

Persistent Fault Model: Generalization, Cryptanalysis and Countermeasures

Viet-Sang Nguyen

Journée C2 at Najac, France

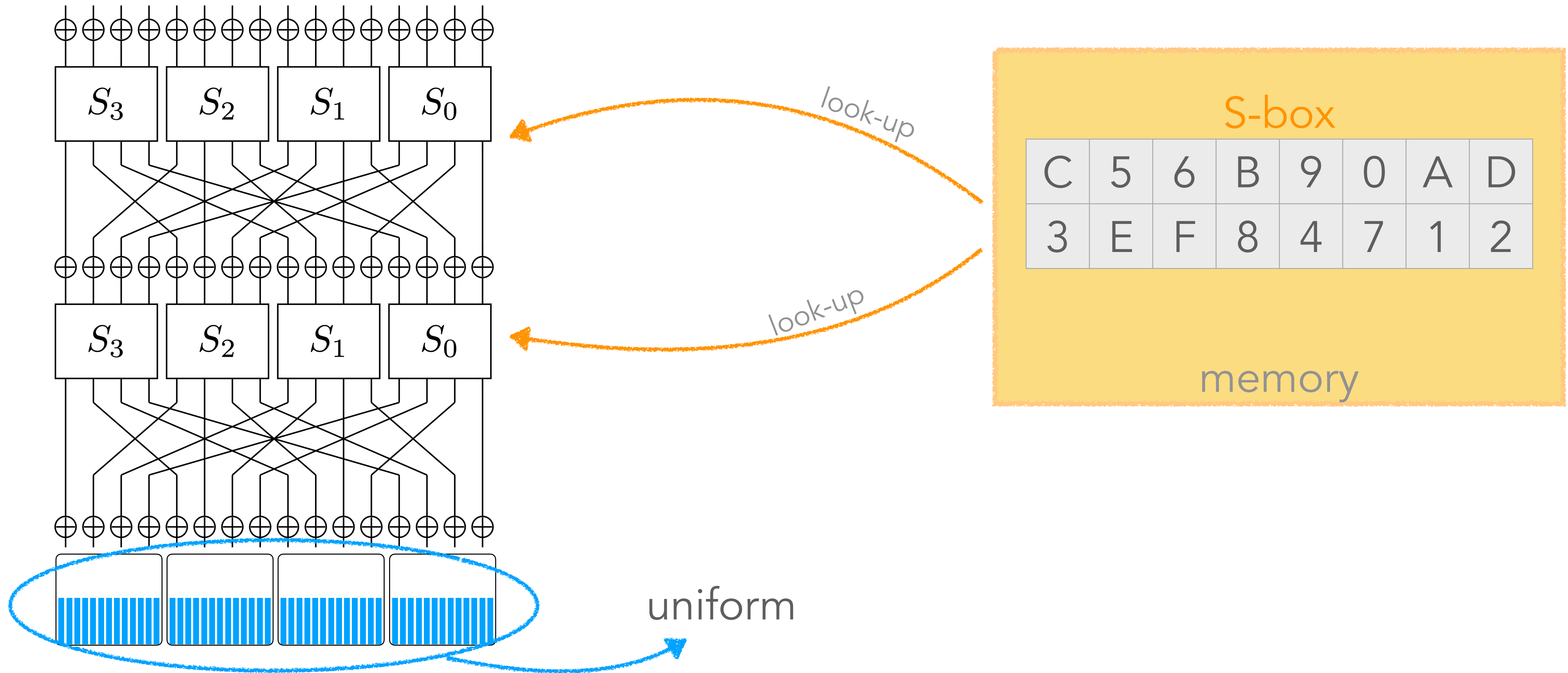
October 16, 2023

(joint work with Vincent Grosso and Pierre-Louis Cayrel)

