Better Steady than Speedy

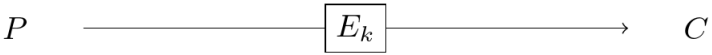
Full Break of SPEEDY-7-192

Christina Boura, **Nicolas David**, Rachelle Heim Boissier, María Naya-Plasencia

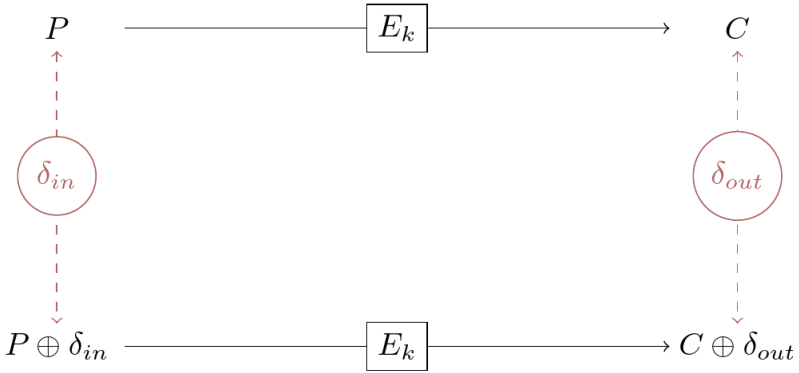
Table Of Contents

Differential Cryptanalysis

Concept

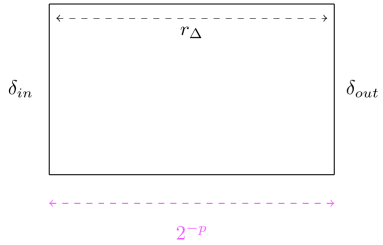


Concept

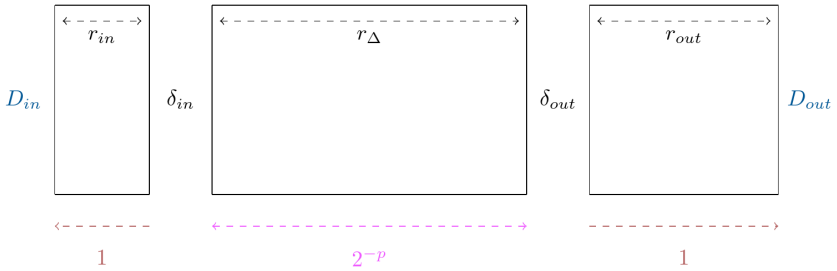


$$\mathbb{P}\left[E_k(P \oplus \delta_{in}) = E_k(P) \oplus \delta_{out}\right]?$$

