Bit Cryptanalysis on Symmetric Ciphers

Xiaoyun Wang

Current Trends in Cryptology, Yaroslavl

2016-06-06

Outline

Outline

XOR differntial attack

- Firstly introduced by Biham and Shamir which becomes a powerful tool in cryptanalysis of block ciphers today
- Aims to analyze how particular XOR differences in plaintext pairs affect the XOR differences of the resultant ciphertext pairs
- It has been developed into many variants which were used to analyze various cipher primitives

Outline

Modular differential attack

- Introduced to analyze hash functions by Wang etc, the adversary cancels the unwanted avalanche arisen from a given input difference by various bit-carry control techniques and finds a specifically optimized differential path
- Then determines a set of sufficient bit conditions to result in the specifically differential path
- Finally the adversary fulfills various techniques including message modifications to guarantee more bit conditions hold, and this improves the success rate of the attack a lot

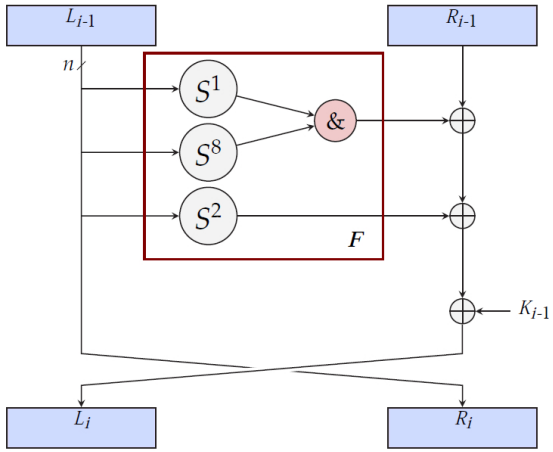
Outline

Conditional differential attack

- Introduced by Knellwolf, Willi Meier and Naya-Plasencia (2010) to analyze NLFSR-Based Cryptosystems
- Combine bit conditions and message modifications with classic differential attack
- Also applied to analyze Grain-128a(with cube attack), Trivium, and Katan etc
- Improved differential attack on Speck by Dinur
- .....

Applied to SIMON: Bitwise Round Function

Following is the joint work of Ning Wang, Xiaoyun Wang, Keting Jia and Jingyun Zhao.



$$\begin{aligned} L_i &= R_{i-1} \oplus F(L_{i-1}) \oplus K_{i-1}, \\ R_i &= L_{i-1}, \end{aligned}$$

Where, $F(x) = (x \lll 1) \cap (x \lll 8) \oplus (x \lll 2)$

Bitwise Round Functions

- Let $L^i = \{X^i_n, X^i_{n+1}, \ldots, X^i_{2n-1}\}$, $R^i = \{X^i_0, X^i_1, \ldots, X^i_{n-1}\}$, and $K^i = \{K^i_0, K^i_1, \ldots, K^i_{n-1}\}$
- Round function

$$X^i_{j+n} = (X^{i-1}_{(j+1)\%n+n} \cap X^{i-1}_{(j+8)\%n+n}) \oplus X^{i-1}_{(j+2)\%n+n} \oplus X^{i-1}_j \oplus K^{i-1}_j$$

$$X^i_j = X^{i-1}_{j+n}, j = 0, 1, ..., n-1$$

Key Schedules

- The key schedules generate a sequence of n_r round subkeys $\{K^0, ..., K^{n_r-1}\}$ from the master key $\{k_0, k_1, ..., k_{m-1}\}$. For different key lengths mn, the key schedules are given as follows, when $i = 0, 1, \ldots, m-1$, $K^i = k_i$; and when $i = m, m+1, \ldots, n_r$

$$\text{if} \quad m = 2, \quad K^i = c \oplus (z_j)_{i-m} \oplus K^{i-m} \oplus (K^{i-m+1} \ggg 3) \oplus (K^{i-m+1} \ggg 4),$$

$$\text{if} \quad m = 3, \quad K^i = c \oplus (z_j)_{i-m} \oplus K^{i-m} \oplus (K^{i-m+2} \ggg 3) \oplus (K^{i-m+2} \ggg 4),$$

$$\text{if} \quad m = 4, \quad K^i = c \oplus (z_j)_{i-m} \oplus K^{i-m} \oplus K^{i-m+1} \oplus (K^{i-m+1} \ggg 1) \oplus (K^{i-m+3} \ggg 3) \oplus (K^{i-m+3} \ggg 4).$$

Important Observation

Given two inputs X^{i-1} and $(X^{i-1})'$ of the i-th round, where $\Delta X^{i-1} = X^{i-1} \oplus (X^{i-1})'$, and an equation $\Delta X^{i+1}_{j+n} = b$, where $b = 0$ or 1, we can find all the solutions of the 2-bit subkey $(K^{i-1}_{(j+1)\%n}, K^{i-1}_{(j+8)\%n})$ which satisfies the equation depending on the following 5 cases.

1. When $(\Delta X^i_{(j+1)\%n+n}, \Delta X^i_{(j+8)\%n+n}) = (0,0)$ and $\Delta X^i_{(j+2)\%n+n} \oplus \Delta X^{i-1}_{j+n} = b \oplus 1$, there is no solution.
2. When $(\Delta X^i_{(j+1)\%n+n}, \Delta X^i_{(j+8)\%n+n}) = (0,0)$ and $\Delta X^i_{(j+2)\%n+n} \oplus \Delta X^{i-1}_{j+n} = b$, there are 4 solutions .
3. When $(\Delta X^i_{(j+1)\%n+n}, \Delta X^i_{(j+8)\%n+n}) = (0,1)$, there are two solutions.
4. When $(\Delta X^i_{(j+1)\%n+n}, \Delta X^i_{(j+8)\%n+n}) = (1,0)$, there are two solutions.
5. When $(\Delta X^i_{(j+1)\%n+n}, \Delta X^i_{(j+8)\%n+n}) = (1,1)$, there are two solutions.

