Related-Key Differential Analysis of the AES

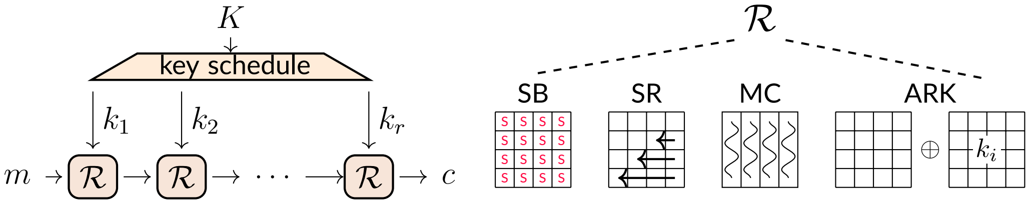
Christina Boura [1] Patrick Derbez [2] Margot Funk [1]

[1]Paris-Saclay University - Versailles University

[2]University of Rennes

March 28, 2024

The Advanced Encryption Standard

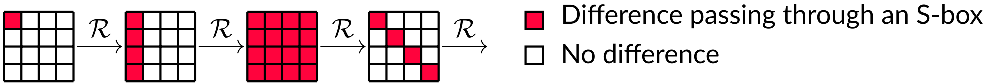


- Standardized in 2001.
- Block size: 128 bits (4×4 matrix of bytes).
- Key size: 128, 192, 256 bits.

Single-key model VS Related-key model

Single-key model

- **Simple** and **powerful** security proofs.
- At least 25 **active S-boxes** / 4 rounds.



4-round truncated differential trail of AES with 25 active S-boxes: $p \leq 2^{-25 \times 6}$

Single-key model VS Related-key model

Related-key model

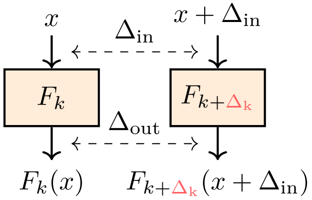
- Biryukov *et al.*, 2009

  ⇝ Related-key attacks on the full AES-192 and AES-256

- Other attacks on the full AES-192 and AES-256.

- Searching for optimal differential trails is more challenging.

related-key differential



Existing methods to find optimal RK differential trails for AES

Search for **truncated trails** and **instantiate** them.

Branch & Bound

Biryukov *et al.* (2010)

$|K| = 128$: several days
$|K| = 192$: several weeks
$|K| = 256$ ✗

Dynamic programming

Fouque *et al.* (2013)

$|K| = 128$: 30 min., 60 GB
$|K| = 192, 256$ ✗

Solver-based search (CP)

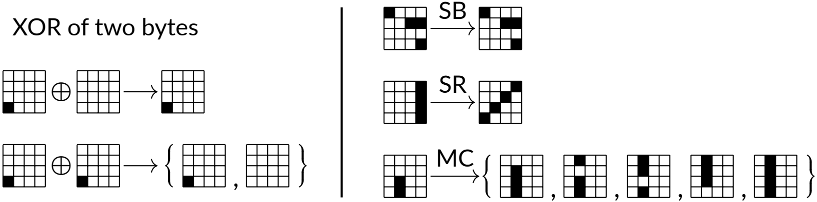
Gerault *et al.* (2018, 2020)
Rouquette *et al.* (2022)

$|K| = 128, 192, 256$ ✓
Fast and memory-efficient

AES differential truncated trails

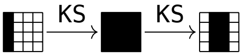
Modeling the AES truncated trails

Basic propagation rules ...



... do not necessarily lead to valid truncated trails.

Ex:



is not instantiable.