Differential Cryptanalysis

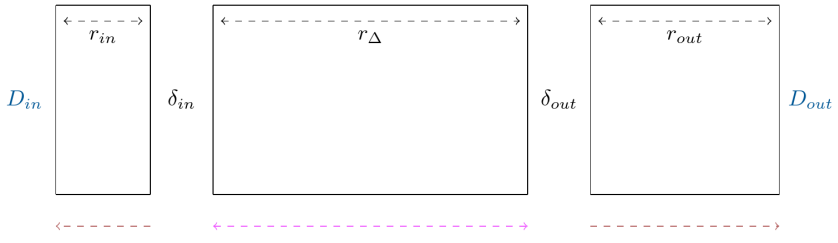


Differential Cryptanalysis



$|D_{in}| = 2^{d_{in}}$
$|D_{out}| = 2^{d_{out}}$
$|\text{State}| = 2^n$

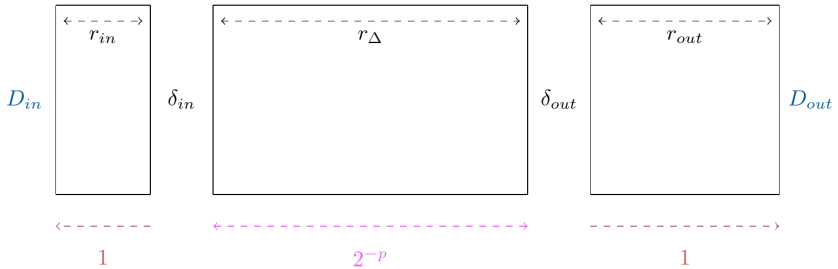
Differential Cryptanalysis



$|D_{in}| = 2^{d_{in}}$
$|D_{out}| = 2^{d_{out}}$
$|\text{State}| = 2^n$

Data Generation:	Calling the encryption oracle.	$\mathcal{D} \approx 2^{p+1}$
Pair Sieving:	Building pairs that follow D_{in} and D_{out}.	$C_{PS} \approx 2^{p+d_{in}+d_{out}-n}$
Key Recovery:	Recovering the encryption key.	$C_{KR} \approx C_{PS} \cdot \mathcal{C}_1$

Differential Cryptanalysis



$|D_{in}| = 2^{d_{in}}$
$|D_{out}| = 2^{d_{out}}$
$|\text{State}| = 2^n$

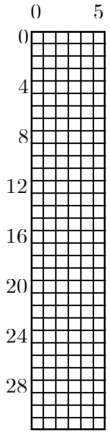
Data Generation:	Calling the encryption oracle.	$\mathcal{D} \approx 2^{p+1}$
Pair Sieving:	Building pairs that follow D_{in} and D_{out}.	$C_{PS} \approx 2^{p+d_{in}+d_{out}-n}$
Key Recovery:	Recovering the encryption key.	$C_{KR} \approx C_{PS} \cdot \mathcal{C}_1$

Minimize $\mathcal{D} + C_{PS} + C_{KR}$.

The SPEEDY Block Cipher

The SPEEDY Block Cipher

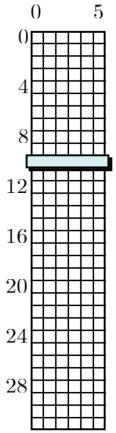
Designed by Leander, Moos, Moradi and Rasoolzadeh (TCHES 21).
State : 6 x 32 bits.
Key Schedule: Bit-permutation



The SPEEDY Block Cipher

Designed by Leander, Moos, Moradi and Rasoolzadeh (TCHES 21).
State : 6 x 32 bits
Key Schedule: Bit-permutation
Round Operation:

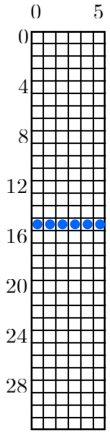
- SB: 6-bit S-Box



The SPEEDY Block Cipher

Designed by Leander, Moos, Moradi and Rasoolzadeh (TCHES 21).
State : 6 x 32 bits
Key Schedule: Bit-permutation
Round Operation:

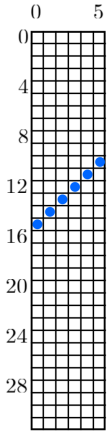
- SB: 6-bit S-Box
- SC: Rotation inside the columns



The SPEEDY Block Cipher

Designed by Leander, Moos, Moradi and Rasoolzadeh (TCHES 21).
State : 6 x 32 bits
Key Schedule: Bit-permutation
Round Operation:

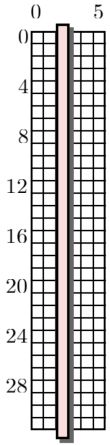
- SB: 6-bit S-Box
- SC: Rotation inside the columns



The SPEEDY Block Cipher

Designed by Leander, Moos, Moradi and Rasoolzadeh (TCHES 21).
State : 6 x 32 bits
Key Schedule: Bit-permutation
Round Operation:

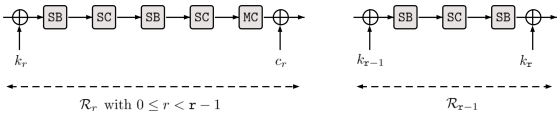
- SB: 6-bit S-Box
- SC: Rotation inside the columns
- MC: MixColumns operation

