Lightweight Cryptanalysis:

Practical Key-Recovery for MANTIS$_5$

Christoph Dobraunig Maria Eichlseder Daniel Kales Florian Mendel

ESC 2017

The Tweakable Block Cipher MANTIS

MANTIS

Motivation

- Low latency
- Tweakable
- Bounds

Design

- PRINCE-like cipher structure
- TWEAKEY tweak schedule
- Midori round transformations

[Bei+16] C. Beierle, J. Jean, S. Kölbl, G. Leander, A. Moradi, T. Peyrin, Y. Sasaki, P. Sasdrich, and S. M. Sim
The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS
CRYPTO 2016

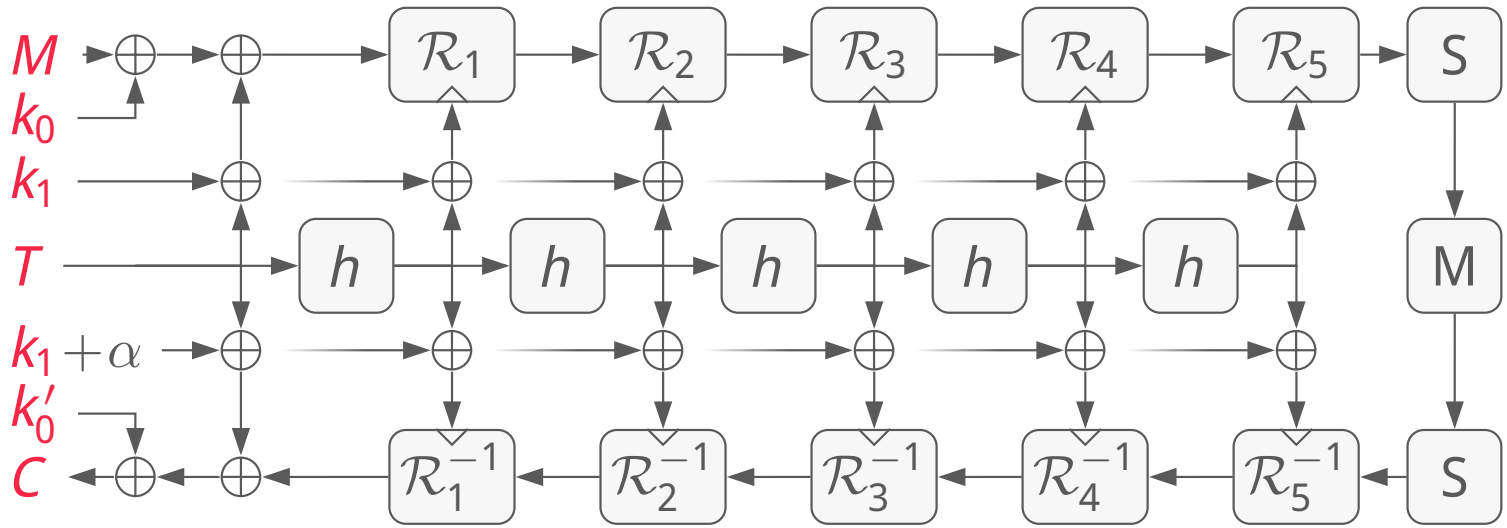
Our results ($\rightarrow$ FSE 2017)

Recover 128-bit key of MANTIS_5 with 2^{30} CP in 1 hour ($< 2^{96}$)

α-Reflective Structure of MANTIS

Inspired by PRINCE [Bor+12]

- 64-bit message block M, tweak T, keys k_0 and k_1
- MANTIS$_r$ means $2r + 2$ S-box layers:



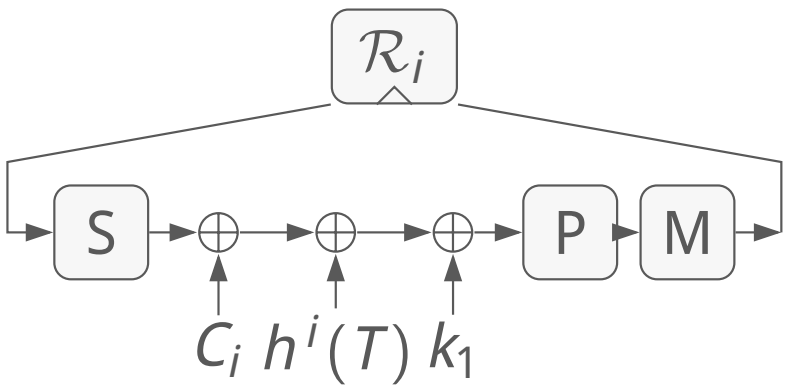
- Decryption = encryption with related key

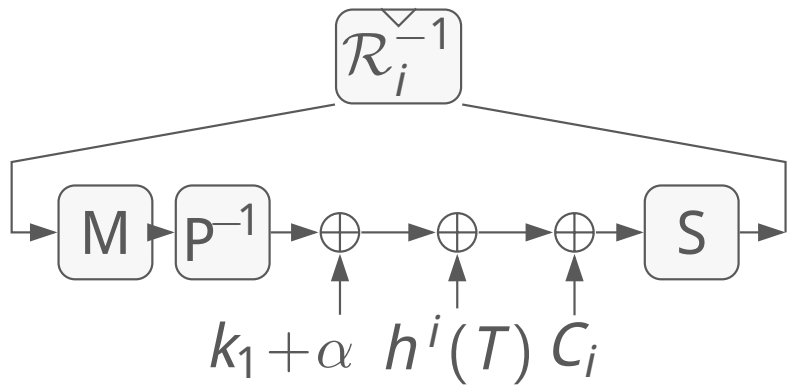
MANTIS Round Functions $\mathcal{R}_i, \mathcal{R}_i^{-1}$

- State S: 4×4 matrix of 4-bit nibble cells S_i:

$$S = \begin{array}{|c|c|c|c|} \hline S_0 & S_1 & S_2 & S_3 \\ \hline S_4 & S_5 & S_6 & S_7 \\ \hline S_8 & S_9 & S_{10} & S_{11} \\ \hline S_{12} & S_{13} & S_{14} & S_{15} \\ \hline \end{array} .$$

- Order in $\mathcal{R}_i$ differs from PRINCE: First permute, then mix





MANTIS Transformations

Inspired by Midori [Ban+15]

- SubCells (S): involutive 4-bit S-box $\mathcal{S}$
- AddConstant$_i$ (C): Xor round constant C_i
- AddTweakey$_i$ (A): Xor key k_1 (for $\mathcal{R}_i$) or $k_1 + \alpha$ (for $\mathcal{R}_i^{-1}$) and permuted tweak $h^i(T)$
- PermuteCells (P): faster diffusion than ShiftRows
- MixColumns (M): involutive near-MDS matrix M over $\mathbb{F}_{2^4}$

