Conditional Cube Attacks on KECCAK-p Based Constructions

Ling Song, Jian Guo, Danping Shi

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

NANYANG TECHNOLOGICAL UNIVERSITY

ASK 2017 @ Changsha, China

Outlines

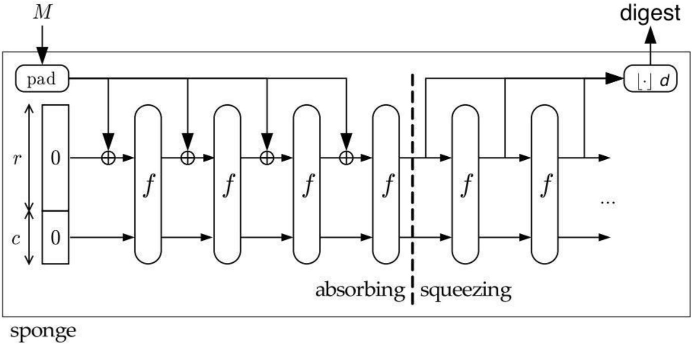
1. KECCAK
2. Conditional Cube Attacks
3. New MILP Model for Searching Conditional Cubes
4. Main Results

Outline

1. KECCAK
2. Conditional Cube Attacks
3. New MILP Model for Searching Conditional Cubes
4. Main Results

SHA-3 (KECCAK) Hash Function

The sponge construction [BDPV11]



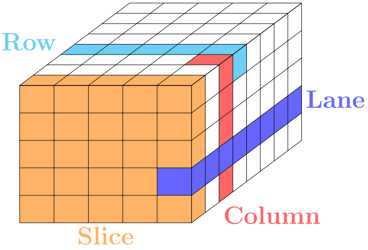
- b-bit permutation f
- Two parameters: bitrate r, capacity c, and $b = r + c$.
- The message is padded and then split into r-bit blocks.

KECCAK Permutation

- 1600 bits: seen as a 5×5 array of 64-bit lanes, $A[x, y], 0 \leq x, y < 5$
- 24 rounds
- each round R consists of five steps:

$$R = \iota \circ \chi \circ \pi \circ \rho \circ \theta$$

- χ : the only nonlinear operation



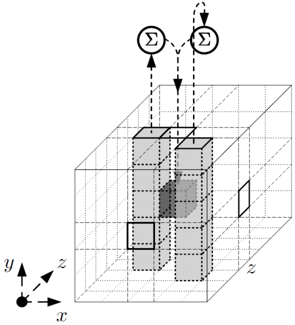
http://www.iacr.org/authors/tikz/

Keccak Permutation

Round function: $\iota \circ \chi \circ \pi \circ \rho \circ \theta$

θ step: adding two columns to the current bit

$$
\begin{aligned}
C[x] =& A[x,0] \oplus A[x,1] \oplus A[x,2] \oplus \\
& A[x,3] \oplus A[x,4] \\
D[x] =& C[x-1] \oplus (C[x+1] \lll 1) \\
A[x,y] =& A[x,y] \oplus D[x]
\end{aligned}
$$



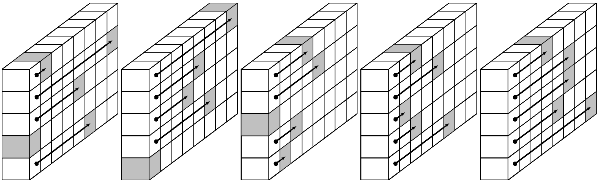
http://keccak.noekeon.org/

- The Column Parity kernel
  - If $C[x] = 0, 0 \leq x < 5$, then the state A is in the CP kernel.