This means that we can reduce the number of subkey bits guessed depending to specific values in selected path.

Differential Attack on 21-Round SIMON32

- 13-round differential with probability $2^{-28.56}$ given by Biryukov

$$D_1 : \ (0000, 0040) \rightarrow (4000, 0000).$$

Differential Attack on 21-Round SIMON32

- Bit conditions on the 4 top rounds and 4 bottom rounds

Round	Input Differences of Each Round
0	0 * 0 1, * * * *, * 0 0 0, 0 * * *, * 1 * *, * * * *, 0 * 0 *, * * * *
1	* 0 **0 0**, **0 1** * *, 0 * 0 **0**, **0 0 0** *, 0 * 0 1, * * * *, * 0 0 0, 0 * * *
2	0 * 0 0, **0 0 0 1**, * 0 0 0, 0 **0 0 0**, * 0 0 0, 0 1 * *, 0 * 0 0, 0 0 0 *
3	**0** 0 0 0, 0 0 **0 0**, 0 **1** 0 0, 0 0 0 **0**, 0 * 0 0, 0 0 0 1, * 0 0 0, 0 0 0 0
4	0 **0** 0 0, 0 0 0 0, **0** 0 0 0, 0 0 0 0, 0 0 0 0, 0 0 0 0, 0 1 0 0, 0 0 0 0
4 $\rightarrow$ 17	13-round differential D_1
17	0 1 0 0, 0 0 0 0, 0 0 0 0, 0 0 0 0, **0** 0 0 0, 0 0 0 0, 0 **0** 0 0, 0 0 0 0
18	* 0 0 0, 0 0 0 0, 0 * 0 0, 0 0 0 1, 0 **1** 0 0, 0 0 0 **0**, **0** 0 0 0, 0 0 **0 0**
19	0 * 0 0, 0 0 0 *, * 0 0 0, 0 1 * *, * 0 0 0, 0 **0 0 0**, 0 * 0 0, **0 0 0 1**
20	* 0 0 0, 0 * * *, 0 * 0 1, * * * *, 0 * 0 **0**, **0 0 0** *, * 0 **0 0**, **0 1** * *
21	0 * 0 *, * * * *, * 1 * *, * * * *, * 0 0 0, 0 * * *, 0 * 0 1, * * * *

Conditions Independent of Subkeys

- There are 10 conditions on the input difference of plaintexts, 10 conditions on ciphertext differences, 16 conditions from round 1 and round 20 are independent of subkey bits

Round	Number of Conditions	Bit Conditions of the i-th Round
0	10	$\Delta X_{16}^0 = 0, \Delta X_{18}^0 = 0, \Delta X_{19}^0 = 1, \Delta X_{25}^0 = 0, \Delta X_{26}^0 = 0,$ $\Delta X_{27}^0 = 0, \Delta X_{28}^0 = 0, \Delta X_{1}^0 = 1, \Delta X_{8}^0 = 0, \Delta X_{10}^0 = 0$
1	8	$\Delta X_{18}^1 = 0, \Delta X_{19}^1 = 0, \Delta X_{20}^1 = 0, \Delta X_{21}^1 = 1,$ $\Delta X_{27}^1 = 0, \Delta X_{28}^1 = 0, \Delta X_{29}^1 = 0, \Delta X_{30}^1 = 0$
20	8	$\Delta X_{3}^{20} = 0, \Delta X_{4}^{20} = 0, \Delta X_{5}^{20} = 0, \Delta X_{6}^{20} = 0,$ $\Delta X_{10}^{20} = 0, \Delta X_{11}^{20} = 0, \Delta X_{12}^{20} = 0, \Delta X_{13}^{20} = 1$
21	10	$\Delta X_{16}^{21} = 0, \Delta X_{18}^{21} = 0, \Delta X_{25}^{21} = 1, \Delta X_{1}^{21} = 0, \Delta X_{2}^{21} = 0,$ $\Delta X_{3}^{21} = 0, \Delta X_{4}^{21} = 0, \Delta X_{8}^{21} = 0, \Delta X_{10}^{21} = 0, \Delta X_{11}^{21} = 1$

How Many Subkey Bits Guessed by Dynamic Techniques

- There are 28 conditions in bold from rounds 2-4 and 17-19 which are related to subkey bits
- There are totally 50 bits of subkey involved in 28 conditions
- **In fact, only 24.26 bits are guessed by dynamic key-guessing techniques**

How to Remove the Redundancy of Subkeys

$\Delta X_{21}^2 = 0$ means,

$$\begin{aligned} \Delta X_{21}^2 &= \Delta X_{22}^1 \cap X_{29}^1 \oplus \Delta X_{23}^1 \oplus \Delta X_{21}^0, \\ X_{29}^1 &= (X_{30}^0 \cap X_{21}^0) \oplus X_{31}^0 \oplus X_{13}^0 \oplus K_{13}^0 \end{aligned}$$

From $X_{29}^1 = 0$, we deduce that,

- $(\Delta X_{22}^1, \Delta X_{23}^1 \oplus \Delta X_{21}^0) = (0, 1)$: No solution of K_{13}^0 (filtered in advance)
- $(\Delta X_{22}^1, \Delta X_{23}^1 \oplus \Delta X_{21}^0) = (0, 0)$: 2 solutions of K_{13}^0 without any computation
- $\Delta X_{22}^1 = 1$, both for $\Delta X_{23}^1 \oplus \Delta X_{21}^0 = 0$ and 1: one solution.

