Improved Fault Analysis on SIMON Block Cipher Family

Hua Chen Jingyi Feng Vincent Rijmen Yunwen Liu
Limin Fan Wei Li

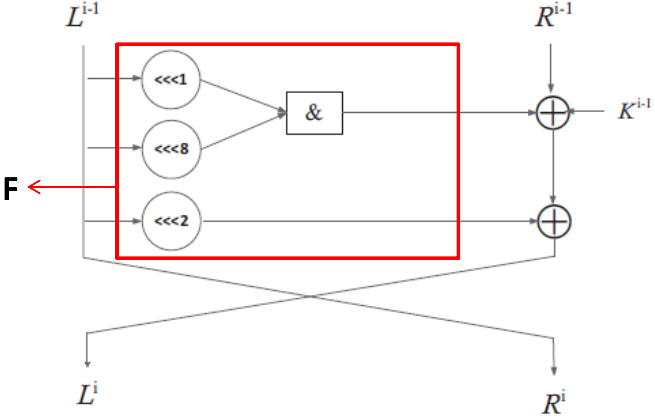
Presenter: Svetla Nikova

ESAT/COSIC, KU Leuven, and iMinds, Belgium

FDTC 2016, 16 Aug, 2016

Motivation

- SIMON is a lightweight block cipher family proposed in 2013.
- It employs a Feistel-type structure with $2n$-bit block size and mn-bit key size.



Motivations

Parameter list for the instances of SIMON family

block size $2n$	key size mn	word size n	key words m	rounds T
32	64	16	4	32
48	72	24	3	36
48	96	24	4	36
64	96	32	3	42
64	128	32	4	44
96	96	48	2	52
96	144	48	3	54
128	128	64	2	68
128	192	64	3	69
128	256	64	4	72

Motivation

- Since SIMON is presented, its implementation security has also caught attention, such as Fault Attack.
- In FDTC 2014, the first Fault Attack against SIMON was presented.
  - Byte and bit injection fault model are both adopted.
  - For the keysize mn, the input of *T-2*-th, *T-3*-th, *T-4*-th,...,*T-m-1*-th round is required to be injected faults respectively.
  - The average number of faults for the byte and bit injection model is respectively $mn/8$ or $mn/2$ if the injection position can be controlled.
  - When the injection position can be selected randomly, the theoretical estimation of injection numbers was not given.

Motivation

- In ICISC 2014, the second Fault Attack against SIMON was presented.
  - Instead of byte or bit fault model, n-bit fault model is adopted.(Each bit of a n-bit word is flipped with the probability 0.5)
  - For the keysize mn, the input of *T-2*-th, *T-3*-th, *T-4*-th, ..., *T-m-1*-th round is still required to be injected faults respectively.
  - A theoretical estimation of average injection numbers was given.

Motivation

- In FDTC 2015, the third Fault Attack against SIMON was proposed.
  - Bit fault model is adopted.
  - For the keysize mn, the first injected round is *T-3*-th round instead of *T-2*-th round and the total number of injected rounds is reduced half.
  - A theoretical estimation of average injection numbers was given.

Motivation

Related work of fault attacks on SIMON:

Related work	Fault model	Number of injected rounds
FDTC 2014	Random byte/bit model	m
ICISC 2014	Random n-bit model	m
FDTC 2015	Random bit model	$\lceil m/2 \rceil$

Our goal:

- Number of injected rounds : 1
- Reduce the injection numbers
- Give the theoretical estimation of injection numbers under random byte fault model, which is not given in former work.

Some properties of SIMON

Property 1 Given a $t(1 \le t \le n)$-bit difference $e = e_0e_1e_2, ...e_{t-1}$, if it is induced into L^0 from the $(s-t+1)$-th to the s-th bit position$(0 \le s \le n-1)$, (that is, $\Delta L^0_{s-t+1}\Delta L^0_{s-t+2}, ..., \Delta L^0_s = e$), then for $1 \le j \le T/2$, after the encryption of r rounds, ΔL^r satisfies:

When $r = 2j-1$,

$$\Delta L^r_i = 0, \quad s \le i \le s + (n - t - 16j + 8) \tag{1}$$

When $r = 2j$,

$$\begin{cases} \dot{\Delta} L^r_i = 0, & s+1 \le i \le s + (n - t - 16j) \\ \Delta L^r_i = e_{t-1}, i = s, \ j < (n-t)/16 \end{cases} \tag{2}$$

Some properties of SIMON

Property 1 gives a kind of differential propagation path.