Invalidate some truncated trails

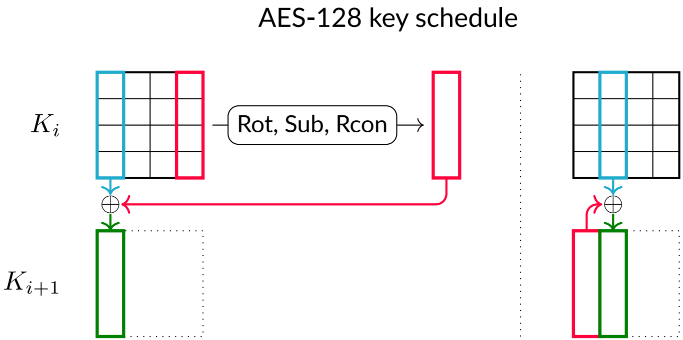
Linear equations $\rightsquigarrow$ Detect inconsistencies of the form $\blacksquare = \sum \square$.

In this work

A **"valid truncated trail"** means a **trail that is consistent with all linear equations** induced by the round function and the key schedule.

Easily checkable with a matrix in row echelon form.

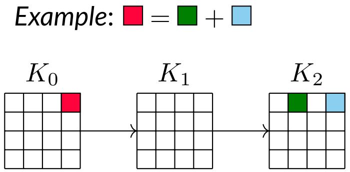
Invalidate some truncated trails



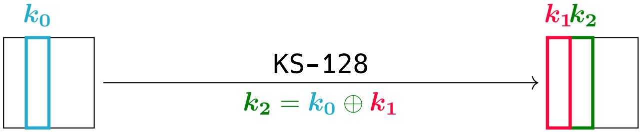
Invalidate some truncated trails

Key bridging

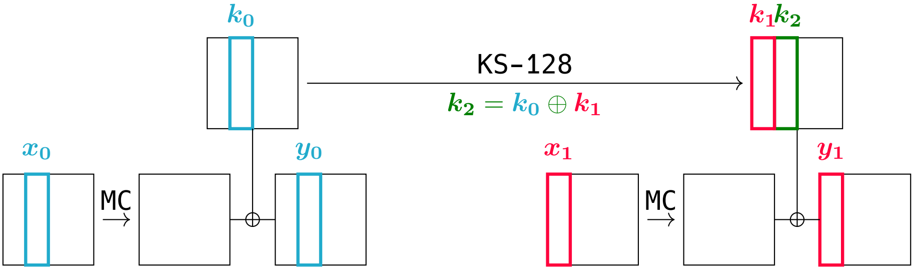
Derive **linear relations** between **distant subkeys**.



Invalidate some truncated trails

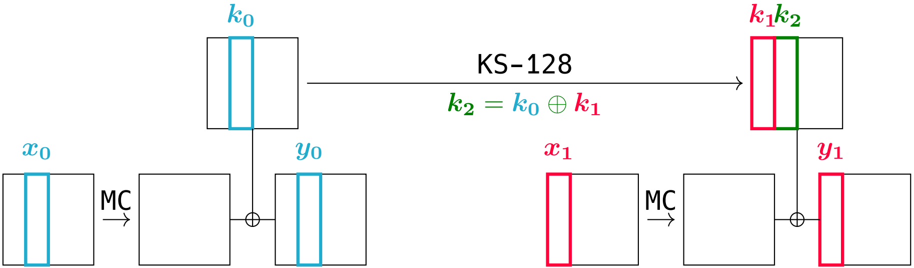


Invalidate some truncated trails



$$\begin{cases} MC(x_0) \oplus k_0 = y_0 \\ MC(x_1) \oplus k_1 = y_1 \end{cases}$$

Invalidate some truncated trails



$$\begin{cases} MC(x_0) \oplus k_0 = y_0 \\ MC(x_1) \oplus k_1 = y_1 \end{cases} \implies MC(x_0 \oplus x_1) = y_0 \oplus y_1 \oplus k_2$$

