Improved Differential and Linear Trail Bounds for Ascon

Solane El Hirch, Silvia Mella, Alireza Mehrdad, Joan Daemen
Radboud University (The Netherlands)

Fast Software Encryption
March 20-24, 2023

Overview

Differential trails

ASCON-p

Scanning the space of trails in ASCON-p

Improved bounds for ASCON

Conclusion

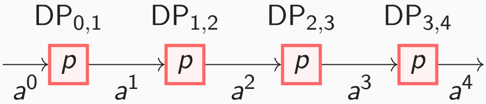
Goal of this work

Techniques that exploit the structure of the linear and non-linear layers in Ascon to improve the bounds of differential and linear trails

In this presentation we focus on differential cryptanalysis

Differential trails

Differential trails in iterated mapping



- r-round trail: $Q = (a^0, a^1, \ldots, a^r)$
- $\mathrm{DP}(Q)$: fraction of all input pairs with difference a^0 that exhibit a^i for $i \leq r$

$$\mathrm{DP}(Q) \approx \mathrm{DP}_p(a^0, a^1) \cdot \mathrm{DP}_p(a^1, a^2) \cdot \cdots \cdot \mathrm{DP}_p(a^{r-1}, a^r)$$

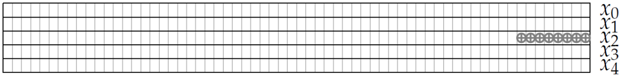
- a^i and a^{i+1} are compatible over p if $\mathrm{DP}_p(a^i, a^{i+1}) > 0$
- The weight of a trail Q is defined as

$$w(Q) = \sum_{i=1}^{r} w(a^{i-1}, a^i), \text{ where } w(a^{i-1}, a^i) = -\log_2(\mathrm{DP}_p(a^{i-1}, a^i))$$

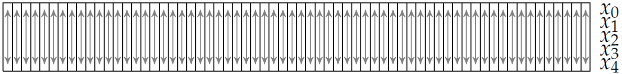
Ascon-p

Ascon-p round transformation

- State of five 64-bit rows $x_0, \cdots, x_4$
- Round transformation $p = p_L \circ p_S \circ p_C$



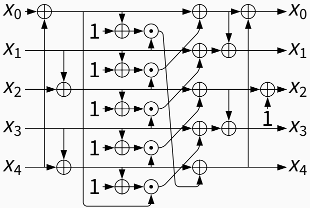
(a) Round constant addition p_C



(b) Substitution layer p_S with 5-bit S-box $\mathcal{S}(x)$

(c) Linear layer with 64-bit diffusion functions $\Sigma_i(x_i)$

(a) 5-bit S-box $\mathcal{S}$ in p_S: χ_5 of Keccak-f [BDPV11] and two mixing steps



(b) Mixing layer p_L:

$$
\begin{aligned}
x_0 &\leftarrow x_0 \oplus (x_0 \ggg 19) \oplus (x_0 \ggg 28)\\
x_1 &\leftarrow x_1 \oplus (x_1 \ggg 61) \oplus (x_1 \ggg 39)\\
x_2 &\leftarrow x_2 \oplus (x_2 \ggg 1) \oplus (x_2 \ggg 6)\\
x_3 &\leftarrow x_3 \oplus (x_3 \ggg 10) \oplus (x_3 \ggg 17)\\
x_4 &\leftarrow x_4 \oplus (x_4 \ggg 7) \oplus (x_4 \ggg 41)
\end{aligned}
$$