KECCAK Permutation

Round function: $\iota \circ \chi \circ \pi \circ \rho \circ \theta$

ρ step: lane level rotations, $A[x, y] = A[x, y] \lll r[x, y]$



http://keccak.noekeon.org/

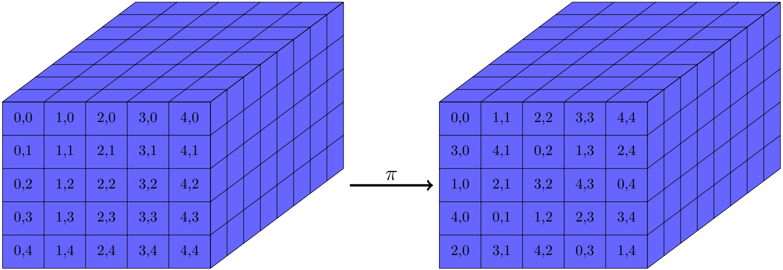
Rotation offsets $r[x, y]$

	$x = 0$	$x = 1$	$x = 2$	$x = 3$	$x = 4$
$y = 0$	0	1	62	28	27
$y = 1$	36	44	6	55	20
$y = 2$	3	10	43	25	39
$y = 3$	41	45	15	21	8
$y = 4$	18	2	61	56	14

Keccak Permutation

Round function: $\iota \circ \chi \circ \pi \circ \rho \circ \theta$

π step: permutation on lanes



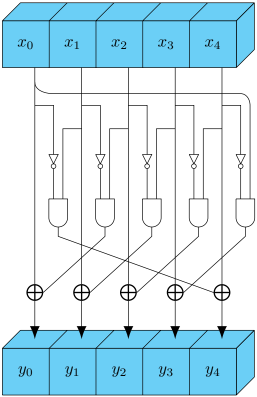
$$A[y, 2*x+3*y] = A[x, y]$$

KECCAK Permutation

Round function: $\iota \circ \chi \circ \pi \circ \rho \circ \theta$

χ step: 5-bit S-boxes, nonlinear operation on rows

$$y_0 = x_0 + (x_1 + 1) \cdot x_2,$$
$$y_1 = x_1 + (x_2 + 1) \cdot x_3,$$
$$y_2 = x_2 + (x_3 + 1) \cdot x_4,$$
$$y_3 = x_3 + (x_4 + 1) \cdot x_0,$$
$$y_4 = x_4 + (x_0 + 1) \cdot x_1.$$

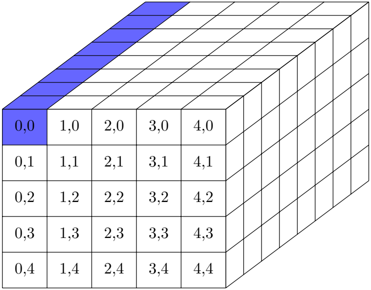


Keccak Permutation

Round function: $\iota \circ \chi \circ \pi \circ \rho \circ \theta$

ι step: adding a round constant to the state

Adding one round-dependent constant to the first "lane", to destroy the symmetry.



$$A[0,0] = A[0,0] \oplus RC[i]$$

KECCAK Permutation

Round function

Internal state A: a 5×5 array of 64-bit lanes

θ step $C[x] = A[x, 0] \oplus A[x, 1] \oplus A[x, 2] \oplus A[x, 3] \oplus A[x, 4]$
$D[x] = C[x - 1] \oplus (C[x + 1] \lll 1)$
$A[x, y] = A[x, y] \oplus D[x]$

ρ step $A[x, y] = A[x, y] \lll r[x, y]$
- The constants $r[x, y]$ are the rotation offsets.

π step $A[y, 2 * x + 3 * y] = A[x, y]$

χ step $A[x, y] = A[x, y] \oplus ((\ A[x + 1, y])\&A[x + 2, y])$

ι step $A[0, 0] = A[0, 0] \oplus RC[i]$
- $RC[i]$ are the round constants.

The only non-linear operation is χ step.