Some properties of SIMON

Property 1 gives a kind of differential propagation path.

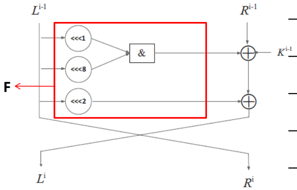
Rounds r	ΔL	ΔR
0	$000000....00e_1e_2e_3...e_t00$	$000000..0000...00000000$
1	$000..0..*********000$	$000000....00e_1e_2e_3...e_t00$
2	$000..**********e_t00$	$000..0..*********000$
3	$00...***********000$	$000..**********e_t00$
4	$00..***********e_t00$	$00...***********000$
$\vdots$	$\vdots$	$\vdots$

The differential propagation path shows:

- If the rightmost bit position of e is s, then before e is fully diffused, the s-th bit difference value of ΔL remains unchanged after even rounds' encryption.
- At the same time, e_t is followed by a number of consecutive 0s.

Some properties of SIMON

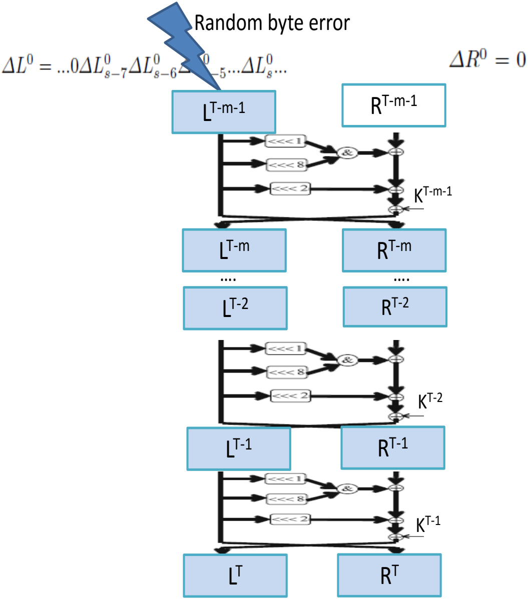
Property 2 For two n-bit differences $X = x_0x_1, ..., x_{n-1}$ and $\Delta X = \Delta x_0 \Delta x_1, ..., \Delta x_{n-1}$, let $\Delta Y = \Delta y_0 \Delta y_1, ..., \Delta y_{n-1} = F(X) \oplus F(X \oplus \Delta X)$, then some bits of $X = x_0x_1x_2, ..., x_{n-1}$ can be deduced through some bit relations between ΔX.



Δx_{i+1}	Δx_{i+8}	x_{i+1}	x_{i+8}
0	0	?	?
0	1	$\Delta y_i \oplus \Delta x_{i+2}$	?
1	0	?	$\Delta y_i \oplus \Delta x_{i+2}$
1	1	?	?

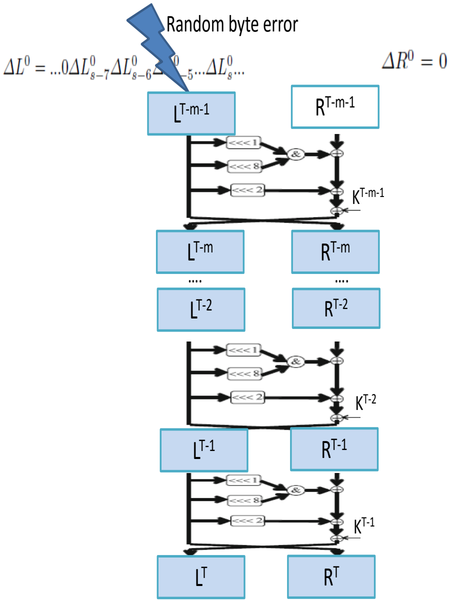
Property 2 can help to recover some bits of intermediate values, which can further reveal some bits of round keys.