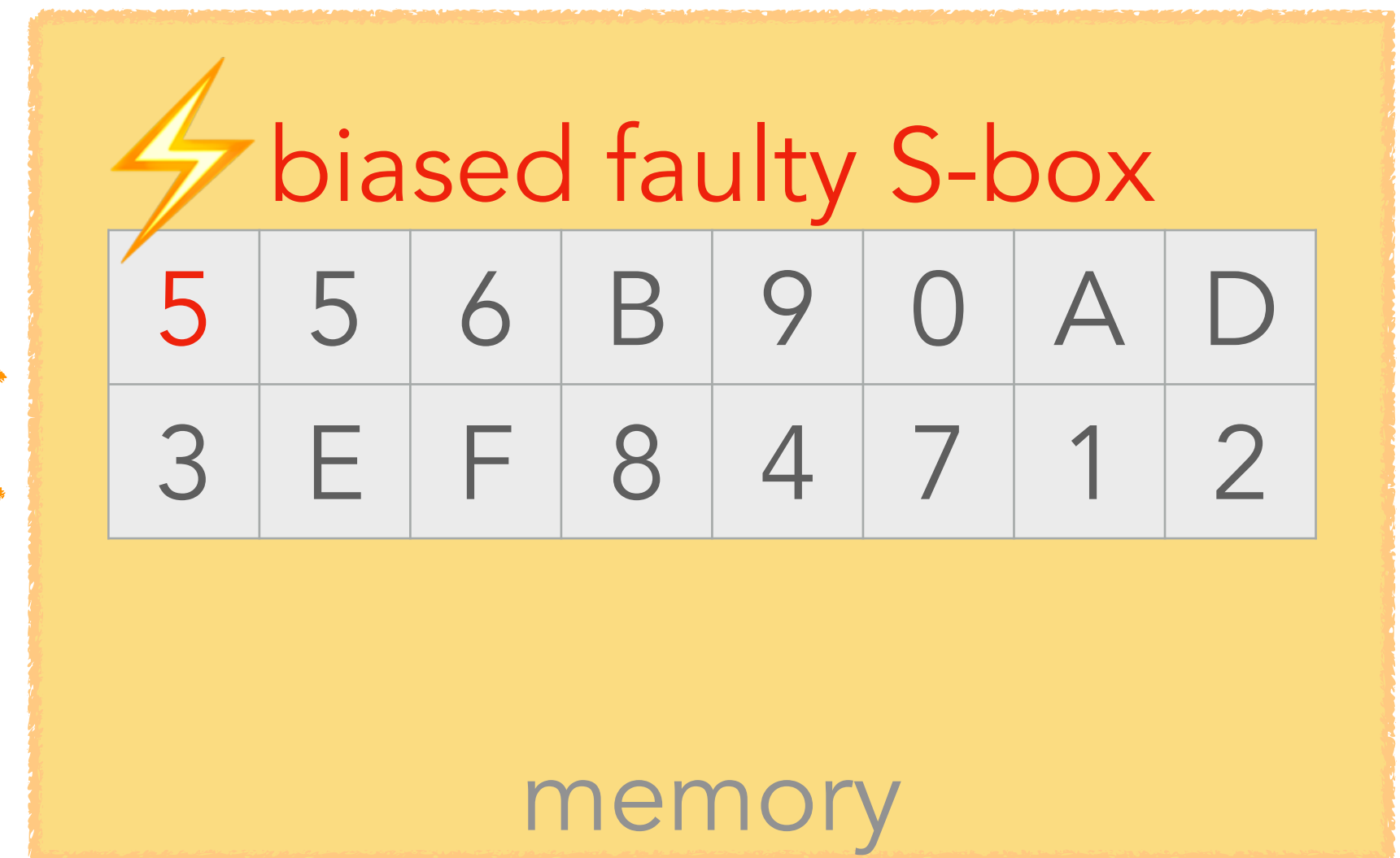
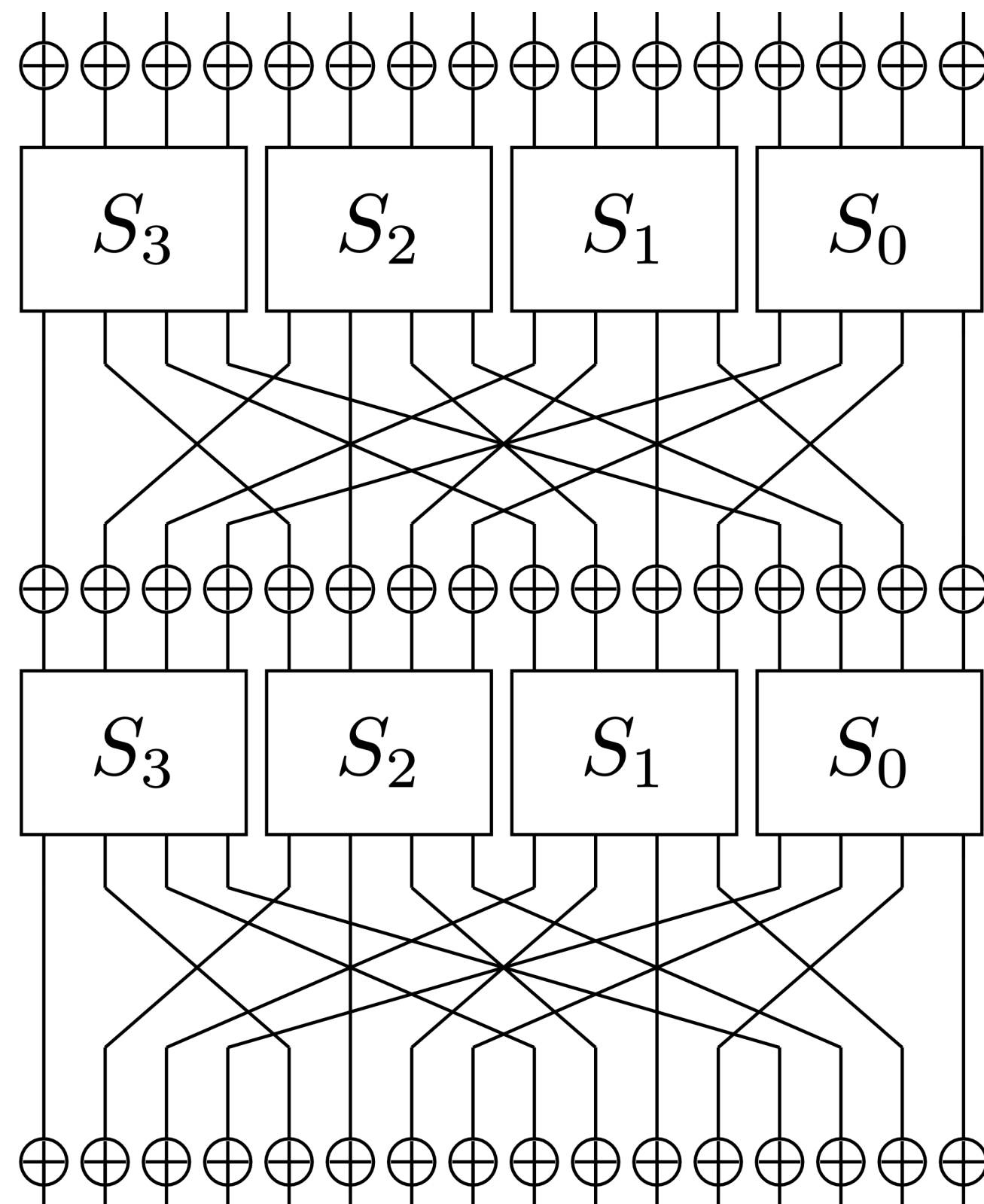