0 or $\geq$ 5 active bytes

Dynamic programming for differential bounds on AES

Dynamic programming for differential bounds

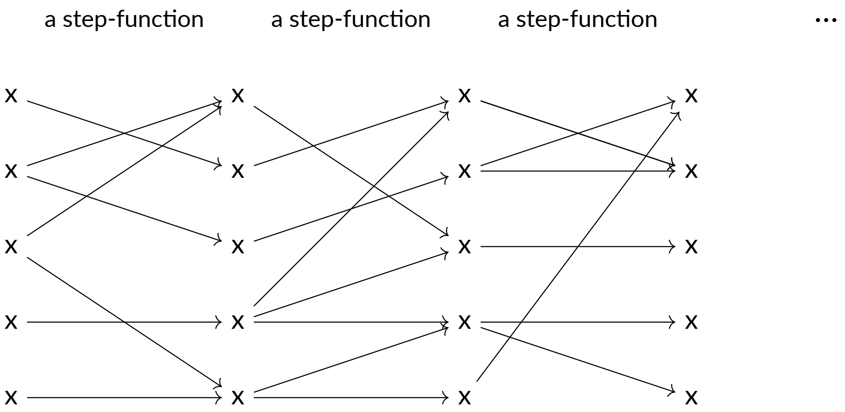
Fouque *et al.*, CRYPTO 2013

- Generic tool based on dynamic programming.
- Complexity easy to understand.
- Application for AES–128 only.

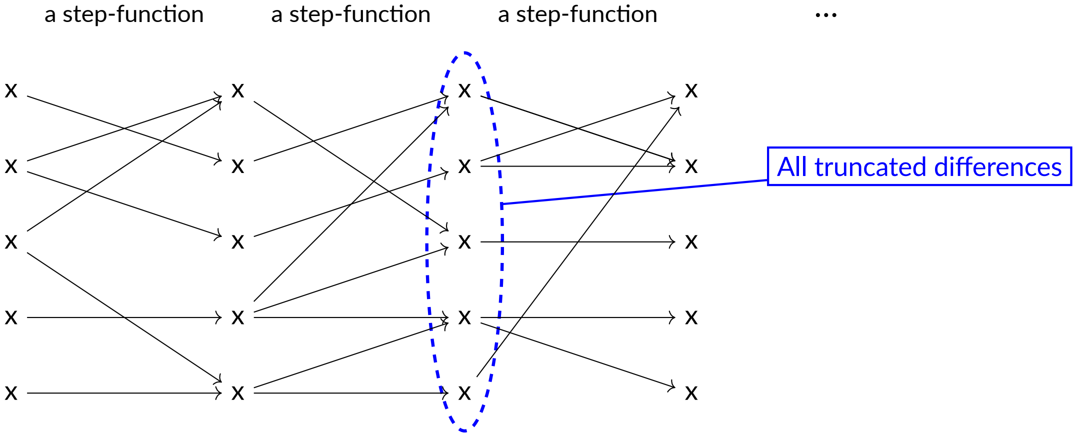
Our work

- **Extend** the work of Fouque *et al.* (2013) for **all versions** of AES.
- **Running time comparable** to that of the **CP approach** of Gerault *et al.* (2018, 2020).

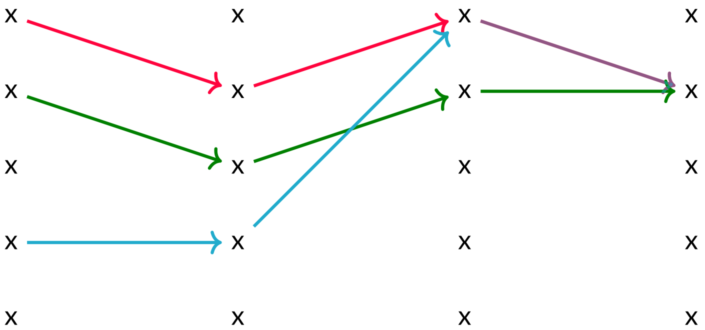
Principle of the dynamic programming algorithm of [FJP13]