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

h

6	5	14	15
0	1	2	3
7	12	13	4
8	9	10	11

(a) Tweak permutation

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

P

0	11	6	13
10	1	12	7
5	14	3	8
15	4	9	2

(b) PermuteCells

$$\mathrm{M} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

(c) MixColumns

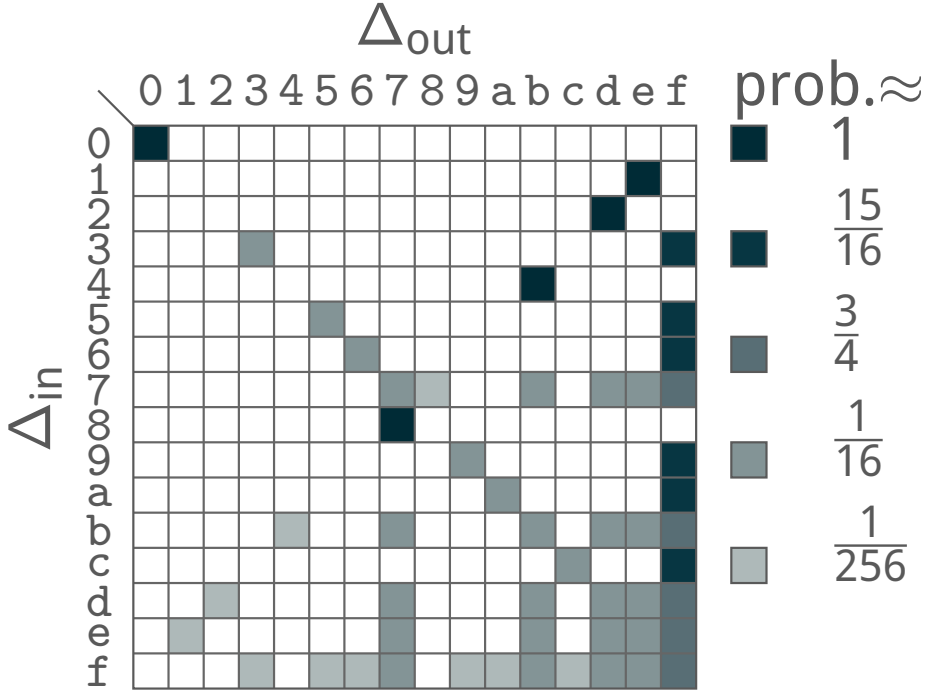
Designers' Analysis and Security Claim

- Min number of active S-boxes (MILP):
  - MANTIS_5: ≥ 34
  - MANTIS_7: ≥ 50
- Max prob of any differential characteristic (MDP 2^{-2}):
  - MANTIS_5: $\leq 2^{-68}$
  - MANTIS_7: $\leq 2^{-100}$
- Security claim: No attacks below...
  - MANTIS_5: D data and $T \leq 2^{126}/D$ time, where $D \leq 2^{30}$ CP
  - MANTIS_7: D data and $T \leq 2^{126}/D$ time

Properties of the MANTIS Transformations

Properties of MixColumns

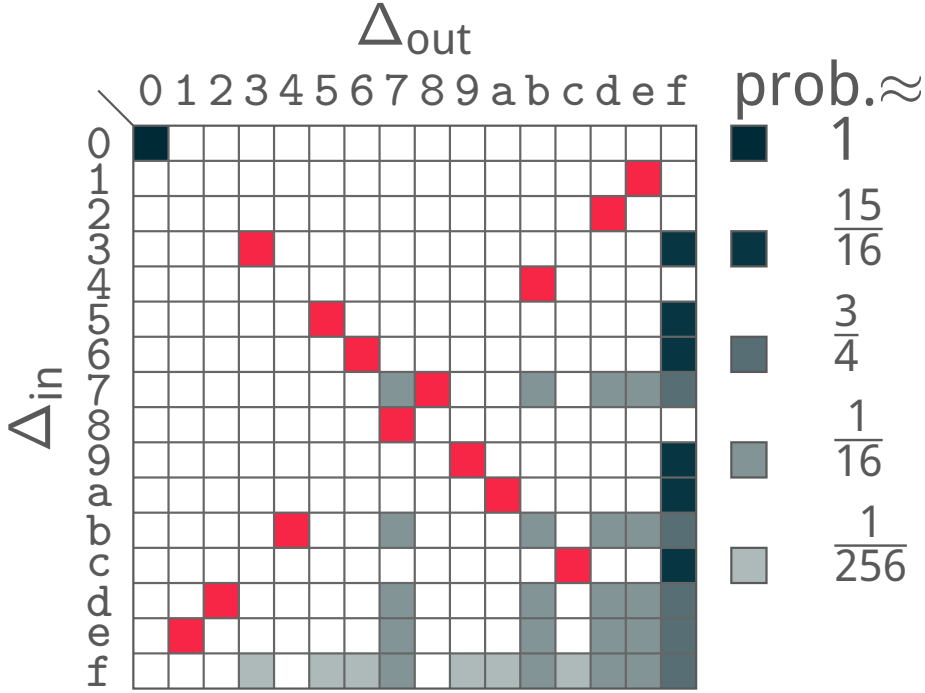
Truncated DDT of MixColumns:



- Binary coefficients

Properties of MixColumns

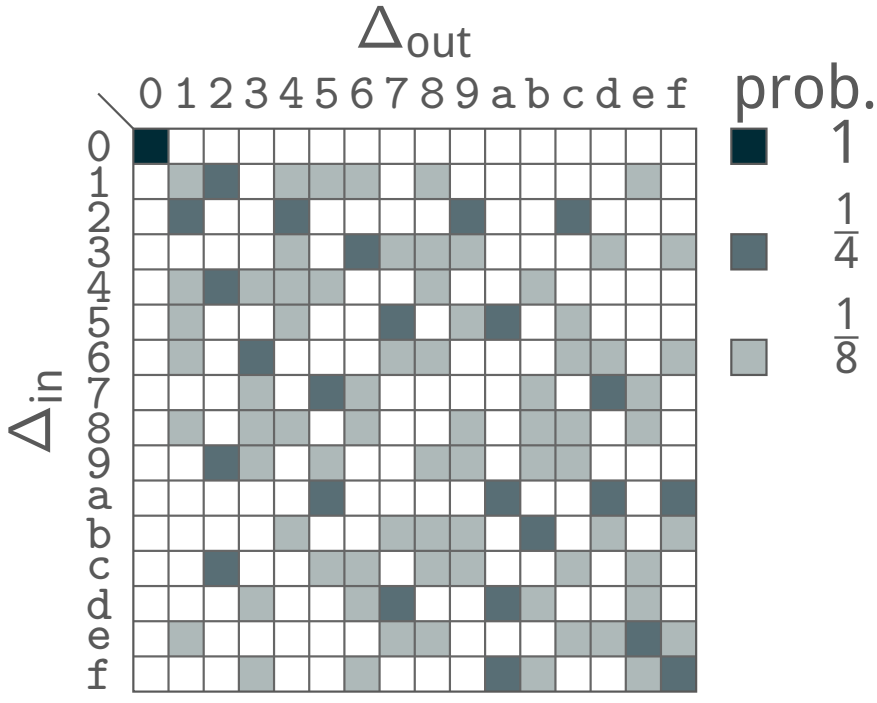
Truncated DDT of MixColumns:



- Binary coefficients
- Branch number 4:
  $1 \rightarrow 3$, $2 \rightarrow 2$, $3 \rightarrow 1$
- Satisfied with $\delta, \delta, \delta, \delta$
- Differential fixed points

Properties of SubCells

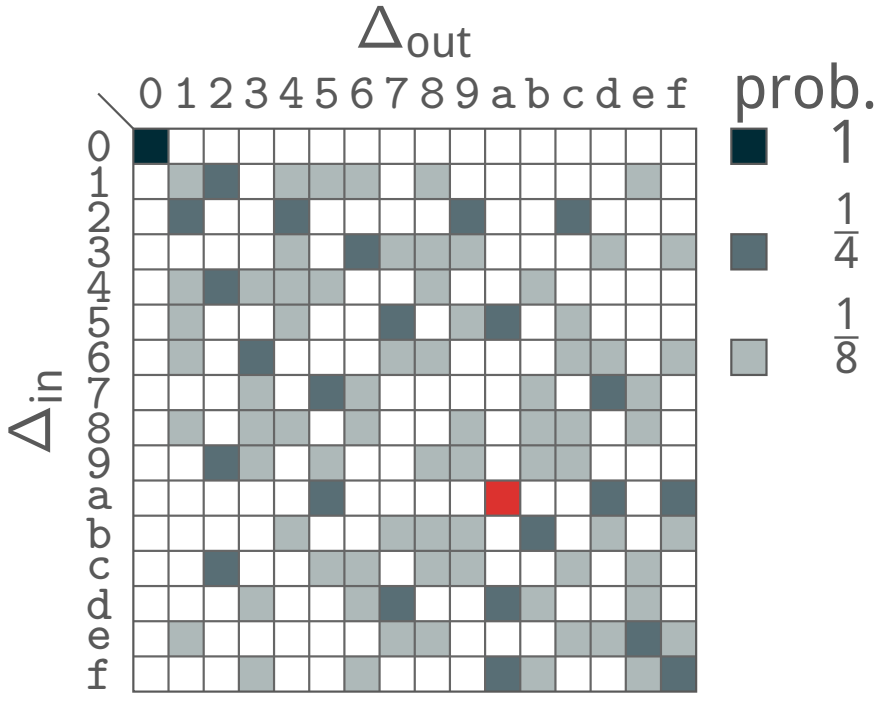
DDT of SubCells:



