Improved Linear (hull) Cryptanalysis of Round-reduced Versions of KATAN

Danping Shi[1,2,3], Lei Hu[1,2⋆], Siwei Sun[1,2], Ling Song[1,2]

[1]State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[2]Data Assurance and Communication Security Research Center, Chinese Academy of Sciences, Beijing 100093, China
[3]University of Chinese Academy of Sciences, Beijing 100093, China
{dpshi, hu, swsun, lsong}@is.ac.cn

Abstract. KATAN is a family of block ciphers published at CHES 2009. Based on the Mixed-integer linear programming (MILP) technique, we propose the first third-party linear cryptanalysis on KATAN. Besides, we evaluate the security of KATAN against the linear attack under the consideration of the dependence of the S-boxes. We present a 131/120-round linear hull attack on KATAN32/48 which are the best known single-key known plaintext attacks. Also, a 94-round linear hull attack on KATAN64 is proposed.

Keywords. KATAN, Mixed-integer linear programming, linear hull, linear cryptanalysis

1 Introduction

Demands for lightweight ciphers used in resource-constrained devices with low cost are increasing in recent years. Many lightweight block ciphers are published in recent years, such as LBlock [1], PRESENT [2], LED [3], PRIDE [4] and SIMON [5].

KATAN is a family of lightweight block ciphers published at CHES 2009 [6]. After its publication, KATAN receives extensive cryptanalysis, for example, the conditional differential cryptanalysis by Knellwolf et al. [7] on 78/70/68-round KATAN32/48/64, differential cryptanalysis by Albrecht et al.[8] on 115-round KATAN32, meet-in-the-middle attack by Isobe et al.[9] on 110/100/94-round KATAN32/48/64, and match box meet-in-the-middle cryptanalysis by Fuhr et al.[10] on 153/129/119-round KATAN32/48/64. All results are presented in Table 1.

Linear attack is an important cryptanalysis technique on modern block ciphers[11]. It aims at finding a linear expression on bits of plaintext, ciphertext, and subkeys, which is different from a random one. The extended linear hull cryptanalysis with same input and output masks is presented by Nyberg [12] in 1995. No new result of linear cryptanalysis on KATAN has been proposed except the linear security analysis given by the designers [6]. And the security evaluation of KATAN with respect to linear cryptanalysis proposed by the designers is not accurate since they did not consider the dependence of the S-boxes. In this paper, We obtain some results about the linear security cryptanalysis on KATAN under consideration of the dependence of the S-boxes, with the similar method published by [13, 14]. Besides, 131/120/94-round linear hull cryptanalysis on KATAN32/48/64 are presented. And the attacks on KATAN32/48 are the best single-key known-plaintext attack. A comparison with existing single-key attacks is listed in Table 1.

Table 1. The analysis results of KATAN based on single-key

Version	Cryptanalysis method	Model	Rounds	Data	Time	Reference
	Differential	CP	78	2^{22}	2^{22}	[7]
	Differental	CP	115	2^{32}	2^{79}	[8]
KATAN32	Match box MITM	CP	153	2^{5}	$2^{78.5}$	[10]
	MITM	KP	110	138	2^{77}	[9]
	Match box MITM	KP	121	4	$2^{77.5}$	[10]
	Linear hull	KP	131	$2^{28.93}$	$2^{78.93}$	Section 4.2
	Differential	CP	70	2^{31}	2^{78}	[7]
	Match box MITM	CP	129	2^{5}	2^{76}	[10]
KATAN48	MITM	KP	100	128	2^{78}	[9]
	Match box MITM	KP	110	4	$2^{77.5}$	[10]
	Linear hull	KP	120	$2^{47.22}$	$2^{75.22}$	Section 4.2
	Differential	CP	68	2^{32}	2^{78}	[7]
	Match box MITM	CP	119	2^{5}	$2^{78.5}$	[10]
KATAN48	MITM	KP	94	116	$2^{77.68}$	[9]
	Match box MITM	KP	102	4	$2^{77.5}$	[10]
	Linear hull	KP	94	2^{57}	2^{78}	Section 4.2

The paper is organized as follows. Section 2 propose the brief description of KATAN. Section 3 shows the searching method of linear masks. The results about the linear (hull) cryptanalysis are given in Section 4. Section 5 is the conclusion.