KECCAK-p Based Constructions

KMAC

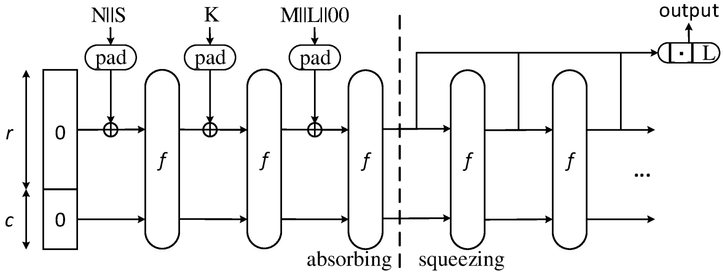
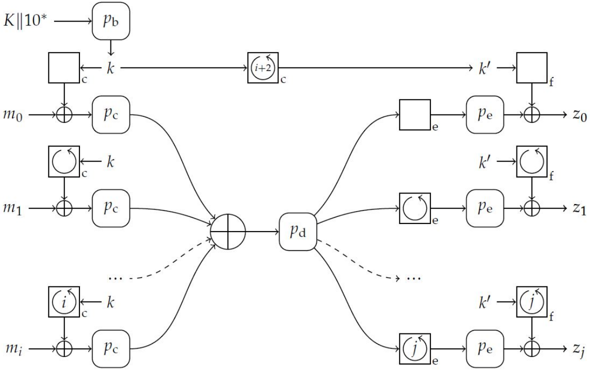


Figure: KMAC processing one message block

- Two versions: KMAC128 and KMAC256
- N and S are public strings.

KECCAK-p Based Constructions

KRAVATTE



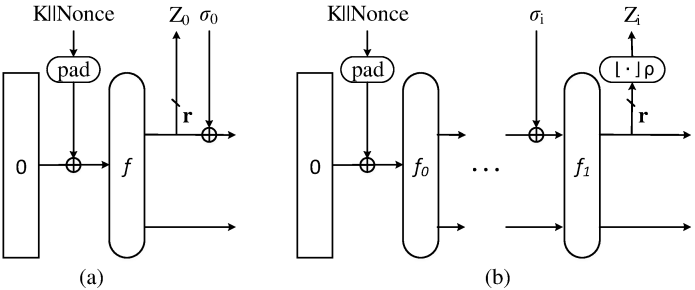
▢ stands for permutations and □ symbolizes rolling functions.

$$p_b = p_c = \text{KECCAK-}p[1600, 6],$$
$$p_d = p_e = \text{KECCAK-}p[1600, 4]^1.$$

[1]Version of 17-Jul-2017.

KECCAK-p Based Constructions

KEYAK and KETJE



(a) KEYAK and (b) KETJE

Outline

Cube Attacks [DS09]

- Given a Boolean polynomial $f(k_0, \ldots, k_{n-1}, v_0, \ldots, v_{m-1})$ and a monomial $t_I = \wedge_{i_r \in I} v_{i_r}$, $I = (i_1, \ldots, i_d)$, f can be written as

$$f(k_0, \ldots, k_{n-1}, v_0, \ldots, v_{m-1}) = t_I \cdot p_{S_I} + q(k_0, \ldots, k_{n-1}, v_0, \ldots, v_{m-1})$$

  - q contains terms that are not divisible by t_I
  - p_{S_I} is called the superpoly of I in f
  - $v_{i_1}, \ldots, v_{i_d}$ are called cube variables. d is the dimension.
- The the cube sum is exactly

$$p_{S_I} = \sum_{(v_{i_1}, \ldots, v_{i_d}) \in C_I} f(k_0, \ldots, k_{n-1}, v_0, \ldots, v_{m-1})$$

- Cube attacks: p_{S_I} is a low-degree polynomial in key bits.
- Cube testers: distinguish p_{S_I} from a random function. E.g., $p_{S_I} = 0$.

Conditional Cube Testers of KECCAK [HWX+17]

- Ordinary cube variables:
  - Do not multiply with any variable in the **first** round.
- Conditional cube variables:
  - Do not multiply with any variable in the **first two** rounds under certain conditions.

Conditional Cube Testers of KECCAK [HWX+17]

- Ordinary cube variables:
  - Do not multiply with any variable in the **first** round.
- Conditional cube variables:
  - Do not multiply with any variable in the **first two** rounds under certain conditions.

Properties

- 2^n-dimensional cubes with 1 conditional cube variable
  - The cube sum is **zero** for $(n+1)$-round KECCAK.