- 4-bit, involutive

Properties of SubCells

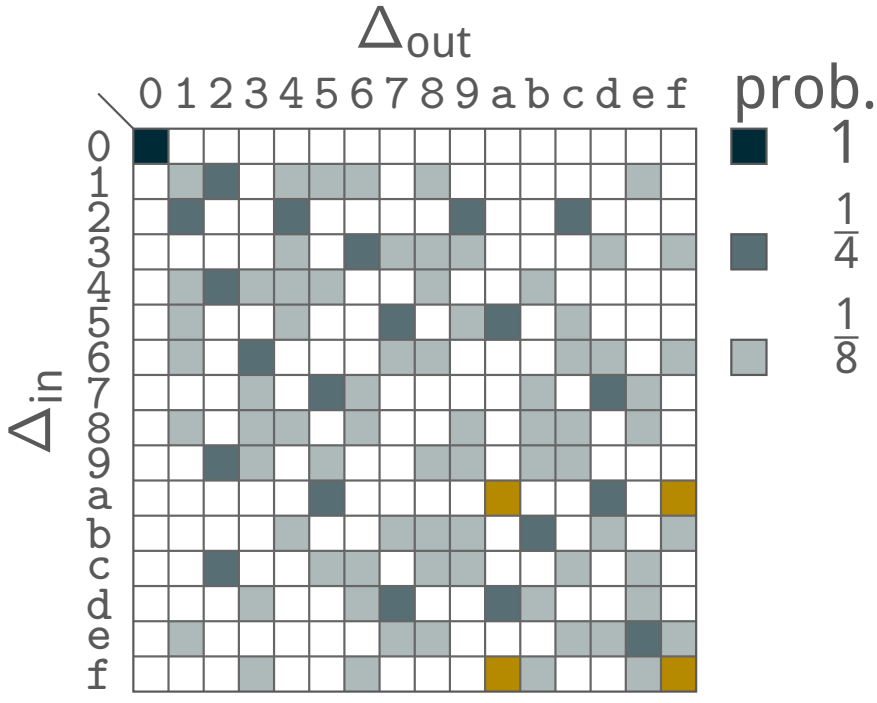
DDT of SubCells:



- 4-bit, involutive
- Differential fixed points:
  - $\mathbb{P}[\mathtt{a} \rightarrow \mathtt{a}] = \frac{1}{4}$

Properties of SubCells

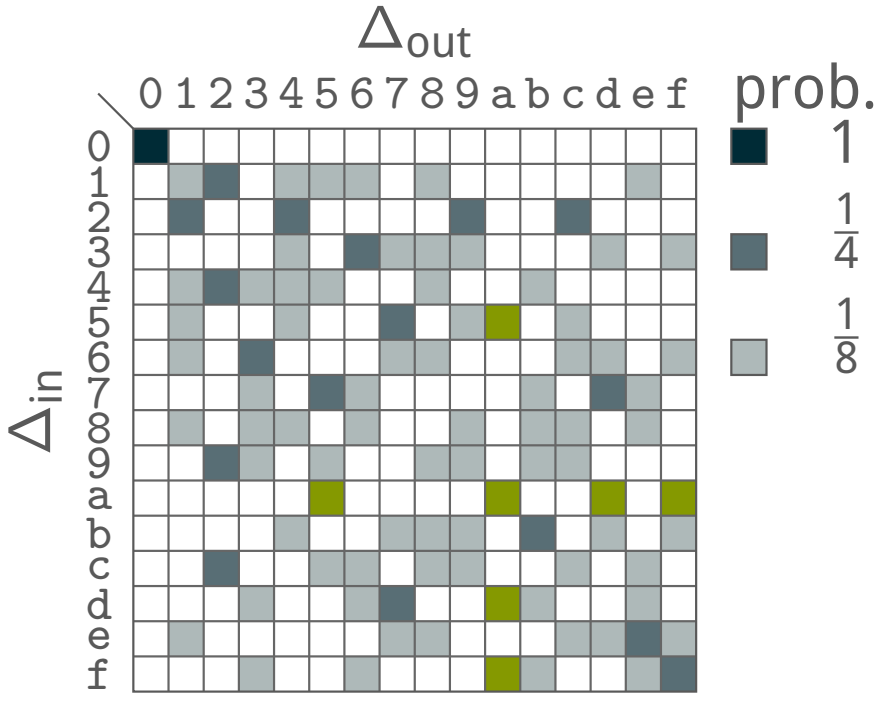
DDT of SubCells:



- 4-bit, involutive
- Differential fixed points:
  - $\mathbb{P}[\mathtt{a} \to \mathtt{a}] = \frac{1}{4}$
  - $\mathbb{P}[\{\mathtt{a}, \mathtt{f}\} \to \{\mathtt{a}, \mathtt{f}\}] = \frac{1}{2}$

Properties of SubCells

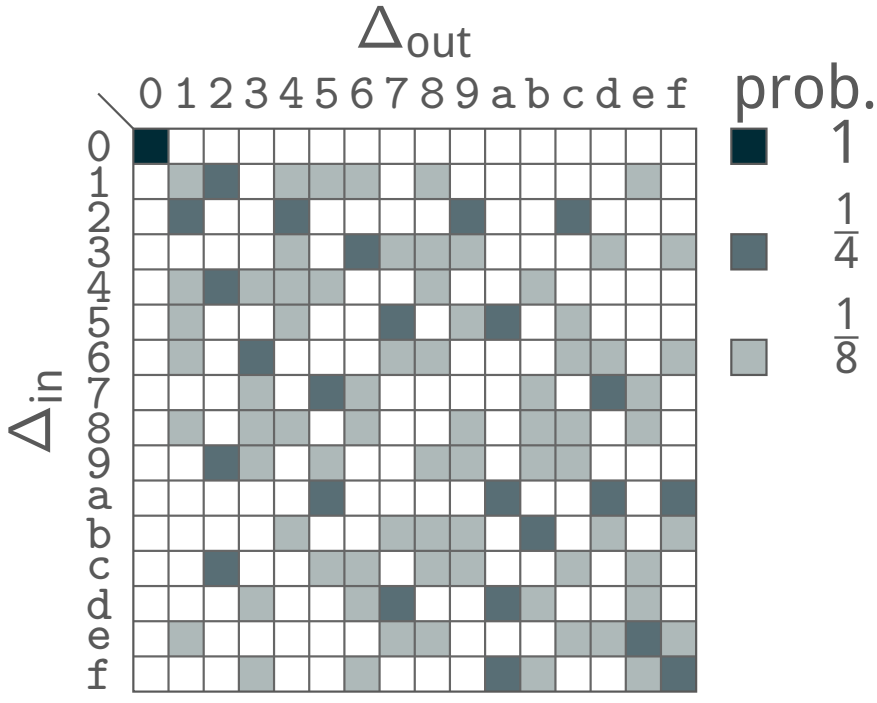
DDT of SubCells:



- 4-bit, involutive
- Differential fixed points:
  - $\mathbb{P}[\mathtt{a} \to \mathtt{a}] = \frac{1}{4}$
  - $\mathbb{P}[\{\mathtt{a}, \mathtt{f}\} \to \{\mathtt{a}, \mathtt{f}\}] = \frac{1}{2}$
  - $\mathbb{P}[\{\mathtt{a}, \mathtt{f}, \mathtt{d}, \mathtt{5}\} \to \mathtt{a}] = \frac{1}{4}$, $\mathbb{P}[\mathtt{a} \to \{\mathtt{a}, \mathtt{f}, \mathtt{d}, \mathtt{5}\}] = 1$

