as a Finalist to the NIST Lightweight Crypto Standardization Process 2021. https://csrc.nist.gov/projects/lightweight-cryptography/finalists.

[DEMSb] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2. Submission to CAESAR: Competition for Authenticated Encryption. Security, Applicability, and Robustness 2016. http://competitions.cr.yp.to/round3/asconv12.pdf.

[DEMS19] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Markus Schofnegger. Algebraic cryptanalysis of variants of frit. In Kenneth G. Paterson and Douglas Stebila, editors, *Selected Areas in Cryptography - SAC 2019 - 26th International Conference, Waterloo, ON, Canada, August 12-16, 2019, Revised Selected Papers*, volume 11959 of *Lecture Notes in Computer Science*, pages 149–170. Springer, 2019.

[DHAK18] Joan Daemen, Seth Hoffert, Gilles Van Assche, and Ronny Van Keer. The design of xoodoo and xoofff. *IACR Trans. Symmetric Cryptol.*, 2018(4):1–38, 2018.

[DHP+] Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, Ronny Van Keer, and Silvia Mella. Xoodyak. Submission as a Finalist to the NIST Lightweight Crypto Standardization Process 2021. https://csrc.nist.gov/projects/lightweight-cryptography/finalists.

[EM91] Shimon Even and Yishay Mansour. A construction of a cipher from a single pseudorandom permutation. In Hideki Imai, Ronald L. Rivest, and Tsutomu Matsumoto, editors, *Advances in Cryptology — ASIACRYPT '91*, pages 210–224, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg.

[GM16] Shay Gueron and Nicky Mouha. Simpira v2: A family of efficient permutations using the AES round function. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *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*, volume 10031 of *Lecture Notes in Computer Science*, pages 95–125, 2016.

[ISS+21] Ryoma Ito, Rentaro Shiba, Kosei Sakamoto, Fukang Liu, and Takanori Isobe. Bit-wise cryptanalysis on AND-RX permutation friet-pc. *J. Inf. Secur. Appl.*, 59:102860, 2021.

[LSL21] Yunwen Liu, Siwei Sun, and Chao Li. Rotational cryptanalysis from a differential-linear perspective - practical distinguishers for round-reduced friet, xoodoo, and alzette. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I*, volume 12696 of *Lecture Notes in Computer Science*, pages 741–770. Springer, 2021.

[Mat93] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.

[QDJZ19] Lingyue Qin, Xiaoyang Dong, Keting Jia, and Rui Zong. Key-dependent cube attack on reduced frit permutation in duplex-ae modes. *IACR Cryptol. ePrint Arch.*, page 170, 2019.

[SBD+18] Thierry Simon, Lejla Batina, Joan Daemen, Vincent Grosso, Pedro Maat Costa Massolino, Kostas Papagiannopoulos, Francesco Regazzoni, and Niels Samwel. Towards lightweight cryptographic primitives with built-in fault-detection. *IACR Cryptol. ePrint Arch.*, page 729, 2018.

[SBD+20] Thierry Simon, Lejla Batina, Joan Daemen, Vincent Grosso, Pedro Maat Costa Massolino, Kostas Papagiannopoulos, Francesco Regazzoni, and Niels Samwel. Friet: An authenticated encryption scheme with built-in fault detection. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I*, volume 12105 of *Lecture Notes in Computer Science*, pages 581–611. Springer, 2020.