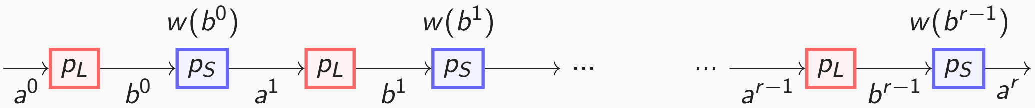
Differential trails in Ascon-p

Rephased round function p: linear layer p_L followed by a non-linear layer p_S

- Trails notation including differences between p_L and p_S:
  $Q = (a^0, b^0, a^1, \cdots, a^{r-1}, b^{r-1}, a^r)$
- $b^i = p_L(a^i)$
- p_L linear $\Rightarrow DP_{p_L}(a^i, b^i) = 1$ and $w_{p_L}(a^i, b^i) = 0$
- Weight of Q

$$w(Q) = \sum_{i=1}^{r} w_{p_S}(b^{i-1}, a^i)$$

Propagation properties of $\mathcal{S}$



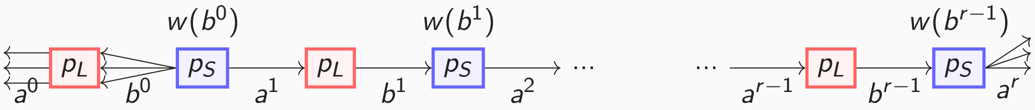
S-box $\mathcal{S}$ based on χ_5 mapping:

- p_S has algebraic degree 2
- The weight of (b^{i-1}, a^i) only depends on $b^{i-1} \rightarrow w_{p_S}(b^{i-1}, a^i) = w_{p_S}(b^{i-1})$
- For an input difference b, the set of compatible differences

$$\mathcal{A}(b) = \{a \in \mathbb{F}_2^5 \colon \exists x \in \mathbb{F}_2^5 \text{ s.t. } \mathcal{S}(x) \oplus \mathcal{S}(x \oplus b) = a\}$$

  is an affine space
- The dimension of the affine space $\mathcal{A}(b)$ is $w_{p_S}(b)$

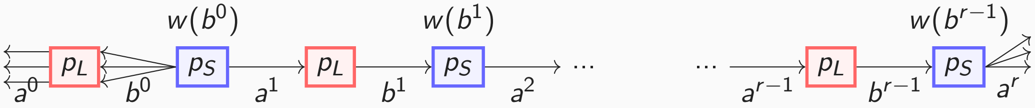
About trail extension



Extension of a trail by one round

- **Forward**: build all trails by appending (b^{r-1}, a^r)
- All trails that share the same differences except a^r have the same weight

$\rightarrow$ **no need to build them to know their weight**

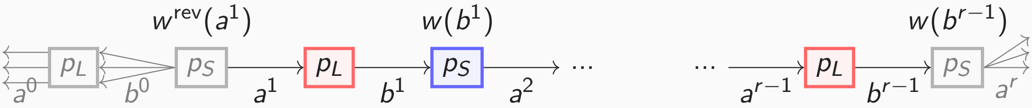


Extension of a trail by one round

- **Backward**: build all trails by prepending (a^0, b^0)
- All trails that share the same differences except (a^0, b^0) don't have the same weight **but** we can easily lower bound their weight

$\rightarrow$ **no need to build them to bound their weight**

Trail cores and minimum reverse weight [DV12]



- Minimum reverse weight:

$$w^{\text{rev}}(a^1) := \min_{b^0} w(b_0)$$

- Trail core: set of trails with same intermediate differences and whose weight is lower bounded by

$$w^{\textbf{rev}}(a^1) + w(b^1) + \ldots + w(b^{r-1})$$

Scanning the space of trails in Ascon-p

Trail search strategy [DV12]

- We can restrict the search to trail cores $\rightarrow$ instead of trails
- Build all r-round trail cores below a target weight T_r
  - If one or more trail cores are found, then the minimum weight among them defines a **tight bound** on the weight of all r-round trails
  - Otherwise, T_r is a **bound** on the weight of all r-round trails
- Start from shorter trail cores and extend

Generating 4 and 8-round trail cores

- Any 4-round trail core of weight $w^{\text{rev}}(a_1) + w(b^1) + w(b^2) + w(b^3) \leq 80$ has
  - $w^{\text{rev}}(a_1) + w(b^1) \leq \frac{80}{2}$ or
  - $w(b^2) + w(b^3) \leq \frac{80}{2}$