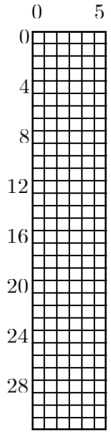


The SPEEDY Block Cipher

Designed by Leander, Moos, Moradi and Rasoolzadeh (TCHES 21).
State : 6 x 32 bits
Key Schedule: Bit-permutation
Round Operation:

- SB: 6-bit S-Box
- SC: Rotation inside the columns
- MC: MixColumns operation





Security Claims And Previous Work

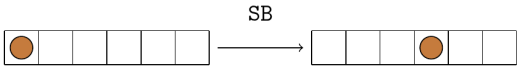
Algorithm	# rounds attacked	Ref.	Data	Time (in C_E)	Memory	Security claim $(\mathcal{T}, \mathcal{D})$
SPEEDY-5-192	3	[1]	$2^{17.6}$	$2^{52.5}$	$2^{25.5}$	$(2^{128}, 2^{64})$
SPEEDY-5-192	5	this work	$2^{101.65}$	$2^{107.8}$	2^{42}	$(2^{128}, 2^{64})$
SPEEDY-6-192	5.5	this work	$2^{121.65}$	$2^{127.8}$	2^{42}	$(2^{128}, 2^{128})$
SPEEDY-7-192	7	this work	$2^{187.28}$	$2^{187.84}$	2^{42}	$(2^{192}, 2^{192})$

"The attacker cannot add more than one round to extend a distinguisher" - SPEEDY designers

[1] Raghvendra Rohit and Santanu Sarkar (Africacrypt 2022)

Differential Attack on SPEEDY-7-192

S-Box: 1-bit Input To 1-bit Output Differences



α/β	1	2	4	8	16	32
1	2	–	4	2	4	2
2	1	2	4	4	2	2
4	–	3	2	–	3	1
8	1	1	3	3	1	1
16	–	–	4	4	3	4
32	1	1	2	3	1	–

Table: probability x 2^{-5}.

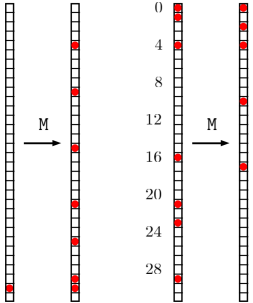
Searching For Good Differentials

Our approach

Precompute all good one-round trails and chain them to create longer trails.

First step: Compute and store in T all $(x, M(x))$ such that both x and $M(x)$ have at most 7 active bits each.

$\Rightarrow$ 164 classes of 32 pairs $(x, M(x))$



One-round Differentials

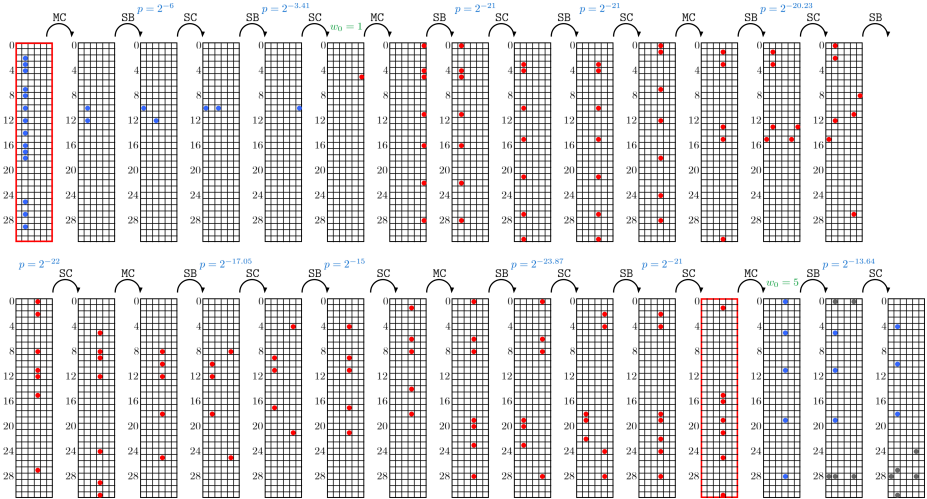
$$\texttt{st[0]} \xrightarrow{\text{MC}} \texttt{st[1]} \xrightarrow{\text{SB}} \texttt{st[2]} \xrightarrow{\text{SC}} \texttt{st[3]} \xrightarrow{\text{SB}} \texttt{st[4]} \xrightarrow{\text{SC}} \texttt{st[5]} \xrightarrow{\text{MC}} \texttt{st[6]}.$$