Properties of SubCells

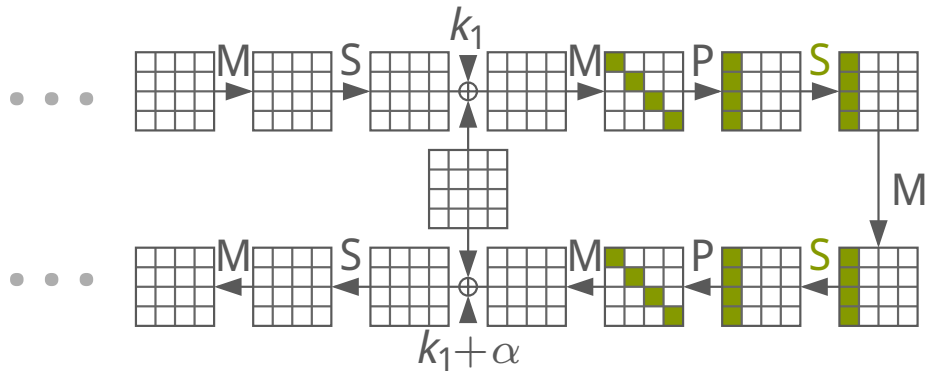
DDT of SubCells:



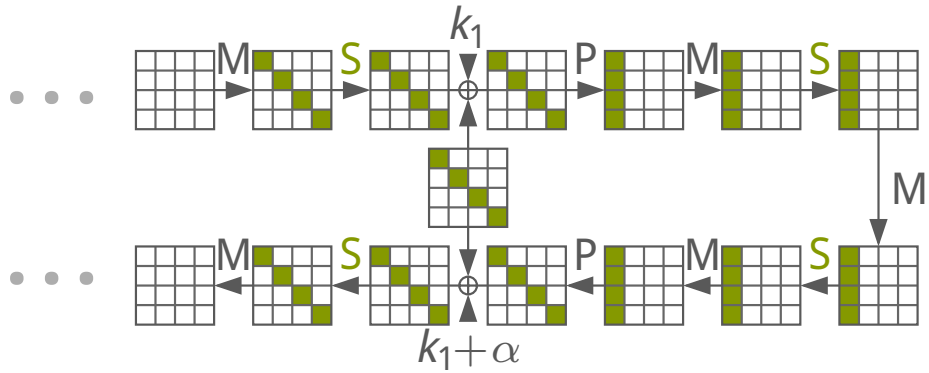
- 4-bit, involutive
- Differential fixed points:
  - $\mathbb{P}[\mathtt{a} \to \mathtt{a}] = \frac{1}{4}$
  - $\mathbb{P}[\{\mathtt{a},\mathtt{f}\} \to \{\mathtt{a},\mathtt{f}\}] = \frac{1}{2}$
  - $\mathbb{P}[\{\mathtt{a},\mathtt{f},\mathtt{d},\mathtt{5}\} \to \mathtt{a}] = \frac{1}{4}$, $\mathbb{P}[\mathtt{a} \to \{\mathtt{a},\mathtt{f},\mathtt{d},\mathtt{5}\}] = 1$
- If (x, x') follows $\{\mathtt{a},\mathtt{f},\mathtt{d},\mathtt{5}\} \to \{\mathtt{a},\mathtt{f}\}$, then so does $(x+\mathtt{a}, x'+\mathtt{a})$

Properties of the Inner Rounds

- Order of operations in PRINCE: Mix-then-Permute



- Order of operations in MANTIS: Permute-then-Mix

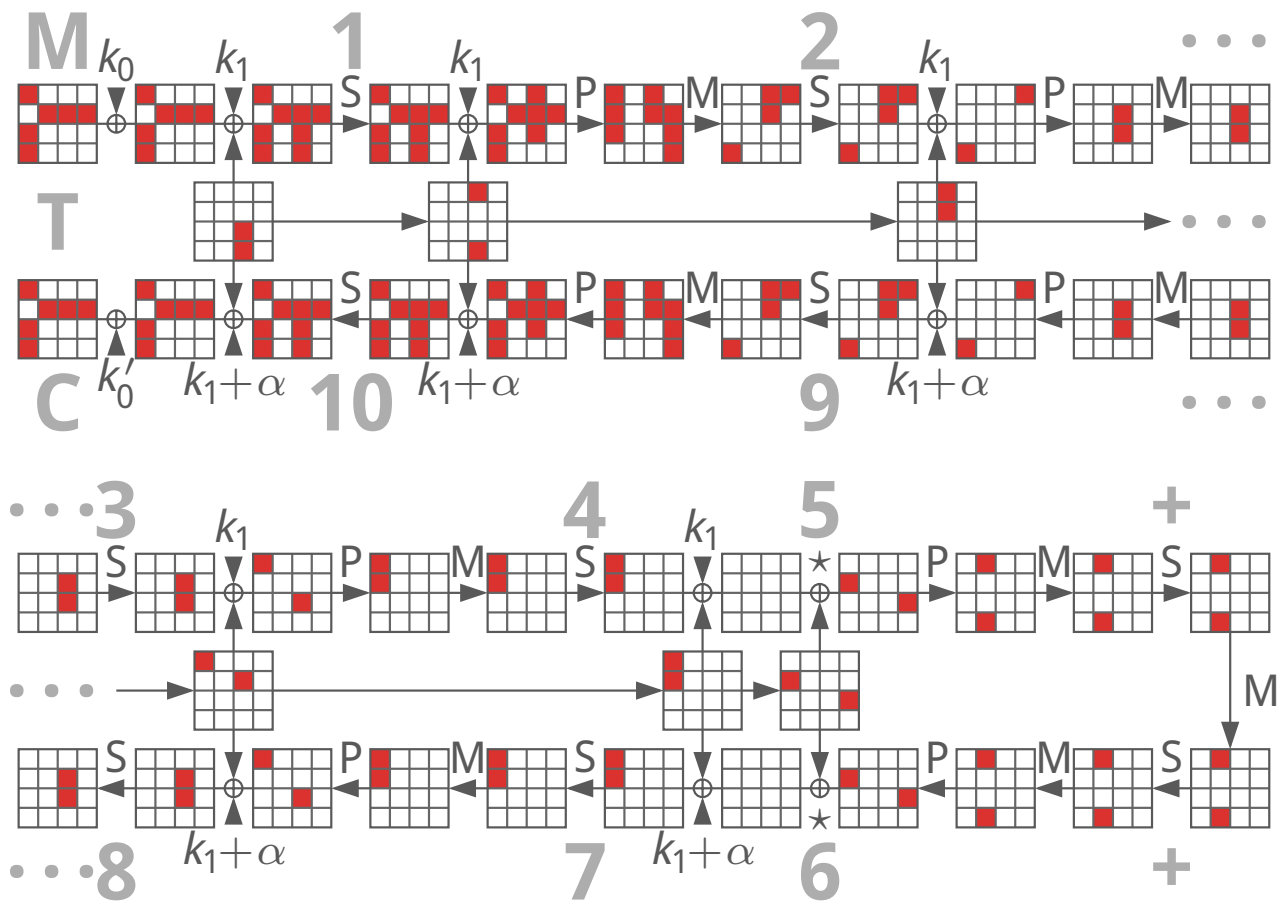


Superboxes over 4 (instead of 2) S-box layers!

A Family of Differential Characteristics

A (Nearly) Optimal Characteristic I

MILP: Truncated char with 34 (or 36) active S-boxes



A (Nearly) Optimal Characteristic II

MILP: Truncated char with 34 (or 36) active S-boxes

Observations:

- MixColumns: All transitions tightly match branch number
- AddTweakey: All differences cancel

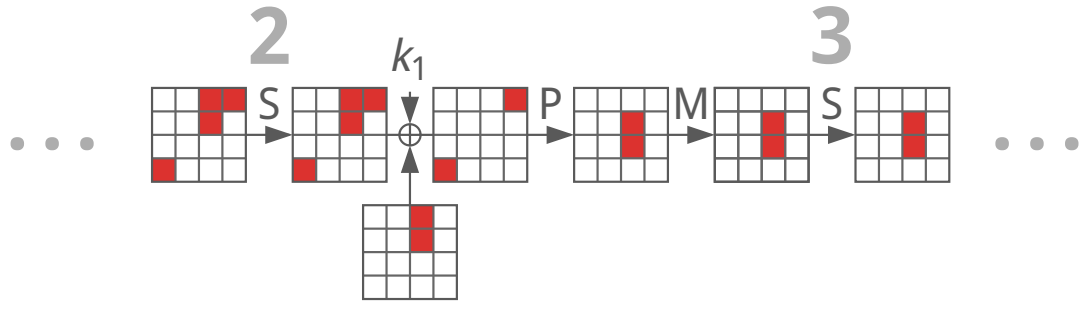
Set all differences to differential fixed point a
$\Rightarrow$ optimal probability 2^{-68} (or 2^{-72})