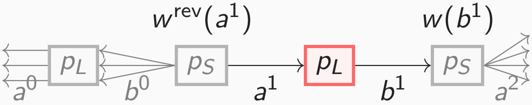
Generating 4 and 8-round trail cores

- Any 4-round trail core of weight $w^{\text{rev}}(a_1) + w(b^1) + w(b^2) + w(b^3) \leq 80$ has
  - $w^{\text{rev}}(a_1) + w(b^1) \leq \frac{80}{2}$ or
  - $w(b^2) + w(b^3) \leq \frac{80}{2}$
- Therefore, any 4-round trail core of weight ≤ 80 can be generated by
  - building all 2-round trail cores with $w^{\text{rev}}(a) + w(b) \leq 40$ and
  - extending them by 2 rounds in the forward and backward direction

Generating 4 and 8-round trail cores

- Any 4-round trail core of weight $w^{\text{rev}}(a_1) + w(b^1) + w(b^2) + w(b^3) \leq 80$ has
  - $w^{\text{rev}}(a_1) + w(b^1) \leq \frac{80}{2}$ or
  - $w(b^2) + w(b^3) \leq \frac{80}{2}$
- Therefore, any 4-round trail core of weight ≤ 80 can be generated by
  - building all 2-round trail cores with $w^{\text{rev}}(a) + w(b) \leq 40$ and
  - extending them by 2 rounds in the forward and backward direction
- Any 8-round trail core of weight ≤ 160 can be generated by
  - building all 4-round trail cores with weight $\leq \frac{160}{2}$ and
  - extending them by 4 rounds in the forward and backward direction

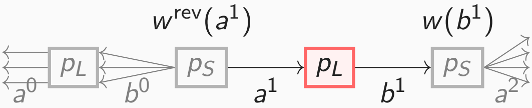
Generating 2-round trail cores as tree traversal



(a^1, b^1) built using the tree-based approach of [MDV17]

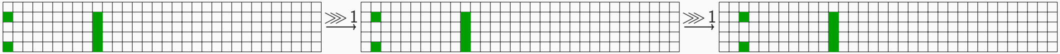
- Two-level tree
- Translation invariance along the horizontal axis
- Canonicity

Generating 2-round trail cores as tree traversal

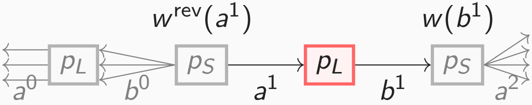


(a^1, b^1) built using the tree-based approach of [MDV17]

- Two-level tree
- Translation invariance along the horizontal axis
- Canonicity



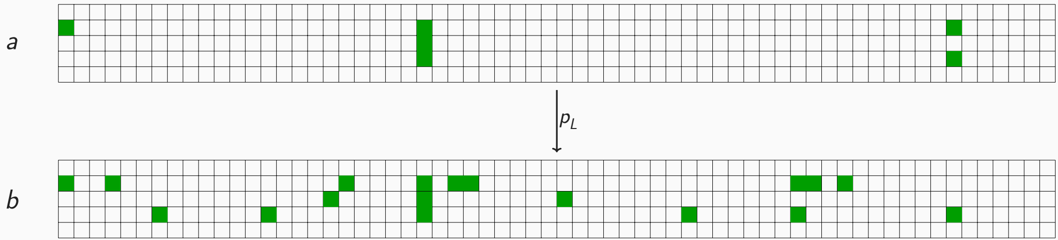
Generating 2-round trail cores as tree traversal