Persistent Fault Attacks

S-box in ciphers

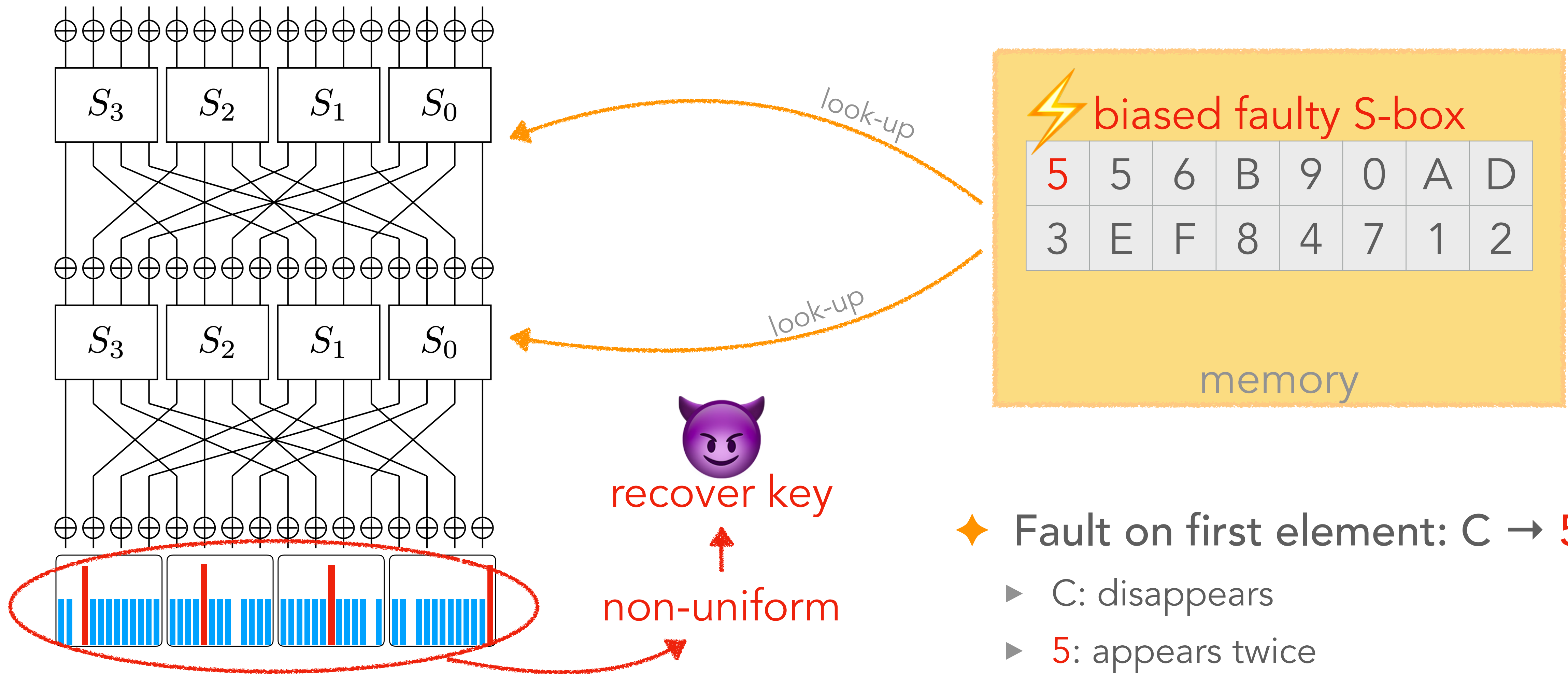


Faulting S-box



- ◆ Fault on first element: C → 5
 - ▶ C: disappears
 - ▶ 5: appears twice

Faulting S-box



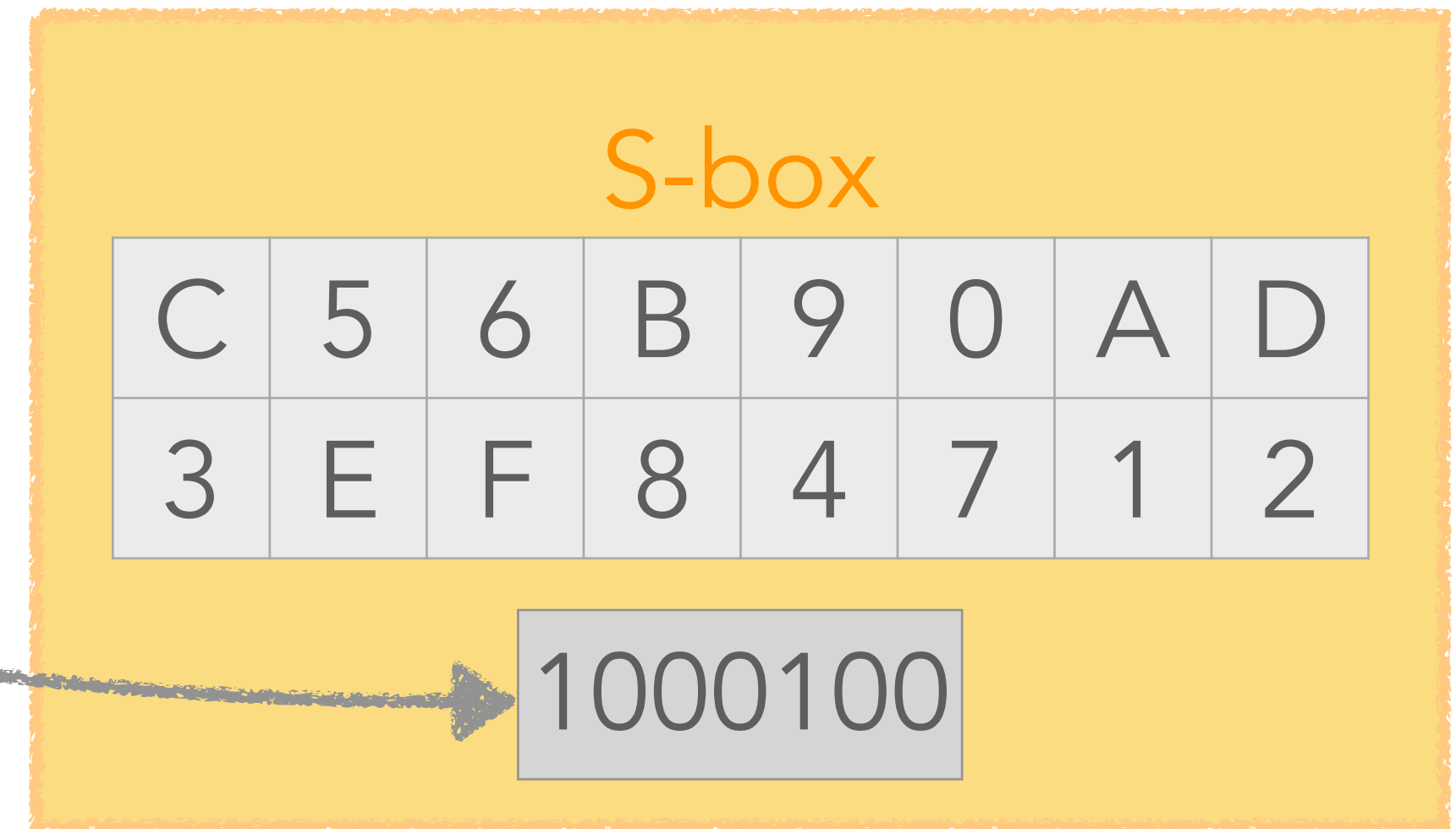
- ◆ Fault on first element: C → 5
 - ▶ C: disappears
 - ▶ 5: appears twice

Many existing attacks

- ◆ [CGR20], [ESP20], [GPT19], [PZRB19], [SBHRBM22], [TL22], [XZYZHR21], [ZHFGTRZG23], [ZLZBHDQR18], [ZZJZBZLGR20]
 - ▶ Different techniques
 - ▶ Reduce #plaintext-ciphertext pairs
- ◆ All rely on **biased faulty S-boxes**

Countermeasures

- ◆ [SM12]: add check sum (CRC)



But wait... 🤔

- ◆ [SM12]: add check sum (CRC)

bypassed !!



What if we fault both S-box and checksum ?

S-box

C	5	6	B	9	0	A	D
3	E	F	8	4	7	1	2

1000100

S-box ⚡

C	5	6	B	9	2	A	D
3	E	F	8	4	7	1	2

0000100 ⚡

Countermeasures




- ◆ [TGB23], [CM19]: detect the "bias"
 - ▶ #appearance (6): 1 ✓
 - ▶ #appearance (3): 1 ✓
 - ▶ #appearance (5): 2 ✗

 biased faulty S-box

5	5	6	B	9	0	A	D
3	E	F	8	4	7	1	2

But wait...

◆ [TGB23], [CM19]: detect the "bias"

- ▶ #appearance (6): 1 
- ▶ #appearance (3): 1 
- ▶ #appearance (5): 2 

bypassed !!



What if we swap 2 elements ?