We computed all such propagations $(\texttt{st[0]}, \texttt{st[6]})$ satisfying the following conditions:

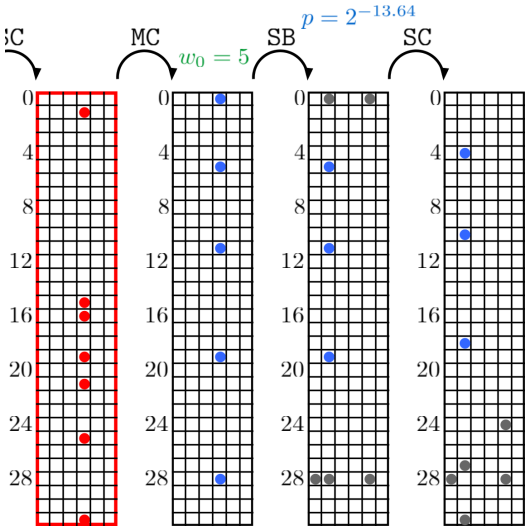
- $\texttt{st[0]}$ has a single active column $\texttt{c}_0$ such that $(\texttt{c}_0, M(\texttt{c}_0)) \in \texttt{T}$,
- $\texttt{st[5]}$ has a single active column $\texttt{c}_5$ such that $(\texttt{c}_5, M(\texttt{c}_5)) \in \texttt{T}$,
- $\texttt{st[2]}$ has at most two active bits per row,
- the probability of the trail $(\texttt{st[0]}, \texttt{st[6]})$ is strictly higher than 2^{-49}.

$48\,923 \times 32 = 1\,565\,536$
one-round trails

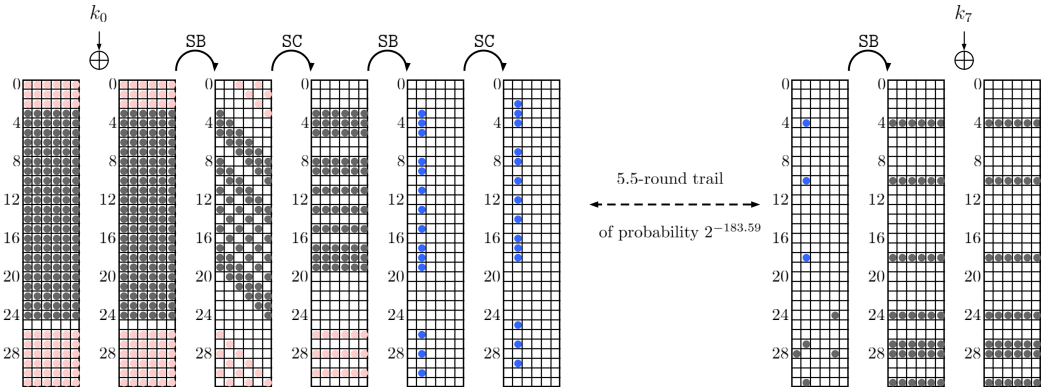
5.5-Round Differential Distinguisher: $2^{-183.59}$



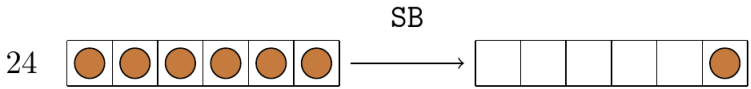
Trade-off



Key Recovery Rounds

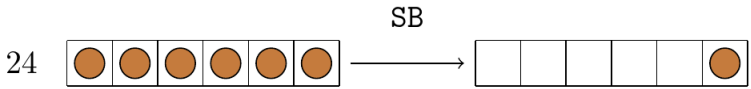


Differential Sieving



Which input differences can follow the truncated differential after the first S-Box ?