Fault Attack on SIMON



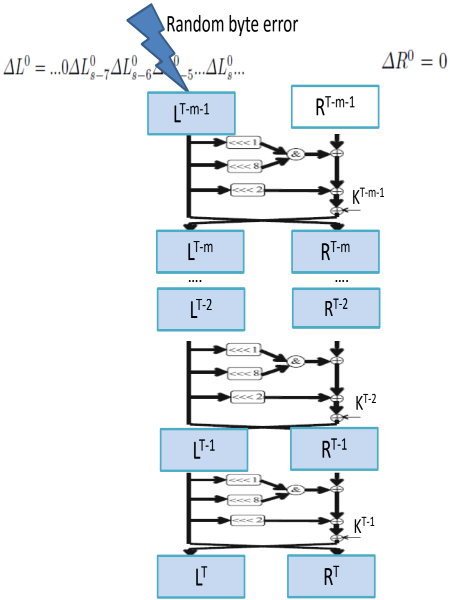
- Fault model: random byte fault
- Fault injection location: L^{T-m-1} (m=2,3 or 4 depending on the key size)

Fault Attack on Simon



Attack procedure:

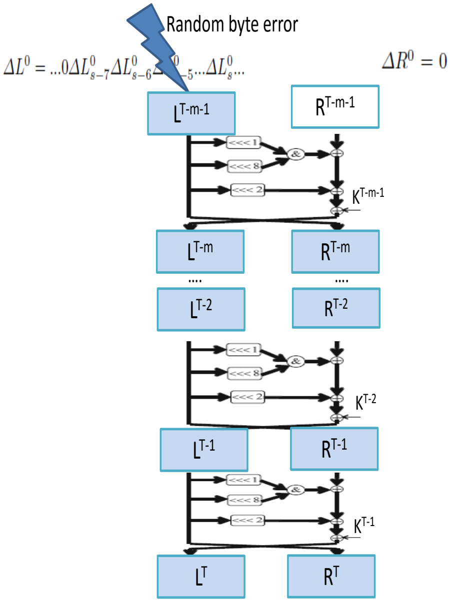
Fault Attack on SIMON



Attack procedure:

1. Select a plaintext and encrypt it correctly.

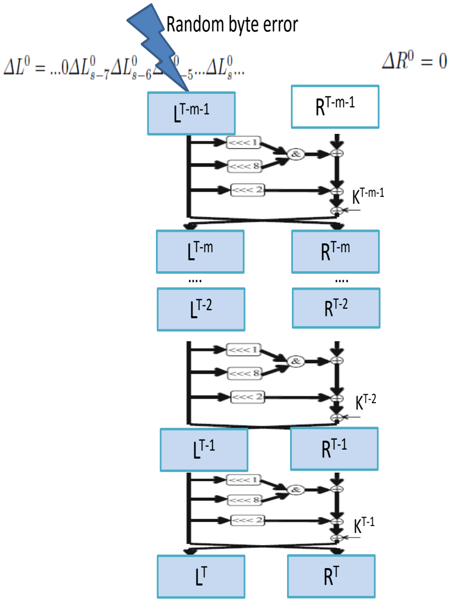
Fault Attack on Simon



Attack procedure:

1. Select a plaintext and encrypt it correctly.
2. Inject a byte fault in L^{T-m-1}.

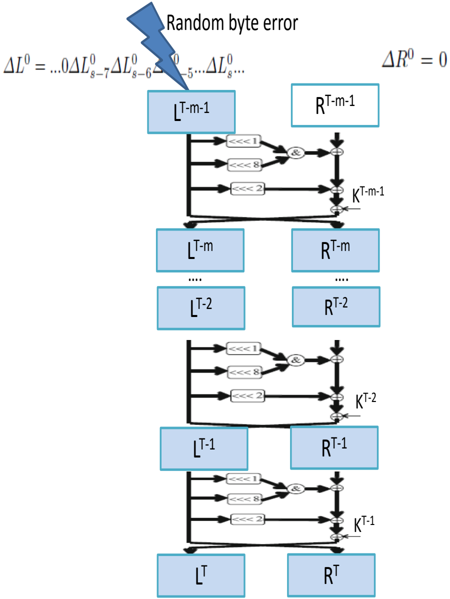
Fault Attack on SIMON



Attack procedure:

1. Select a plaintext and encrypt it correctly.
2. Inject a byte fault in L^{T-m-1}.
3. ΔL^{T-1} and ΔR^{T-1} can be easily obtained from the structure of Feistel.