 biased faulty S-box

5	5	6	B	9	0	A	D
3	E	F	8	4	7	1	2

non-biased faulty S-box

C	5	6	B	9	0	A	D
3	F	E	8	4	7	1	2

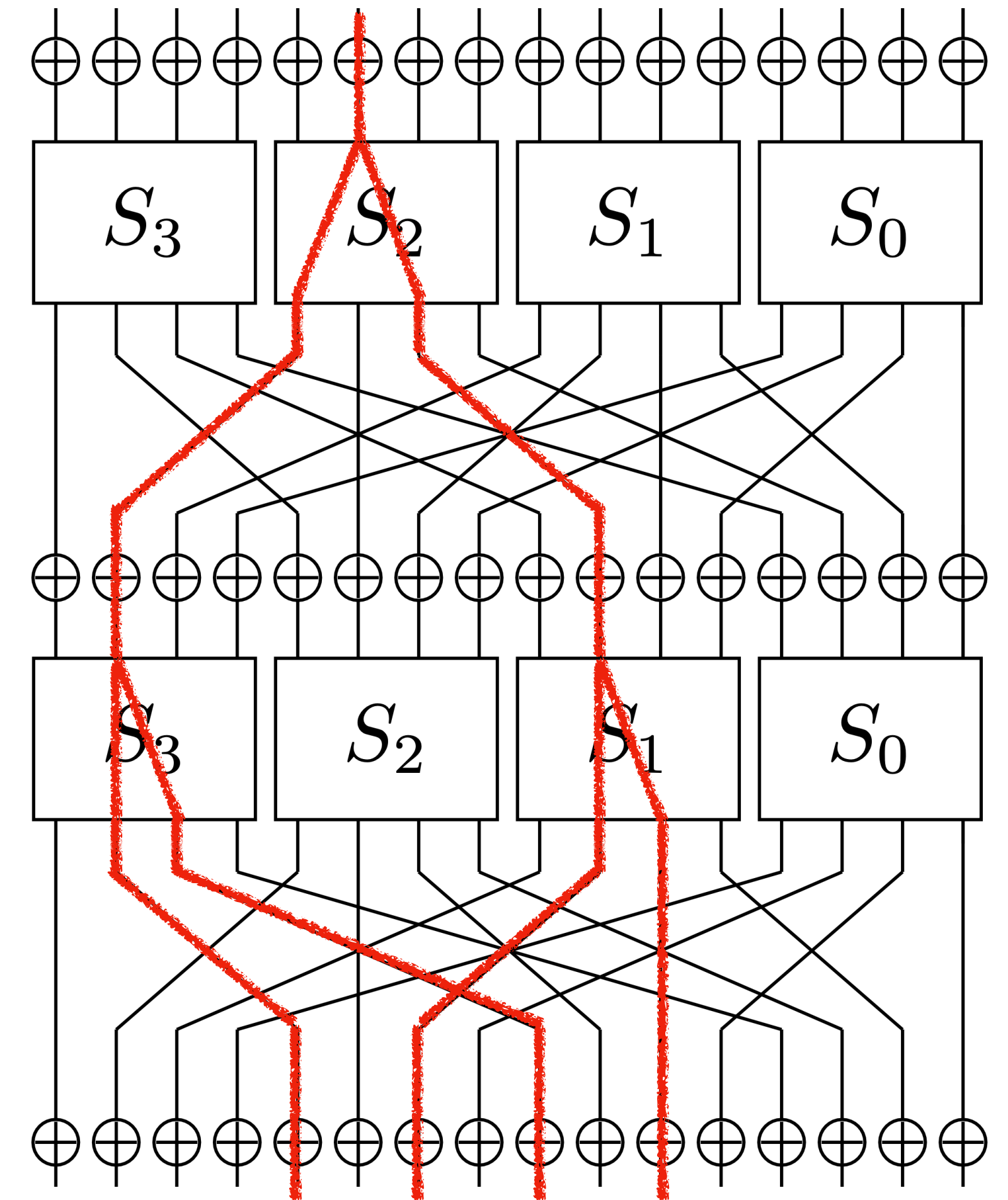




**After bypassing,
still possible to recover key**

Classical linear attack

- ◆ [Matsui94] exploits weakness of S-box
 - ▶ Statistical analysis on many plaintext-ciphertext pairs
- ◆ We apply on PRESENT (80-bit key)
 - ▶ Recover a bits (advantage)
 - ▶ Brute-force $80 - a$ bits



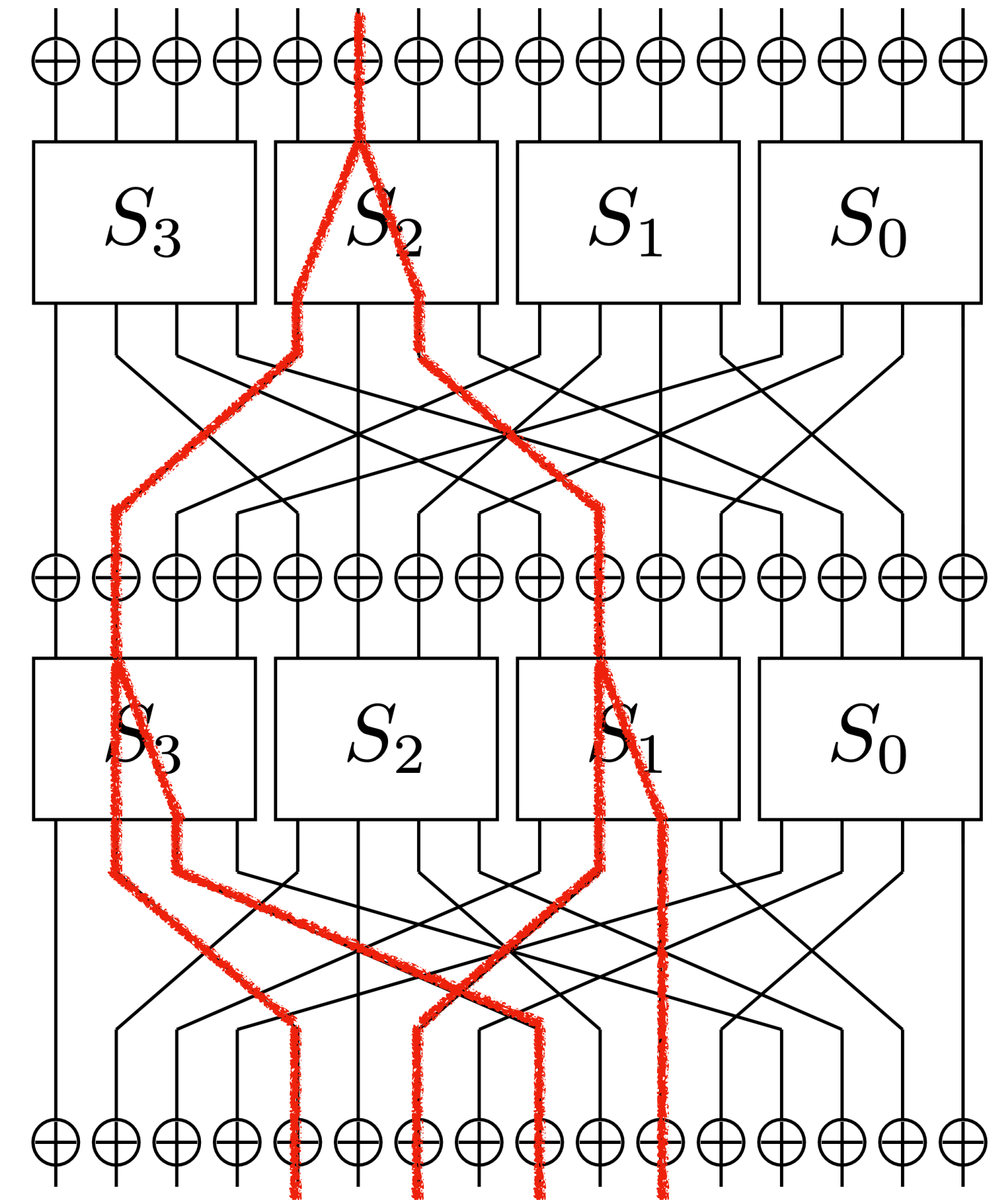
Classical linear attack

- ◆ [Matsui94] exploits weakness of S-box
 - ▶ Statistical analysis on many plaintext-ciphertext pairs
- ◆ We apply on PRESENT (80-bit key)
 - ▶ Recover a bits (advantage)
 - ▶ Brute-force $80 - a$ bits



Data complexity N ?

Success probability P_S ?