Differential Sieving



Which input differences can follow the truncated differential after the first S-Box ?

1. Compute this set of differences.
2. Discard all the pairs that do not fall into this set.

→ For all the input rows, we can perform a differential sieving.

Total sieving of 2^{-11}

Complexity Interlude

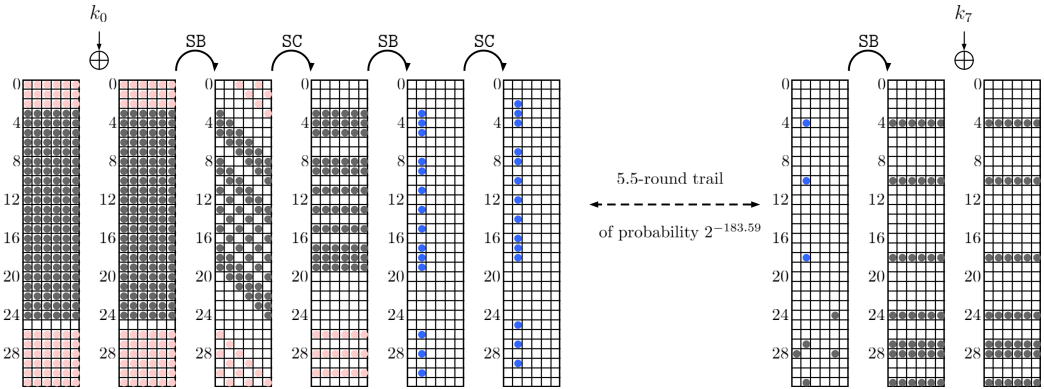
- Data : $\mathcal{D} = 2^{p+1} = 2^{184.59}$
- Pair Sieving : $C_{PS} = 2^{p+d_{in}+d_{out}-n} = 2^{208.5}$

Complexity Interlude

- Data : $\mathcal{D} = 2^{p+1} = 2^{184.59}$
- Pair Sieving : $C_{PS} = 2^{p+d_{in}+d_{out}-n} = 2^{208.5}$

Complexity trade-off : $\uparrow p$ and $\downarrow d_{in}$

Key Recovery Rounds

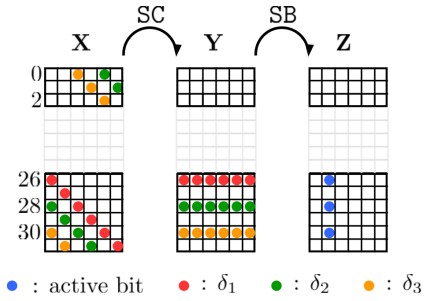


Complexity Trade-off

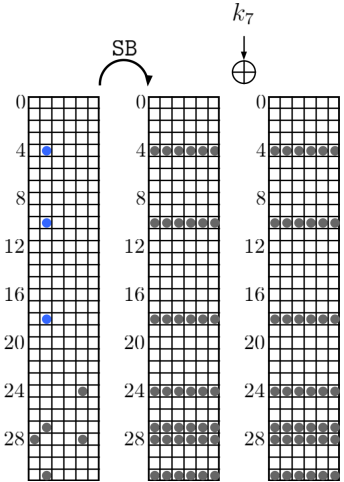
Consider the differences that activate at most 3 rows in input.

- Add $2^{-2.69}$ factor to p.
- Drastically reduces d_{in}.