Only one case has two solutions

How to Remove the Redundancy of Subkeys

$\Delta X_{30}^2 = 0$ means,

$$\begin{aligned}
\Delta X_{30}^2 &= (\Delta X_{31}^1 \cap X_{22}^1) \oplus (X_{31}^1 \cap \Delta X_{22}^1) \\
&= \oplus(\Delta X_{31}^1 \cap \Delta X_{22}^1) \oplus \Delta X_{16}^1 \oplus \Delta X_{30}^0, \\
X_{31}^1 &= (X_{16}^0 \cap X_{23}^0) \oplus X_{17}^0 \oplus X_{15}^0 \oplus K_{15}^0, \\
X_{22}^1 &= (X_{23}^0 \cap X_{30}^0) \oplus X_{24}^0 \oplus X_{6}^0 \oplus K_{6}^0.
\end{aligned}$$

How to Remove the Redundancy of Subkeys

Find solutions of (K_{15}^0, K_6^0) from the values of $(\Delta X_{31}^1$, ΔX_{22}^1 , $\Delta X_{16}^1 \oplus \Delta X_{30}^0)$,

- (0,0,1): no solutions
- (0,0,0): no subkey bit in ΔX_{30}^2, and 4 solutions.
- (0,1,0) or (0,1,1): one solution of K_{15}^0, 2 solutions.
- (1,0,0) or (1,0,1): one solution of K_6^0, two solutions.
- (1,1,0) or (1,1,1): one solution of $K_{15}^0 \oplus K_6^0$, two solutions.

Only 0.75 bits key- guessing on average rather than 2 bits(subkey bits involved) in traditional attacks

Bit Conditions to Construct Data Structures

1. Divided the plaintexts into 2^{18} structures (10 necessary conditions on plaintexts, 8 necessary conditions on the first three rounds): 2^{14} plaintexts for each.
   - Build the following 8 equations $X_j^1 = (X^0_{(j+1)\%n+n} \cap X^0_{(j+8)\%n+n}) \oplus X^0_{(j+2)\%n+n} \oplus X^0_{j-n}$, where $j = 18, 19, 20, 21, 27, 28, 29, 30$.
   - Because there are 10 conditions on plaintexts, we fixed 10 bit X_i^0 as constants, and obtained each structure by traversing 14 free plaintext bits and solving the above equations system.

Build data structures by solving bit equations

Summary of Differential attacks on SIMON: 2-4 More Rounds

Cipher	Rounds	Attacked Rounds	Time	Data	Reference
SIMON32/64	32	18	2^{46}	$2^{31.2}$	Abed et al. in FSE2014
		19	2^{34}	2^{31}	Biryukov et al. in FSE2014
		21	$2^{55.25}$	2^{31}	New Results
SIMON48/72	36	19	2^{52}	2^{46}	Abed et al. in FSE2014
		20	2^{52}	2^{46}	Biryukov et al. in FSE2014
		23	$2^{63.25}$	2^{47}	New Results
SIMON48/96	36	19	2^{76}	2^{46}	Abed et al. in FSE2014
		20	2^{75}	2^{46}	Biryukov et al. in FSE2014
		24	$2^{87.25}$	2^{47}	New Results
SIMON64/96	42	26	2^{94}	2^{63}	Abed et al. in FSE2014
		26	2^{89}	2^{63}	Biryukov et al. in FSE2014
		28	$2^{82.75}$	2^{63}	New Results
SIMON64/128	44	26	2^{126}	2^{63}	Abed et al. in FSE2014
		26	2^{121}	2^{63}	Biryukov et al. in FSE2014
		29	$2^{114.75}$	2^{63}	New Results
SIMON96/96	52	35	$2^{93.3}$	$2^{93.2}$	Abed et al. in FSE2014
		37	2^{95}	2^{95}	New Results
SIMON96/144	54	35	$2^{101.1}$	$2^{93.2}$	Abed et al. in FSE2014
		37	$2^{130.75}$	2^{95}	New Results
SIMON128/128	68	46	$2^{125.7}$	$2^{125.6}$	Abed et al. in FSE2014
		49	2^{127}	2^{127}	New Results
SIMON128/192	69	46	$2^{142.0}$	$2^{125.6}$	Abed et al. in FSE2014
		49	$2^{183.25}$	2^{127}	New Results
SIMON128/256	72	46	$2^{206.0}$	$2^{125.6}$	Abed et al. in FSE2014

Dynamic Key-guessing in linear attacks

Following is the joint work of Huaifeng Chen and Xiaoyun Wang, FSE 2016

Figure: When k_G is known, the set of x can be splitted to two sets. f is independent of k_B in set S_A and independent of k_A in set S_B.