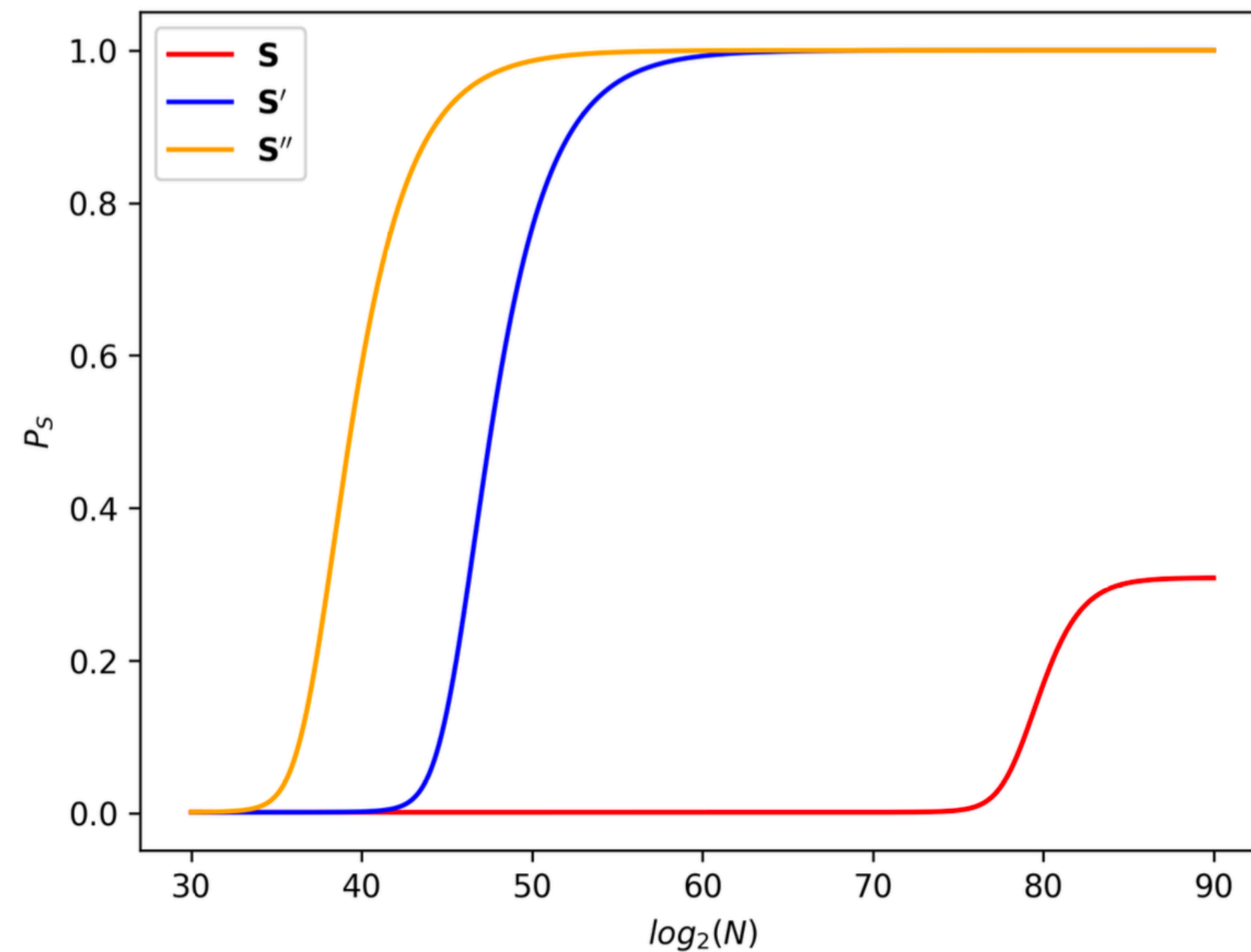
Linear attack with faulty S-boxes

	x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Orig.	$S'(x)$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2
2 swaps	$S'(x)$	C	5	6	B	9	0	A	3	D	E	F	8	4	7	1	2
3 swaps	$S''(x)$	C	5	8	B	9	0	A	D	3	6	F	E	4	7	1	2

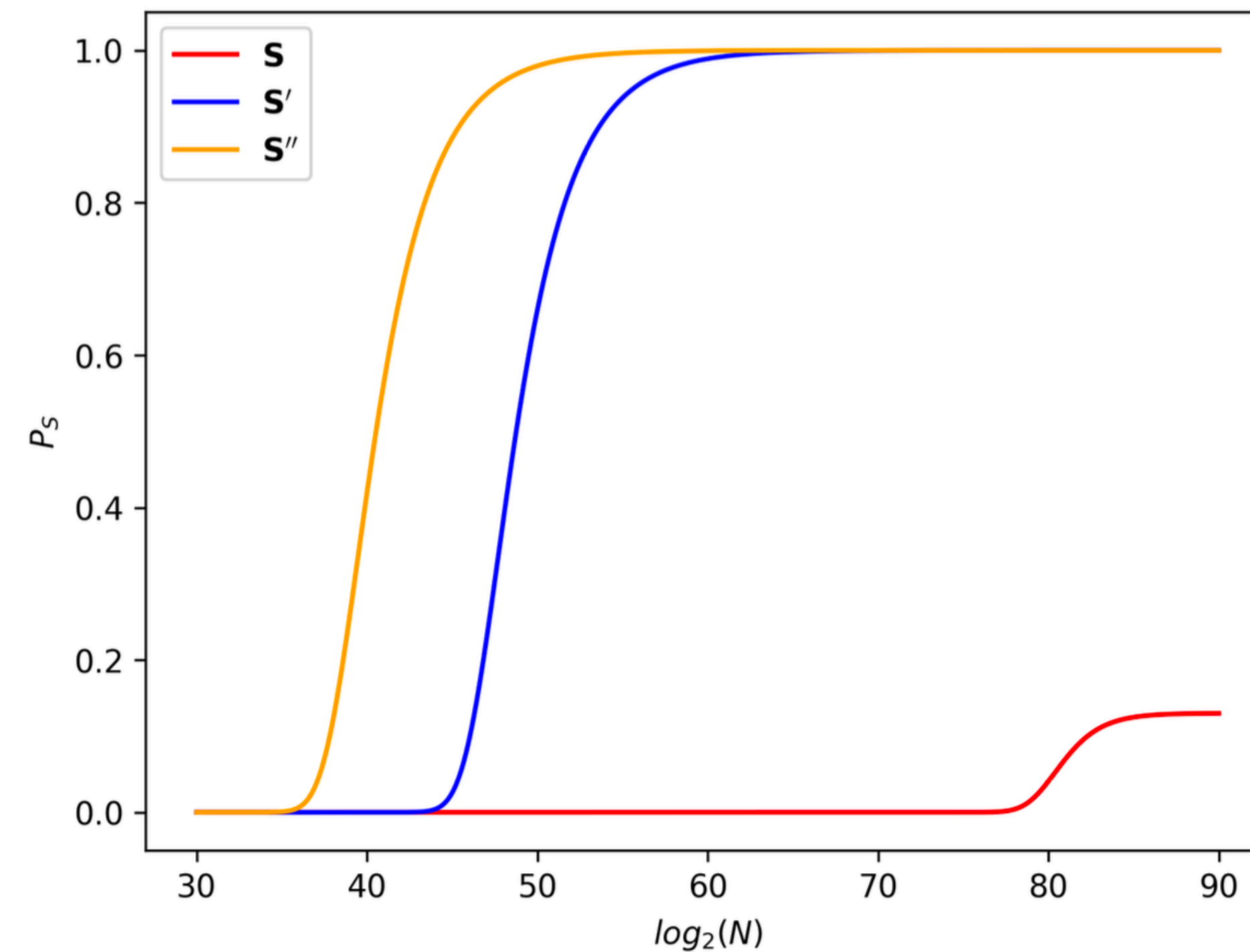
Linear attack with faulty S-boxes

	x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Orig.	$S(x)$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2
2 swaps	$S'(x)$	C	5	6	B	9	0	A	3	D	E	F	8	4	7	1	2
3 swaps	$S''(x)$	C	5	8	B	9	0	A	D	3	6	F	E	4	7	1	2