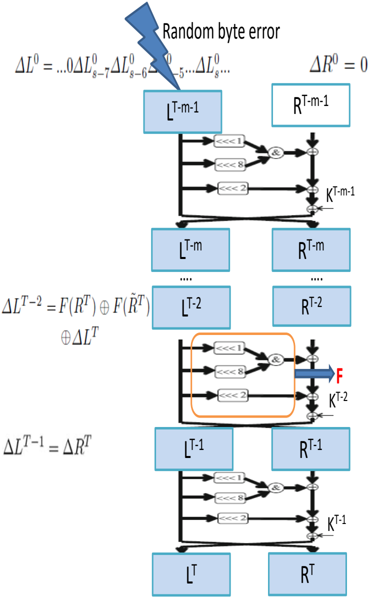
Fault Attack on SIMON



Attack procedure:

1. Select a plaintext and encrypt it correctly.
2. Inject a byte fault in L^{T-m-1}.
3. ΔL^{T-1} and ΔR^{T-1} can be easily obtained from the structure of Feistel.
4. By using property 1, the attacker can determine the rightmost bit injection position with the value 1. (e.g, if $\Delta L^0_s = 1$, then s can be determined).

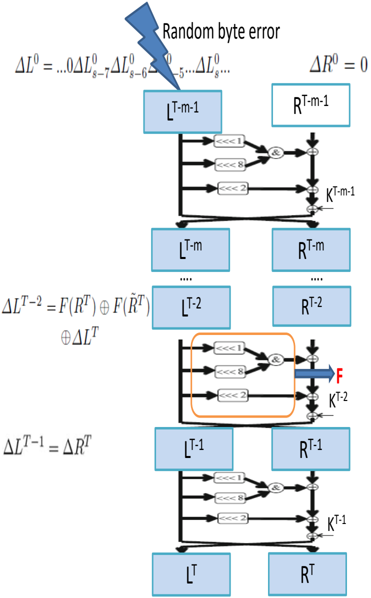
Fault Attack on SIMON



Attack procedure:

5. Compute ΔL^{T-2} and $\Delta L^{T-1} \oplus \Delta R^{T-2}$. ΔL^{T-2}, ΔL^{T-1} can be easily obtained. The whole value of ΔR^{T-2} is unknown, but some bits are 0s according to property 1. So $\Delta L^{T-1} \oplus \Delta R^{T-2}$ can be partially deduced.

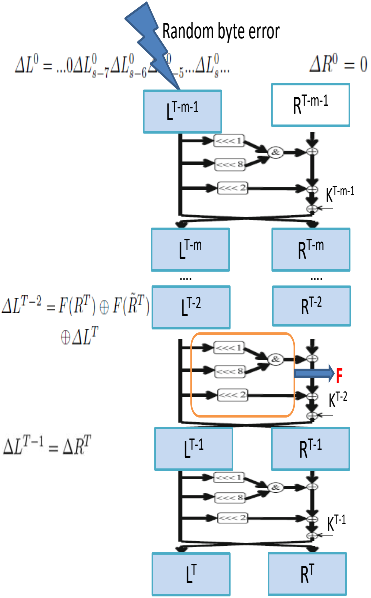
Fault Attack on SIMON



Attack procedure:

5. Compute ΔL^{T-2} and $\Delta L^{T-1} \oplus \Delta R^{T-2}$. ΔL^{T-2}, ΔL^{T-1} can be easily obtained. The whole value of ΔR^{T-2} is unknown, but some bits are 0s according to property 1. So $\Delta L^{T-1} \oplus \Delta R^{T-2}$ can be partially deduced.
6. By using property 2, some bits of L^{T-2} can be recovered, which can directly deduce some bits of K^{T-1}.