Example:$y = f(x, k) = (x_0 \oplus k_0)\&(x_1 \oplus k_1)$, compute the bias of y where $V[x]$ denotes the numer of x. GUESS k_0 and SPLIT the x into two sets

- For the x with $x_0 = k_0$, initialize a counter T_0 and set $T_0 = V[0||x_0] + V[1||x_0]$ (1 addition)
- For the x with $x_0 = k_0 \oplus 1$, initialize a counter T_1 and set $T_1 = V[0||x_0] - V[1||x_0]$ (1 addition)
- COMBINATION $B(y) = T_0 + (-1)^{k_1} T_1$ (2 additions)
- TIME: $(2^2 \times 2^2 = 2^4) \rightarrow (2 \times (1 + 1 + 2) = 2^3)$

Linear Hulls of Simon

- For Simon32/64, the linear hull used in the attack is

$$X^i_{L,5} \rightarrow X^{i+13}_{R,13}$$

with potential $\bar{\epsilon}^2 = 2^{-30.19}$ in [2].

BS	Input Active Bits	Output Active Bits	ALH
32	$X^i_{L,6}$	$X^{i+13}_{R,14}$	$2^{-31.69}$
	$X^i_{L,5}$	$X^{i+13}_{R,13}$	$2^{-30.19}$
	$X^i_{L,0}$	$X^{i+14}_{L,8}, X^{i+14}_{R,6}$	$2^{-32.56}$
48	$X^i_{L,7}, X^i_{L,11}, X^i_{L,19}, X^i_{R,9}, X^i_{R,17}$	$X^{i+15}_{L,5}, X^{i+15}_{R,3}, X^{i+15}_{R,7}, X^{i+15}_{R,11}, X^{i+15}_{R,19}$	$2^{-44.11}$
	$X^i_{L,6}, X^i_{L,14}, X^i_{L,18}, X^i_{L,22}, X^i_{R,16}$	$X^{i+15}_{L,4}, X^{i+15}_{L,20}, X^{i+15}_{R,6}, X^{i+15}_{R,18}, X^{i+15}_{R,20}, X^{i+15}_{R,22}$	$2^{-42.28}$
	$X^i_{L,1}, X^i_{L,5}, X^i_{L,21}, X^i_{R,23}$	$X^{i+16}_{L,1}, X^{i+16}_{L,5}, X^{i+16}_{R,23}$	$2^{-44.92}$
64	$X^i_{L,20}, X^i_{L,24}, X^i_{R,22}$	$X^{i+21}_{L,22}, X^{i+21}_{R,20}, X^{i+21}_{R,24}$	$2^{-62.53}$
	$X^i_{L,6}$	$X^{i+21}_{L,0}, X^{i+21}_{R,2}, X^{i+21}_{R,6}, X^{i+21}_{R,30}$	$2^{-60.72}$
	$X^i_{L,3}, X^i_{L,27}, X^i_{L,31}, X^i_{R,29}$	$X^{i+22}_{L,3}, X^{i+22}_{R,1}, X^{i+22}_{R,2}$	$2^{-63.83}$
96	$X^i_{L,2}, X^i_{L,34}, X^i_{L,38}, X^i_{L,42}, X^i_{R,36}$	$X^{i+30}_{L,2}, X^{i+30}_{L,42}, X^{i+30}_{L,46}, X^{i+30}_{R,0}, X^{i+30}_{R,40}$	$2^{-94.2}$
128	$X^i_{L,2}, X^i_{L,58}, X^i_{L,62}, X^i_{R,60}$	$X^{i+41}_{L,60}, X^{i+41}_{R,0}, X^{i+41}_{R,2}, X^{i+41}_{R,58}, X^{i+41}_{R,62}$	$2^{-126.6}$

* BS means the block size of SIMON; #R means the number of rounds for the linear hull

[1] Mohamed Ahmed Abdelraheem *et al.* IACR Cryptology ePrint Archive 2014/681, 2014

[2] Danping Shi *et al.*IACR Cryptology ePrint Archive 2014/973, 2014

[3] Siwei Sun *et al.* IACR Cryptology ePrint Archive 2014/747, 2014

Boolean representation of $X^i_{L,5}$ and $X^{i+13}_{R,13}$

- One round backward for $X^i_{L,5}$

$$X^i_{L,5} = \underbrace{(X^{i-1}_{L,4} \& X^{i-1}_{L,13}) \oplus X^{i-1}_{L,3} \oplus X^{i-1}_{R,5}}_{x_0} \oplus \underbrace{K^{i-1}_5}_{k_0} = x_0 \oplus k_0.$$

- Four rounds backward for $X^i_{L,5}$: $X^i_{L,5} = f(x,k)$ where

$$\begin{aligned} f(x,k) =& x_0 \oplus k_0 \oplus ((x_1 \oplus k_1)\&(x_2 \oplus k_2)) \oplus ((x_3 \oplus k_3)\&(x_4 \oplus k_4)) \oplus \\ & [(x_5 \oplus k_5 \oplus ((x_6 \oplus k_6)\&(x_7 \oplus k_7)))\&(x_8 \oplus k_8 \oplus ((x_9 \oplus k_9)\&(x_7 \oplus k_7)))] \oplus \\ & \{(x_{10} \oplus k_{10} \oplus ((x_6 \oplus k_6)\&(x_7 \oplus k_7)) \oplus \\ & [(x_{11} \oplus k_{11} \oplus ((x_{12} \oplus k_{12})\&(x_{13} \oplus k_{13})))\&(x_{14} \oplus k_{14} \oplus ((x_3 \oplus k_3)\&(x_{13} \oplus k_{13})))])\& \\ & (x_{15} \oplus k_{15} \oplus ((x_7 \oplus k_7)\&(x_9 \oplus k_9)) \oplus \\ & [(x_{14} \oplus k_{14} \oplus ((x_{13} \oplus k_{13})\&(x_3 \oplus k_3)))\&(x_{16} \oplus k_{16} \oplus ((x_3 \oplus k_3)\&(x_4 \oplus k_4)))])\}, \end{aligned}$$