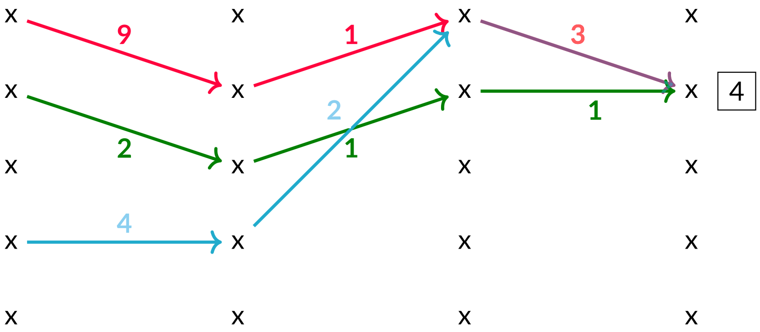
Principle of the dynamic programming algorithm of [FJP13]



Principle of the dynamic programming algorithm of [FJP13]



Principle of the dynamic programming algorithm of [FJP13]

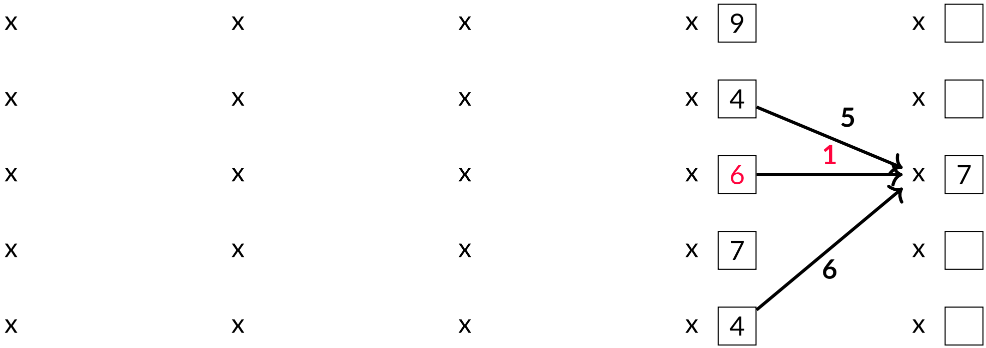


min active S-boxes

Principle of the dynamic programming algorithm of [FJP13]

x	x	x	x $\boxed{9}$
x	x	x	x $\boxed{4}$
x	x	x	x $\boxed{6}$
x	x	x	x $\boxed{7}$
x	x	x	x $\boxed{4}$

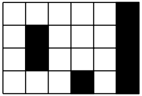
Principle of the dynamic programming algorithm of [FJP13]



Adapting the dynamic programming algorithm of [FJP13]

1. **Reduce the memory complexity.**

Truncated difference

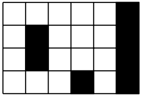


$\lvert K \rvert$	128	192	256
$\#$	2^{32}	2^{40} ✗	2^{48} ✗

Adapting the dynamic programming algorithm of [FJP13]

1. **Reduce the memory complexity.**

Truncated difference



$\lvert K \rvert$	128	192	256
#	2^{32}	2^{40} ✗	2^{48} ✗

Compressed difference

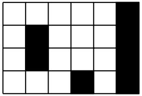
0	2	0	1	0	4

$\lvert K \rvert$	128	192	256
#	$2^{18.58}$	$2^{23.22}$	$2^{27.86}$

Adapting the dynamic programming algorithm of [FJP13]

