Éléments de cryptanalyse

Habilitation à Diriger des Recherches

Yann Rotella

Université Paris-Saclay

6 février 2025

Laboratoire de mathématiques de Versailles - CNRS UMR 8100

BASICS

1. Information is valuable.

Basics

1. Information is valuable.
2. Malicious people exist.

BASICS

1. Information is valuable.
2. Malicious people exist.
3. We need ciphers.

BASICS

1. Information is valuable.
2. Malicious people exist.
3. We need ciphers.
4. We can’t prove that a cipher is secure.

Basics

1. Information is valuable.
2. Malicious people exist.
3. We need ciphers.
4. We can't prove that a cipher is secure.

We do cryptanalysis

Overview of Contributions

Cryptanalysis of primitives

Pyjamask-96 [ToSC:DRS20]

GEA 1/2 [EC:BDLLRRRS21]

Keccak [ToSC:HNR21]

Panther [AfC:BHR22]

Troïka [SAC:BMR22]

Designs

Subterranean 2.0 [ToSC:DMMR20]

LwPR [C:HMMRSU23]

Transistor [BBBBCLPPRR]

Compression functions [C:FRD23]

Duplex-based modes [EC:GHKR23]

Cryptanalysis of modes

OUTLINE

Cryptanalysis of the GPRS Encryption Algorithms GEA-1 and GEA-2

Beierle, Derbez, Leander, Leurent, Raddum, R,
Rupprecht and Stennes

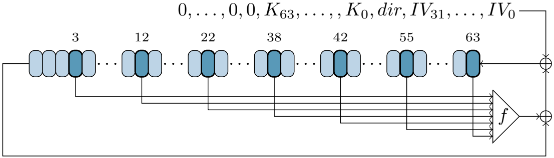
EUROCRYPT 2023

GEA-1 : CONTEXT (BEFORE 2020)

- Proprietary stream cipher, designed by ETSI in 1998
- GPRS (General Packet Radio Service)
- No public specification available
- Reverse engineered (partly) by Nohl and Melette (2011)

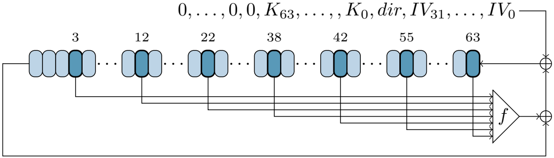
GEA-1 : INITIALISATION

S, 225 times :



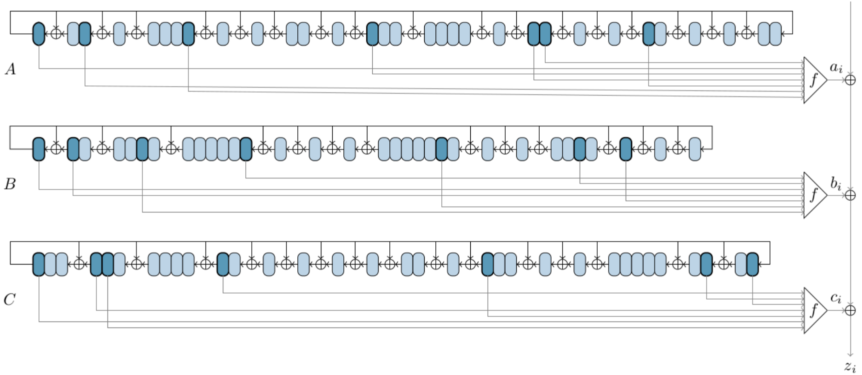
GEA-1 : INITIALISATION

S, 225 times :

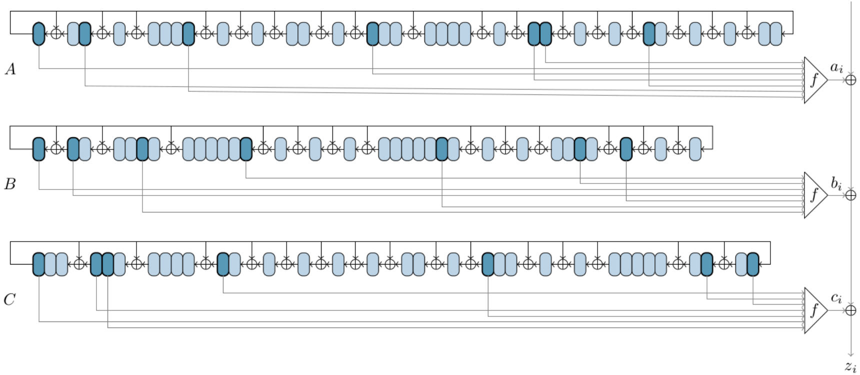


We call s the secret value contained in S.

GEA-1 : STRUCTURE



GEA-1 : Structure



$s \in \mathbb{F}_2^{64}$, 64 initialisation clocks :

$$A \leftarrow s_0 s_1 \cdots s_{63}$$

$$B \leftarrow s_{16} s_{17} \cdots s_{15}$$

$$C \leftarrow s_{32} s_{33} \cdots s_{31}$$

GEA-1 : OBSERVATION

The relation between S and A, B and C is linear.

GEA-1 : OBSERVATION

The relation between S and A, B and C is linear.
$\exists\ M_A \in \mathcal{M}(31 \times 64)$, $M_B \in \mathcal{M}(32 \times 64)$, $M_C \in \mathcal{M}(33 \times 64)$ such that

$$\alpha = M_A s$$
$$\beta = M_B s$$
$$\gamma = M_C s$$

GEA-1 : OBSERVATION

The relation between S and A, B and C is linear.
$\exists\ M_A \in \mathcal{M}(31 \times 64)$, $M_B \in \mathcal{M}(32 \times 64)$, $M_C \in \mathcal{M}(33 \times 64)$ such that

$$\alpha = M_A s$$
$$\beta = M_B s$$
$$\gamma = M_C s$$

$$\dim(\ker(M_A) \cap \ker(M_C)) = 24$$

GEA-1 : Attack

$$\mathbb{F}_2^{64} = (\ker(M_A) \cap \ker(M_C)) \oplus \ker(M_B) \oplus V$$

$$\alpha = M_A(t+u+v) = M_A(u+v)$$
$$\beta = M_B(t+u+v) = M_B(t+v)$$
$$\gamma = M_C(t+u+v) = M_C(u+v)$$

GEA-1 : Attack

$$\mathbb{F}_2^{64} = (\ker(M_A) \cap \ker(M_C)) \oplus \ker(M_B) \oplus V$$

$$\alpha = M_A(t+u+v) = M_A(u+v)$$
$$\beta = M_B(t+u+v) = M_B(t+v)$$
$$\gamma = M_C(t+u+v) = M_C(u+v)$$

`for` $v \in V$,

1. `Compute and sort` $\mathcal{L} = (z_i + f(\beta^i(t,v)))_{t,i}$
2. `for` u, `look for` $(f(\alpha^i(u,v)) + f(\gamma^i(u,v)))_{u,i}$ `in` $\mathcal{L}$

GEA-1 : ATTACK

$$\mathbb{F}_2^{64} = (\ker(M_A) \cap \ker(M_C)) \oplus \ker(M_B) \oplus V$$

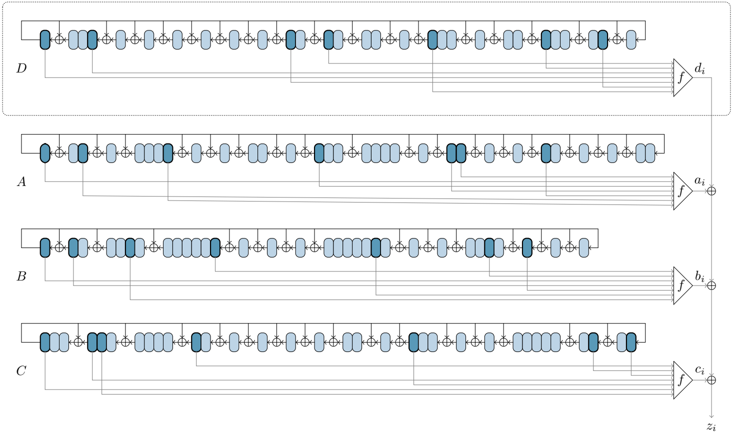
$$\alpha = M_A(t+u+v) = M_A(u+v)$$
$$\beta = M_B(t+u+v) = M_B(t+v)$$
$$\gamma = M_C(t+u+v) = M_C(u+v)$$

`for` $v \in V$,

1. `Compute and sort` $\mathcal{L} = (z_i + f(\beta^i(t,v)))_{t,i}$
2. `for` u, `look for` $(f(\alpha^i(u,v)) + f(\gamma^i(u,v)))_{u,i}$ `in` $\mathcal{L}$

Cost of the attack 2^{40}

GEA-2 : IMPROVEMENTS



GEA-2 : GUESS AND DETERMINE

Number of monomials :

$$1 + \sum_{i=1}^{4} \binom{29}{i} + \binom{31}{i} + \binom{32}{i} + \binom{33}{i} = 152\,682$$

GEA-2 : Guess and Determine

Number of monomials :

$$1 + \sum_{i=1}^{4} \binom{29}{i} + \binom{31}{i} + \binom{32}{i} + \binom{33}{i} = 152\,682$$

But 12 800 bits per frame.