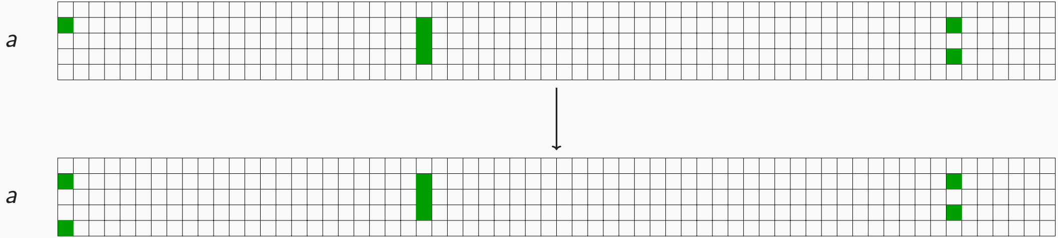
(a^1, b^1) built using the tree-based approach of [MDV17]

- Two-level tree
- Translation invariance along the horizontal axis
- Canonicity
- Score function
- Alternative representation of the linear layer

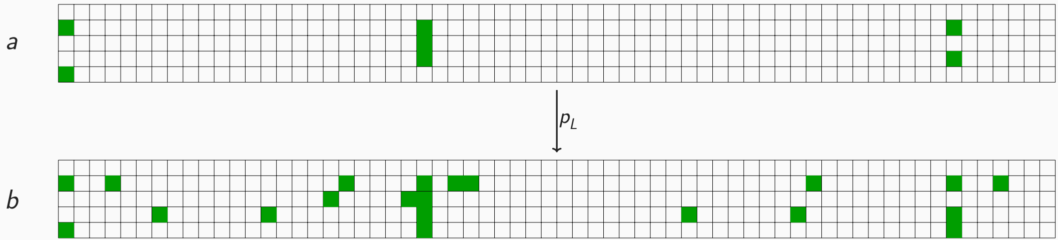
Traversing the 2-round trail cores tree



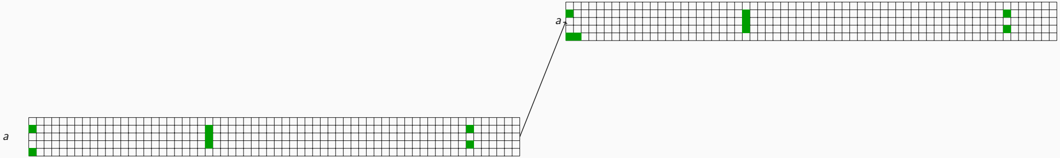
Traversing the 2-round trail cores tree



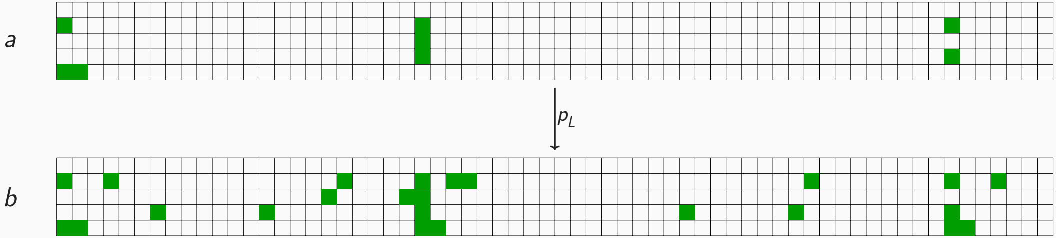
Traversing the 2-round trail cores tree



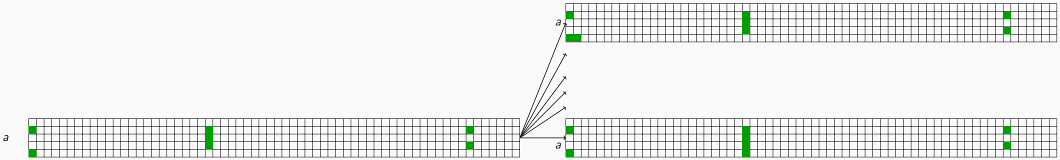
Traversing the 2-round trail cores tree



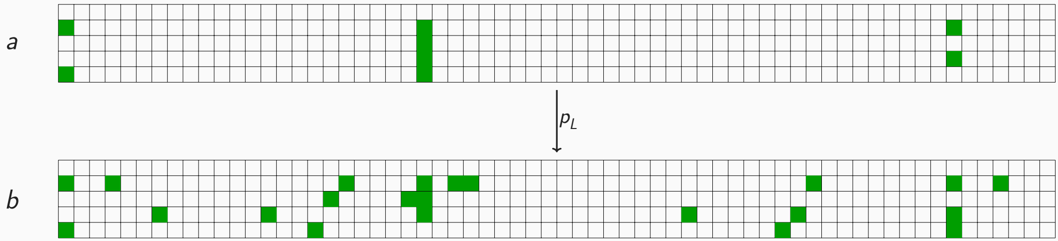
Traversing the 2-round trail cores tree