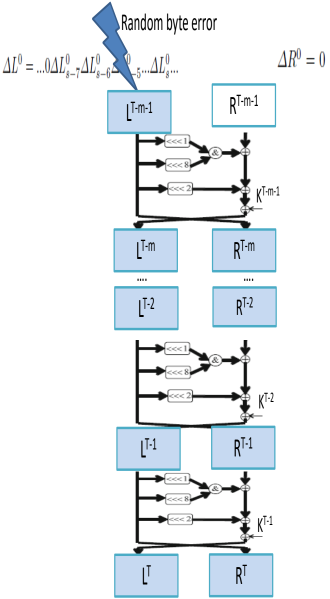
Fault Attack on SIMON



Attack procedure:

5. Compute ΔL^{T-2} and $\Delta L^{T-1} \oplus \Delta R^{T-2}$. ΔL^{T-2}, ΔL^{T-1} can be easily obtained. The whole value of ΔR^{T-2} is unknown, but some bits are 0s according to property 1. So $\Delta L^{T-1} \oplus \Delta R^{T-2}$ can be partially deduced.
6. By using property 2, some bits of L^{T-2} can be recovered, which can directly deduce some bits of K^{T-1}.
7. By repeating Step 1 to Step 6, the whole value of K^{T-1} can be extracted gradually.

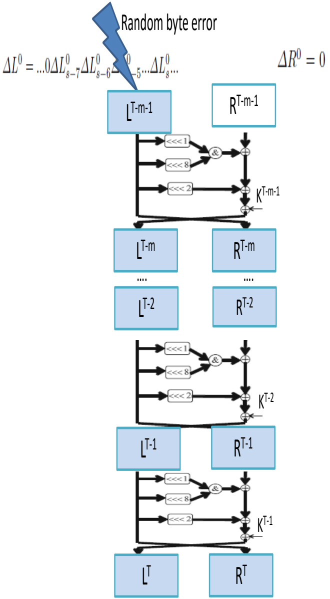
Fault Attack on SIMON



Attack procedure:

8 To recover the whole master key, K^{T-2} also requires to be recovered when $m = 2$. By partially decrypting the ciphertexts with K^{T-1}, L^{T-1} and R^{T-1} can be obtained.

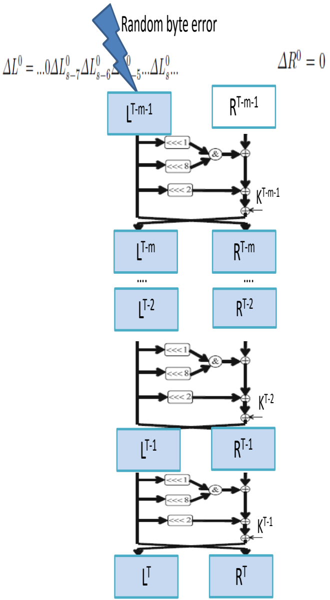
Fault Attack on SIMON



Attack procedure:

8. To recover the whole master key, K^{T-2} also requires to be recovered when $m = 2$. By partially decrypting the ciphertexts with K^{T-1}, L^{T-1} and R^{T-1} can be obtained.
9. By executing the similar steps as Step 2 to Step 7, K^{T-2} can be recovered.

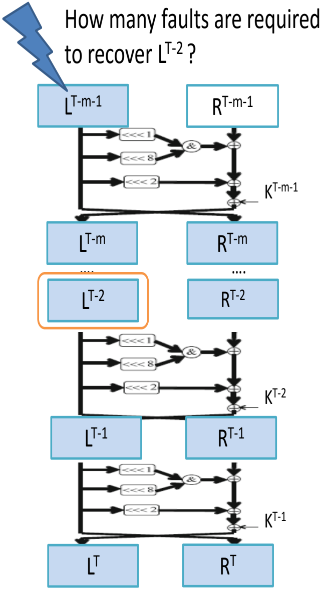
Fault Attack on SIMON



Attack procedure:

8. To recover the whole master key, K^{T-2} also requires to be recovered when $m = 2$. By partially decrypting the ciphertexts with K^{T-1}, L^{T-1} and R^{T-1} can be obtained.
9. By executing the similar steps as Step 2 to Step 7, K^{T-2} can be recovered.
10. For $m = 3$ or $m = 4$, additional round keys require to be recovered, and they can be revealed by the similar steps as Step 8 to Step 9.