GEA-2 : Guess and Determine

Number of monomials :

$$1 + \sum_{i=1}^{4} \binom{29}{i} + \binom{31}{i} + \binom{32}{i} + \binom{33}{i} = 152\,682$$

But 12 800 bits per frame.

Guess $n_A + n_B + n_C + n_D$ bits :

$$1 + \sum_{i=1}^{4} \binom{29 - n_D}{i} + \binom{31 - n_A}{i} + \binom{32 - n_B}{i} + \binom{33 - n_C}{i}$$

GEA-2 : ATTACK

1. Guess n_A and n_D bits

GEA-2 : ATTACK

1. Guess n_A and n_D bits
2. Derive linear equations $(\langle m_i, s_{A+D} \rangle)_{1 \leq i \leq n}$ that do not depend on any bits of A or D (number of monomials < 12800).

GEA-2 : ATTACK

1. Guess n_A and n_D bits
2. Derive linear equations $(\langle m_i, s_{A+D}\rangle)_{1\leq i\leq n}$ that do not depend on any bits of A or D (number of monomials < 12800).
3. Apply list-merging to

$$t = \langle m_1, z\rangle \oplus \langle m_1, s_{A+D}\rangle, \langle m_2, z\rangle \oplus \langle m_2, s_{A+D}\rangle, \ldots$$

and

$$f_1 : \beta \mapsto \langle m_1, s_B(\beta)\rangle, \ldots, \langle m_c, s_B(\beta)\rangle$$

and

$$f_2 : \gamma \mapsto \langle m_1, s_C(\gamma)\rangle, \ldots, \langle m_c, s_C(\gamma)\rangle$$

GEA 1/2 : SUMMARY AND FUTURE WORK

Summary :

- `GEA-1` attack in 2^{40}
- `GEA-2` attack in $2^{45.1}$

GEA 1/2 : Summary and future work

Summary :

- `GEA-1` attack in 2^{40}
- `GEA-2` attack in $2^{45.1}$

Improved by :

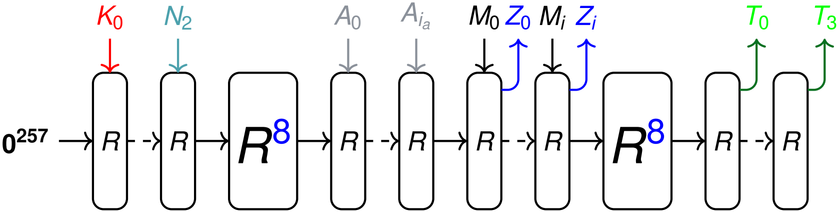
- Amzaleg and Dinur in 2022
- Avoine, Carpent, Claverie, Devine and Leblanc-Albarel in 2024

The Subterranean 2.0 Cipher Suite

Daemen, Maat Costa Massolino, Mehrdad and R.

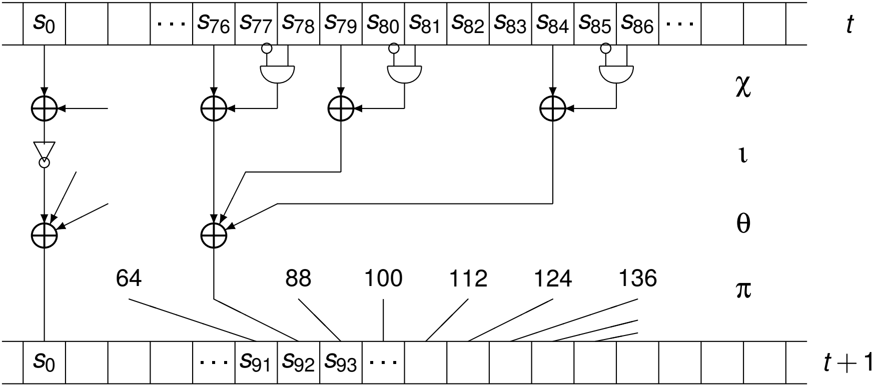
NIST-lwc and ToSC 2020

Subterranean 2.0 : Mode



- $|K_j| = |N_j| = |A_j| = |M_j| = 33 = 32 + 1$: 32 bits of message, 1 bit for padding
- $|Z_j| = |T_j| = 32$
- State size of 257 bits

SUBTERRANEAN 2.0 : ROUND FUNCTION



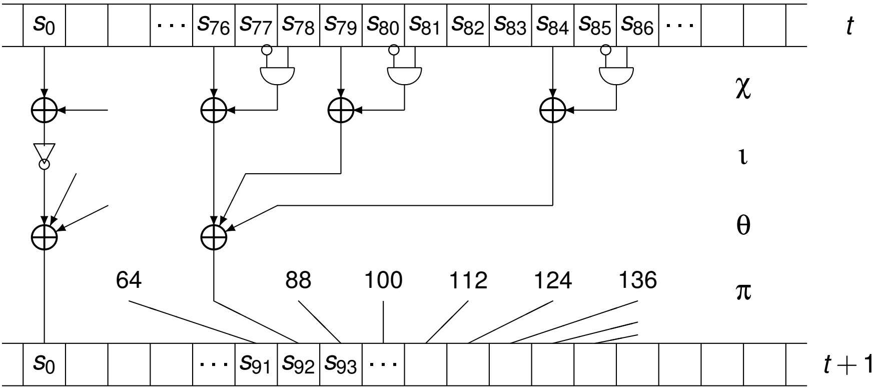
$\chi : s_i \leftarrow s_i + (s_{i+1} + 1)s_{i+2}$

$\iota : s_i \leftarrow s_i + \delta_i$

$\theta : s_i \leftarrow s_i + s_{i+3} + s_{i+8}$

$\pi : s_i \leftarrow s_{12i}$

SUBTERRANEAN 2.0 : ROUND FUNCTION



$\chi : s_i \leftarrow s_i + (s_{i+1} + 1)s_{i+2}$

$\iota : s_i \leftarrow s_i + \delta_i$

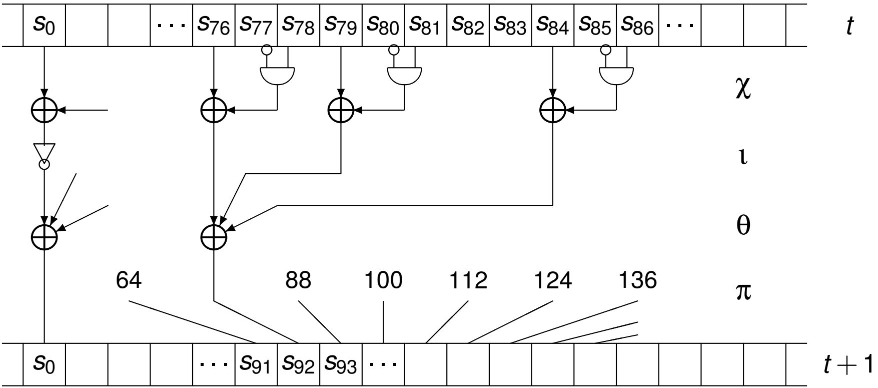
$\theta : s_i \leftarrow s_i + s_{i+3} + s_{i+8}$