1. **Reduce the memory complexity.**

Truncated difference



$\lvert K \rvert$	128	192	256
#	2^{32}	2^{40} ✗	2^{48} ✗

Compressed difference

0	2	0	1	0	4

$\lvert K \rvert$	128	192	256
#	$2^{18.58}$	$2^{23.22}$	$2^{27.86}$

2. **Integrate constraints over several rounds in a second step.**

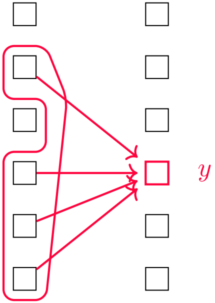
Remarks

- Propagation rules for compressed differences

  $\rightsquigarrow$ new incompatibilities possible

- Improvements to compute the arrays

$prec(y) = \{x \mid x \to y\}$

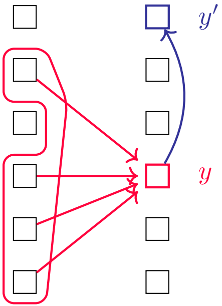


Naive time complexity: $\sum_y |prec(y)|$

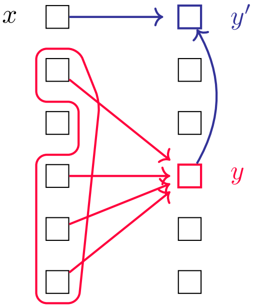
Remarks

- For some values y, y' for SR, $\mathrm{ARK} \circ \mathrm{MC}$

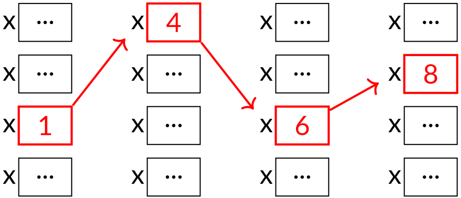
$$prec(y) = prec(y')$$



$$prec(y') = prec(y) \cup \{x\}$$



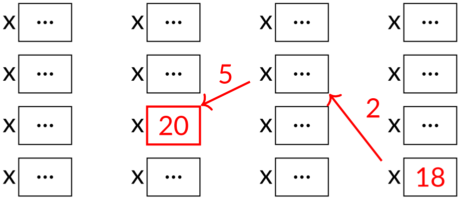
Integrate constraints over several rounds



min

$\hookrightarrow$ bound not necessarily tight!

Integrate constraints over several rounds



Trail with less than 22 active S-boxes?

1. Search for a **compressed trail** with n active S-boxes.
   - depth-first search approach in the backward direction
   - check some linear relations, at least partially
2. Turn it, if possible, into a **truncated trail**.

Complexity

- To construct the arrays:

	Time complexity	Memory (Bytes)
AES-128	$r \times 2^{22.89}$	$(9r - 9) \times 2^{18.58}$
AES-192	$r \times 2^{27.53}$	$(3r - 3) \times 2^{23.22}$
AES-256	$r \times 2^{32.18}$	$(3r - 4) \times 2^{27.86}$

- The total complexity depends on the number of trails found during the second step.

Running time

Algorithm	R	Min nb of active S-boxes	CP [RGMS22] Time	Dynam. Prog. Real Time (User Time)
AES-128	4	1	31s	1s (1s)
	5	17	2h24m24s	40s (5m6s)
AES-192	5	5	8	1s (5s)
	6	10	17s	1s (8s)
	7	14	46s	1s (9s)
	8	18	1m23s	1m35s (12m37s)
	9	24	30m	4d5h (20d4h)
AES-256	9	15	5m46s	32s (3m24s)
	10	16	2m39s	34s (3m31s)
	11	20	5m30s	42s (4m30s)
	12	20	4m37s	42s (4m16s)
	13	24	7m	52s (5m24s)
	14	24	9m17s	50s (5m5s)

Conclusion

- Our *ad hoc* algorithm is competitive.
- It works because the AES is very structured.

$\rightsquigarrow$ The search space is much smaller that one could have expected.

$\rightsquigarrow$ Hard to adapt to less structured ciphers?

Conclusion

- Our *ad hoc* algorithm is competitive.
- It works because the AES is **very structured**.

⇝ The search space is much smaller that one could have expected.

⇝ Hard to adapt to less structured ciphers?

- Other result:

⇝ **differential MITM attack against 13 rounds of** AES–256, with 2 related keys.