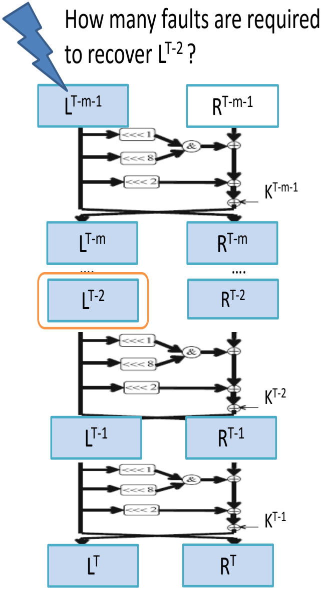
Data Complexity Analysis



Calculation procedure:

1. Calculate the probability that $\Delta L_i^{T-2} = 1$ with the fault value e injected from the $(s-7)$-th to s-th bit.

Data Complexity Analysis



Calculation procedure:

1. Calculate the probability that $\Delta L_i^{T-2} = 1$ with the fault value e injected from the $(s-7)$-th to s-th bit.
2. According to property 2, calculate the probability that L_i^{T-2} can be recovered after the fault injection. (Denoted by $U_{i,s,e}$.)

Data Complexity Analysis

3 Calculate the number of the fault injections required to recover all the bits of L^{T-2}(Denoted by f_n)

Data Complexity Analysis

3 Calculate the number of the fault injections required to recover all the bits of L^{T-2}(Denoted by f_n)

- Denote by q_i the probability that L_i^{T-2} is recovered considering all the (s, e) combinations.

$$q_i = \frac{1}{255n} \sum_{s=0}^{n-1} \sum_{e=1}^{255} U_{i,s,e}$$

Data Complexity Analysis

3 Calculate the number of the fault injections required to recover all the bits of L^{T-2}(Denoted by f_n)

- Denote by q_i the probability that L_i^{T-2} is recovered considering all the (s, e) combinations.

$$q_i = \frac{1}{255n} \sum_{s=0}^{n-1} \sum_{e=1}^{255} U_{i,s,e}$$

- q_i^l represents the probability that L_i^{T-2} is recovered after l fault injections.

$$q_i^l = 1 - (1 - q_i)^l$$

Data Complexity Analysis

3 Calculate the number of the fault injections required to recover all the bits of L^{T-2}(Denoted by f_n)

- Denote by q_i the probability that L_i^{T-2} is recovered considering all the (s, e) combinations.

$$q_i = \frac{1}{255n} \sum_{s=0}^{n-1} \sum_{e=1}^{255} U_{i,s,e}$$

- q_i^l represents the probability that L_i^{T-2} is recovered after l fault injections.

$$q_i^l = 1 - (1 - q_i)^l$$

- Finally,

$$f_n = \sum_{l=1}^{\infty} (Q^l - Q^{l-1}) l, \quad Q^0 = 1$$

Data Complexity Analysis

4 After L^{T-2} is recovered, K^{T-1} can be deduced directly. In addition, the same correct and faulty ciphertexts to recover L^{T-2} are also used to recover L^{T-3},...,L^{T-m-1}, which corresponds to K^{T-2},...,K^{T-m} respectively. So the total number of the fault injections to extract the master key is about f_n.

Data Complexity Analysis

4 After L^{T-2} is recovered, K^{T-1} can be deduced directly. In addition, the same correct and faulty ciphertexts to recover L^{T-2} are also used to recover L^{T-3},...,L^{T-m-1}, which corresponds to K^{T-2},...,K^{T-m} respectively. So the total number of the fault injections to extract the master key is about f_n.