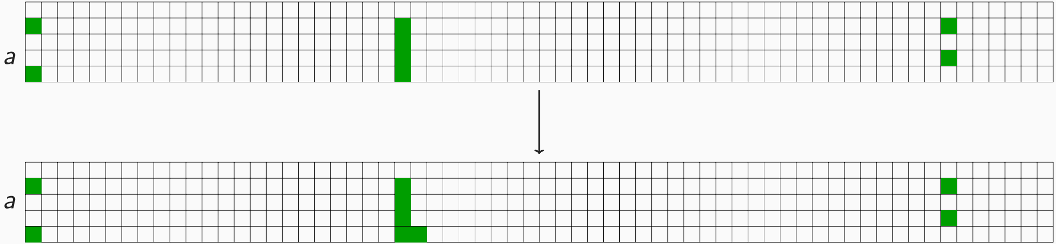
Traversing the 2-round trail cores tree



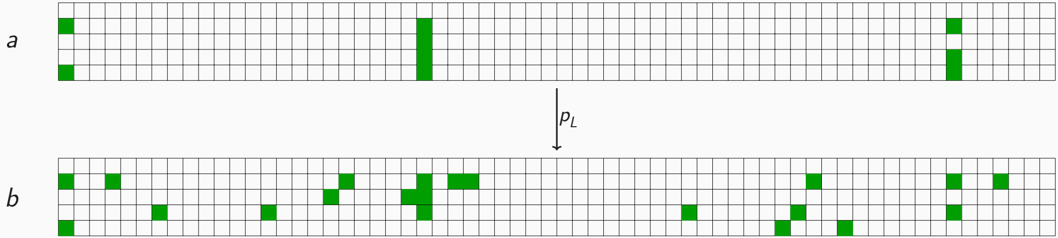
Traversing the 2-round trail cores tree



Traversing the 2-round trail cores tree



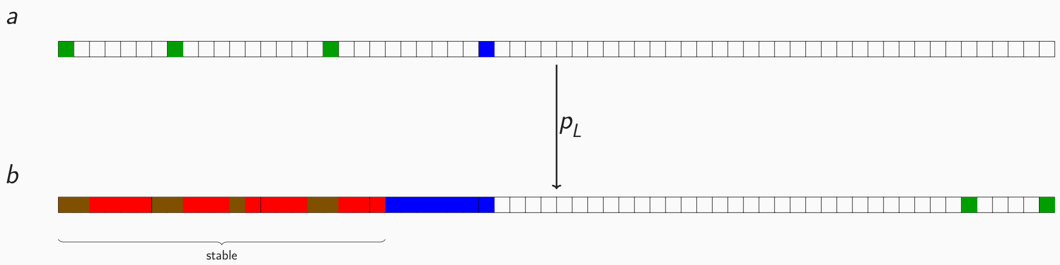
Pruning the tree: score function



Lower bound on the weight of a node and all its descendants

- ☞ Based on the number of active columns → each contribute at least 2
- ☞ Refining the score of b: bits in all active rows but the last one remain active
  - Consider their contribution to the weight
  - Minimum weight of all possible columns that can be obtained by adding bits in the last active row