$\pi : s_i \leftarrow s_{12i}$

For j from 0 to 32,

Absorption : $s_{12^{4j}} \leftarrow s_{12^{4j}} + x_j$

Subterranean 2.0 : Round function



$\chi : s_i \leftarrow s_i + (s_{i+1}+1)s_{i+2}$

$\iota : s_i \leftarrow s_i + \delta_i$

$\theta : s_i \leftarrow s_i + s_{i+3} + s_{i+8}$

$\pi : s_i \leftarrow s_{12i}$

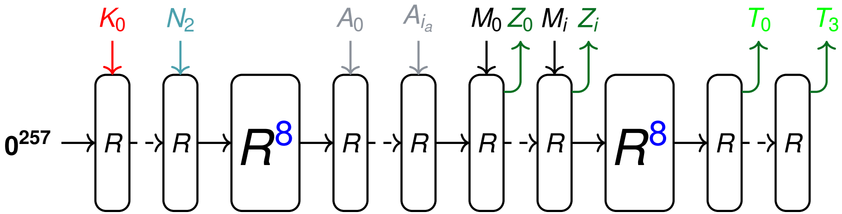
For j from 0 to 32,

Absorption : $s_{12^{4j}} \leftarrow s_{12^{4j}} + x_j$

For j from 0 to 31,

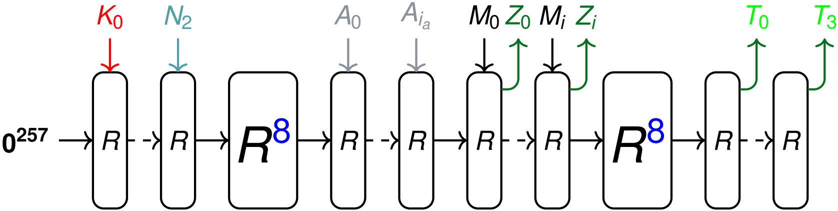
Extraction : $z \leftarrow z || (s_{12^{4j}} + s_{-12^{4j}})$

Subterranean 2.0 : Arguments



”Strong” permutation for initialisation and finalisation

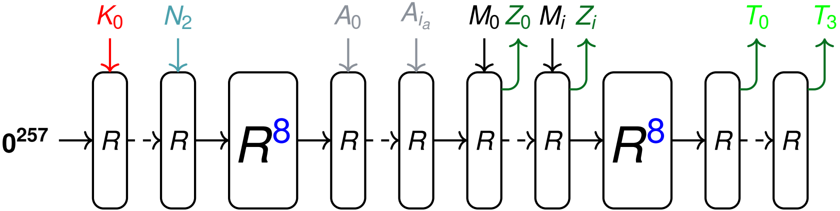
SUBTERRANEAN 2.0 : ARGUMENTS



"Strong" permutation for initialisation and finalisation, corroborated by :

- Liu, Isobe and Meier in 2019 with cubes
- El Hirch, Mehrdad, Mella, Grassi, Daemen in 2022 and 2023 for differential trail search

SUBTERRANEAN 2.0 : ARGUMENTS

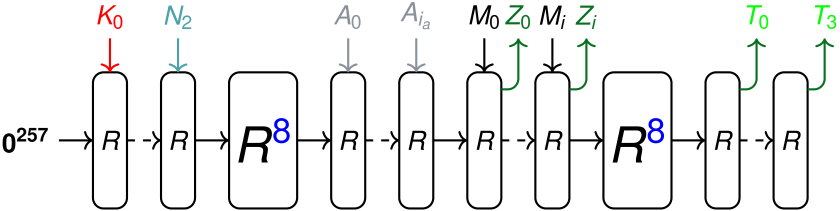


"Strong" permutation for initialisation and finalisation, corroborated by :

- Liu, Isobe and Meier in 2019 with cubes
- El Hirch, Mehrdad, Mella, Grassi, Daemen in 2022 and 2023 for differential trail search

"Light" permutation in the middle :

SUBTERRANEAN 2.0 : ARGUMENTS



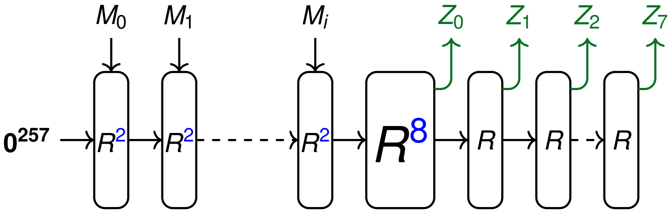
"Strong" permutation for initialisation and finalisation, corroborated by :

- Liu, Isobe and Meier in 2019 with cubes
- El Hirch, Mehrdad, Mella, Grassi, Daemen in 2022 and 2023 for differential trail search

"Light" permutation in the middle :

- Efficiency
- Wise choice of bit positions

SUBTERRANEAN 2.0 : HASHING



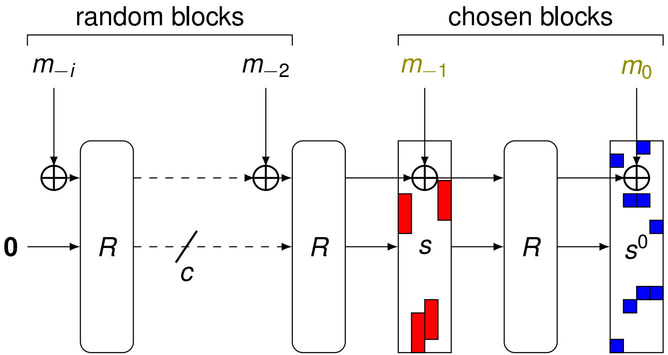
- $|M_j| = 9 = 8 + 1$, 8 bits of message, 1 padding
- $|Z_j| = 32$, NIST : 8 output blocks

Subterranean 2.0 : Inner collisions

- Generic attack costs $2^{c/2} = 2^{(257-8)/2} = 2^{124.5}$

SUBTERRANEAN 2.0 : INNER COLLISIONS

- Generic attack costs $2^{c/2} = 2^{(257-8)/2} = 2^{124.5}$



SUBTERRANEAN 2.0 : SYSTEM FOR COLLISIONS

$$
\begin{cases}
q_{124}(s) + q_{124}(s') & = & b_5 s_{133} + b'_5 s'_{133} \\
q_{125}(s) + q_{125}(s') & = & b_5 s_{135} + b'_5 s'_{135} \\
q_{126}(s) + q_{126}(s') & = & b_5 + b'_5 + b_2 s_{135} + b'_2 s'_{135} \\
q_{127}(s) + q_{127}(s') & = & b_2 s_{137} + b'_2 s'_{137} \\
q_{128}(s) + q_{128}(s') & = & b_2 + b'_2 \\
q_{129}(s) + q_{129}(s') & = & b_5 s_{133} + b'_5 s'_{133} \\
q_{130}(s) + q_{130}(s') & = & b_5 s_{135} + b'_5 s'_{135} \\
q_{131}(s) + q_{131}(s') & = & b_5 + b'_5 + b_2 s_{135} + b'_2 s'_{135} \\
q_{132}(s) + q_{132}(s') & = & b_5 s_{133} + b'_5 s'_{133} + b_2 s_{137} + b'_2 s'_{137} \\
q_{133}(s) + q_{133}(s') & = & b_5 s_{135} + b'_5 s'_{135} + b_2 + b'_2 \\
q_{134}(s) + q_{134}(s') & = & b_5 + b'_5 + b_2 s_{135} + b'_2 s'_{135} \\
q_{135}(s) + q_{135}(s') & = & b_2 s_{137} + b'_2 s'_{137} \\
q_{136}(s) + q_{136}(s') & = & b_2 + b'_2
\end{cases}
$$

SUBTERRANEAN 2.0 : SYSTEM FOR COLLISIONS

$$
\left\{
\begin{array}{rcl}
q'_{124}(s) + q'_{124}(s') & = & 0 \\
q'_{125}(s) + q'_{125}(s') & = & 0 \\
q'_{126}(s) + q'_{126}(s') & = & 0 \\
q'_{127}(s) + q'_{127}(s') & = & 0 \\
q'_{128}(s) + q'_{128}(s') & = & 0 \\
q'_{129}(s) + q'_{129}(s') & = & 0 \\
q'_{130}(s) + q'_{130}(s') & = & 0 \\
q'_{131}(s) + q'_{131}(s') & = & 0 \\
q'_{132}(s) + q'_{132}(s') & = & b_5 s_{133} + b'_5 s'_{133} \\
q'_{133}(s) + q'_{133}(s') & = & b_5 s_{135} + b'_5 s'_{135} \\
q_{134}(s) + q_{134}(s') & = & b_5 + b'_5 + b_2 s_{135} + b'_2 s'_{135} \\
q_{135}(s) + q_{135}(s') & = & b_2 s_{137} + b'_2 s'_{137} \\
q_{136}(s) + q_{136}(s') & = & b_2 + b'_2
\end{array}
\right.
$$

Subterranean 2.0 : System for collisions

$$\begin{cases} q'_{124}(s) + q'_{124}(s') & = & 0 \\ q'_{125}(s) + q'_{125}(s') & = & 0 \\ q'_{126}(s) + q'_{126}(s') & = & 0 \\ q'_{127}(s) + q'_{127}(s') & = & 0 \\ q'_{128}(s) + q'_{128}(s') & = & 0 \\ q'_{129}(s) + q'_{129}(s') & = & 0 \\ q'_{130}(s) + q'_{130}(s') & = & 0 \\ q'_{131}(s) + q'_{131}(s') & = & 0 \\ q'_{132}(s) + q'_{132}(s') & = & b_5 s_{133} + b'_5 s'_{133} \\ q'_{133}(s) + q'_{133}(s') & = & b_5 s_{135} + b'_5 s'_{135} \\ q_{134}(s) + q_{134}(s') & = & b_5 + b'_5 + b_2 s_{135} + b'_2 s'_{135} \\ q_{135}(s) + q_{135}(s') & = & b_2 s_{137} + b'_2 s'_{137} \\ q_{136}(s) + q_{136}(s') & = & b_2 + b'_2 \end{cases}$$