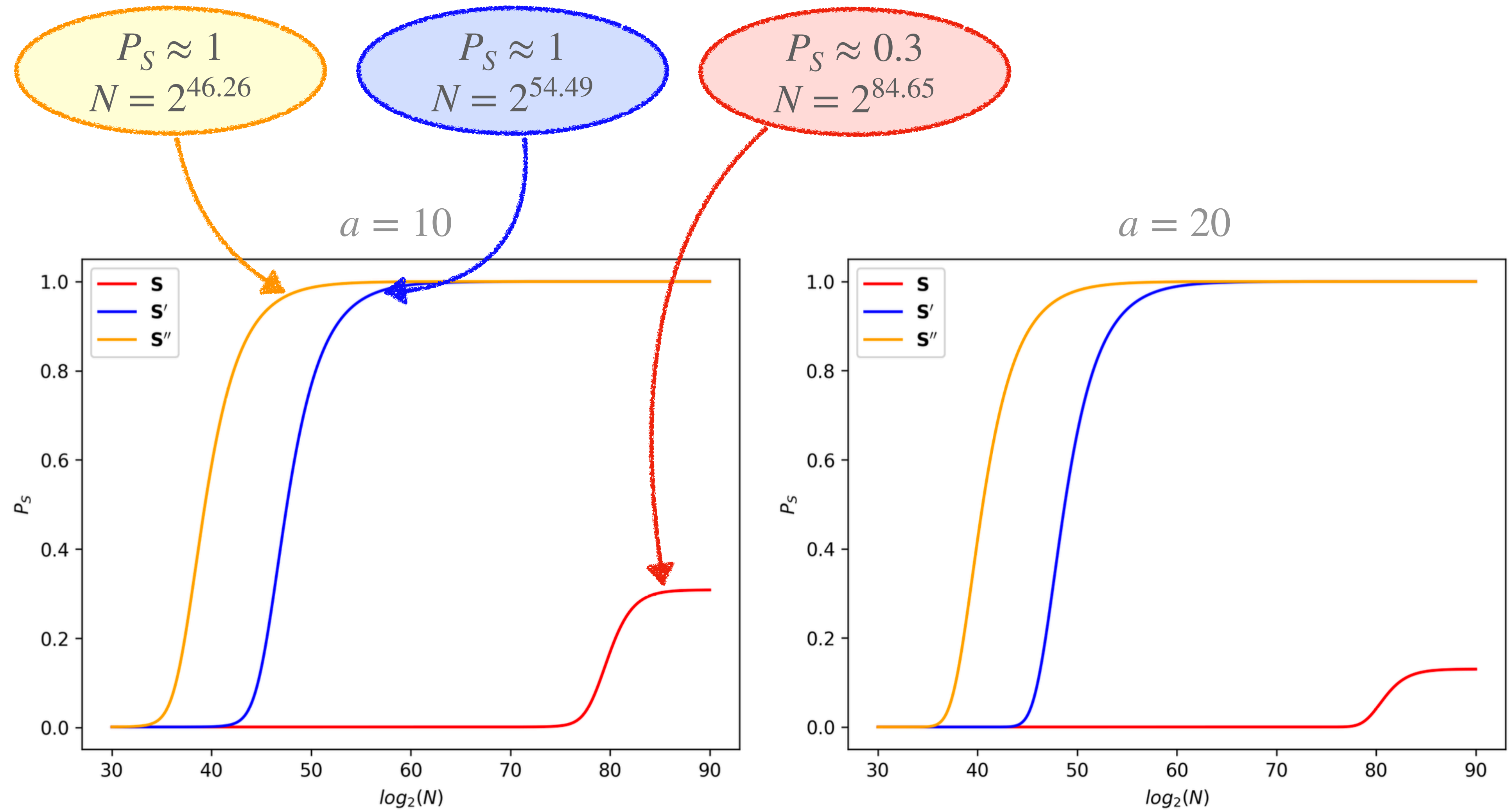
$a = 10$



$a = 20$



Linear attack with faulty S-boxes



Comparison with advanced attacks

Rounds	S-box	Data	Time	Memory	P_S	Source
27	S	$2^{63.8}$	2^{77}	2^{70}	0.95	[ZZ15]
27	S	$2^{63.8}$	$2^{77.5}$	2^{48}	0.95	[BTV18]
27	S	$2^{63.4}$	2^{72}	2^{44}	0.95	[FN20]
28	S	2^{64}	-	2^{89}	0.95	[FN20]
31	S'	$2^{54.49}$	2^{70}	2^{24}	0.95	This work
31	S''	$2^{46.26}$	2^{70}	2^{24}	0.95	This work



Generalize a Strong Model

Our proposed model

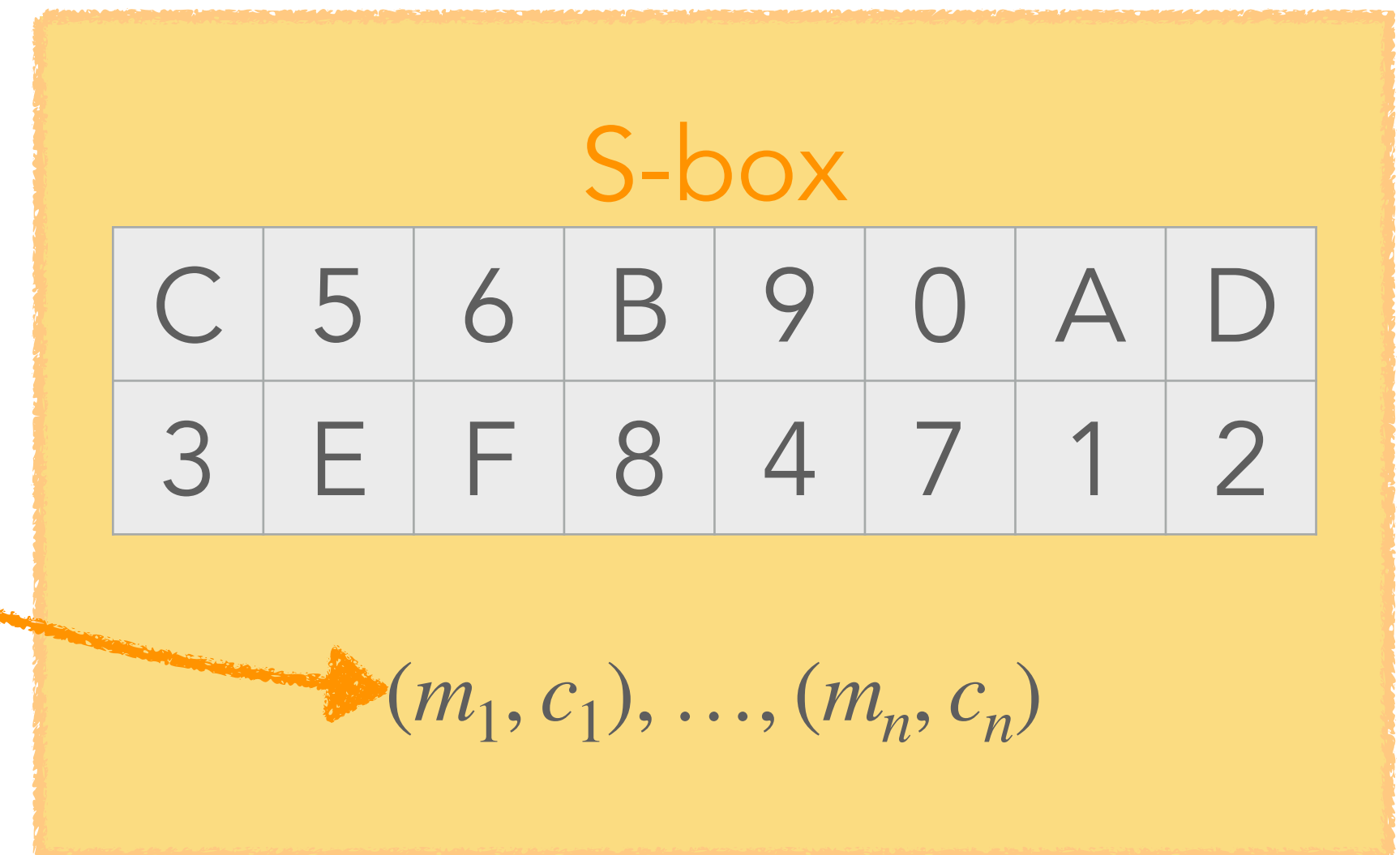
	Previous models	Our model
Biased faulty S-box	✓	✓
Non-biased faulty S-box	✗	✓
Key schedule	✗	✓
Implementations without look-up tables for S-box	✗	✓
Faulting checksum for countermeasure	✗	✓