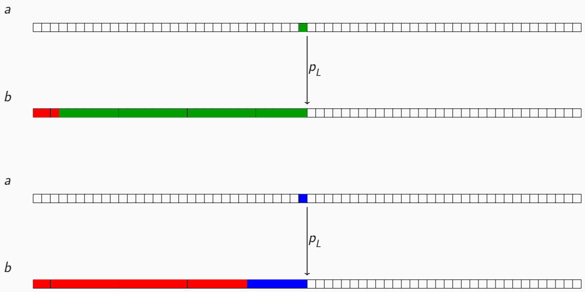
Pruning the tree: score function (continued)



☞ Score in the last active row: adding bits affect specific bit positions $\rightarrow$ for row 2 adding bit at position j affects bits in $[j - 6 \bmod 64, j]$ [MMGD22]

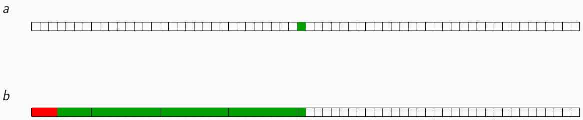
- Subterranean operates on a 257-bit states
- $\theta : x_i \leftarrow x_i \oplus (x_i \lll 3) \oplus (x_i \lll 8)$
- Stable bits: active bits present in all descendants of a node

Pruning the tree: the alternative row representation

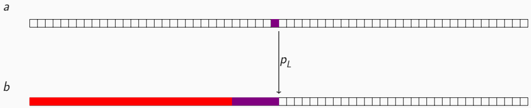


- $p_L : b_j \leftarrow a_j \oplus a_{j+s} \oplus a_{j+t}$
- $p_L : b_j \leftarrow a_j \oplus a_{j+19} \oplus a_{j+28} \rightarrow [0, 28]$
- $p_L : b_j \leftarrow a_j \oplus a_{j+1} \oplus a_{j+6} \rightarrow [0, 6]$

Pruning the tree: the alternative row representation

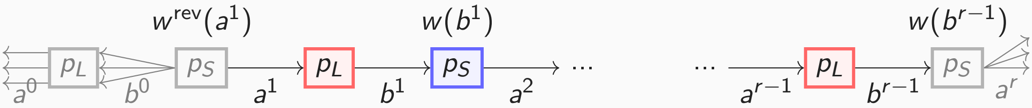


- $p_L : b_j \leftarrow a_j \oplus a_{j+s} \oplus a_{j+t}$
- Row 0: $[0, 28]$



- Alternative coordinate k:
  $k = j \cdot q \bmod 64$ with q odd
- $p'_L : b'_k \leftarrow a'_k \oplus a'_{k+sq} \oplus a'_{k+tq}$
- $p_L = \pi_{q^{-1}} \circ p'_L \circ \pi_q$ where
  $\pi_q(j) = q \cdot j \bmod 64$

☞ Bit positions at 0,1,5 for row 0:
$\rightarrow [0, 5]$



Build a^r (forward extension) or b^0 (backward extension) through a tree search

- p_L applies a linear function to each row independently: we can determine the bits that remain active at a^0 and b^r after p_L^{-1} and p_L respectively
- **Score:** lower bounds the weight of the $(r+1)$-round trail cores obtained

Lower bound on $w(b^r)$ while building a^r

- Approach used in XOODOO [DHVK18]
- Stable bits at b^r are represented by a stability mask $\mathcal{M}$ and $b^r \wedge \mathcal{M}$ gives:
  - The stable bits of b^r
  - The column of b^r active in all its descendants
  - Active column contribute at least 2 to the weight