Conditional Cube Testers of KECCAK [HWX+17]

- Ordinary cube variables:
  - Do not multiply with any variable in the **first** round.
- Conditional cube variables:
  - Do not multiply with any variable in the **first two** rounds under certain conditions.

Properties

- 2^n-dimensional cubes with 1 conditional cube variable
  - The cube sum is **zero** for $(n+1)$-round KECCAK.
- If the conditions involve the key, the conditional cube can be used to recover the key.
- Time complexity of the key recovery: $\mathbf{\frac{k}{t} \cdot 2^{2^n+t}}$, where t is the number of key bits involved in the conditions.

Outline

How to keep the first χ linear

The expression of $b = \chi(a)$ is of algebraic degree 2:

$b_i = a_i + \overline{a_{i+1}} \cdot a_{i+2}$, for $i = 0, 1, \ldots, 4$.

How to keep the first χ linear

The expression of $b = \chi(a)$ is of algebraic degree 2:

$b_i = a_i + \overline{a_{i+1}} \cdot a_{i+2}$, for $i = 0, 1, \ldots, 4$.

Observation

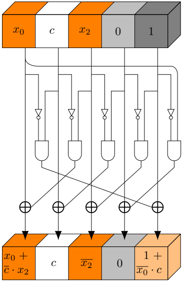
When there is no neighbouring variables in the input of an Sbox, then the application of χ does NOT increase algebraic degree.

How to keep the first χ linear

The expression of $b = \chi(a)$ is of algebraic degree 2:
$b_i = a_i + \overline{a_{i+1}} \cdot a_{i+2}$, for $i = 0, 1, \ldots, 4$.

Observation

When there is no neighbouring variables in the input of an Sbox, then the application of χ does NOT increase algebraic degree.



√

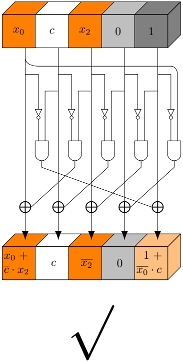
How to keep the first χ linear

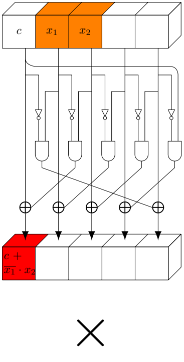
The expression of $b = \chi(a)$ is of algebraic degree 2:

$b_i = a_i + \overline{a_{i+1}} \cdot a_{i+2}$, for $i = 0, 1, \ldots, 4$.

Observation

When there is no neighbouring variables in the input of an Sbox, then the application of χ does NOT increase algebraic degree.





Linear Structure [GLS16]

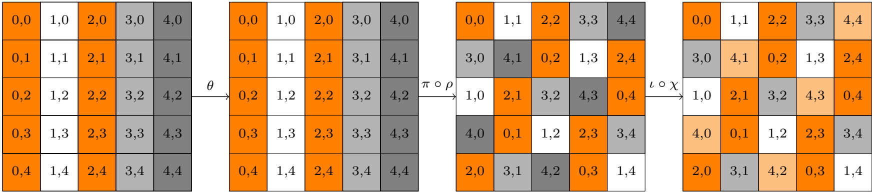


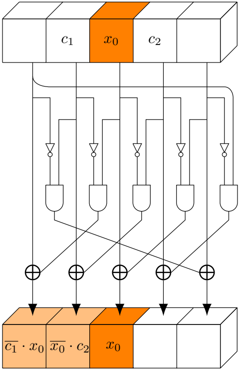
Figure: 1-round linear structure of KECCAK-p with the degrees of freedom up to 512, where ■: variables; ■: algebraic degree at most 1; ■: 1; ■: 0.

- All variables do not multiply with each other in the first round.
- **BUT** we need at least one conditional variable.

The Conditional Cube variable

Requirement of the second χ

- If an input bit of the **second** χ contains the conditional variable, then its neighbouring bits should be constants.
- These neighbouring bits are denoted as $s_0, s_1, \ldots$
- Each s_i is calculated from 11 output bits of the first round.