Relaxing (Clustering) Characteristics I

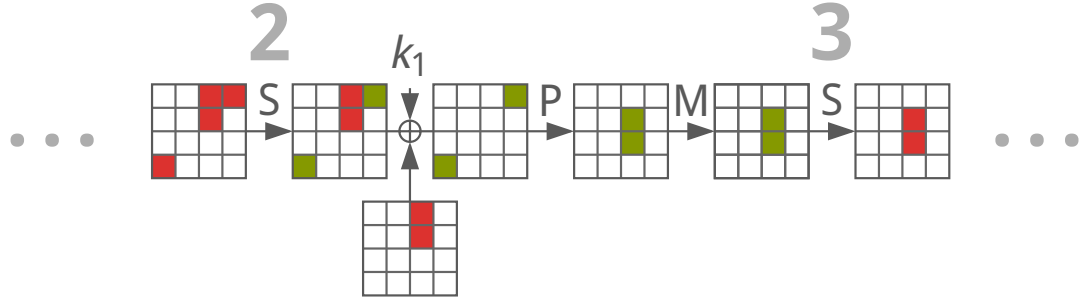
$\mathtt{a} \xrightarrow{S} \mathtt{a} \xrightarrow{S} \mathtt{a}$ can be relaxed to $\mathtt{a} \xrightarrow{S} \{\mathtt{a}, \mathtt{f}, \mathtt{d}, \mathtt{5}\} \xrightarrow{S} \mathtt{a}$ or $\mathtt{a} \xrightarrow{S} \{\mathtt{a}, \mathtt{f}\} \xrightarrow{S} \{\mathtt{a}, \mathtt{f}\}$

(if M cooperates)

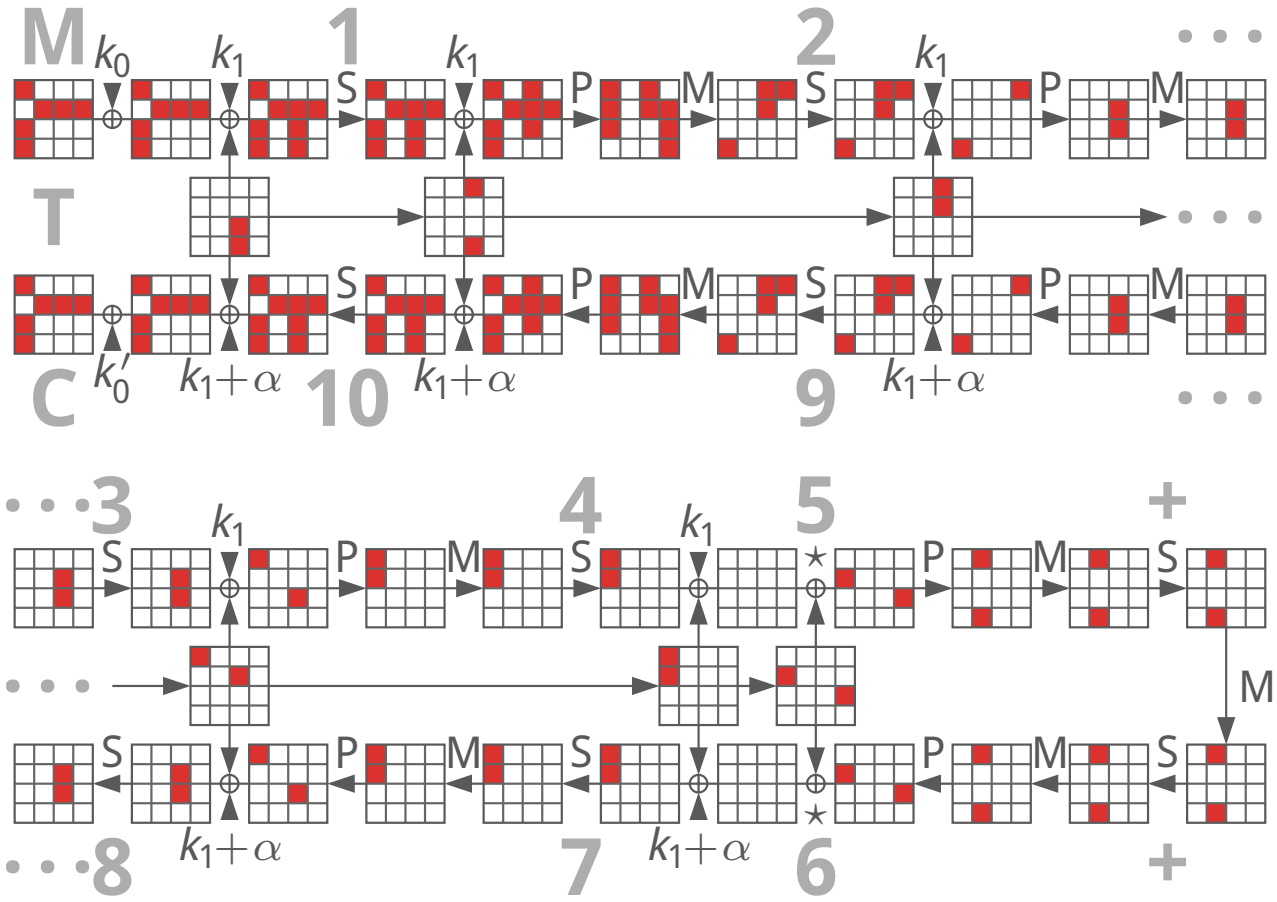
- Round 2 with probability 2^{-8}:



- Round 2 with probability 2^{-6} (Round 3 unchanged 2^{-4}):



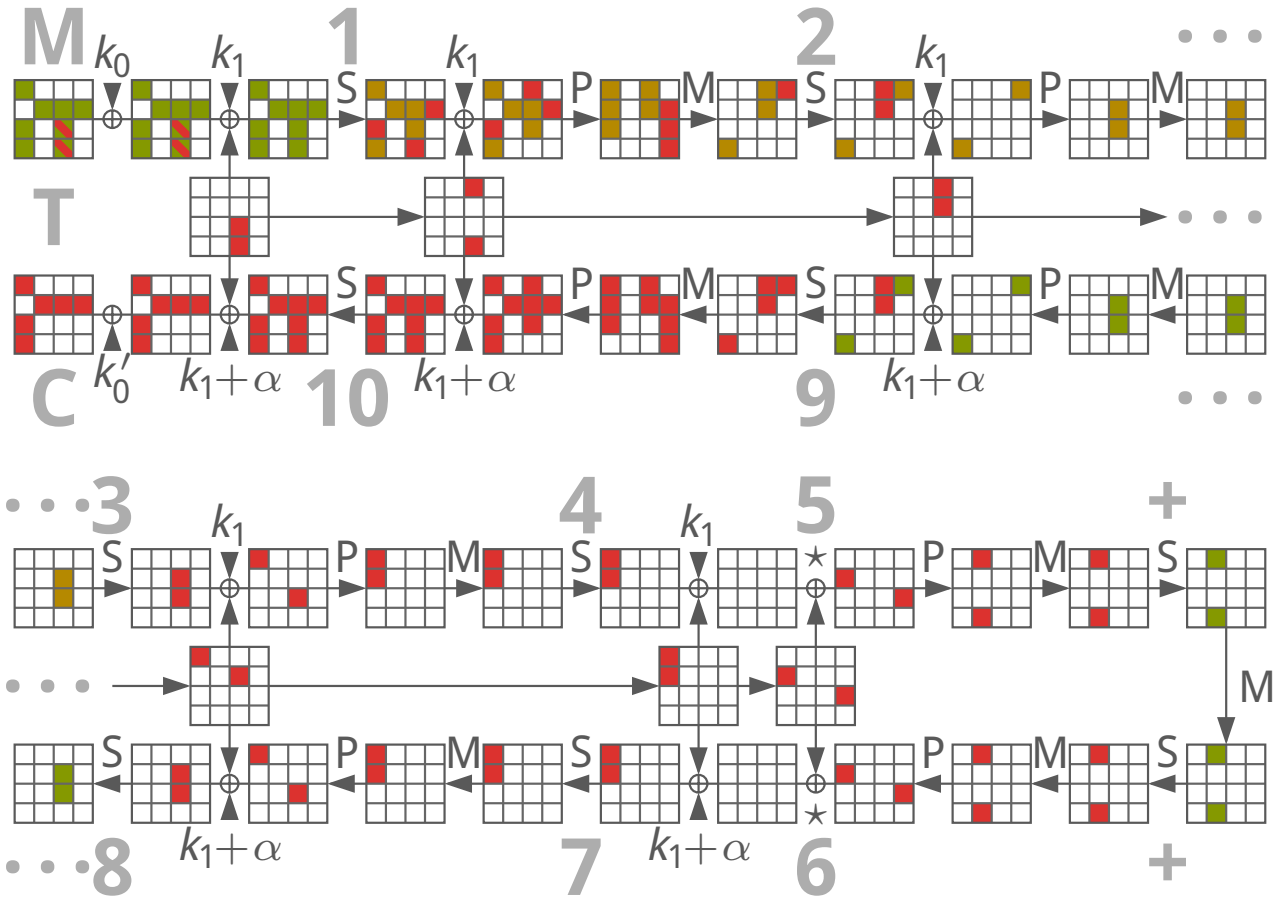
Relaxing (Clustering) Characteristics II