Attack in 2^{116}

Subterranean 2.0 : Summary and future work

Summary :

- `Subterranean 2.0`: hardware efficiency and still secure

SUBTERRANEAN 2.0 : SUMMARY AND FUTURE WORK

Summary :

- `Subterranean 2.0`: hardware efficiency and still secure
- Techniques can be applied to other constructions (e.g. to KECCAK, Heim Boissier and R in 2020)

Subterranean 2.0 : Summary and future work

Summary :

- `Subterranean 2.0`: hardware efficiency and still secure
- Techniques can be applied to other constructions (e.g. to Keccak, Heim Boissier and R in 2020)

On the existence of biases between consecutive blocks :

SUBTERRANEAN 2.0 : SUMMARY AND FUTURE WORK

Summary :

- `Subterranean 2.0`: hardware efficiency and still secure
- Techniques can be applied to other constructions (e.g. to KECCAK, Heim Boissier and R in 2020)

On the existence of biases between consecutive blocks :

- Unexpected behavior found by Song, Tu, Shi and Hu in 2021

Subterranean 2.0 : Summary and future work

Summary :

- `Subterranean 2.0`: hardware efficiency and still secure
- Techniques can be applied to other constructions (e.g. to Keccak, Heim Boissier and R in 2020)

On the existence of biases between consecutive blocks :

- Unexpected behavior found by Song, Tu, Shi and Hu in 2021
- `Panther` design (Bhargavi, Srinivasan, Lakshmy, 2021), broken by Boura, Heim Boissier and R. in 2022

SUBTERRANEAN 2.0 : SUMMARY AND FUTURE WORK

Summary :

- `Subterranean 2.0`: hardware efficiency and still secure
- Techniques can be applied to other constructions (e.g. to KECCAK, Heim Boissier and R in 2020)

On the existence of biases between consecutive blocks :

- Unexpected behavior found by Song, Tu, Shi and Hu in 2021
- `Panther` design (Bhargavi, Srinivasan, Lakshmy, 2021), broken by Boura, Heim Boissier and R. in 2022
- Absence proven in `Transistor`, Baudrin, Belaïd, Bon, Boura, Canteaut, Leurent, Paillier, Perrin, Rivain and R.

Algebraic and Higher-Order Differential Cryptanalysis of Pyjamask-96

Dobraunig, R. and Schoone

ToSC 2020

Higher-order derivatives : Principle

- Dates back to 1994 by Lai and Knudsen

Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$.

Higher-order derivatives : Principle

- Dates back to 1994 by Lai and Knudsen

Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$.
For any linear space V, with $\dim(V) \geq \deg(f) + 1$,

$$g : x \mapsto \sum_{v \in V} f(x+v) = 0$$

Higher-order derivatives : Principle

- Dates back to 1994 by Lai and Knudsen

Let $f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$.
For any linear space V, with $\dim(V) \geq \deg(f) + 1$,

$$g : x \mapsto \sum_{v \in V} f(x+v) = 0$$

Bounds on the degree :

Higher-order derivatives : Principle

- Dates back to 1994 by Lai and Knudsen

Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$.
For any linear space V, with $\dim(V) \geq \deg(f) + 1$,

$$g : x \mapsto \sum_{v \in V} f(x+v) = 0$$

Bounds on the degree :

- Trivial Upper bound : $\deg(F \circ G) \leq \deg(F) \times \deg(G)$

Higher-order derivatives : Principle

- Dates back to 1994 by Lai and Knudsen

Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$.
For any linear space V, with $\dim(V) \geq \deg(f) + 1$,

$$g : x \mapsto \sum_{v \in V} f(x+v) = 0$$

Bounds on the degree :

- Trivial Upper bound : $\deg(F \circ G) \leq \deg(F) \times \deg(G)$
- Upper bound by Boura, Canteaut and De Cannière in 2011

HIGHER-ORDER DERIVATIVES : PRINCIPLE

- Dates back to 1994 by Lai and Knudsen

Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$.
For any linear space V, with $\dim(V) \geq \deg(f) + 1$,

$$g : x \mapsto \sum_{v \in V} f(x+v) = 0$$

Bounds on the degree :

- Trivial Upper bound : $\deg(F \circ G) \leq \deg(F) \times \deg(G)$
- Upper bound by Boura, Canteaut and De Cannière in 2011
- Lower bounds by Hebborn, Lambin, Leander, Todo in 2020

Higher-order derivatives : Principle

- Dates back to 1994 by Lai and Knudsen

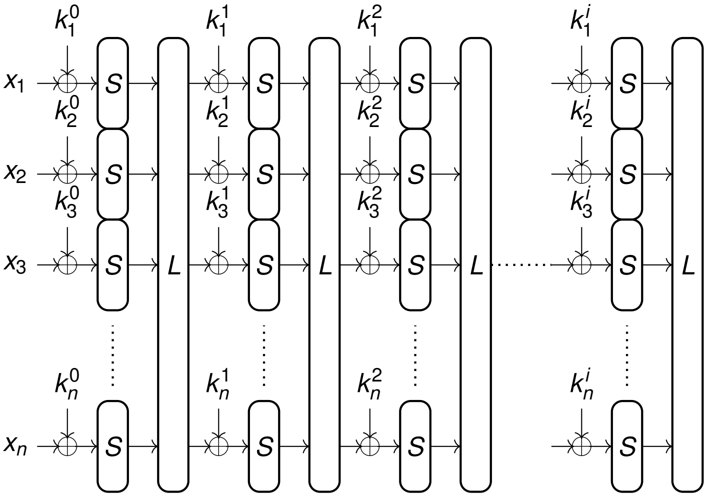
Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$.
For any linear space V, with $\dim(V) \geq \deg(f) + 1$,

$$g : x \mapsto \sum_{v \in V} f(x + v) = 0$$

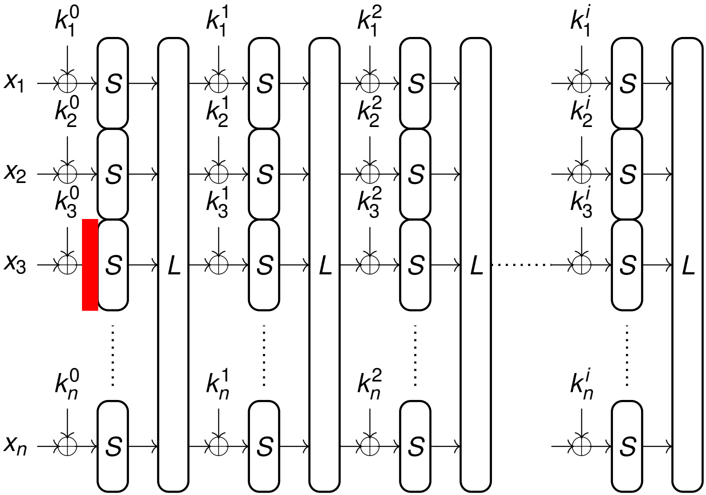
Bounds on the degree :

- Trivial Upper bound : $\deg(F \circ G) \leq \deg(F) \times \deg(G)$
- Upper bound by Boura, Canteaut and De Cannière in 2011
- Lower bounds by Hebborn, Lambin, Leander, Todo in 2020
- Division property, many improvements since Todo in 2015