  $x_0, \ldots, x_{16}$ can be computed from X^{i-4}, $k_0, \ldots, k_{16}$ are the xor-sums of some bits of $K^{i-4}, \ldots, K^{i-1}$ (See the following table)

- Four rounds forward for $X^{i+13}_{R,13}$: similar to that above

Table: 4 rounds before $X^i_{L,5}$ for SIMON32

x_0	$X^{i-4}_{L,13} \oplus (X^{i-4}_{L,14} \& X^{i-4}_{L,7}) \oplus X^{i-4}_{R,15} \oplus X^{i-4}_{L,1} \oplus X^{i-4}_{L,5}$	k_0	$K^{i-4}_{15} \oplus K^{i-3}_{1} \oplus K^{i-3}_{5} \oplus K^{i-2}_{3} \oplus K^{i-1}_{5}$
x_1	$X^{i-4}_{L,14} \oplus (X^{i-4}_{L,15} \& X^{i-4}_{L,8}) \oplus X^{i-4}_{R,0}$	k_1	K^{i-4}_{0}
x_2	$X^{i-4}_{L,7} \oplus (X^{i-4}_{L,8} \& X^{i-4}_{L,1}) \oplus X^{i-4}_{R,9}$	k_2	K^{i-4}_{9}
x_3	$X^{i-4}_{L,2} \oplus (X^{i-4}_{L,3} \& X^{i-4}_{L,12}) \oplus X^{i-4}_{R,4}$	k_3	K^{i-4}_{4}
x_4	$X^{i-4}_{L,11} \oplus (X^{i-4}_{L,12} \& X^{i-4}_{L,5}) \oplus X^{i-4}_{R,13}$	k_4	K^{i-4}_{13}
x_5	$X^{i-4}_{L,14} \oplus (X^{i-4}_{L,15} \& X^{i-4}_{L,8}) \oplus X^{i-4}_{R,0} \oplus X^{i-4}_{L,2}$	k_5	$K^{i-4}_{0} \oplus K^{i-3}_{2}$
x_6	$X^{i-4}_{L,15} \oplus (X^{i-4}_{L,0} \& X^{i-4}_{L,9}) \oplus X^{i-4}_{R,1}$	k_6	K^{i-4}_{1}
x_7	$X^{i-4}_{L,8} \oplus (X^{i-4}_{L,9} \& X^{i-4}_{L,2}) \oplus X^{i-4}_{R,10}$	k_7	K^{i-4}_{10}
x_8	$X^{i-4}_{L,7} \oplus (X^{i-4}_{L,8} \& X^{i-4}_{L,1}) \oplus X^{i-4}_{R,9} \oplus X^{i-4}_{L,11}$	k_8	$K^{i-4}_{9} \oplus K^{i-3}_{11}$
x_9	$X^{i-4}_{L,1} \oplus (X^{i-4}_{L,2} \& X^{i-4}_{L,11}) \oplus X^{i-4}_{R,3}$	k_9	K^{i-4}_{3}
x_{10}	$X^{i-4}_{L,14} \oplus (X^{i-4}_{L,15} \& X^{i-4}_{L,8}) \oplus X^{i-4}_{R,0} \oplus (X^{i-4}_{L,3} \& X^{i-4}_{L,12}) \oplus X^{i-4}_{R,4}$	k_{10}	$K^{i-4}_{0} \oplus K^{i-3}_{2} \oplus K^{i-4}_{4} \oplus K^{i-2}_{4}$
x_{11}	$X^{i-4}_{L,15} \oplus (X^{i-4}_{L,0} \& X^{i-4}_{L,9}) \oplus X^{i-4}_{R,1} \oplus X^{i-4}_{L,3}$	k_{11}	$K^{i-4}_{1} \oplus K^{i-3}_{3}$
x_{12}	$X^{i-4}_{L,0} \oplus (X^{i-4}_{L,1} \& X^{i-4}_{L,10}) \oplus X^{i-4}_{R,2}$	k_{12}	K^{i-4}_{2}
x_{13}	$X^{i-4}_{L,9} \oplus (X^{i-4}_{L,10} \& X^{i-4}_{L,3}) \oplus X^{i-4}_{R,11}$	k_{13}	K^{i-4}_{11}
x_{14}	$X^{i-4}_{L,8} \oplus (X^{i-4}_{L,9} \& X^{i-4}_{L,2}) \oplus X^{i-4}_{R,10} \oplus X^{i-4}_{L,12}$	k_{14}	$K^{i-4}_{10} \oplus K^{i-3}_{12}$
x_{15}	$X^{i-4}_{L,7} \oplus (X^{i-4}_{L,8} \& X^{i-4}_{L,1}) \oplus X^{i-4}_{R,9} \oplus (X^{i-4}_{L,12} \& X^{i-4}_{L,5}) \oplus X^{i-4}_{R,13}$	k_{15}	$K^{i-4}_{9} \oplus K^{i-3}_{11} \oplus K^{i-4}_{13} \oplus K^{i-2}_{13}$
x_{16}	$X^{i-4}_{L,1} \oplus (X^{i-4}_{L,2} \& X^{i-4}_{L,11}) \oplus X^{i-4}_{R,3} \oplus X^{i-4}_{L,5}$	k_{16}	$K^{i-4}_{3} \oplus K^{i-3}_{5}$