2^{-72}

a

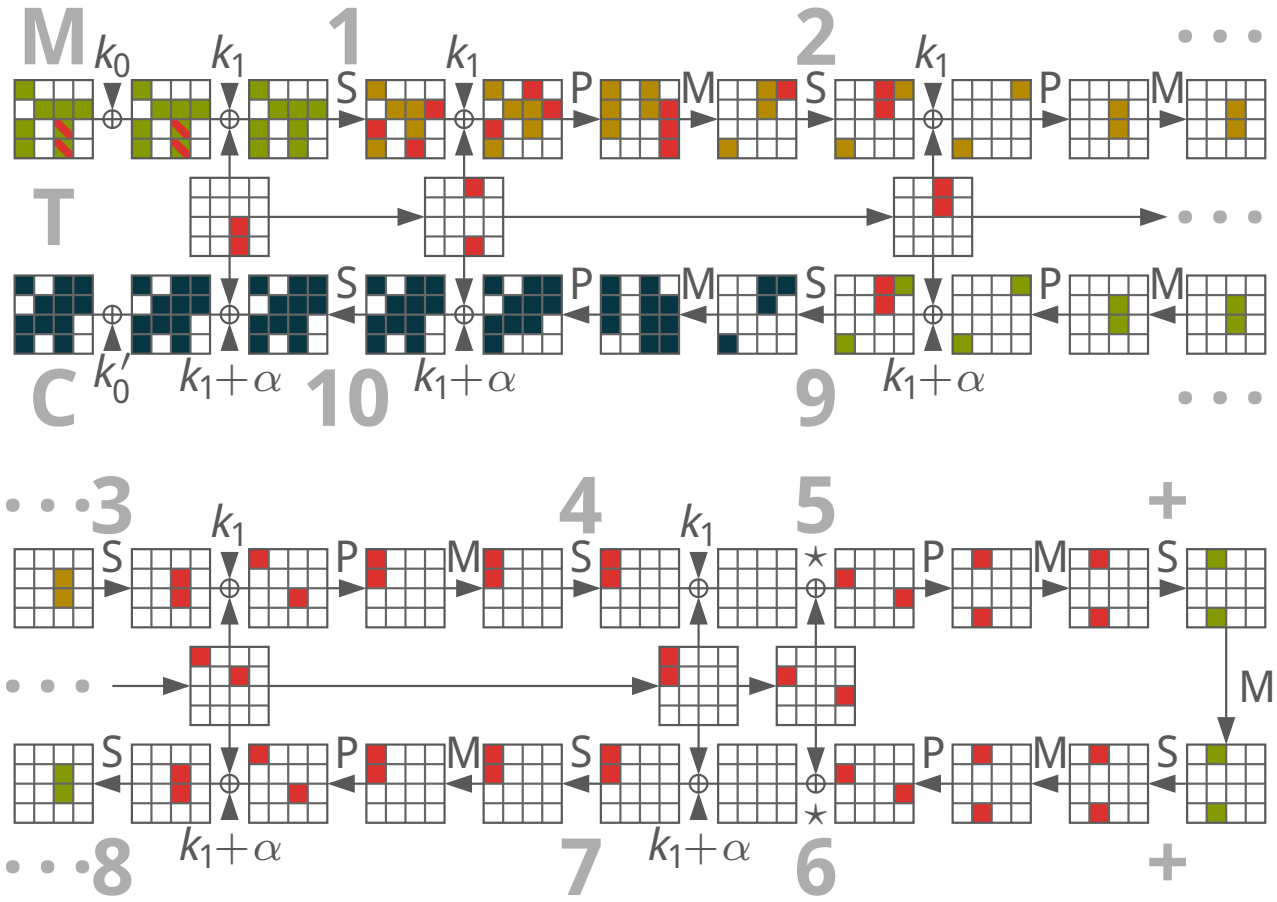
Relaxing (Clustering) Characteristics II



$2^{-64.51}$

a $\{a, f\}$ $\{a, f, d, 5\}$

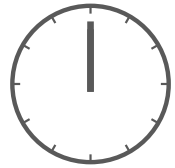
Relaxing (Clustering) Characteristics II