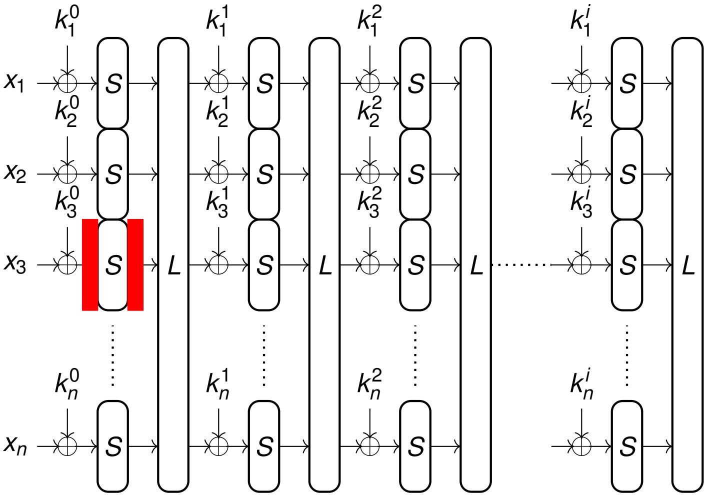
Higher-order differential attacks



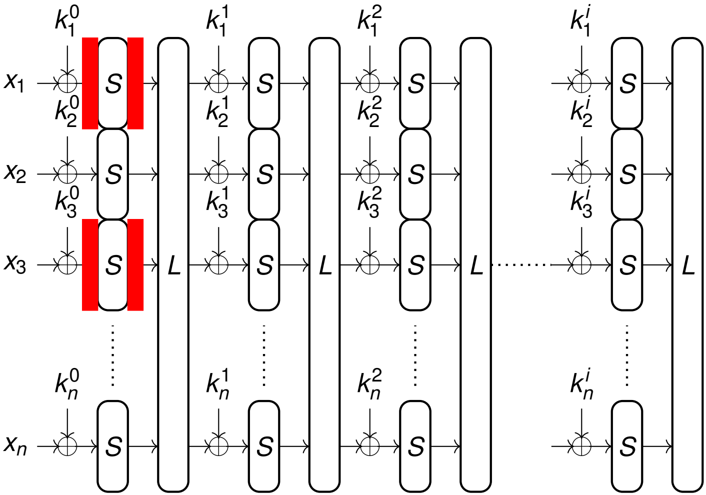
Higher-order differential attacks



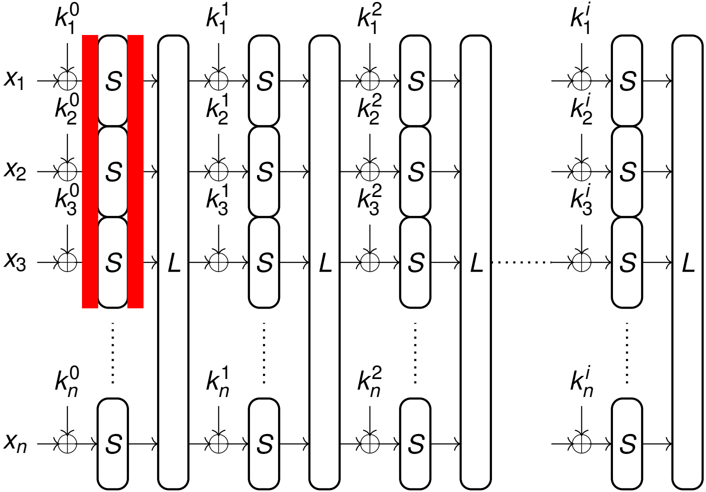
Higher-order differential attacks



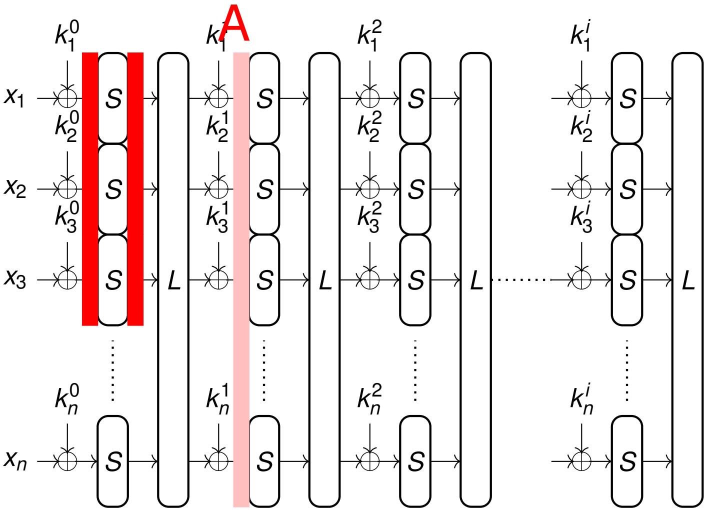
Higher-order differential attacks



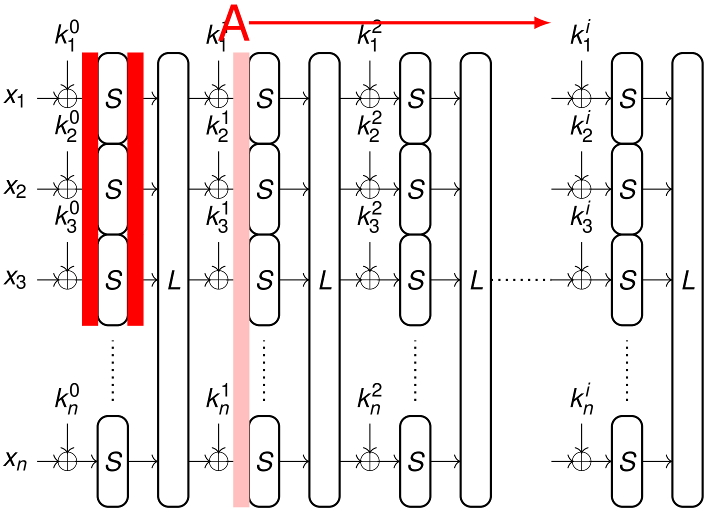
Higher-order differential attacks



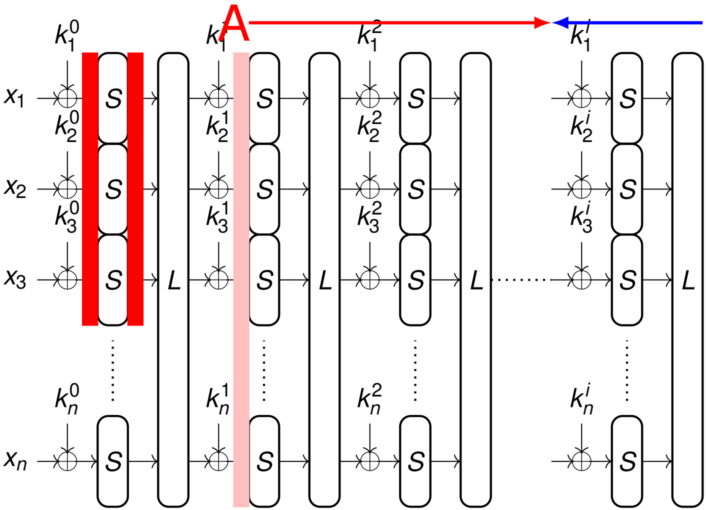
Higher-order differential attacks



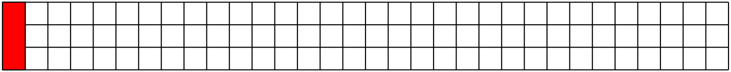
HIGHER-ORDER DIFFERENTIAL ATTACKS



Higher-order differential attacks



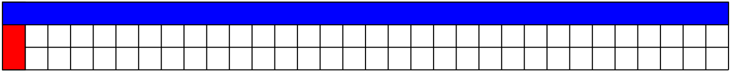
Pyjamask (Goudarzi, Jean, Kölbl, Peyrin, Rivain, Sasaki, Siang Meng Sim, NIST 2019)



The S-box layer :

- 96-bit : quadratic S-box on 3 bits
- 128-bit : S-box of degree 3, on 4 bits

PYJAMASK (GOUDARZI, JEAN, KÖLBL, PEYRIN, RIVAIN, SASAKI, SIANG MENG SIM, NIST 2019)



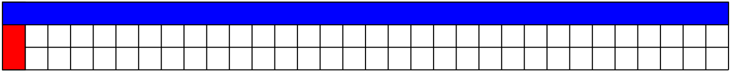
The S-box layer :

- 96-bit : quadratic S-box on 3 bits
- 128-bit : S-box of degree 3, on 4 bits

The linear layer :

- Circulant matrices of size 32 on each row

Pyjamask (Goudarzi, Jean, Kölbl, Peyrin, Rivain, Sasaki, Siang Meng Sim, NIST 2019)



The S-box layer :

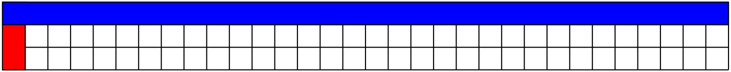
- 96-bit : quadratic S-box on 3 bits
- 128-bit : S-box of degree 3, on 4 bits

The linear layer :

- Circulant matrices of size 32 on each row

14 rounds

Pyjamask (Goudarzi, Jean, Kölbl, Peyrin, Rivain, Sasaki, Siang Meng Sim, NIST 2019)



The S-box layer :