Notice: $x_{10} = x_3 \oplus x_5, x_{15} = x_4 \oplus x_8$

Compute the bias of f

- 15 independent bits for x since $x_{10} = x_3 \oplus x_5, x_{15} = x_4 \oplus x_8$
- Compress x_0 since $f(x, k)$ is linear with $x_0 \oplus k_0$, 14 independent bits remain
- Guess k_1, k_3, k_7. CASE $(x_1 \oplus k_1, x_3 \oplus k_3, x_7 \oplus k_7)$
  - $(0, 0, 0)$
    - $f = f_{00}$

$$f_{00} = ((x_5 \oplus k_5)\&(x_8 \oplus k_8)) \oplus \{(x_{10} \oplus k_{10} \oplus [(x_{11} \oplus k_{11} \oplus ((x_{12} \oplus k_{12})\&(x_{13} \oplus k_{13})))\&(x_{14} \oplus k_{14})])\&(x_{15} \oplus k_{15} \oplus [(x_{14} \oplus k_{14})\&(x_{16} \oplus k_{16})])\}$$

    - 8 independent bits of x remain $(x_{10} = x_3 \oplus x_5)$
    - 7 additions to compress the other bits of x
    - Getting new counters needs $2^8 \times 7$ additions
  - $(0, 0, 1), \ldots, (1, 1, 1)$ similar to $(0, 0, 0)$

Compute the bias of f_{00}

- Guess k_5, k_{14} and split the x into 4 sets

Table: Simplification for f_{00} after guessing k_5, k_{14}

Guess	Value	f_{00}	Related K
k_5, k_{14}	0,0	$(x_{10} \oplus k_{10})\&(x_{15} \oplus k_{15})$	
	0,1	$(x_{10,11} \oplus k_{10,11} \oplus ((x_{12} \oplus k_{12})\&(x_{13} \oplus k_{13})))\&(x_{15,16} \oplus k_{15,16})$	
	1,0	$(x_{10} \oplus k_{10})\&(x_{15} \oplus k_{15})$	k_8
	1,1	$(x_{10,11} \oplus k_{10,11} \oplus ((x_{12} \oplus k_{12})\&(x_{13} \oplus k_{13})))\&(x_{15,16} \oplus k_{15,16})$	k_8

- Case $(0,0), (1,0)$
  - Getting new counters: $2^6 - 2$ additions
  - Similar to the example, however there is dependence between x_5 and x_{10}, computing the bias needs no more than 2^2 additions
- Case $(0,1), (1,1)$
  - Getting new counters: $2^6 - 2^4$ additions
  - Computing the bias needs about $2^{5.64}$ additions
- Total time $2^2 \times ((2^6 - 2 + 2^2 + 2^6 - 2^4 + 2^{5.64}) \times 2 + 2^8) \approx 2^{11.19}$

Attacks on Simon32/64

- Time of computing bias of f is about

$$((2^8 \times 7 + 2^{11.19}) \times 8 + 2^{13} \times 7) \times 2^3 = 2^{19.46}$$

- Data $N = 2\bar{\epsilon}^{-2} = 2^{31.19}$, advantage $a = 8$,
  success probability 0.477 [SB02]
- 21-round (4+13+4) attack: $2^{35.84} Additions + 2^{56} Encryptions$
- 22-round (4+13+4) attack: $2^{48.84} Additions + 2^{56} Encryptions$
- 23-round (4+13+4) attack: $2^{61.84} Additions + 2^{56.3} Encryptions$

Table: Summary of Linear Hull Attacks on SIMON

Cipher	Attacked rounds	Data	Time	Reference
SIMON32/64	21	$2^{30.56}$	$2^{55.56}$	Abdelraheem et al
	21	-	-	[?] Shi et al
	23	$\mathbf{2^{31.19}}$	$\mathbf{2^{61.84}A + 2^{56.3}E}$	**New result**
SIMON48/72	20	$2^{44.11}$	$2^{70.61}$	Abdelraheem et al
	24	$\mathbf{2^{47.92}}$	$\mathbf{2^{67.89}A + 2^{65.34}E}$	**New result**
SIMON48/96	21	$2^{44.11}$	$2^{70.61}$	Abdelraheem et al
	21	-	-	Shi et al
	23	$2^{47.92}$	$2^{92.92}$	Sun et al
	25	$\mathbf{2^{47.92}}$	$\mathbf{2^{89.89}A + 2^{88.28}E}$	**New result**
SIMON64/96	27	$2^{62.53}$	$2^{88.53}$	Abdelraheem et al
	30	$\mathbf{2^{63.53}}$	$\mathbf{2^{93.62}A + 2^{88.13}E}$	**New result**
SIMON64/128	29	$2^{62.53}$	$2^{123.53}$	Abdelraheem et al
	29	-	-	Shi et al
	31	$\mathbf{2^{63.53}}$	$\mathbf{2^{119.62}A + 2^{120.00}E}$	**New result**
SIMON96/96	**37**	$\mathbf{2^{95.2}}$	$\mathbf{2^{67.94}A + 2^{88}E}$	**New result**
SIMON96/144	36	$2^{94.2}$	$2^{123.5}$	Abdelraheem et al
	38	$\mathbf{2^{95.2}}$	$\mathbf{2^{98.94}A + 2^{136.00}E}$	**New result**
SIMON128/128	**49**	$\mathbf{2^{127.6}}$	$\mathbf{2^{87.77}A + 2^{120}E}$	**New result**
SIMON128/192	48	$2^{126.6}$	$2^{187.6}$	Abdelraheem et al
	51	$\mathbf{2^{127.6}}$	$\mathbf{2^{155.77}A + 2^{184.00}E}$	**Our result**
SIMON128/256	50	$2^{126.6}$	$2^{242.6}$	Abdelraheem et al
	53	$\mathbf{2^{127.6}}$	$\mathbf{2^{239.77}A + 2^{248.01}E}$	**New result**

* '-' means not given; A means addition; E means encryption;

The following is the joint work of Senyang Huang, Meiqin Wang, Jingyuan Zhao, Xiaoyun Wang.