Strong Countermeasure

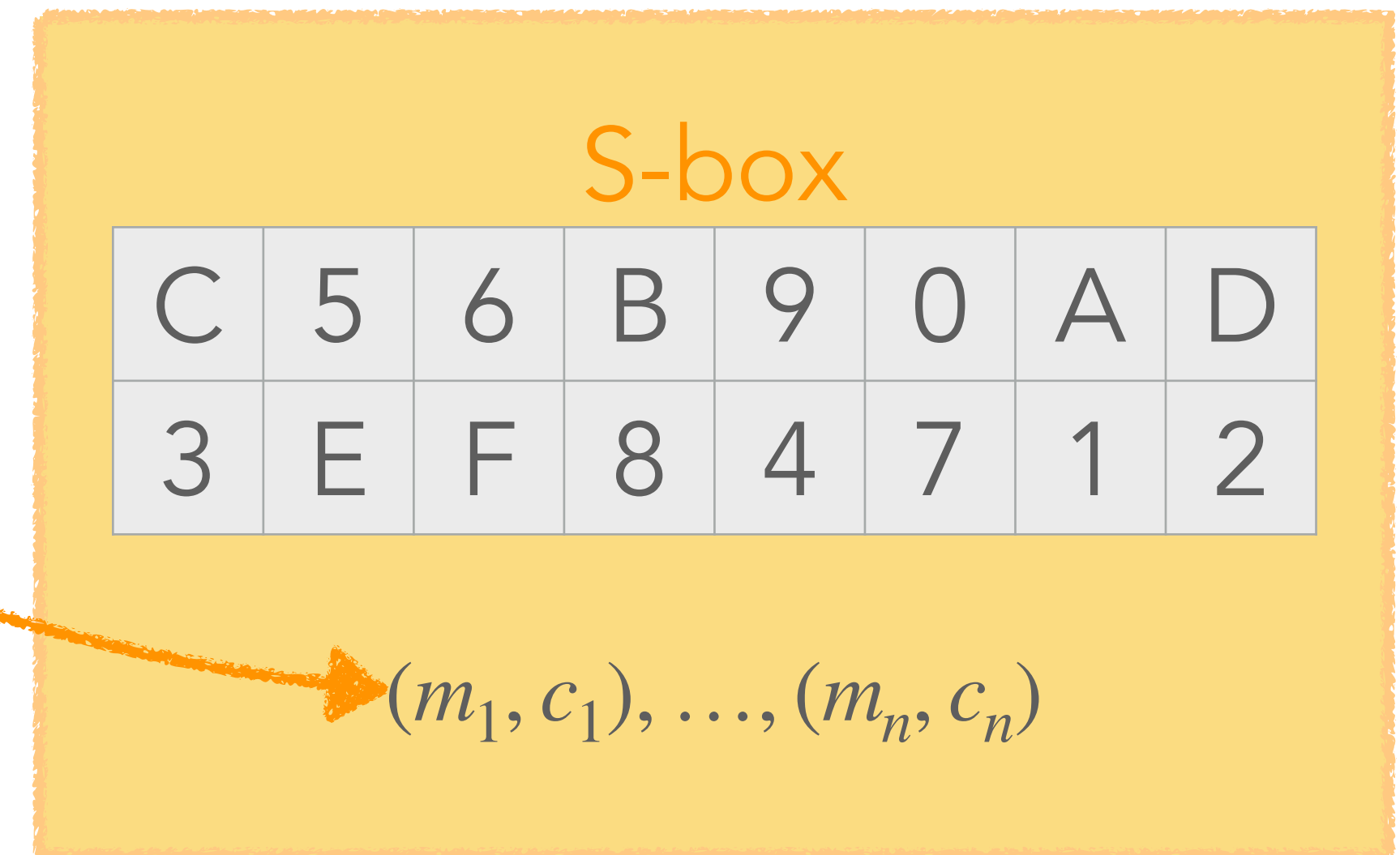
S-box in ciphers

- ◆ Store correct plaintext-ciphertext pairs
- ◆ Check encryption's correctness



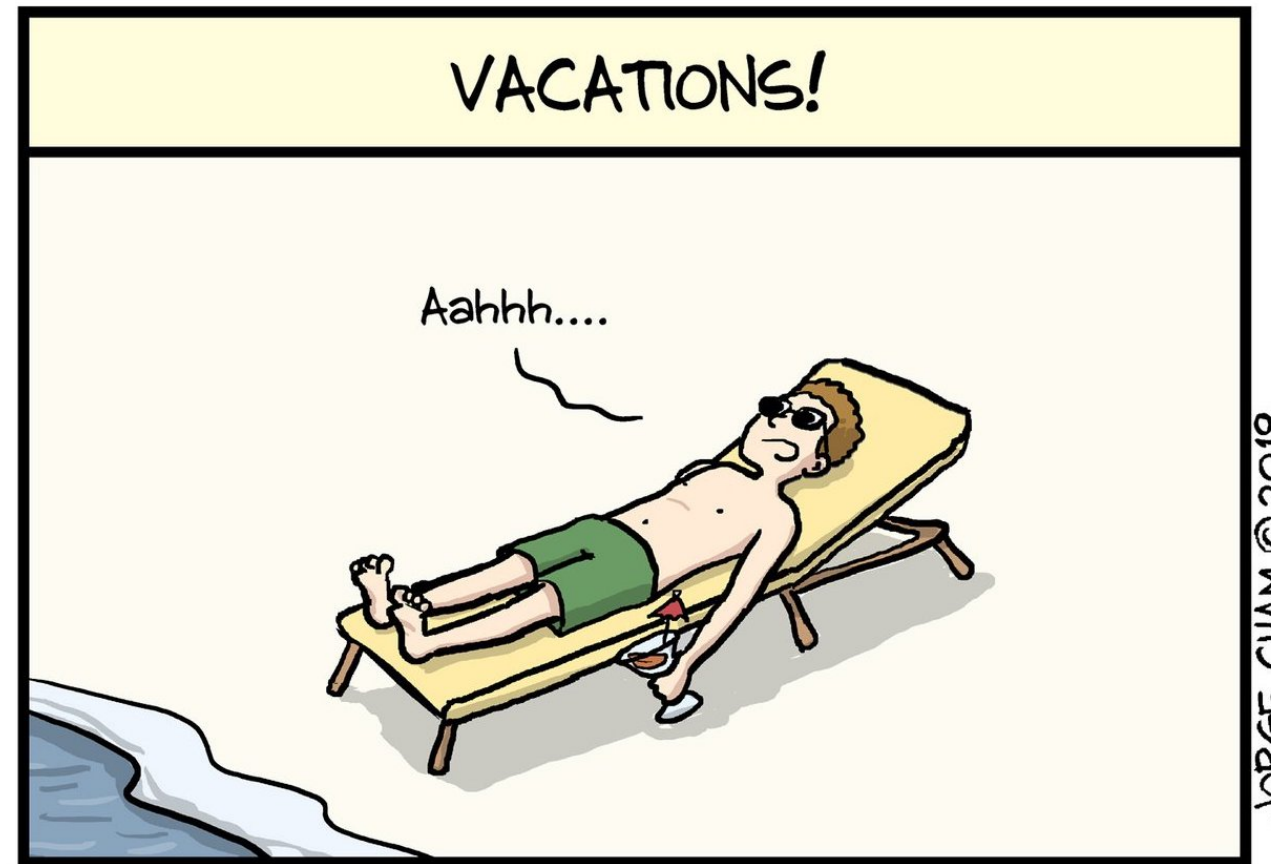
S-box in ciphers

- ◆ Store correct plaintext-ciphertext pairs
- ◆ Check encryption's correctness

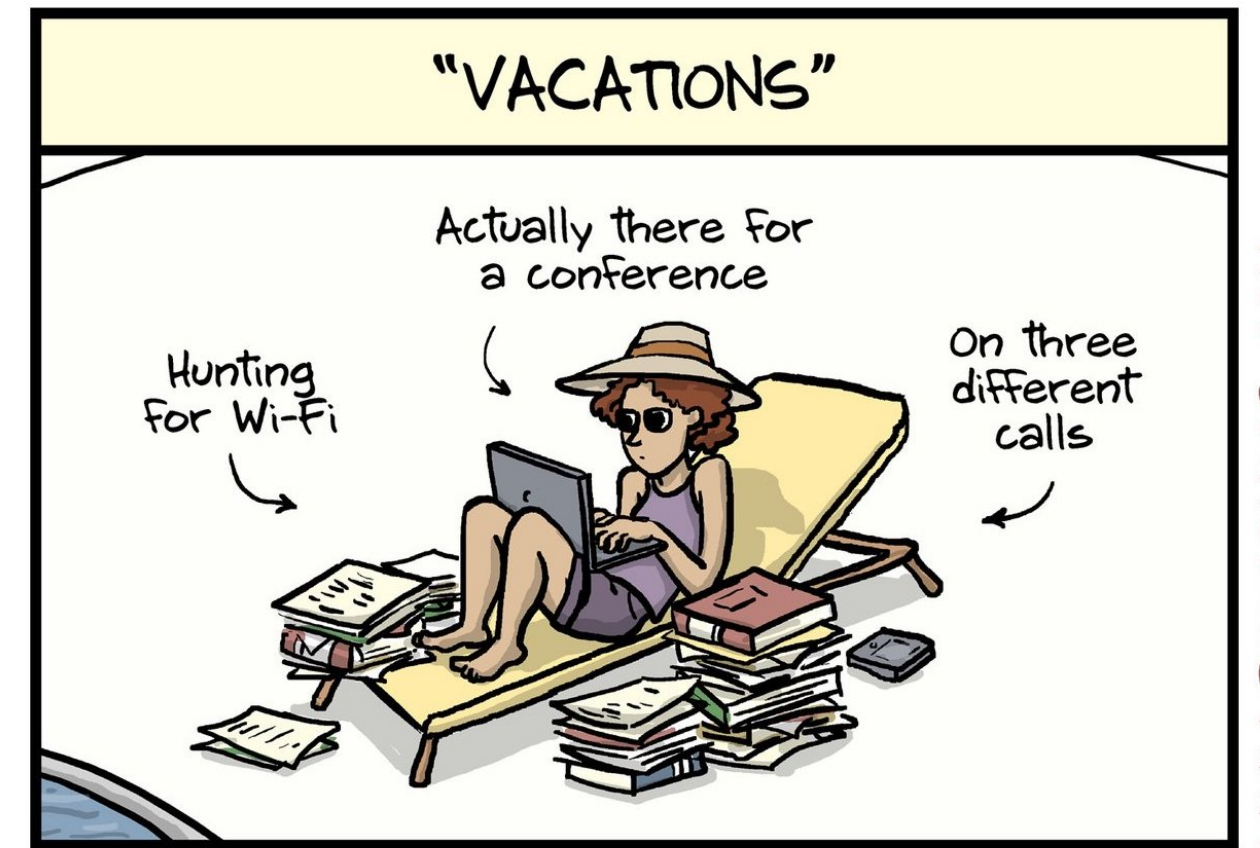


	Resistent
Biased faulty S-box	✓
Non-biased faulty S-box	✓
Faults in implementations without look-up tables	✓
Faulting checksum for countermeasure	✓

WHAT REAL PEOPLE TAKE:



WHAT ACADEMICS TAKE:



Thank you! 🙏

Any questions? 🤔

References

- ◆ [CGR20] Carré, Guilley, Rioul: “Persistent fault analysis with few encryptions”, COSADE 2020
- ◆ [ESP20] Engels, Schellenberg, Paar: “SPFA: SFA on multiple persistent faults”, FDTC 2020
- ◆ [GPT19] Gruber, Probst, Tempelmeier: “Persistent fault analysis of OCB, DEOXYs and COLM”, FDTC 2019
- ◆ [PZRB19] Pan, Zhang, Ren, Bhasin: “One fault is all it needs: Breaking higher-order masking with persistent fault analysis”, DATE 2019
- ◆ [SBHRBM22] Soleimany, Bagheri, Hadipour, Ravi, Bhasin, Mansouri: “Practical multiple persistent faults analysis”, CHES 2022
- ◆ [TL22] Tang, Liu: “MPFA: An efficient multiple faults-based persistent fault analysis method for low-cost FIA”, TCAD 2022