- Data : $2^{187.28}$
- Pair Sieving : $2^{186.42}$



Key Recovery: Free Key Bits



Key Recovery: Early Abort

1. Consider a propagation condition.

Key Recovery: Early Abort

1. Consider a propagation condition.
2. Guess the needed key bits that we do not know yet: 2^{g_i}.

Key Recovery: Early Abort

1. Consider a propagation condition.
2. Guess the needed key bits that we do not know yet: 2^{g_i}.
3. Discard the (pair,key) that do not follow the condition: 2^{-f_i}.

Key Recovery: Early Abort

1. Consider a propagation condition.
2. Guess the needed key bits that we do not know yet: 2^{g_i}.
3. Discard the (pair,key) that do not follow the condition: 2^{-f_i}.
4. Repeat until we know all the key.

Key Recovery: Early Abort

1. Consider a propagation condition.
2. Guess the needed key bits that we do not know yet: 2^{g_i}.
3. Discard the (pair,key) that do not follow the condition: 2^{-f_i}.
4. Repeat until we know all the key.

$$C_{KR} = 2^{g_0} + 2^{g_0 - f_0}(2^{g_1} + 2^{g_1 - f_1}(\cdots(2^{g_l})\cdots)).$$

Key Recovery: Early Abort

1. Consider a propagation condition.
2. Guess the needed key bits that we do not know yet: 2^{g_i}.
3. Discard the (pair,key) that do not follow the condition: 2^{-f_i}.
4. Repeat until we know all the key.

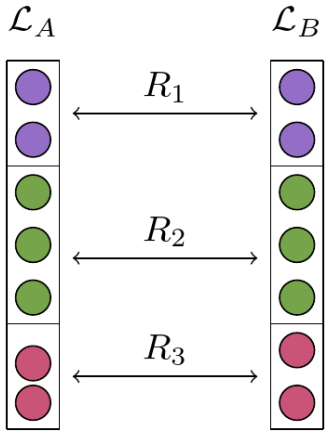
$$C_{KR} = 2^{g_0} + 2^{g_0 - f_0}(2^{g_1} + 2^{g_1 - f_1}(\cdots(2^{g_l})\cdots)).$$

→ The sooner we sieve, the better the complexity.

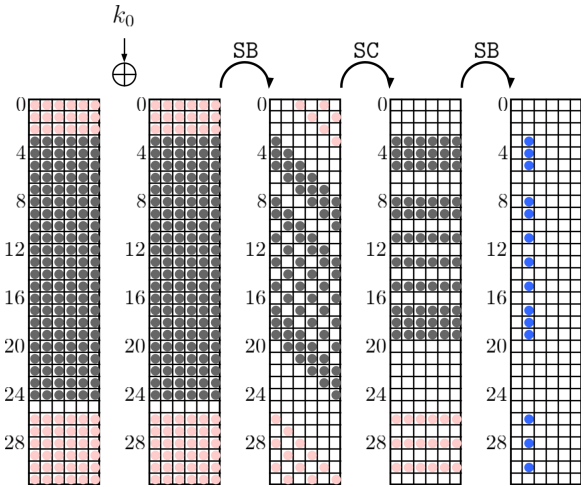
Parallel Matching

Introduced by M. Naya-Plasencia (Crypto 2011).

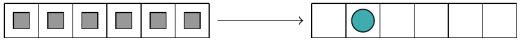
Input: $\mathcal{L}_A, \mathcal{L}_B$
R st. $R(a,b) \Leftrightarrow R_i(a_i, b_i)$
Output: $S = \{(a,b) | R(a,b)\}$



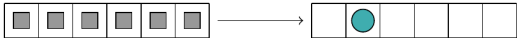
Key Recovery: 2nd S-Box



Key Recovery: 2nd S-Box



Key Recovery: 2nd S-Box



- Precompute the state before the S-Box by considering

$$x \mapsto \left(S^{-1}(x), S^{-1}(x \oplus [0,1,0,0,0,0])\right).$$

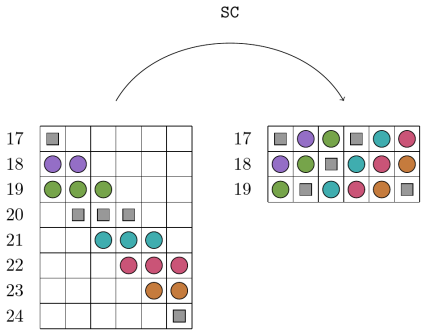
- Store the 2^6 pairs in a table.