2 Brief description of KATAN

KATAN is a family of block ciphers with 32, 48, or 64-bit block length, listed by KATAN32, KATAN48 or KATAN64. All versions share the same 80-bit master key. For each version, the plaintext is load in two registers L_1 and L_2, where the length of L_1 and L_2 for each version are listed in Table 2. For KATAN32, in each round, the registers L_1 and L_2 are shifted to the left with 1 position, and two new computed bits by two nonlinear functions $f_a(\cdot)$ and $f_b(\cdot)$ are loaded in the least significant bits of L_1 and L_2, where the least significant denoted by index 0 is in the right of the register. The ciphertext is obtained after 254 rounds. The f_a and f_b are defined as follows

$$f_a(L_1) = L_1[x_1] \oplus L_1[x_2] \oplus (L_1[x_3] \wedge L_1[x_4]) \oplus (L_1[x_5] \wedge IR) \oplus k_a$$
$$f_b(L_2) = L_2[y_1] \oplus L_2[y_2] \oplus (L_2[y_3] \wedge L_2[y_4]) \oplus (L_2[y_5] \wedge L_2[y_6]) \oplus k_b,$$

where IR is irregular known update rule, k_a and k_b are two subkey bits. The bits x_i and y_i are listed in Table 2.

Table 2. The parameters for KATAN

version	$\lvert L_1\rvert$	$\lvert L_2\rvert$	x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	y_4	y_5	y_6
KATAN32	13	19	12	7	8	5	3	18	7	12	10	8	3
KATAN48	19	29	18	12	15	7	6	28	19	21	13	15	6
KATAN64	25	39	24	15	20	11	9	38	25	33	21	14	9

For KATAN48, the shift update of the register and the nonlinear function f_a, f_b are applied twice with same round subkeys in each round, while the nonlinear functions and update of the register are applied three times for KATAN64. Besides, the length of the registers L_1 and L_2 for KATAN32/48/64 are different, which is listed in Table 2.

We only consider single key cryptanalysis in this paper and thus the key schedule is omitted here. More details on KATAN can be found in paper [6].

3 The Linear Cryptanalysis of KATAN

3.1 Notations

$x^r[i]$: the i-th bit in r-th round of the register L_1
$y^r[i]$: the i-th bit in r-th round of the register L_2
k_a^r: the r-th round subkey used in f_a
k_b^r: the r-th round subkey used in f_b
α_x: the masks of the variable x

3.2 Definition of the linear cryptanalysis

Denote f be a boolean function, the correlation ϵ_f of f is defined by

$$Pr(f(x) = 0) - Pr(f(x) = 1).$$

The linear cryptanalysis is evaluated by the correlation.

The *potential* introduced by Nyberg[12] is used to evaluate the linear hull cryptanalysis. Give the input and output masks α and β for a block cipher $C = f(P, K)$, the *potential* $ALH(\alpha, \beta)$ is defined by

$$ALH(\alpha, \beta) = \sum_{\gamma} (Pr(\alpha \cdot P + \beta \cdot C + \gamma \cdot K = 0) - 1/2)^2.$$

3.3 Dependence of sbox

For simplicity, each operation $\wedge$ is called a 2-1-bit sbox. The sbox is active if the output mask is non-zero. Due to IR is known, $L_1[x_5] \wedge IR$ is a linear operation in each round, not a sbox. Usually the sboxes are independent. But due to the fact that two sboxes may share the same input as only few bits are registered in each round, the sboxes for KATAN are dependent.

In our linear cryptanalysis, we take the dependence of the sboxes into consideration. Because of the dependence of the sboxes, the correlation of the linear approximation will be computed directly instead of applying the pilling-up lemma similar to paper[13, 14]. For example, consider two active sboxes of L_1 in 0-th round and 4-th round, the two approximations with zero input mask and non-zero output mask are $x^0[5] \wedge x^0[8]$ and $x^4[5] \wedge x^4[8]$. The two approximations have the same correlation(absolute) 2^{-1}. The correlation of $x^0[5] \wedge x^0[8] + x^4[5] \wedge x^4[8]$ is 2^{-2} if the two approximation are independent by pilling-up lemma. But due to $x^4[8] = x^0[5]$, the correlation of $x^0[5] \wedge x^0[8] + x^4[5] \wedge x^4[8]$ is 2^{-1}.

In our paper, the dependence of the sbox is taken into consideration. After XOR all approximations of active sboxes, then the correlation is computed directly instead of applying the pilling-up lemma. The computing method in the following is similar to the paper [13, 14] proposed.

Clearly, the XOR-ed approximation of all active sboxes is a quadratic function. For any quadratic boolean function $f(t) = Q(t) + L(t)$, where $Q(t)$ is the sum of quadratic term $t[i] \wedge t[j]$ and $L(t)$ is linear combination of $t[i]$, there exists a non-singular transform $s = A \cdot t$ such that $g(s) = f(A^{-1} \cdot t) = Q(s) + L(s)$ is also a quadratic boolean function, sharing the same correlation of f. Besides, suppose $Q(s) = s[i_1] \wedge s[i_2] + s[i_3] \wedge s[i_4] \cdots + s[i_{m-1}] \wedge s[i_m]$, all subscripts $i_1, i_2, i_3, i_4, \cdots, i_{m-1}, i_m$ are not coincident. This kind quadratic function g is called the standard function of f here. The correlation of the standard quadratic function is easily computed as follows. Denote $L(s) = s[j_1] + s[j_2] + \cdots + s[j_{n-1}] + s[j_n]$. If $\{j_1, j_2, \cdots, j_n\} \subseteq \{i_1, i_2, \cdots, i_m\}$, the correlation $\epsilon_g = m/2$, else $\epsilon_g = 0$.