New MILP Model

Mixed integer linear programming (MILP) takes an *objective function* *obj* and a set of inequalities $M \cdot X < b$ over real numbers as input and finds solutions optimizing *obj*.

Let $a[x][y][z]$ be the state:

$$a \xrightarrow{\pi \circ \rho \circ \theta} b \xrightarrow{\chi} c$$

$A[x][y][z] = 1$ if $a[x][y][z]$ contains a cube variable:

$$A \xrightarrow{\pi \circ \rho \circ \theta} B \xrightarrow{\chi} C$$

$V[x][y][z] = 1$ indicates a bit condition.

Modeling the First χ

Patterns of the Diffusion of χ

$$c[x] = b[x] + \overline{b[x+1]} \cdot b[x+2]^{1}$$

$b[x]$	$b[x+1]$	$b[x+2]$	$c[x]$

[1] Omit coordinates $[y][z]$.

Modeling the First χ

Patterns of the Diffusion of χ

$$c[x] = b[x] + \overline{b[x+1]} \cdot b[x+2]^1$$

$b[x]$	$b[x+1]$	$b[x+2]$	$c[x]$
constant	constant	constant	constant

[1]Omit coordinates $[y][z]$.

Modeling the First χ

Patterns of the Diffusion of χ

$$c[x] = b[x] + \overline{b[x+1]} \cdot b[x+2]^1$$

$b[x]$	$b[x+1]$	$b[x+2]$	$c[x]$
constant	constant	constant	constant
var	*	*	var

[1]Omit coordinates $[y][z]$.

Modeling the First χ

Patterns of the Diffusion of χ

$$c[x] = b[x] + \overline{b[x+1]} \cdot b[x+2]^1$$

$b[x]$	$b[x+1]$	$b[x+2]$	$c[x]$
constant	constant	constant	constant
var	*	*	var
constant	constant	var	var (deg ≤ 1)

[1] Omit coordinates $[y][z]$.

Modeling the First χ

Patterns of the Diffusion of χ

$$c[x] = b[x] + \overline{b[x+1]} \cdot b[x+2]^1$$

$b[x]$	$b[x+1]$	$b[x+2]$	$c[x]$
constant	constant	constant	constant
var	*	*	var
constant	constant	var	var (deg ≤ 1)
constant	1	var	constant

[1]Omit coordinates $[y][z]$.

Modeling the First χ

Patterns of the Diffusion of χ

$$c[x] = b[x] + \overline{b[x+1]} \cdot b[x+2]$$ [1]

$b[x]$	$b[x+1]$	$b[x+2]$	$c[x]$
constant	constant	constant	constant
var	*	*	var
constant	constant	var	var $(\deg \leq 1)$
constant	1	var	constant
⋮	⋮	⋮	⋮

[1] Omit coordinates $[y][z]$.

Modeling the First χ

Patterns of the Diffusion of χ

$$B[x] = \begin{cases} 0, & b[x] \text{ is a constant;} \\ 1, & b[x] \text{ is a var.} \end{cases} \qquad V[x] = \begin{cases} 0, & \text{no condidtion on } b[x]; \\ 1, & b[x] \text{ is restricted to } 0/1. \end{cases}$$

Table: Diffusion of variables through χ. Symbol '*' denotes arbitrary value.

$B[x]$	$B[x+1]$	$B[x+2]$	$V[x+1]$	$V[x+2]$	$C[x]$
0	0	0	*	*	0
1	0	0	*	*	1
1	0	1	*	0	1
0	0	1	0	0	1
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	0	1	0

Modeling the First χ

Inequalities Describing the Diffusion of χ

- By generating the convex hull of the set of patterns, we get

$$
\begin{aligned}
B[x] - B[x+1] - B[x+2] - V[x+1] - V[x+2] - C[x] &\geq -2 \\
-B[x] - B[x+1] + V[x+2] + C[x] &\geq 0 \\
-B[x+2] - V[x+2] &\geq -1 \\
B[x] + B[x+1] + B[x+2] - C[x] &\geq 0 \\
-B[x] + C[x] &\geq 0 \\
-B[x+1] - B[x+2] + V[x+1] + V[x+2] + C[x] &\geq 0 \\
-B[x] - B[x+1] &\geq -1
\end{aligned}
$$

Modeling the Second χ

Two Cases for the Second χ

- Each neighbouring bit s_i of the conditional variables is calculated from 11 bits of $c[x][y][z]$.