Key Recovery: 2nd S-Box

Assume we know [18,19,21,22,23] before SC.

- Row 17 : $2^{6-2\times4} = 2^{-2}$
- Row 18 : $2^{6-2\times5} = 2^{-4}$
- Row 19 : $2^{6-2\times4} = 2^{-2}$

Overall filter: 2^{-8}



Sieve is possible without knowing the whole row
$\rightarrow$ fewer key guesses needed to start sieving.

Key Recovery: 1st S-Box

To recover [18,19,21,22,23]:

- $[19, 21] : 2^{2.02+2.07} = 2^{4.09}$
- $[18, 22, 23] : 2^{2.17+1+0.51} = 2^{3.68}$

Parallel matching: $2^{4.94}$ (instead of $2^{7.77}$)

row							Key determined	Key left	Differential Filter	Fixed bits	First S-box Cost
0	0	1	2	3	4	5	2	4	*	*	*
1	6	7	8	9	10	11	1	5	*	*	*
2	12	13	14	15	16	17	1	5	*	*	*
3	18	19	20	21	22	23	1	5	0.42	4	1.42
4	24	25	26	27	28	29	3	3	0.48	4	-0.52
5	30	31	32	33	34	35	1	5	0.07	3	2.07
6	36	37	38	39	40	41	2	4	0.07	3	1.07
7	42	43	44	45	46	47	0	6	0.07	3	3.07
8	48	49	50	51	52	53	2	4	0	2	2
9	54	55	56	57	58	59	3	3	0.02	2	1.02
10	60	61	62	63	64	65	1	5	0.05	3	2.05
11	66	67	68	69	70	71	1	5	0.07	3	2.07
12	72	73	74	75	76	77	1	5	0.12	3	2.12
13	78	79	80	81	82	83	2	4	0.02	2	2.02
14	84	85	86	87	88	89	2	4	0.07	3	1.07
15	90	91	92	93	94	95	0	6	0.09	3	3.09
16	96	97	98	99	100	101	1	5	0.07	3	2.07
17	102	103	104	105	106	107	0	6	0.09	3	3.09
18	108	109	110	111	112	113	3	3	0	2	1
19	114	115	116	117	118	119	2	4	0.02	2	2.02
20	120	121	122	123	124	125	1	5	0	2	3
21	126	127	128	129	130	131	1	5	0.07	3	2.07
22	132	133	134	135	136	137	1	5	0.17	3	2.17
23	138	139	140	141	142	143	2	4	0.51	4	0.51
24	144	145	146	147	148	149	1	5	1.42	5	1.42
25	150	151	152	153	154	155	0	6	*	*	*
26	156	157	158	159	160	161	1	5	*	*	*
27	162	163	164	165	166	167	2	4	*	*	*
28	168	169	170	171	172	173	1	5	*	*	*
29	174	175	176	177	178	179	1	5	*	*	*
30	180	181	182	183	184	185	1	5	*	*	*
31	186	187	188	189	190	191	1	5	*	*	*

Conclusion

- $1 \rightarrow 1$ S-Box transitions lead to high probability trail.
- Sieving techniques and trade-offs
- Refined key recovery

Data $2^{187.28}$, Time $2^{187.84}$, Memory 2^{42}

Contradicts the designers claim

Conclusion

- $1 \rightarrow 1$ S-Box transitions lead to high probability trail.
- Sieving techniques and trade-offs
- Refined key recovery

Data $2^{187.28}$, Time $2^{187.84}$, Memory 2^{42}

Contradicts the designers claim

Thank you for your attention