For example, $f(t) = t[1] \wedge t[2] + t[1] \wedge t[3] + t[2] \wedge t[4] + t[2]$. Do non-singular transform $s[1] = t[1] + t[4], s[2] = t[2] + t[3], s[3] = t[3], s[4] = t[4]$. Then the standard function $g(s) = s[1] \wedge s[2] + s[3] \wedge s[4] + s[2] + s[3]$ is obtained. Besides, the correlation of f is same with g, which is 2^{-2}, due to $\{2, 3\} \subseteq \{1, 2, 3, 4\}$.

In this paper, the correlation of each linear characteristic is computed directly after XOR all approximations of active sboxes, instead of applying pilling-up lemma. Firstly, obtain the standard function of the approximation. Secondly, the correlation is easily obtained from the standard function. The calculating method is suitable for other ciphers with the similar sboxes of KATAN.

3.4 Automatic enumeration of of characteristic with MILP

Similar with paper[13–15], we obtain the linear characteristic by the automatic enumeration with Mixed-integer linear Programming Modelling(MILP). The method denotes each mask bit by a 0-1 variable, then describes the cipher by linear constraints and optimized the a objective function. Specially, following is the MILP modelling.

Constraints for linear operation

Constraints for bitwise XOR and branching structure are same with paper [13, 16] in the following.

1. For XOR operation $z = x \oplus y$, then masks $\alpha_x = \alpha_y = \alpha_z$.
2. For three branching structure $z = x = y$, then masks $\alpha_x \oplus \alpha_y \oplus \alpha_z = 0$

Constraints for sbox

For sbox $z = x \wedge y$, then masks $2\alpha_z \geq \alpha_x + \alpha_y$.

Constraints with dealing dependence of sbox

For each original variable t of the registers, denote a 0-1 variable V_t to indicate wether the variable t is the input of one active sbox, where $V_t = 1$ if is. The original variables are $|L_1| + |L_2|$ initial variables of registers and the two added new variables loaded in the LSB of registers in each round. And each variable t may be the input of at most three sboxes. For each variable t, the added constraints for V_t are introduced with the number n_t of sboxes t effected. Suppose the output masks of the n_t sboxes are $\alpha_i, i \in 1, \cdots, n_t$. Then the constraints are

$$n_t \cdot V_t \geq \alpha_1 + \alpha_2 + \cdots + \alpha_{n_t}$$
$$\alpha_1 + \alpha_2 + \cdots + \alpha_{n_t} \geq V_t$$

For example, the original variable $x^0[5]$ of the register L_1 for KATAN32. The variable effected 2 sboxes, 0-th and 4-th round sbox of L_1.

Objective function

As showed previous, the XOR-ed approximation of all active sboxes is a quadratic function. Usually, the more variables exists in the quadratic terms, the bigger correlation is. Then the objective function is minimize the number of the variables exists in the quadratic function, i.e.

$$\sum_x V_x.$$

3.5 The computation of the correlation

After obtaining the masks searched by the method presented in Section 3.4, the accurate correlation is computed by the method showed in Section 3.3.

4 Results

The linear cryptanalysis given by the designer [6] did not consider the dependence of the sboxes. The 126-round security cryptanalysis eliminated by 42-round linear approximation is not accurate due to the dependence of the sboxes. With taking the dependence of sboxes into consideration, we obtained some new results for the security cryptanalysis, and some results of linear hull cryptanalysis by the MILP modelling shown in Section 3.

4.1 Results for linear characteristic

For KATAN32, the correlation for the best 42-round linear approximation is still 2^{-5}, the same with result presented by designer [6]. But the linear characteristic obtained is under the consideration of the dependence of the sbox. Besides, a best 84-round linear characteristic with correlation 2^{-15} is presented in Appendix. Originally, the 84-round linear characteristic is directly eliminated as no more than 2^{-5*2} by pilling-up lemma, while the dependence of the sboxes does not suit the condition of pilling-up lemma. In this paper, the 84-round linear characteristic presented has considered the dependence, and demonstrates KATAN32 is secure against linear cryptanalysis.