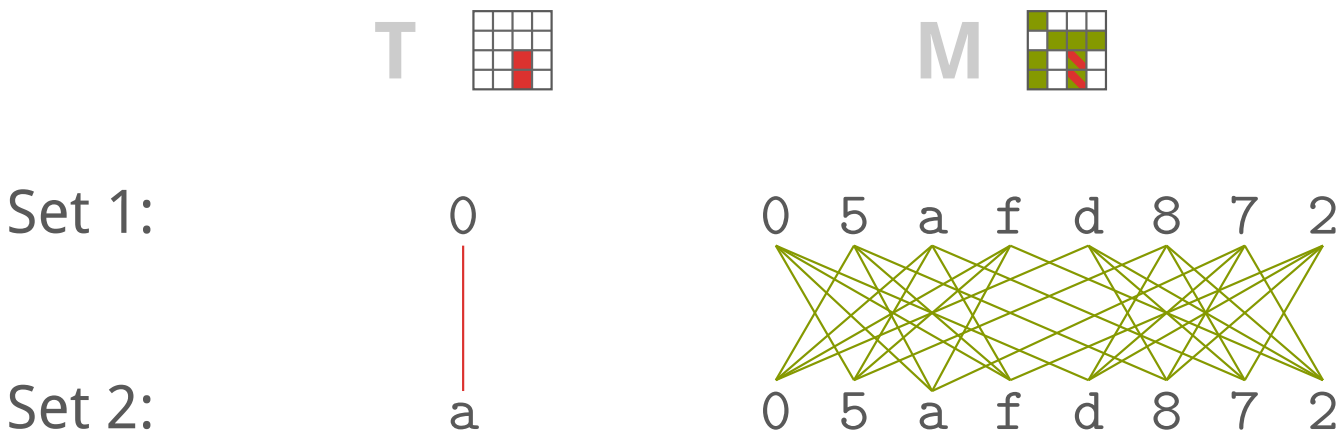
$2^{-40.51}$

a $\{a, f\}$ $\{a, f, d, 5\}$ don't care

Initial Structure for Data Limit $D \leq 2^{30}$



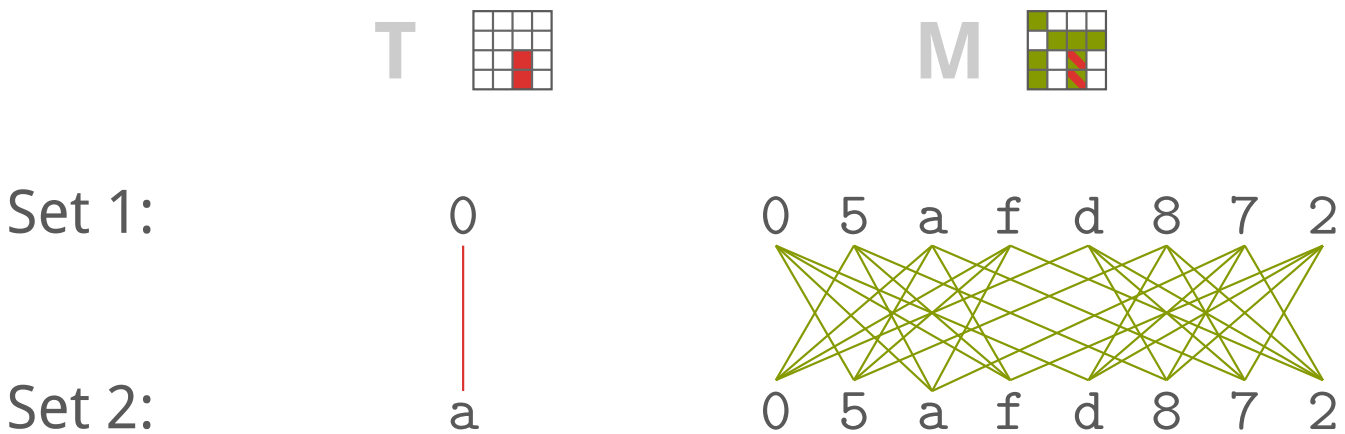
Efficiently generate differences $\{\mathtt{a}, \mathtt{f}, \mathtt{d}, \mathtt{5}\}$ (note $\mathtt{a} + \mathtt{5} = \mathtt{f}$):



$(8 \cdot 4)^8 =$ 2^{40} pairs from $2 \cdot 8^8 =$ 2^{25} CP

Initial Structure for Data Limit $D \leq 2^{30}$

Efficiently generate differences $\{\mathtt{a}, \mathtt{f}, \mathtt{d}, \mathtt{5}\}$ (note $\mathtt{a} + \mathtt{5} = \mathtt{f}$):



$k \cdot (8 \cdot 4)^8 = k \cdot 2^{40}$ pairs from $k \cdot 2 \cdot 8^8 = k \cdot 2^{25}$ CP

Staged Key Recovery Attack

Key Recovery – Overview

1. Filter for useful pairs: **C** = 
2. Recover 44-bit final key $k_0' + k_1$
3. Filter for right pairs: **9** = 
4. Recover 32-bit initial key $k_0 + k_1$
5. Combine and complete k_0 and k_1

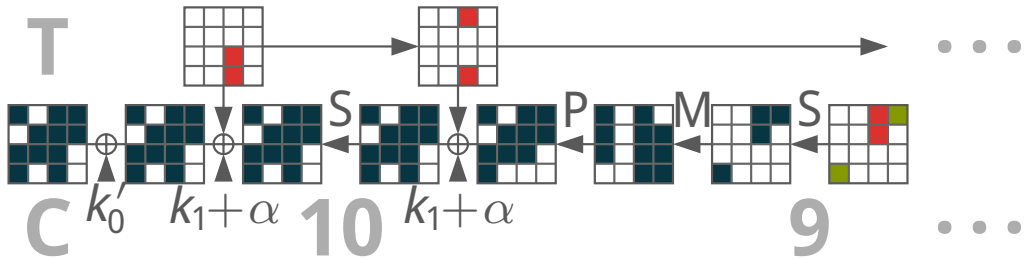
1 Filter for Useful Pairs

- Generate 2^{41} pairs from 2^{26} CP
- We expect $2^{41-40.51} > 1$ right pair

- Necessary condition: $\mathbf{C}$ = 
- Reduction to about $2^{41-22} = 2^{19}$ pairs I
- Still expect > 1 right pair

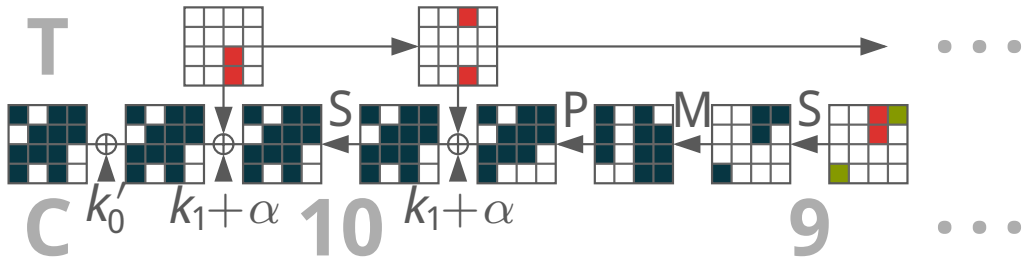
2 Recover 44-bit Final Key

- We need to guess a 44-bit key $k_0' + k_1$ and test (2^{-30})



2 Recover 44-bit Final Key

- We need to guess a 44-bit key $k'_0 + k_1$ and test (2^{-30})

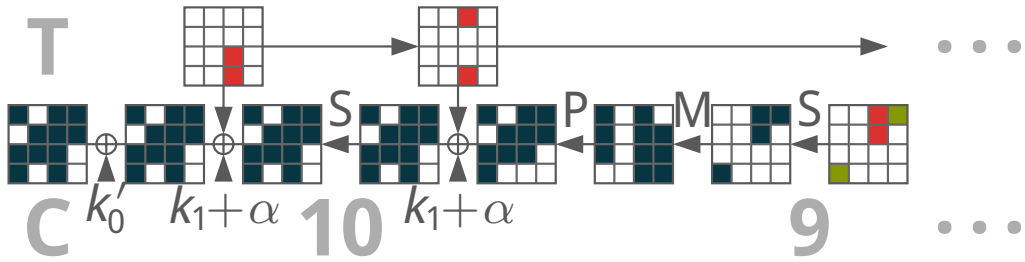


- For each pair i, about $2^{44-30} = 2^{14}$ key candidates $\mathcal{B}^i$ remain:

$$\mathcal{B}^i = \mathcal{C}^i_{0,5,10} \times \mathcal{C}^i_{14} \times \mathcal{C}^i_{3,6,9,12} \times \mathcal{C}^i_{2,7,8}.$$

2 Recover 44-bit Final Key

- We need to guess a 44-bit key $k_0' + k_1$ and test (2^{-30})



- For each pair i, about $2^{44-30} = 2^{14}$ key candidates $\mathcal{B}^i$ remain:

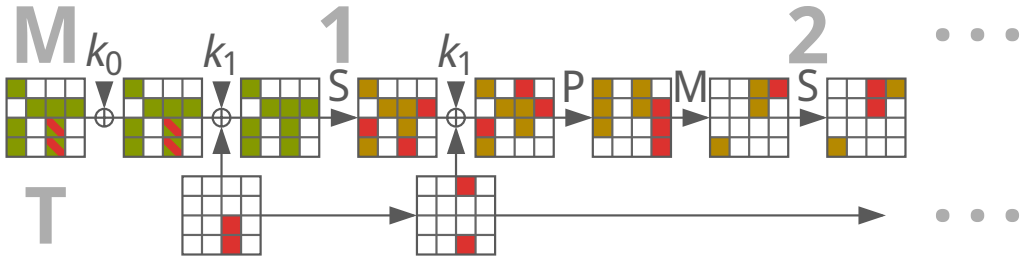
$$\mathcal{B}^i = \mathcal{C}^i_{0,5,10} \times \mathcal{C}^i_{14} \times \mathcal{C}^i_{3,6,9,12} \times \mathcal{C}^i_{2,7,8}.$$

- Reduces key space by factor 2^{11}: Repeat $4\times$ and compute

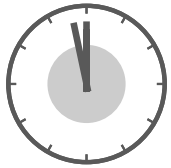
$$\bigcap_{r=1}^{4} \bigcup_{\substack{i \in I_r \\ |I_r| \approx 2^{19}}} \mathcal{C}^{(r,i)}_{0,5,10} \times \mathcal{C}^{(r,i)}_{14} \times \mathcal{C}^{(r,i)}_{3,6,9,12} \times \mathcal{C}^{(r,i)}_{2,7,8}.$$

3 Filter for right pairs
4 Recover 32-bit initial key

- We expect > 4 right pairs to remain
- We need to guess a 32-bit key $k_0 + k_1$ and test ($2^{-15.51}$)



5 Combine and complete k_0 and k_1



- We have now collected 76 linear equations for k_0 and k_1
- Collect 14 more equations:

1. Recover $S_0 + S_5 + S_{10}$ of k_1: We target cell S_{12} at the beginning of Round 2. From our previously recovered key bits, we know the values of cells S_0, S_5, S_{10} at the beginning of Round 1. Our target cell is the sum of these known values, plus an unknown cell $S_0 + S_5 + S_{10}$ of k_1. Checking the correct S-box transition for all 4 valid pairs is expected to eliminate all but the correct cell value (otherwise, we can additionally check the transition in Round 9). This adds 1 linearly independent equation to the system.