- 96-bit : quadratic S-box on 3 bits
- 128-bit : S-box of degree 3, on 4 bits

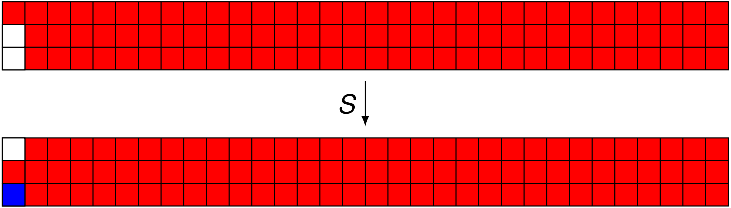
The linear layer :

- Circulant matrices of size 32 on each row

14 rounds

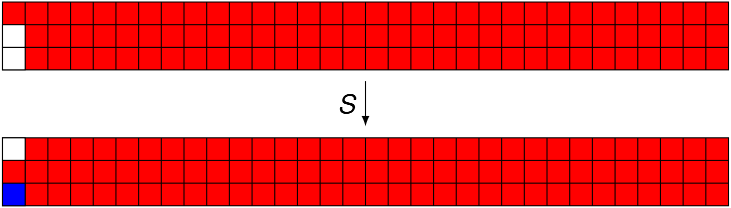
Round	1	2	3	4	5	6	7	8	9	10
96-bit	2	4	8	16	32	64	80	88	92	**94**
128-bit	3	9	27	81	112	122	126	127	127	127

PYJAMASK : 11 ROUNDS DISTINGUISHER



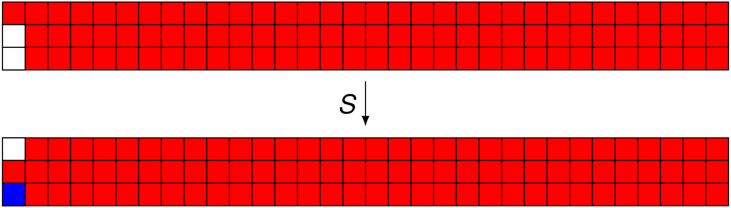
- Value in blue depends on three key bits (7 monomials)

PYJAMASK : 11 ROUNDS DISTINGUISHER



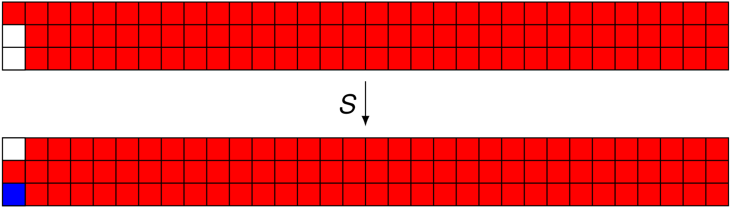
- Value in blue depends on three key bits (7 monomials)
- 7 possible directions and 32 S-boxes

PYJAMASK : 11 ROUNDS DISTINGUISHER



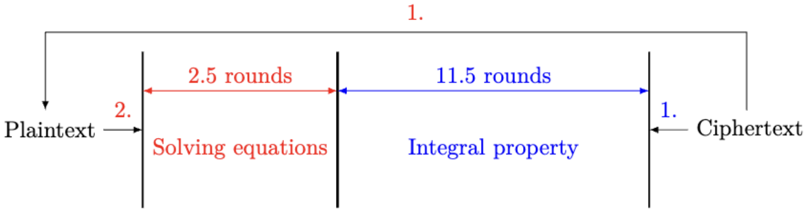
- Value in blue depends on three key bits (7 monomials)
- 7 possible directions and 32 S-boxes
- There is 3 possible shifts

PYJAMASK : 11 ROUNDS DISTINGUISHER

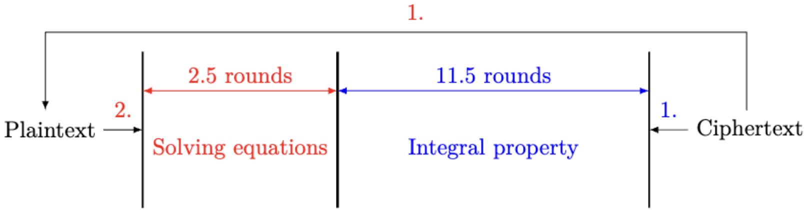


- Value in blue depends on three key bits (7 monomials)
- 7 possible directions and 32 S-boxes
- There is 3 possible shifts
- Gives $(3 \cdot 7 - 7) \cdot 32 = 448$ equations

PYJAMASK : THE DEVIL IS IN THE DETAILS

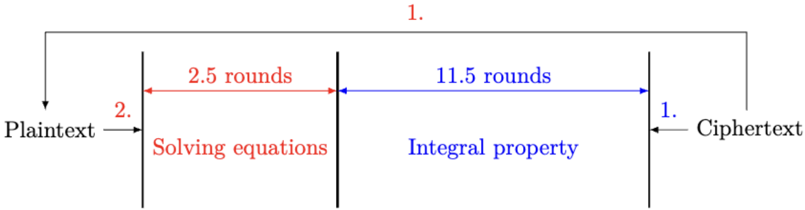


PYJAMASK : THE DEVIL IS IN THE DETAILS



$$\sum_{j=1}^{2^{94}} S \circ A_{k^{(2)}} \circ L \circ \widehat{S} \circ A_{k^{(1)}} \circ L \circ \widehat{S} \circ A_{k^{(0)}} \left(P^{(j)} \right)$$

PYJAMASK : THE DEVIL IS IN THE DETAILS



$$\sum_{j=1}^{2^{94}} S \circ A_{k^{(2)}} \circ L \circ \widehat{S} \circ A_{k^{(1)}} \circ L \circ \widehat{S} \circ A_{k^{(0)}} \left(P^{(j)} \right)$$

$$\sum_{i=1}^{8} \binom{96+128}{i} \approx 2^{47}$$

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\sum_{j=1}^{2^{94}} S \circ A_{K^{(2)}} \circ L \circ \widehat{S} \circ A_{K^{(1)}} \circ L \circ \widehat{S} \circ A_{K^{(0)}} \left(P^{(j)} \right)$$

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\sum_{j=1}^{2^{94}} S \circ A_{K^{(2)}} \circ L \circ \widehat{S} \circ A_{K^{(1)}} \circ L \circ \widehat{S} \circ A_{K^{(0)}} \left(P^{(j)} \right)$$

$$\begin{aligned} S(P+K)_1 &= (p_0+k_0)(p_1+k_1)+p_0+k_0+p_1+k_1+p_2+k_2 \\ &= S(P)_1+S(K)_1+p_0k_1+p_1k_0 \end{aligned}$$

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\sum_{j=1}^{2^{94}} S \circ A_{K^{(2)}} \circ L \circ \widehat{S} \circ A_{K^{(1)}} \circ L \circ \widehat{S} \circ A_{K^{(0)}} \left(P^{(j)} \right)$$

$$\begin{aligned} S(P+K)_1 &= (p_0+k_0)(p_1+k_1) + p_0 + k_0 + p_1 + k_1 + p_2 + k_2 \\ &= S(P)_1 + S(K)_1 + p_0 k_1 + p_1 k_0 \end{aligned}$$

$$A_{K^1} \circ L \circ S(P+K_0) = L \circ S(P) + L \circ S(K^0) + K^1 + \sum_{i \in I} p_i k_{f(i)} + p_{f(i)} k_i$$

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\sum_{j=1}^{2^{94}} S \circ A_{K^{(2)}} \circ L \circ \widehat{S} \circ A_{K^{(1)}} \circ L \circ \widehat{S} \circ A_{K^{(0)}} \left(P^{(j)} \right)$$

$$\begin{aligned} S(P+K)_1 &= (p_0+k_0)(p_1+k_1)+p_0+k_0+p_1+k_1+p_2+k_2 \\ &= S(P)_1+S(K)_1+p_0k_1+p_1k_0 \end{aligned}$$

$$A_{K^1} \circ L \circ S(P+K_0) = L \circ S(P) + L \circ S(K^0) + K^1 + \sum_{i \in I} p_i k_{f(i)} + p_{f(i)} k_i$$

$$\kappa := L \circ S(K^0) + K^1$$

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\sum_{j=1}^{2^{94}} S \circ A_{K^{(2)}} \circ L \circ \widehat{S} \circ A_{K^{(1)}} \circ L \circ \widehat{S} \circ A_{K^{(0)}} \left(P^{(j)} \right)$$

$$\begin{aligned} S(P+K)_1 &= (p_0+k_0)(p_1+k_1) + p_0 + k_0 + p_1 + k_1 + p_2 + k_2 \\ &= S(P)_1 + S(K)_1 + p_0 k_1 + p_1 k_0 \end{aligned}$$

$$A_{K^1} \circ L \circ S(P+K_0) = L \circ S(P) + L \circ S(K^0) + K^1 + \sum_{i \in I} p_i k_{f(i)} + p_{f(i)} k_i$$

$$\kappa := L \circ S(K^0) + K^1$$

$$\pi := L \circ S(P)$$

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\kappa := L \circ S(K^0) + K^1$$

$$\pi := L \circ S(P)$$

$$\sum_{j=1}^{2^{94}} S \circ A_{k^{(2)}} \circ L \circ \widehat{S} \circ A_{k^{(1)}} \circ L \circ \widehat{S} \circ A_{k^{(0)}} \left(P^{(j)} \right)$$