For KATAN48, the correlation for the best 43-round linear approximation is 2^{-8}, while the previous best given by designer has correlation 2^{-9}. For KATAN64, the correlation for the best 37-round linear approximation is 2^{-10}, the same with the result shown by designer. Due to the computing resources, the more accurate security analysis for KATAN48/64 are not obtained. Some characteristics are listed in Appendix.

4.2 Results for linear hull

By setting the input and output masks presented in Table 3, some results about linear hull for some versions are found with some added conditions, and some best linear hull attack are mounted by these linear hulls.

Table 3. The input and output masks for linear hull

version	Input masks of register L_1	Input masks of register L_2
KATAN32	1000010001000	0000000000000010000
KATAN48	0000000000000000000	00000000000000000100000000100
KATAN64	0100000000100000100000000	000000000000000000100000000100010000000
version	output masks of register L_1	output masks of register L_2
KATAN32	0010000000000	1000000000000000000
KATAN48	1000000001000000001	00000000100000000000000000000
KATAN64	1000001001001000100000000	100010000100000100101000100100000100100

For KATAN32, the 84-round linear hull with *potential* $2^{-27.93}$ are obtained with added condition $\sum_x V_x \leq 44$. A 131-round attack with 21-round forward and 26-round backward are mounted with the linear hull. The included bits of the registers are listed in Table 5. The included key bits are listed in Table 4. And 48-bit key need to be guessed, while another 10-bit key are not.

For KATAN48, the 90-round linear hull with *potential* $2^{-46.22}$ are obtained with added condition $\sum_x V_x \leq 65$. With the linear hull, we mount a 120-round attack with 16-round forward and 14-round backward. And 28-bit key need to be guessed, while another 8-bit key are not. The included bits of the registers are listed in Table 6 and the included key bits are listed in Table 4.

For KATAN64, the 76-round linear hull with *potential* 2^{-57} are obtained with added condition $\sum_x V_x \leq 77$. With the linear hull, we mount a 94-round attack with 12-round forward and 6-round backward. And 20-bit key need to be guessed, while another 13-bit key are not. The included bits of the registers are listed in Table 7 and the included key bits are listed in Table 4.

The data complexity N of the linear hull attack is set by $2ALH^{-1}$. Suppose the length of the guessed-key is l_k, the the time complexity is $N * 2^{l_k}$. The complexity is summarized in Table 1.

Table 4. The included key bits

version	round of guessed-key for k_a
KATAN32	13, 9, 8, 6, 4, 3, 2, 1, 0, 111, 114, 115, 116, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130
KATAN48	7, 4, 3, 2, 1, 0, 107, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119,
KATAN64	5, 4, 3, 2, 1, 0, 89, 90, 91, 92,93
version	round of guessed-key for k_b
KATAN32	10, 7, 5, 4, 3, 2, 1, 0, 111, 113, 115, 117, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130
KATAN48	10, 6, 3, 2, 1, 0, 113, 116, 117, 118
KATAN64	5, 4, 2, 1, 0, 89, 91, 92, 93
version	non-guessed key
KATAN32	$k_a^5, k_a^{16}, k_a^{107}, k_a^{112}, k_a^{117}, k_b^8, k_b^{13}, k_b^{17}, k_b^{105}, k_b^{116}$
KATAN48	$k_a^5, k_a^7, k_a^{53}, k_b^2, k_b^4, k_b^{55}, k_b^{57}, k_b^{59}$
KATAN64	$k_a^5, k_a^8, k_a^9, k_a^{88}, k_a^{90}, k_a^{91}, k_a^{93}, k_b^2, k_b^7, k_b^9, k_b^{88}, k_b^{89}, k_b^{92}$

5 Conclution

We first considered the linear hull cryptanalysis on KATAN. The cryptanlaysis for KATAN32/48 is the best single-key known plaintext attack. Besides, we evaluated the security analysis on linear cryptanalysis taking the dependence of the sboxes into consideration.

References

1. Wenling Wu and Lei Zhang. Lblock: A lightweight block cipher. In Javier Lopez and Gene Tsudik, editors, *Applied Cryptography and Network Security*, volume 6715 of *Lecture Notes in Computer Science*, pages 327–344. Springer Berlin Heidelberg, 2011.
2. A. Bogdanov, L.R. Knudsen, G. Leander, C. Paar, A. Poschmann, M.J.B. Robshaw, Y. Seurin, and C. Vikkelsoe. Present: An ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer Berlin Heidelberg, 2007.
3. Jian Guo, Thomas Peyrin, Axel Poschmann, and Matt Robshaw. The led block cipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems ?CHES 2011*, volume 6917 of *Lecture Notes in Computer Science*, pages 326–341. Springer Berlin Heidelberg, 2011.
4. Martin R. Albrecht, Benedikt Driessen, Elif Bilge Kavun, Gregor Leander, Christof Paar, and Tolga Yalçin. Block ciphers - focus on the linear layer (feat. PRIDE). In *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference,*