SIMON$2n/mn$	f_n
SIMON64/96	27.97
SIMON96/96	33.57
SIMON96/144	46.93
SIMON128/128	48.23
SIMON128/192	67.18
SIMON128/256	89.21

Applicability and Extendibility Analysis

- For SIMON with $n = 96$ or 128, our attack also works when faults are injected in the location earlier than the $(T - m - 1)$-th round.

Applicability and Extendibility Analysis

- For SIMON with $n = 96$ or 128, our attack also works when faults are injected in the location earlier than the $(T - m - 1)$-th round.
- For SIMON32/64, SIMON48/72, SIMON48/96 and SIMON64/128, our attack can not extract the whole master key with a fault injected into only one intermediate round.

Applicability and Extendibility Analysis

- For SIMON with $n = 96$ or 128, our attack also works when faults are injected in the location earlier than the $(T - m - 1)$-th round.
- For SIMON$32/64$, SIMON$48/72$, SIMON$48/96$ and SIMON$64/128$, our attack can not extract the whole master key with a fault injected into only one intermediate round.
- Besides random byte fault model, our attack is also applicable to random t-bit fault model with the similar attack procedure.

PC verification

PC verification

- Experimental number of the fault injections

SIMON$2n/mn$	Random n-bit model	Random bit model		Random byte model	
	ICISC 2014	FDTC 2014	FDTC 2015	FDTC 2014	**This paper**
SIMON64/96	10.45	189.44	126.29	39	31.57
SIMON96/96	7.46	210.24	105.12	42	35.08
SIMON96/144	11.19	315.36	210.24	63	50.84
SIMON128/128	7.82	299.68	149.84	60	50.55
SIMON128/192	11.73	449.52	299.68	90	72.88
SIMON128/256	15.64	599.36	299.68	120	104.82

PC verification

- Experimental number of the fault injections

SIMON$2n/mn$	Random n-bit model	Random bit model		Random byte model	
	ICISC 2014	FDTC 2014	FDTC 2015	FDTC 2014	**This paper**
SIMON64/96	10.45	189.44	126.29	39	31.57
SIMON96/96	7.46	210.24	105.12	42	35.08
SIMON96/144	11.19	315.36	210.24	63	50.84
SIMON128/128	7.82	299.68	149.84	60	50.55
SIMON128/192	11.73	449.52	299.68	90	72.88
SIMON128/256	15.64	599.36	299.68	120	104.82

- Round locations of the fault injections

SIMON$2n/mn$	Random n-bit model	Random bit model		Random byte model	
	ICISC 2014	FDTC 2014	FDTC 2015	FDTC 2014	**This paper**
SIMON64/96	L^{38},L^{39},L^{40}	L^{38},L^{39},L^{40}	L^{38},L^{39}	L^{38},L^{39},L^{40}	L^{38}
SIMON96/96	L^{49},L^{50}	L^{49},L^{50}	L^{49}	L^{49},L^{50}	L^{49}
SIMON96/144	L^{50},L^{51},L^{52}	L^{50},L^{51},L^{52}	L^{50},L^{51}	L^{50},L^{51},L^{52}	L^{50}
SIMON128/128	L^{65},L^{66}	L^{65},L^{66}	L^{65}	L^{65},L^{66}	L^{65}
SIMON128/192	L^{65},L^{66},L^{67}	L^{65},L^{66},L^{67}	L^{65},L^{66}	L^{65},L^{66},L^{67}	L^{65}
SIMON128/256	L^{67},L^{68},L^{69},L^{70}	L^{67},L^{68},L^{69},L^{70}	L^{67},L^{69}	L^{67},L^{68},L^{69},L^{70}	L^{67}

Summary

- Compared with the previous work, our attack successfully reduces the number of injected round locations to 1 for six instances of SIMON.

Summary

- Compared with the previous work, our attack successfully reduces the number of injected round locations to 1 for six instances of SIMON.
- We also give a theoretical estimation of data complexity, which shows less fault injections are required in our attack compared with other attacks under the same fault model.

Summary

- Compared with the previous work, our attack successfully reduces the number of injected round locations to 1 for six instances of SIMON.
- We also give a theoretical estimation of data complexity, which shows less fault injections are required in our attack compared with other attacks under the same fault model.
- Our method can also be extended to the random t-bit model.

Thank you!