- ◆ [XZYZHR21] Xu, Zhang, Yang, Zhao, He, Ren: “Pushing the limit of PFA: Enhanced persistent fault analysis on block ciphers”, TCAD 2021
- ◆ [ZHFGTRZG23] Zhang, Huang, Feng, Gong, Tao, Ren, Zhao, Gou: “Efficient persistent fault analysis with small number of chosen plaintexts”, CHES 2023
- ◆ [ZLZBHDQR18] Zhang, Lou, Zhao, Bhasin, He, Ding, Qureshi, Ren: “Efficient persistent fault analysis with small number of chosen plaintexts”, CHES 2018
- ◆ [ZZJZBZLGR20] Zhang, Zhang, Jiang, Zhu, Bhasin, Zhao, Liu, Gu, Ren: “Persistent fault attack in practice”, CHES 2020
- ◆ [TGB23] Tissot, Grosso, Bossuet: “BALoo: First and efficient countermeasure dedicated to persistent fault attacks”, IOLTS 2023
- ◆ [CB19] Caforio, Banik: “A study of persistent fault analysis”, SPACE 2019
- ◆ [SM12] Schmidt, Medwed: “Countermeasures for symmetric key ciphers”, chapter in Fault Analysis in Cryptography 2012
- ◆ [Matsui94] Matsui: “The first experimental cryptanalysis of the data encryption standard”, CRYPTO 1994
- ◆ [FN20] Flórez-Gutiérrez, Naya-Plasencia: “Improving key-recovery in linear attacks: Application to 28-round PRESENT”, EUROCRYPT 2020
- ◆ [ZZ15] Zheng, Zhang: “FFT-based multidimensional linear attack on PRESENT using the 2-bit-fixed characteristic”, SCN15
- ◆ [BTV18] Bogdanov, Tishhauser, Vejre: “Multivariate profiling of hulls for linear cryptanalysis”, FSE 2018