Table 5. The included bits of the registers for KATAN32

Round	bits of register L_1	bits of register L_2
0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
1	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18
2	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18
3	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18
4	0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 15, 17, 18
5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 15, 16, 18
6	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 14, 16, 17
7	1, 2, 3, 4, 5, 6, 7, 10, 11	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 15, 17, 18
8	0, 2, 3, 4, 5, 6, 7, 8, 11, 12	1, 2, 3, 5, 6, 7, 8, 9, 10, 12, 13, 16, 18
9	0, 1, 3, 4, 5, 7, 8, 12	0, 2, 3, 4, 6, 7, 8, 9, 10, 11, 14, 17
10	1, 2, 4, 5, 6, 9	0, 1, 3, 4, 5, 7, 8, 9, 10, 11, 12, 15, 18
11	0, 2, 3, 5, 6, 7, 10	16, 1, 2, 4, 5, 6, 8, 10, 12
12	1, 3, 4, 6, 7, 8, 11	17, 2, 3, 5, 6, 7, 9, 11, 13
13	2, 4, 5, 7, 8, 9, 12	18, 3, 4, 6, 7, 8, 10, 12, 14
14	0, 10, 3, 5, 6	0, 4, 5, 7, 9, 15
15	1, 11, 4, 6, 7	16, 1, 5, 6, 8, 10
16	8, 2, 12, 5, 7	17, 2, 6, 7, 9, 11
17	8, 3	0, 18, 3, 7, 8, 10, 12
18	0, 9, 4	1
19	1, 10, 5	2
20	2, 11, 6	3
21	3, 12, 7	4
105	10	18
106	0, 11	8, 9, 11, 4, 13
107	1, 12	9, 10, 12, 5, 14
108	8, 9, 2, 6	0, 6, 10, 11, 13, 15
109	9, 10, 3, 7	16, 1, 7, 11, 12, 14
110	8, 10, 11, 4	17, 2, 8, 12, 13, 15
111	9, 11, 12, 5	16, 18, 3, 9, 13, 14
112	0, 6, 8, 9, 10, 12	0, 4, 8, 9, 10, 11, 13, 14, 15, 17
113	1, 4, 6, 7, 8, 9, 10, 11	0, 1, 5, 9, 10, 11, 12, 14, 15, 16, 18
114	0, 2, 5, 7, 8, 9, 10, 11, 12	1, 2, 4, 6, 8, 9, 10, 11, 12, 13, 15, 16, 17
115	1, 3, 4, 6, 8, 9, 10, 11, 12	0, 2, 3, 5, 7, 9, 10, 11, 12, 13, 14, 16, 17, 18
116	0, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 3, 4, 6, 8, 9, 10, 11, 12, 13, 14, 15, 17, 18
117	0, 1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 18
118	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17
119	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
120	0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
121	0, 1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
122	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
123	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
124	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
125	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
126	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
127	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
128	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
129	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
130	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
131	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18

Table 6. The included bits of the registers for KATAN48

Round	bits of register L_1
0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
1	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 17, 18
2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17
3	1, 3, 4, 5, 6, 7, 9, 10, 11, 12, 14, 15, 17, 18
4	0, 1, 3, 5, 6, 9, 11, 12, 14, 17
5	2, 3, 5, 8, 11, 14
6	16, 1, 4, 5, 7, 10, 13
7	0, 1, 3, 6, 7, 9, 12, 15, 18
8	2, 3, 5, 8, 11, 14
9	16, 1, 4, 5, 7, 10, 13
10	18, 3, 6, 7, 9, 12, 15
11	8, 0, 11, 5
12	10, 7, 2, 13
13	4, 15, 12, 9
14	17, 11, 6, 14
15	
16	
106	0, 9, 18
107	17, 2, 11, 14, 9
108	16, 8, 11, 4, 13
109	18, 13, 10, 6, 15
110	17, 8, 9, 12, 14, 15
111	1, 7, 8, 10, 11, 13, 14, 16, 17
112	3, 8, 9, 10, 12, 13, 15, 16, 18
113	5, 8, 9, 10, 11, 12, 14, 15, 17, 18
114	0, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17
115	0, 2, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 18
116	2, 4, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
117	0, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
118	0, 1, 2, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
119	1, 2, 3, 4, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
120	1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18
Round	bits of register L_2
0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 24, 26, 27, 28
1	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 23, 24, 26, 28
2	0, 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19, 20, 21, 22, 25, 26, 28
3	0, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 17, 18, 19, 20, 21, 22, 24, 27, 28
4	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 16, 17, 19, 20, 23, 24, 26
5	0, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19, 21, 22, 25, 26, 28
6	2, 4, 5, 6, 9, 10, 11, 12, 13, 14, 15, 17, 18, 19, 20, 21, 24, 27, 28
7	4, 6, 8, 11, 12, 13, 14, 17, 19, 21, 26
8	1, 6, 8, 10, 13, 14, 15, 16, 19, 21, 23, 28
9	16, 18, 3, 25, 10, 12
10	18, 20, 5, 27, 12, 14
11	1
12	3
13	5
14	7
15	0, 9
16	2, 11
106	20
107	1, 22
108	0, 24, 3
109	2, 26, 5
110	1, 4, 28, 7
111	0, 17, 3, 21, 6, 23, 8, 9, 15
112	0, 2, 5, 8, 10, 11, 17, 19, 23, 25
113	1, 2, 4, 7, 10, 12, 13, 19, 21, 25, 27
114	0, 1, 3, 4, 6, 7, 9, 12, 14, 15, 16, 20, 21, 22, 23, 27
115	0, 2, 3, 5, 6, 7, 8, 9, 11, 14, 16, 17, 18, 20, 22, 23, 24, 25
116	1, 2, 4, 5, 7, 8, 9, 10, 11, 13, 16, 18, 19, 20, 22, 24, 25, 26, 27
117	0, 1, 3, 4, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 18, 20, 21, 22, 24, 26, 27, 28
118	0, 1, 2, 3, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 17, 18, 20, 21, 22, 23, 24, 26, 28
119	0, 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 28
120	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28