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\kappa := L \circ S(K^0) + K^1$$

$$\pi := L \circ S(P)$$

$$\sum_{j=1}^{2^{94}} S \circ A_{k^{(2)}} \circ L \circ \widehat{S} \circ A_{k^{(1)}} \circ L \circ \widehat{S} \circ A_{k^{(0)}} \left(P^{(j)}\right)$$

$$= \sum_{p \in \mathcal{P}} \sum_{(u,u',v,v') \in (\mathbb{F}_2^n)^4} p^u \pi^{u'} k^v \kappa^{v'}$$

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\kappa := L \circ S(K^0) + K^1$$

$$\pi := L \circ S(P)$$

$$\sum_{j=1}^{2^{94}} S \circ A_{k^{(2)}} \circ L \circ \widehat{S} \circ A_{k^{(1)}} \circ L \circ \widehat{S} \circ A_{k^{(0)}} \left(P^{(j)}\right)$$

$$= \sum_{p \in \mathcal{P}} \sum_{(u,u',v,v') \in (\mathbb{F}_2^n)^4} p^u \pi^{u'} k^v \kappa^{v'}$$

N_{total}	N_{eval}	$N_{solving}$	$N_{keybits}$
7642713	3910569	3829480	154

PYJAMASK : THE DEVIL IS IN THE DETAILS

$$\kappa := L \circ S(K^0) + K^1$$

$$\pi := L \circ S(P)$$

$$\sum_{j=1}^{2^{94}} S \circ A_{k^{(2)}} \circ L \circ \widehat{S} \circ A_{k^{(1)}} \circ L \circ \widehat{S} \circ A_{k^{(0)}} \left(P^{(j)}\right)$$

$$= \sum_{p \in \mathcal{P}} \sum_{(u,u',v,v') \in (\mathbb{F}_2^n)^4} p^u \pi^{u'} k^v \kappa^{v'}$$

N_{total}	N_{eval}	$N_{solving}$	$N_{keybits}$
7 642 713	3 910 569	3 829 480	154

Cost of the attack : 2^{114}

PYJAMASK : SUMMARY AND FUTURE WORK

Summary :

- Time : 2^{114}

Pyjamask : Summary and future work

Summary :

- Time : 2^{114}
- Data : 2^{96}

PYJAMASK : SUMMARY AND FUTURE WORK

Summary :

- Time : 2^{114}
- Data : 2^{96}
- `Pyjamask-AEAD` : 7 rounds only

PYJAMASK : SUMMARY AND FUTURE WORK

Summary :

- Time : 2^{114}
- Data : 2^{96}
- `Pyjamask-AEAD` : 7 rounds only

Improvements :

Pyjamask : Summary and future work

Summary :

- Time : 2^{114}
- Data : 2^{96}
- `Pyjamask-AEAD` : 7 rounds only

Improvements :

- Improvements by Cui, Hu, Wang and Wang in 2022

Pyjamask : Summary and future work

Summary :

- Time : 2^{114}
- Data : 2^{96}
- `Pyjamask-AEAD` : 7 rounds only

Improvements :

- Improvements by Cui, Hu, Wang and Wang in 2022
- Pass more than one round in the beginning

PYJAMASK : SUMMARY AND FUTURE WORK

Summary :

- Time : 2^{114}
- Data : 2^{96}
- `Pyjamask-AEAD` : 7 rounds only

Improvements :

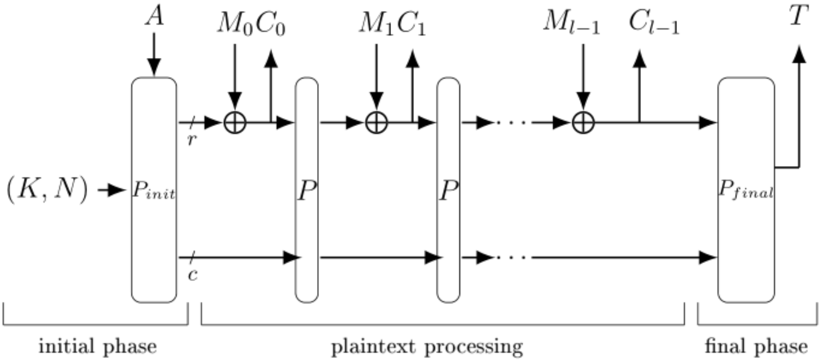
- Improvements by Cui, Hu, Wang and Wang in 2022
- Pass more than one round in the beginning
- Key dependency in the distinguisher

Generic Attack on Duplex-Based AEAD Modes Using Random Function Statistics

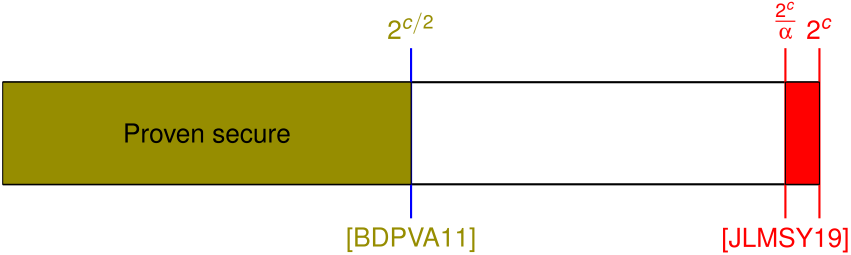
Gilbert, Heim Boissier, Khati, R.

EUROCRYPT 2023

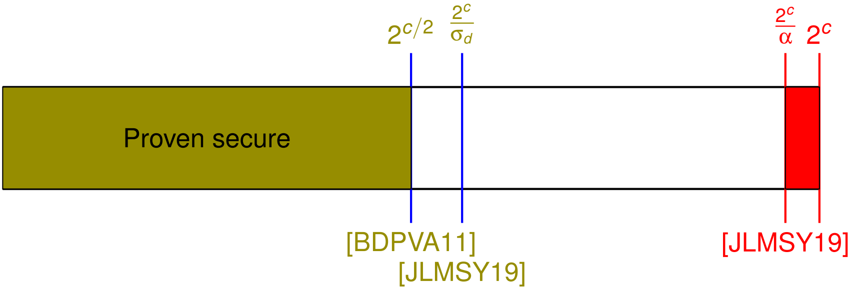
Duplex mode (Bertoni, Daemen, Peeters and Van Assche, 2012)



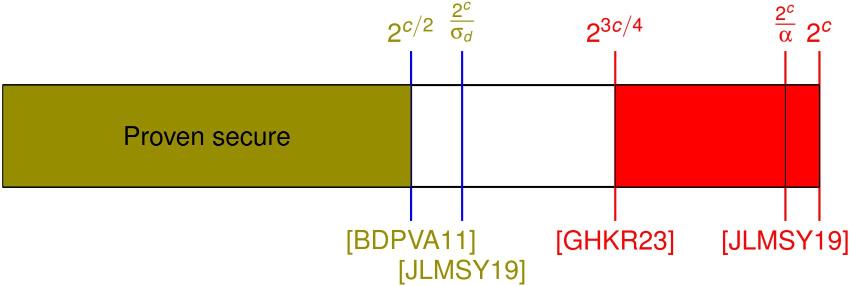
Duplex mode : Security



Duplex mode : Security



DUPLEX MODE : SECURITY



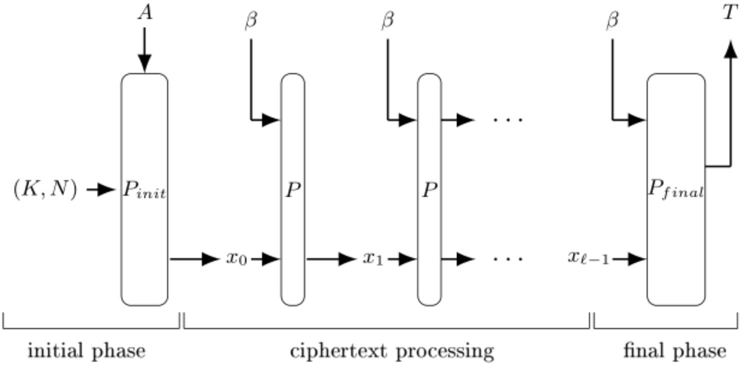
Duplex mode : Main observation

$C_\beta^\ell = \beta_\ell = \underbrace{\beta||\cdots||\beta}_{\ell}$:

Duplex mode : Main observation

$$C_\beta^\ell = \beta_\ell = \underbrace{\beta || \cdots || \beta}_{\ell} :$$

$$\begin{array}{rcccc} f_\beta & : & \mathbb{F}_2^c & \longrightarrow & \mathbb{F}_2^c \\ & & x & \longmapsto & \lfloor P(\beta || x) \rfloor_c \end{array}$$



Definition : Exceptional function

An exceptional function $f : X \rightarrow X$ is said exceptional if its largest component is big and its cycle is small.