  Case 1 For these 11 bits, none of them are variables, i.e., $C[x][y][z] = 0$;

  Case 2 There are variables among the 11 bits and the XOR of these bits forms a linear equation which consumes 1 bit degree of freedom.

- Introduce S_i for s_i

$$S_i = \begin{cases} 0, & \text{for Case 1;} \\ 1, & \text{for Case 2.} \end{cases}$$

Modeling the Second χ

Patterns and Inequalities for the Second χ

If $c[x][y][z]$ is needed for calculating s_i, then $c[x][y][z]$ should not contain terms with uncertain coefficients.

- Patterns that exclude terms with uncertain coefficients:

S_i	$B[x]$	$B[x+1]$	$B[x+2]$	$V[x+1]$	$V[x+2]$
0	*	*	*	*	*
1	0	0	0	*	*
1	1	0	0	*	*
1	1	0	1	**1**	0
1	0	0	1	1	0
1	0	1	0	0	1

Modeling the Second χ

Patterns and Inequalities for the Second χ

- Inequalities:

$$\begin{aligned}
-S_i - B[x+1] - B[x+2] &\geq -2 \\
-S_i + B[x] - B[x+1] + V[x+2] &\geq -1 \\
-S_i - B[x+2] + V[x+1] &\geq -1 \\
-S_i - B[x+1] - V[x+1] &\geq -2 \\
-S_i - B[x+2] - V[x+2] &\geq -2 \\
-S_i - B[x] - B[x+1] &\geq -2
\end{aligned}$$

Modeling the Search for Conditional Cubes

- Modeling the linear layer is simple.
- Set the dimension of the target cube to 2^n.
- Objective

$$\text{Minimize}: \quad \sum V[x][y][z].$$

Outline

Application of the New Model

The new model is applicable to keyed KECCAK modes, including

- Constructions with fully unknown internal state
  - KMAC, KRAVATTE (**first attacks**)
- Constructions with partially known internal state
  - KETJE, KEYAK (**improved attacks**)

KMAC and KRAVATTE

Target	Key Size	Capacity	n_r Rounds	Complexity	Reference
KMAC128	128	256	7	2^{76}	this
KMAC256	256	512	9	2^{147}	
KRAVATTE	128	-	8	2^{65}	this
	256	-	9	2^{129}	
KECCAK-MAC	128	256/512	7	2^{72}	[HWX+17]
		768	7	2^{75}	[LBW+17]
		1024	6	$\mathbf{2^{58.3}}$	
		1024	6	$\mathbf{2^{41}}$	this

KEYAK and KETJE

Target	Key Size	n_r Rounds	Complexity	Nonce respected	Reference
Lake KEYAK	128	**6**	2^{37}	Yes	[DMP+15]
	128	8	2^{74}	No	[HWX+17]
	128	**8**	$2^{71.01}$	Yes	this
	256	9	$2^{137.05}$	Yes	
River KEYAK	128	8	2^{77}	Yes	
KETJE Major	128	7	2^{83}	Yes	[LBW+17]
	128	7	$\mathbf{2^{71.24}}$	Yes	this
KETJE Minor	128	7	2^{81}	Yes	[LBW+17]
	128	7	$\mathbf{2^{73.03}}$	Yes	this
KETJE SR v1	128	7	2^{115}	Yes	[DLWQ17]
	128	7	$\mathbf{2^{92}}$	Yes	this

In conclusion:

1. Model the non-linear layer completely, and nest the two nonlinear layers in two rounds together.
2. First attacks on KMAC and KRAVATTE, and improved attacks on KEYAK and KETJE.

In conclusion:

1. Model the non-linear layer completely, and nest the two nonlinear layers in two rounds together.

2. First attacks on KMAC and KRAVATTE, and improved attacks on KEYAK and KETJE.

Thank you for your attention!