Table 7. The included bits of the registers L_1 for KATAN64

Round	bits of register L_1
0	0, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15, 18, 19, 22, 23
1	0, 1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 21, 22
2	0, 2, 3, 4, 5, 6, 9, 10, 11, 13, 14, 15, 18, 19, 20, 24
3	0, 1, 3, 5, 6, 7, 8, 9, 12, 13, 14, 16, 17, 18, 21, 22
4	3, 4, 6, 8, 9, 10, 11, 12, 15, 17, 19, 20, 21, 24
5	2, 6, 7, 9, 11, 13, 15, 18, 20, 22, 24
6	1, 18, 5, 9, 10, 14
7	17, 4, 21, 8, 12, 13
8	16, 2, 20, 7, 24, 11, 15
9	10, 19, 5, 14, 23
10	8, 17, 2
11	11, 20, 5
12	8, 14, 23
88	8, 24, 18, 12, 15
89	2, 21, 23, 11, 18, 14, 15
90	0, 1, 5, 13, 14, 17, 18, 21, 22, 24
91	3, 4, 8, 12, 14, 16, 17, 18, 20, 21, 23, 24
92	1, 2, 6, 7, 11, 12, 13, 14, 15, 17, 18, 19, 20, 21, 22, 23, 24
93	0, 1, 4, 5, 9, 10, 12, 13, 14, 15, 16, 17, 18, 20, 21, 22, 23, 24
94	0, 2, 3, 4, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24
Round	bits of register L_2
0	0, 1, 2, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 28, 30, 31, 32, 33, 35, 36, 37, 38
1	0, 1, 3, 4, 5, 7, 8, 9, 10, 12, 14, 15, 16, 17, 19, 20, 21, 23, 24, 25, 28, 29, 31, 33, 34, 36, 38
2	0, 3, 4, 6, 7, 8, 10, 11, 12, 13, 15, 17, 18, 19, 20, 23, 24, 27, 28, 31, 32, 36, 37
3	2, 3, 6, 7, 9, 11, 13, 14, 15, 18, 35, 20, 21, 22, 26, 30, 31
4	0, 5, 6, 9, 10, 12, 14, 16, 17, 18, 21, 23, 24, 25, 29, 33, 34, 38
5	32, 2, 3, 37, 8, 9, 13, 15, 19, 20, 21, 24, 26, 27
6	0, 2, 35, 5, 6, 11, 12, 16, 18, 22, 24, 29, 30
7	32, 33, 3, 5, 38, 8, 9, 14, 15, 19, 21, 25, 27
8	18, 35, 22, 6, 8, 11, 30
9	33, 2, 38, 9, 11, 14, 21, 25
10	1, 5, 14
11	8, 17, 4
12	11, 20, 7
88	2, 5, 38, 11, 34, 14, 18, 20, 23, 29
89	32, 2, 36, 37, 8, 12, 14, 17, 21, 23, 24, 26, 28, 5
90	1, 34, 35, 5, 8, 10, 11, 15, 16, 17, 20, 22, 23, 24, 26, 27, 29, 31
91	0, 32, 34, 4, 37, 38, 8, 11, 13, 14, 18, 19, 20, 23, 2, 25, 26, 27, 29, 30
92	1, 2, 3, 5, 7, 11, 12, 14, 16, 17, 21, 22, 23, 24, 26, 27, 28, 29, 30, 32, 33, 35, 36, 37
93	0, 1, 2, 4, 5, 6, 8, 10, 11, 14, 15, 16, 17, 19, 20, 22, 23, 24, 25, 26, 27, 29, 30, 31, 32, 33, 34, 35, 36, 38
94	0, 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 17, 18, 19, 20, 22, 23, 24, 25, 26, 27, 28, 29, 30, 32, 33, 34, 35, 36, 37, 38

Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I, pages 57–76, 2014.

5. Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The simon and speck families of lightweight block ciphers. *IACR Cryptology ePrint Archive*, 2013:404, 2013.
6. Christophe De Cannière, Orr Dunkelman, and Miroslav Knezevic. KATAN and KTANTAN - A family of small and efficient hardware-oriented block ciphers. In *Cryptographic Hardware and Embedded Systems - CHES 2009, 11th International Workshop, Lausanne, Switzerland, September 6-9, 2009, Proceedings*, pages 272–288, 2009.
7. Simon Knellwolf, Willi Meier, and María Naya-Plasencia. Conditional differential cryptanalysis of nlfsr-based cryptosystems. In *Advances in Cryptology - ASIACRYPT 2010 - 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings*, pages 130–145, 2010.
8. Martin R. Albrecht and Gregor Leander. An all-in-one approach to differential cryptanalysis for small block ciphers. In *Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers*, pages 1–15, 2012.
9. Takanori Isobe and Kyoji Shibutani. All subkeys recovery attack on block ciphers: Extending meet-in-the-middle approach. In *Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers*, pages 202–221, 2012.

10. Thomas Fuhr and Brice Minaud. Match box meet-in-the-middle attack against KATAN. In *Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers*, pages 61–81, 2014.
11. Mitsuru Matsui. Linear cryptanalysis method for des cipher. In Tor Helleseth, editor, *Advances in Cryptology ?EUROCRYPT ?3*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer Berlin Heidelberg, 1994.
12. Kaisa Nyberg. Linear approximation of block ciphers. In *Advances in CryptologyEUROCRYPT'94*, pages 439–444. Springer, 1995.
13. Siwei Sun, Lei Hu, Meiqin Wang, Peng Wang, Kexin Qiao, Xiaoshuang Ma, Danping Shi, Ling Song, and Kai Fu. Towards finding the best characteristics of some bit-oriented block ciphers and automatic enumeration of (related-key) differential and linear characteristics with predefined properties. *IACR Cryptology ePrint Archive*, 2014:747, 2014.
14. Danping Shi, Lei Hu, Siwei Sun, Ling Song, Kexin Qiao, and Xiaoshuang Ma. Improved linear (hull) cryptanalysis of round-reduced versions of SIMON. *IACR Cryptology ePrint Archive*, 2014:973, 2014.
15. Siwei Sun, Lei Hu, Peng Wang, Kexin Qiao, Xiaoshuang Ma, and Ling Song. Automatic security evaluation and (related-key) differential characteristic search: Application to simon, present, lblock, des(l) and other bit-oriented block ciphers. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology ?ASIACRYPT 2014*, volume 8873 of *Lecture Notes in Computer Science*, pages 158–178. Springer Berlin Heidelberg, 2014.
16. Andrey Bogdanov and Vincent Rijmen. Linear hulls with correlation zero and linear cryptanalysis of block ciphers. *Designs, Codes and Cryptography*, 70(3):369–383, 2014.