2. Recover $S_6 + S_{12}$ of k_1: We target cells S_2, S_6 at the beginning of Round 2. Each of the two is the sum of the same two unknown, constant values (cell S_3 and cell S_9 after AddTweakey of Round 1), a known, variable value, and a cell of k_1 (S_6 or S_{12}, respectively). By checking the S-box transitions and then eliminating the two unknown constants, we recover the cell sum $S_6 + S_{12}$ of k_1. This adds 1 linearly independent equation to the system.

3. Recover $S_2 + S_7 + S_8$ of k_1: We target cell S_3 at the end of Round 9. Similar to (1), the transition depends on a sum of k_1 cells, $S_2 + S_7 + S_8$. From (1), we can derive the exact target difference, so the transition probability is at most 2^{-2}, and we expect only the one correct cell value to remain. This adds 4 linearly independent equations to the system.

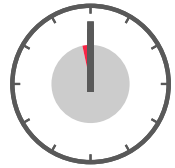
4. Guess 1 bit: If we guess only 1 bit of k_0 now (e.g., in cell S_{12}), this will fully determine the values of cells S_2, S_5, S_6, S_7, S_8, S_{12} of k_0 and k_1.

5. Recover S_3 of k_1: We target cells S_6 and S_{10} at the beginning of Round 3. Due to the previous MixColumns operation, the internal difference between cells S_6 and S_{10} is equal to the internal difference between cells S_3 and S_{12} after the previous AddTweakey operation, which is known except for the addition of key cell S_3 of k_1. On the other hand, since we require that both target cells belong to the same set of 4 possible values for a valid transition, this cuts down the possible values for S_3 of k_1 to less than half. After repeating for all 4 valid pairs and, if necessary, similarly for the transition in Round 8, we expect only the correct candidate to remain. This adds 4 linearly independent equations to the system.

6. Recover S_9 of k_1: We target cells S_2 and S_6 at the end of Round 9. The transition depends on the values of cells S_3, S_6, S_9, S_{12} before AddTweakey of Round 10, which are all known by now except for the addition of key cell S_9 of k_1. Determining S_9 adds another 4 linearly independent equations to the system.

- Brute-force the remaining 38 bits

5 Combine and complete k_0 and k_1



- We have now collected 76 linear equations for k_0 and k_1
- Collect 14 more equations:

1. Recover $S_0 + S_5 + S_{10}$ of k_1: We target cell S_{12} at the beginning of Round 2. From our previously recovered key bits, we know the values of cells S_0, S_5, S_{10} at the beginning of Round 1. Our target cell is the sum of these known values, plus an unknown cell $S_0 + S_5 + S_{10}$ of k_1. Checking the correct S-box transition for all 4 valid pairs is expected to eliminate all but the correct cell value (otherwise, we can additionally check the transition in Round 9). This adds 1 linearly independent equation to the system.

2. Recover $S_6 + S_{12}$ of k_1: We target cells S_2, S_6 at the beginning of Round 2. Each of the two is the sum of the same two unknown, constant values (cell S_3 and cell S_9 after AddTweakey of Round 1), a known variable value, and a cell of k_1 (S_6 or S_{12}, respectively). By checking the S-box transitions and then eliminating the two unknown constants, we recover the cell sum $S_6 + S_{12}$ of k_1. This adds 1 linearly independent equation to the system.

3. Recover $S_2 + S_7 + S_8$ of k_1: We target cell S_2 at the end of Round 9. Similar to (1), the transition depends on a sum of k_1 cells, $S_2 + S_7 + S_8$. From (1), we can derive the exact target difference, so the transition probability is at most 2^{-2}, and we expect only the one correct cell value to remain. This adds 4 linearly independent equations to the system.

4. Guess 1 bit: If we guess only 1 bit of k_1 now (e.g., in cell S_{12}), this will fully determine the values of cells S_2, S_5, S_6, S_7, S_8, S_{12} of k_0 and k_1.

5. Recover S_3 of k_1: We target cells S_3 and S_{10} at the beginning of Round 3. Due to the previous MixColumns operation, the internal difference between cells S_3 and S_{10} is equal to the internal difference between cells S_3 and S_{12} after the previous AddTweakey operation, which is known except for the addition of key cell S_3 of k_1. On the other hand, since we require that both target cells belong to the same set of 4 possible values for a valid transition, this cuts down the possible values for S_3 of k_1 to less than half. After repeating for 4 valid pairs and, if necessary, similarly for the transition in Round 8, we expect only the correct candidate to remain. This adds 4 linearly independent equations to the system.

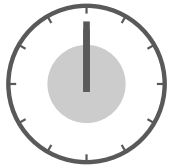
6. Recover S_9 of k_1: We target cells S_2 and S_6 at the end of Round 9. The transition depends on the values of cells S_3, S_6, S_9, S_{12} before AddTweakey of Round 10, which are all known by now except for the addition of key cell S_9 of k_1. Determining S_9 adds another 4 linearly independent equations to the system.

- Brute-force the remaining 38 bits

SAT Solver

Conclusions

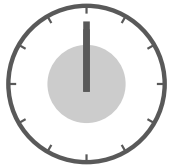
Practical Verification



- Estimates and validity confirmed
- Two issues, though:
  1. Variance: Right pairs appear in clusters.
     Some repetitions have no right pairs, some have many...
     Fix: Adjust generation of pairs (increase to 2^{30} CP)
  2. Equivalent key candidates: Both k^* and $k^* + \mathtt{a}$ pass test

  Both caused by the same property of SubCells:
  If (x, x') follows $\{\mathtt{a}, \mathtt{f}, \mathtt{d}, 5\} \rightarrow \{\mathtt{a}, \mathtt{f}\}$, then so does $(x + \mathtt{a}, x' + \mathtt{a})$

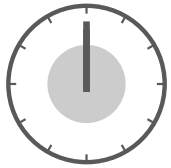
Applicability to MANTIS$_7$



- ✓ Same claim for $DT \leq 2^{126}$, no data limit D
- ✓ Optimal characteristic with probability 2^{-100} (instead of 2^{-68})
- ✓ Similar clustering effects

- ✗ Too few active S-boxes in **M** and **C**
- ✗ Small state size requires high probability

Conclusion



- Security margin of MANTIS may be too small...
- Caused by typical “lightweight” properties
  - Differential fixed points
  - Lightweight tweakey schedule
  - Superbox effect in inner rounds
  - Data limit not as effective as expected (multiple differentials)
  - No margin for key recovery

Bibliography

[Bor+12] J. Borghoff, A. Canteaut, T. Güneysu, E. B. Kavun, M. Knezevic, L. R. Knudsen, G. Leander, V. Nikov, C. Paar, C. Rechberger, P. Rombouts, S. S. Thomsen, et al.
PRINCE – A Low-Latency Block Cipher for Pervasive Computing Applications
ASIACRYPT 2012

[JNP14] J. Jean, I. Nikolić, and T. Peyrin
Tweaks and Keys for Block Ciphers: The TWEAKEY Framework
ASIACRYPT 2014

[Ban+15] S. Banik, A. Bogdanov, T. Isobe, K. Shibutani, H. Hiwatari, T. Akishita, and F. Regazzoni
Midori: A Block Cipher for Low Energy
ASIACRYPT 2015

[Bei+16] C. Beierle, J. Jean, S. Kölbl, G. Leander, A. Moradi, T. Peyrin, Y. Sasaki, P. Sasdrich, and S. M. Sim
The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS
CRYPTO 2016