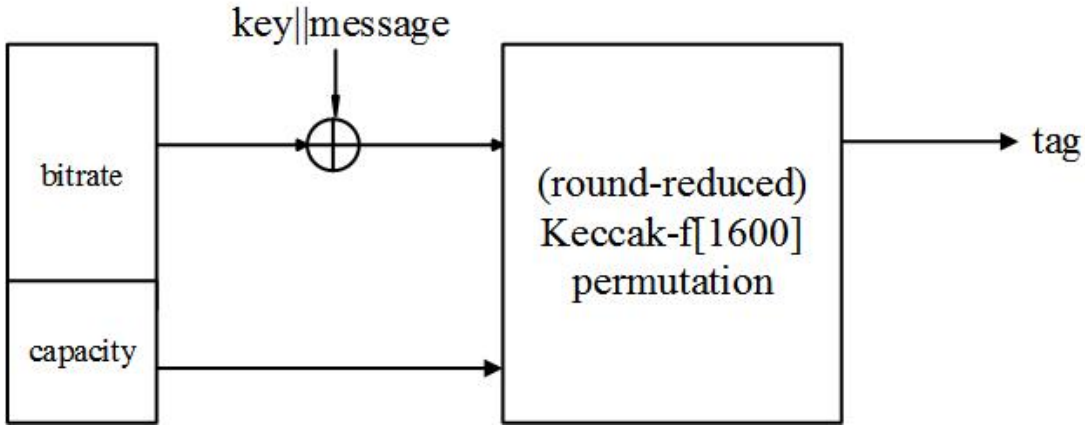


Figure: construction of Keccak-MAC

- Degree of operation χ in Keccak Sponge function is two. n-round Keccak-MAC is of $2^n + 1$ degree at most.
- Keccak sponge function produces very fast diffusion that one bit may influences 33 bits in the next round.

Therefore, adding bit conditions to reduce degree of output polynomial is inflexible.

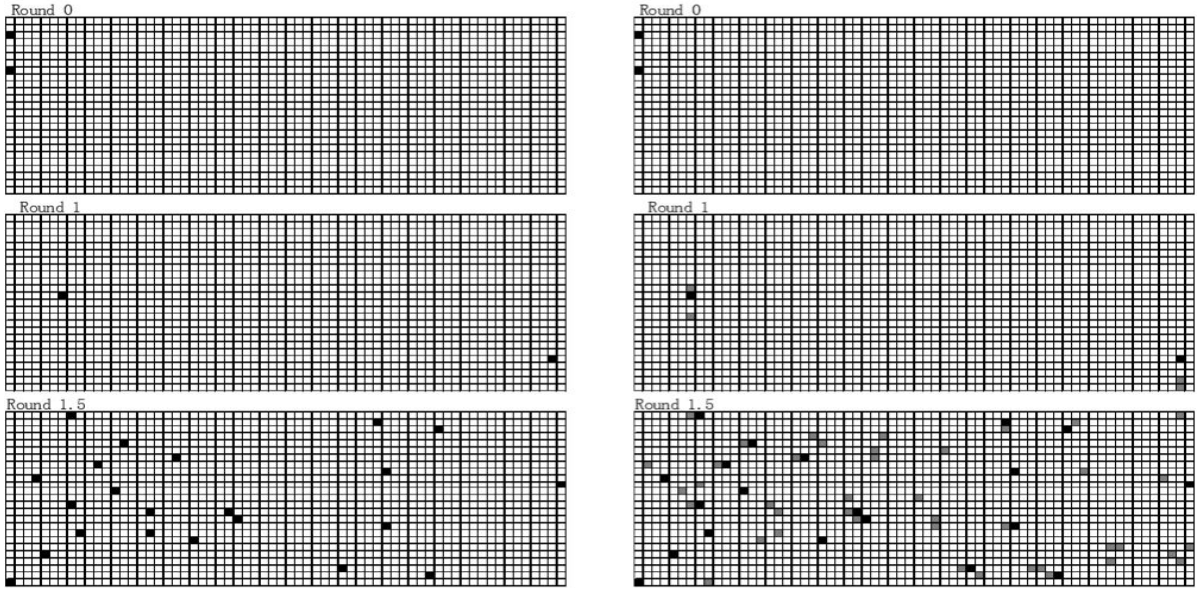


Figure: Propagation of a variable with and without bit conditions

Our strategy:

- Add bit conditions to slow down the propagation.
- Select variables not multiplied with each other in the second round under the control of bit conditions.
- Prove that the target term does not exist in the output polynomial controlled under some bit conditions.