Duplex mode : Forgery attack

Definition : Exceptional function
An exceptional function $f : X \to X$ is said exceptional if its largest component is big and its cycle is small.

Precomputation : find β such that f_β is exceptional.

Duplex mode : Forgery attack

Definition : Exceptional function
An exceptional function $f : X \to X$ is said exceptional if its largest component is big and its cycle is small.

Precomputation : find β such that f_β is exceptional.

Online : (N, A, C, T) with N, A arbitrary and $C = C_\beta^\ell$ with $\ell = \lambda 2^{\frac{c}{2}}$, with T computed from cycle points of the function.

DUPLEX MODE : FORGERY ATTACK

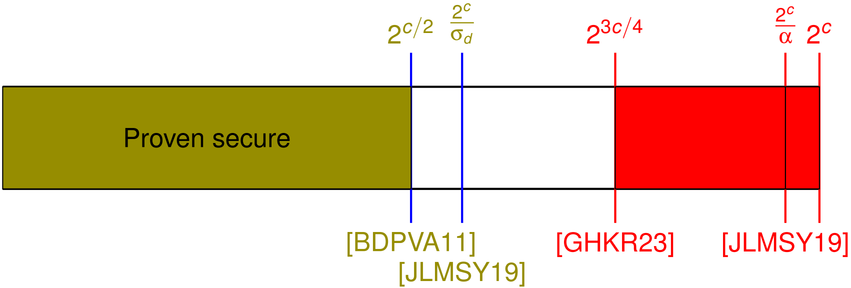
Definition : Exceptional function
An exceptional function $f : X \to X$ is said exceptional if its largest component is big and its cycle is small.

Precomputation : find β such that f_β is exceptional.

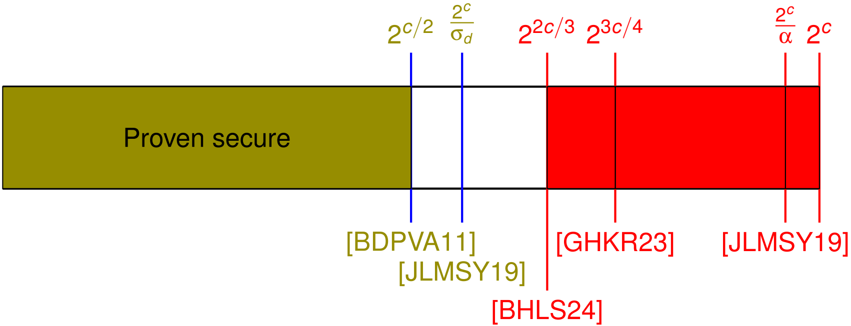
Online : (N, A, C, T) with N, A arbitrary and $C = C_\beta^\ell$ with $\ell = \lambda 2^{\frac{c}{2}}$, with T computed from cycle points of the function.

Forgery attack in $O(2^{\frac{3c}{4}})$

DUPLEX MODE : IMPROVEMENTS



Duplex mode : Improvements



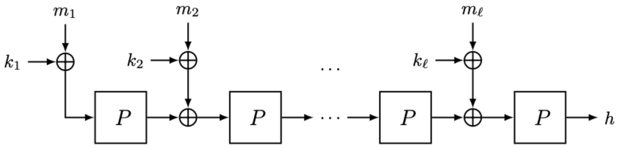
On the Security of Keyed Hashing Based on Public Permutations

Fuchs, R., Daemen

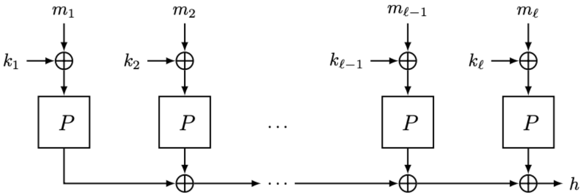
CRYPTO 2023

Comparison of keyed compression functions

The serial construction :

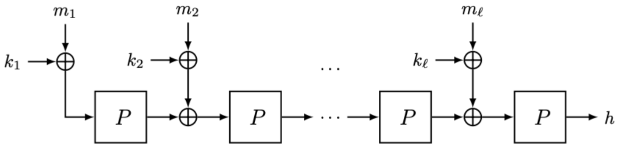


The parallel construction :

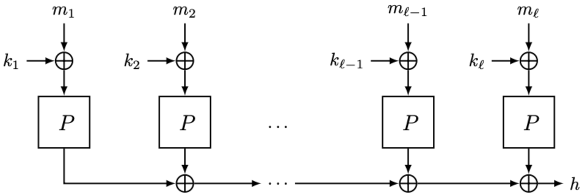


Comparison of keyed compression functions

The serial construction :



The parallel construction :



Security model : the attacker only gets $E_{K'}(h)$.

Comparison of keyed compression functions

The relevant security criteria are

- The differential uniformity for the serial case :

$$\mathrm{MDP}_f = \max_{a \neq 0, b} \mathrm{DP}_f(a, b)$$

- The maximum in norm two for the parallel case :

$$\mathrm{MNDP}_f = \max_{a \neq 0} \sum_b \mathrm{DP}_f^2(a, b)$$

Comparison of keyed compression functions

The relevant security criteria are

- The differential uniformity for the serial case :

$$\mathrm{MDP}_f = \max_{a \neq 0, b} \mathrm{DP}_f(a, b)$$

- The maximum in norm two for the parallel case :

$$\mathrm{MNDP}_f = \max_{a \neq 0} \sum_b \mathrm{DP}_f^2(a, b)$$

And

$$\mathrm{MDP}_f \geq \mathrm{MNDP}_f$$

KEYED HASHING : SUMMARY AND FUTURE WORK

Summary :

- The parallel construction provides better security
- Use affine spaces to obtain a quadratic gain

Open question :

- How to estimate accurately the MDP_f and the MNDP_f ?

Conclusion and Perspectives

Conclusion and perspectives

Cryptanalysis with polynomials :

- Collisions, key-recovery and distinguisher

Conclusion and perspectives

Cryptanalysis with polynomials :

- Collisions, key-recovery and distinguisher
- Find more fine-grained information than the degree

Cryptanalysis with polynomials :

- Collisions, key-recovery and distinguisher
- Find more fine-grained information than the degree
- Algorithms for analyzing the ANF

Conclusion and perspectives

Cryptanalysis with polynomials :

- Collisions, key-recovery and distinguisher
- Find more fine-grained information than the degree
- Algorithms for analyzing the ANF
- What about other polynomial representations ?

Conclusion and perspectives

Cryptanalysis with polynomials :

- Collisions, key-recovery and distinguisher
- Find more fine-grained information than the degree
- Algorithms for analyzing the ANF
- What about other polynomial representations ?

With a focus on :

Conclusion and perspectives

Cryptanalysis with polynomials :

- Collisions, key-recovery and distinguisher
- Find more fine-grained information than the degree
- Algorithms for analyzing the ANF
- What about other polynomial representations ?

With a focus on :

- Low-data cryptanalysis

Conclusion and perspectives

Cryptanalysis with polynomials :

- Collisions, key-recovery and distinguisher
- Find more fine-grained information than the degree
- Algorithms for analyzing the ANF
- What about other polynomial representations ?

With a focus on :

- Low-data cryptanalysis
- For reproductibility of cryptanalysis results

Conclusion and perspectives

Cryptanalysis with polynomials :

- Collisions, key-recovery and distinguisher
- Find more fine-grained information than the degree
- Algorithms for analyzing the ANF
- What about other polynomial representations ?

With a focus on :

- Low-data cryptanalysis
- For reproductibility of cryptanalysis results
- Constrained models such as WPRFs

1. We need cryptography.

1. We need cryptography.
2. We need cryptanalysis.

1. We need cryptography.
2. We need cryptanalysis.
3. Cryptanalysis is fun!

1. We need cryptography.
2. We need cryptanalysis.
3. Cryptanalysis is fun !

I do cryptanalysis