6 Appendix

Table 8. The 84-round linear characteristic for KATAN32

Round	input masks of register L_1	input masks of register L_2
0	1000010001000	0000000000000010000
1	0000000000000	0000000000000100001
2	0000000000000	0000000000001000010
3	0000000000000	0000000000010000100
4	0000000000000	0000000000100001000
5	0000000000000	0000000001000010000
6	0000000000000	0000000010000100000
7	0000000000000	0000000100001000000
8	0000000000000	0000001000010000000
9	0000000000000	0000010000100000000
10	0000000000000	0000100001000000000
11	0000000000000	0001000010000000000
12	0000000000000	0010000100000000000
13	0000000000000	0100001000000000000
14	0000000000000	1000010000000000000
15	0000000000001	0000100000100000000
16	0000000000010	0001000001000000000
17	0000000000100	0010000010000000000
18	0000000001000	0100000100000000000
19	0000000010000	1000001000000000000
20	0000000100001	0000010000100000000
21	0000001000010	0000100001000000000
22	0000010000100	0001000010000000000
23	0000100001000	0010000100000000000
24	0001000010000	0100001000000000000
25	0010000100000	1000010000000000000
26	0100001000001	0000100000100000000
27	1000010000010	0001000001000000000
28	0000000000100	0010000010000000001
29	0000000001000	0100000100000000010
30	0000000010000	1000001000000000100
31	0000000100001	0000000000100001000
32	0000001000010	0000000001000010000
33	0000010000100	0000000010000100000
34	0000100001000	0000000100001000000
35	0001000010000	0000001000010000000
36	0010000100000	0000010000100000000
37	0100001000000	0000100001000000000
38	1000010000000	0001000010000000000
39	0000000000000	0010000100000000001
40	0000000000000	0100001000000000010
41	0000000000000	1000010000000000100
42	0000000000001	0000100000100001000
43	0000000000010	0001000001000010000
44	0000000000100	0010000010000100000
45	0000000001000	0100000100001000000
46	0000000010000	1000001000010000000
47	0000000100001	0000000000000000000
48	0000001000010	0000000000000000000
49	0000010000100	0000000000000000000
50	0000100001000	0000000000000000000
51	0001000010000	0000000000000000000
52	0010000100000	0000000000000000000
53	0100001000000	0000000000000000000
54	1000010000000	0000000000000000000
55	0000000000000	0000000000000000001
56	0000000000000	0000000000000000010
57	0000000000000	0000000000000000100
58	0000000000000	0000000000000001000
59	0000000000000	0000000000000010000
60	0000000000000	0000000000000100000
61	0000000000000	0000000000001000000
62	0000000000000	0000000000010000000
63	0000000000000	0000000000100000000
64	0000000000000	0000000001000000000
65	0000000000000	0000000010000000000
66	0000000000000	0000000100000000000
67	0000000000000	0000001000000000000
68	0000000000000	0000010000000000000
69	0000000000000	0000100000000000000
70	0000000000000	0001000000000000000
71	0000000000000	0010000000000000000
72	0000000000000	0100000000000000000
73	0000000000000	1000000000000000000
74	0000000000001	0000000000100000000
75	0000000000010	0000000001000000000
76	0000000000100	0000000010000000000
77	0000000001000	0000000100000000000
78	0000000010000	0000001000000000000
79	0000000100000	0000010000000000000
80	0000001000000	0000100000000000000
81	0000010000000	0001000000000000000
82	0000100000000	0010000000000000000
83	0001000000000	0100000000000000000
84	0010000000000	1000000000000000000

Table 9. The 84-round linear characteristic for KATAN48

Round	input masks of register L_1	input masks of register L_2
0	0000000001000001001	00000100000000010000000000000
1	0000000100000100100	00010000000001000000000000000
2	0000010000010010000	01000000000100000000000000000
3	0001000001001000001	00000000000000000000000000000
4	0100000100100000100	00000000000000000000000000000
5	0000000010010010000	00000000000000000000000000001
6	0000001001001000000	00000000000000000000000000100
7	0000100100100000000	00000000000000000000000010000
8	0010010010000000000	00000000000000000000001000000
9	1001001000000000000	00000000000000000000100000000
10	0000000000000000000	00000000000000000010000000010
11	0000000000000000000	00000000000000001000000001000
12	0000000000000000000	00000000000000100000000100000
13	0000000000000000000	00000000000010000000010000000
14	0000000000000000000	00000000001000000001000000000
15	0000000000000000000	00000000100000000100000000000
16	0000000000000000000	00000010000000010000000000000
17	0000000000000000000	00001000000001000000000000000
18	0000000000000000000	00100000000100000000000000000
19	0000000000000000000	10000000010000000000000000000
20	0000000000000000010	00000000000000000000000000000
21	0000000000000001000	00000000000000000000000000000
22	0000000000000100000	00000000000000000000000000000
23	0000000000010000000	00000000000000000000000000000
24	0000000001000000000	00000000000000000000000000000
25	0000000100000000000	00000000000000000000000000000
26	0000010000000000000	00000000000000000000000000000
27	0001000000000000000	00000000000000000000000000000
28	0100000000000000000	00000000000000000000000000000
29	0000010000010000000	00000000000000000000000000001
30	0001000001000000000	00000000000000000000000000100
31	0100000100000000000	00000000000000000000000010000
32	0000000000000000000	00000000000000000000001000001
33	0000000000000000000	00000000000000000000100000100
34	0000000000000000000	00000000000000000010000010000
35	0000000000000000000	00000000000000001000001000000
36	0000000000000000000	00000000000000100000100000000
37	0000000000000000000	00000000000010000010000000000
38	0000000000000000000	00000000001000001000000000000
39	0000000000000000000	00000000100000100000000000000
40	0000000000000000000	00000010000010000000000000000
41	0000000000000000000	00001000001000000000000000000
42	0000000000000000000	00100000100000000000000000000
43	0000000000000000000	10000010000000000000000000000