Selected Variables	keccak[2][0]=keccak[7][0]=v_0, keccak[2][1]=keccak[7][1]=v_1, keccak[2][2]=keccak[7][2]=v_2, keccak[2][3]=keccak[7][3]=v_3, keccak[2][22]=keccak[7][22]=v_{22}, keccak[2][23]=keccak[7][23]=v_{23}, keccak[2][44]=keccak[7][44]=v_{44}, keccak[2][45]=keccak[7][45]=v_{45}, keccak[3][15]=keccak[8][15]=v_{79}
Bit Conditions	keccak[2][4]= $k_5 + k_{69} + v_{197} + v_{324} + 1$, keccak[2][59]= $k_{60} + v_{252} + v_{379} + 1$, keccak[2][7]= $k_8 + k_{72} + v_{200} + 1 + v_{327}$, keccak[2][47]= $v_{174} + v_{367}$, keccak[2][6]= $k_7 + k_{71} + v_{199} + v_{326} + 1$, keccak[2][61]= $k_{62} + v_{254} + v_{381} + 1$, keccak[2][9]= $v_{136} + v_{329} + v_{393}$, keccak[2][46]= $v_{173} + v_{366}$, keccak[2][5]=$k_6 + k_{70} + v_{198} + v_{325} + 1$, keccak[2][60]=$k_{61} + v_{253} + v_{380} + 1$, keccak[2][8]= $v_{135} + v_{328} + v_{392}$, keccak[2][48]= $v_{175} + v_{368}$, keccak[4][6]= $k_8 + k_{72} + v_{200} + v_{391} + 1$, keccak[2][62]= $k_{63} + v_{255} + v_{382} + 1$, keccak[2][10] = $v_{137} + v_{330} + v_{394}$, keccak[2][26] = $k_{27} + k_{91} + v_{219} + v_{346} + 1$, keccak[2][17] = $k_{18} + v_{210} + v_{337} + 1$, keccak[2][29] = $v_{156} + v_{349} + v_{413}$, keccak[4][3] = $k_5 + k_{69} + v_{197} + 1$, keccak[2][27] = $k_{28} + k_{92} + v_{220} + v_{347} + 1$, keccak[2][30] = $v_{157} + v_{350} + v_{414}$, keccak[2][25] = $v_{152} + v_{345}$, keccak[5][49] = $k_{49} + k_{113} + v_{175} + 1$, keccak[2][51] = $v_{178} + v_{371} + v_{435}$, keccak[4][25] = $k_{27} + k_{91} + v_{219} + 1$, keccak[2][40] = $k_{41} + v_{233} + v_{360} + 1$, keccak[2][52]= $v_{179} + v_{372} + v_{436}$, keccak[2][49]=$k_{50} + k_{114} + v_{242} + v_{369} + 1$, keccak[3][45] = $k_{110} + v_{302} + v_{429} + 1$, keccak[3][23] = $k_{22} + v_{214} + v_{407}$, keccak[3][31] = $k_{30} + v_{222} + v_{415}$
Fixed Value	keccak[4][44]=0, keccak[4][2] = 1
Guessed Key Bits	$k_{60}, k_{18}, k_{62}, k_{61}, k_{22}, k_{110}, k_6 + k_{70}$, $k_{50} + k_{114}, k_{41}, k_{28} + k_{92}, k_7 + k_{71}, k_5 + k_{69}$, $k_{63}, k_{49} + k_{113}, k_{27} + k_{91}, k_8 + k_{72}, k_{30}$

Table: Parameters set for attack 5-round Keccak-MAC-512

With all the conditions hold in the former frame,

- The nine selected variables never multiply with each other by the end of the second round.
- The output polynomial on the nine variables is of eight degree after three extra round.
- Target term $v_0v_1v_2v_3v_{22}v_{23}v_{44}v_{45}v_{79}$ will never appear in the output polynomial.
- Only 17 key bits are related in the conditions.

By the same way, we could find 17 variables never multiplied with each other by the end of the second round for Keccak-MAC-384. Multiplied term of the 17 variables will not appear after 6-round Keccak-MAC-384 in the output polynomial by the same reason. In this situation, 32 key bits are contained in the bit conditions.

Rounds	Capacity	Time	Data	Memory	Referance
5	576	2^{35}	2^{35}	negligible	EUROCRYPT'15
6	256	2^{66}	2^{64}	2^{32}	EUROCRYPT'15
7	256	2^{97}	2^{64}	2^{32}	EUROCRYPT'15
5	576/1024	2^{24}	2^{24}	negligible	our result
6	256/768	2^{40}	2^{40}	negligible	our result
7	256/512	2^{72}	2^{72}	negligible	our result

Table: Summary of key recovery attacks on Keccak-MAC

Rounds	Capacity	Time	Data	Memory	Referance
7	256	2^{76}	2^{75}	2^{43}	EUROCRYPT'15
7	256	2^{42}	2^{42}	negligible	our result
8	256	2^{74}	2^{74}	negligible	our result

Table: Summary of key recovery attacks on Keyak

Rounds	Capacity	Time	Data	Memory	Referance
4	448/512	2^{25}	2^{24}	negligible	INDOCRYPT'11
6	448	2^{52}	2^{52}	negligible	AFRICACRYPT'1
5	448/512/1024	2^{9}	2^{9}	negligible	our result
6	448/512/768	2^{17}	2^{17}	negligible	our result
7	448	2^{33}	2^{33}	negligible	our result

Table: Summary of distinguishing attacks on Keccak sponge function

Conclusion

- Bit cryptanalysis including the XOR differential attack with bit conditions or modular differential attacks
- Usually available to dynamic key-guessing techniques, and remove the redundancy of subkeys by bit conditions which are exact methods
- More powerful than searching the differential path even with automatic searching techniques

Thanks for your attention!

Questions?