Score: twice the number of active columns of $b^r \wedge \mathcal{M}$

Backward Extension: Score Function

- Two methods: **compatible patterns** (used for KECCAK-p) and **envelope space**
- Contribution of a^0: use stability mask $\mathcal{M}$ to determine the stable bits of a^0
- Lower bound on $w^{rev}(a^0) + w(b^0)$ while building b^0
- Functions score_a and score_b bound $w^{rev}(a^0)$ and $w(b^0)$ respectively

Score: $\text{score}_a + \text{score}_b$

1. Extension using **compatible patterns**
    a. score_b is computed as for KECCAK-p [DV12, MDV17]: the sum of the minimum weight of each column in b^0
    b. score_a based on the stable bits of a^0
    c. Effective method for small number of active columns in a^1

1. Extension using **compatible patterns**
2. Extension using the **envelope space**
   a. score_b is twice the number of active columns in a^1
   b. Build an envelope space for each active column position in a^1: envelope space $0 + \langle \mathsf{e}_0, \mathsf{e}_1, \mathsf{e}_2, \mathsf{e}_3, \mathsf{e}_4 \rangle$
   c. Envelope space $\mathcal{E}$: union of all these envelope space
   d. Scan $\mathcal{E}$ in a tree-based fashion: score_a based on the stable bits of a^0
   e. Effective method when many active columns in a^1

Improved bounds for Ascon

Improved bounds for Ascon

# Rounds	probability p of differential trails				squared correlation c^2 of linear trails			
	Bound	Best known	New bound	Time	Bound	Best known	New bound	Time
1	2^{-2}	2^{-2}			2^{-2}	2^{-2}		
2	2^{-8}	2^{-8}			2^{-8}	2^{-8}		
3	2^{-40}	2^{-40}			2^{-26}	2^{-28}	2^{-28}	$<$ 4sec
4	2^{-72}	2^{-107}	2^{-86}	13 days	2^{-72}	2^{-98}	2^{-88}	110 days
6	2^{-108}	2^{-305}	2^{-129}	+6 days	2^{-108}		2^{-132}	+21 days
8	2^{-144}		2^{-172}	+0	2^{-144}		2^{-176}	+0
12	2^{-216}		2^{-258}	+0	2^{-216}		2^{-264}	+0

Comparison with other works:

- This work: prove bound of 2^{-86} in 13 CPU days and 2^{-88} in 110 CPU days
- In [EME22] and [MR22]: cost estimation is 6688 CPU days and 3898 CPU days to prove the bound of 2^{-80}

Conclusion

Conclusion

- Dedicated tools for trail search for Ascon
- Proved the tight bound for 3-rounds for linear trails
- Improved bounds for differential and linear trails over 4, 6, 8, 12 rounds

Thank you for your attention!

Guido Bertoni, Joan Daemen, Michael Peeters, and Gilles Van Assche.
The keccak reference, January 2011.

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.

Joan Daemen and Gilles Van Assche.
Differential propagation analysis of keccak.
In Anne Canteaut, editor, *Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers*, volume 7549 of *Lecture Notes in Computer Science*, pages 422–441. Springer, 2012.

Johannes Erlacher, Florian Mendel, and Maria Eichlseder.
Bounds for the security of ascon against differential and linear cryptanalysis.

IACR Trans. Symmetric Cryptol., 2022(1):64–87, 2022.

Silvia Mella, Joan Daemen, and Gilles Van Assche.
New techniques for trail bounds and application to differential trails in keccak.
IACR Trans. Symmetric Cryptol., 2017(1):329–357, 2017.

Alireza Mehrdad, Silvia Mella, Lorenzo Grassi, and Joan Daemen.
Differential trail search in cryptographic primitives with big-circle chi - application to subterranean.
to appear in IACR Trans. Symmetric Cryptol., 2022.

Rusydi H. Makarim and Raghvendra Rohit.
Towards tight differential bounds of ascon: A hybrid usage of smt and milp.
IACR Transactions on Symmetric Cryptology, 2022(3):303–340, 2022.