New Impossible Differential Characteristic of SPECK64 using MILP

HoChang Lee*, HyungChul Kang*, Deukjo Hong**,
Jaechul Sung***, Seokhie Hong*

*Graduate School of Information Security, Korea University.

**Department of Information Technology, Chonbuk National University.

***Department of Mathematics, University of Seoul.

abstract

Impossible differential attack is one of powerful methods for analyzing block ciphers. When designing block ciphers, it must be safe for impossible differential attacks. In case of impossible differential attack, the attack starts from finding the impossible differential characteristic. However, in the case of the ARX-based block cipher, these analyzes were difficult due to the addition of modulus. In this paper, we introduce 150 new six-round impossible differential characteristics of ARX-basef block cipher, SPECK64, using Mixed Integer Linear Programming (MILP) base impossible differential characteristic search proposed by Cui [3] etc.

I. Introduction

Impossible differential attack introduced by Knudsen [6] and Biham et al. [7] is an attack method that reduces the key space by removing the incorrect key by using the differential characteristic with probability 0. In impossible differential attacks, finding impossible differential characteristics of long rounds is one of the important parts of attack. However, in the case of ARX base block cipher consisting of modulus addition, bit circular movement and XOR operation, it is difficult to find impossible differential characteristics by modulo addition. Biryukov et al. [5] and Muoha et al. [8] proposed such an ARX-based block cipher, proposed an automatic differential character search method. Cui et al. [3] proposed a method to search for impossible differential characteristics of ARX-based block ciphers by applying MILP-based method [2] such as Fu.

The Mixed-Integer Linear Programming (MILP) problem is a problem that optimizes the value of the objective function with a value that satisfies the linear constraint equation. A linear constraint expression can be created for each bit and used for difference analysis.

In this paper, using the method proposed by Cui et al., We used MILP-based method and searched for impossible differential characteristics for SPECK 64. As a result, We found impossible differnetial characteristics of 6-round. Section 2 describes the MILP-based search method, and Section 3 shows the results applied to SPECK 64. Finally, in chapter 4, we can not conclude.

table 1. 13 Linear inequality expressing modular addition

$$\begin{aligned}
&\beta[i]-\gamma[i]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq 0,\\
&\alpha[i]-\beta[i]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq 0,\\
&-\alpha[i]+\gamma[i]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq 0,\\
&-\alpha[i]-\beta[i]-\gamma[i]-(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq -3,\\
&\alpha[i]+\beta[i]+\gamma[i]-(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq 0,\\
&-\beta[i]+\alpha[i+1]+\beta[i+1]+\gamma[i+1]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq 0,\\
&\beta[i]+\alpha[i+1]-\beta[i+1]+\gamma[i+1]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq 0,\\
&\beta[i]-\alpha[i+1]+\beta[i+1]+\gamma[i+1]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq 0,\\
&\alpha[i]+\alpha[i+1]+\beta[i+1]-\gamma[i+1]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq 0,\\
&\gamma[i]-\alpha[i+1]-\beta[i+1]-\gamma[i+1]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq -2,\\
&-\beta[i]+\alpha[i+1]-\beta[i+1]-\gamma[i+1]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq -2,\\
&-\beta[i]-\alpha[i+1]+\beta[i+1]-\gamma[i+1]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq -2,\\
&-\beta[i]-\alpha[i+1]-\beta[i+1]+\gamma[i+1]+(\neg eq(\alpha[i],\beta[i],\gamma[i])) \geqq -2.
\end{aligned}$$

II. MILP-based differential search

In this chapter, we explain notation and the concept of MILP-based impossible differential characteristic search.

2.1. Notations

In this paper, the following notation is used.

$\oplus$: XOR

$\neg$: Complement

$\alpha[i]$: The i th bit from the right of a

Δ_x: A set of all input differences

Δ_y: A set of all output differences

$\ggg a(\lll a)$: Bit circular movement a times as large as the right (left)

$L[i]$: The Left word of input value of i th round of SPECK

$R[i]$: The right word of the input value of the i th round of SPECK

l_i^j: The j th bit from the right of $L[i]$

r_i^j: The j th bit from the right of $R[i]$

2.2. MILP base impossible differential feature search

Cui et al. [3] proposed a Mixed-Integer Linear Programming (MILP) base impossible differential characteristic search method for ARX-based block ciphers. The MILP problem is a problem of optimizing the value of the objective function for values satisfying the linear constraint equation. In order to search for the MILP base impossible differential characteristic, the objective function is set to the expression expressing the probability of the differential characteristic, and the linear constraint expression is configured to constitute the cipehr system. Therefore, with respect to the set cryptosystem, the optimum differential characteristic probability constituting the cryptosystem corresponding to the answer to the MILP problem is obtained. However, if the answer to the MILP problem can not be obtained for a specific input or output differential, it means that the differential characteristic can not be configured in the specified cryptosystem for that input and output differential value. Therefore, the input and output differenctials become invalid differential characteristics of

Algorithm 1. Impossible differential feature search

```
// $\Delta_x$ : A set of all input differences
// $\Delta_y$ : A set of all output differences
// round : Round length of impossible difference characteristic you want to find
def mycallback(model,where) :
    if where == GRB.Callback.MIP :
        best = model.cbGet(GRB.Callback.MIP_OBSJBST)
    if best >= 0 :
        m.terminate()
for $i$ in $\Delta_x$ :
    for $j$ in $\Delta_y$ :
        Create an ″idc.lp″ file in the difference between the $i$ th input $j$ difference and the output
        m = read("idc.lp")
        m.optimize(mycallback)
        if m.status == 3 :
            Save difference between $i$ th input difference and $j$ th output
```

the given cryptosystem. Find a case where you can not get the answer to the MILP problem and search for MILP-based impossible differential characteristics.

In order to set the objective function, Lipmaa etc [1] proposed the two organizations and its arrangement is as follows.

Theorem 1. Existence of differential [1]

The necessary and sufficient condition that the differential $(\alpha,\beta\rightarrow\gamma)$ has the probability that it is not 0 is the following two conditions.

1.$\alpha[0]\oplus\beta[0]\oplus\gamma[0]=0$

2.$\alpha[i-1]=\beta[i-1]=\gamma[i-1]=\alpha[i]\oplus\beta[i]\oplus\gamma[i]$

($\alpha[i-1]=\beta[i-1]=\gamma[i-1]$, $i\in[1,n-1]$ 경우)

Theorem 2. Probability of differential [1]

When the differential $(\alpha,\beta\rightarrow\gamma)$ has a probability that it is not 0, the probability is $2^{-\sum_{i=0}^{n-2}\neg eq(\alpha[i],\beta[i],\gamma[i])}$. At this time, it is $eq(\alpha[i],\beta[i],\gamma[i])=\begin{cases}1 & \alpha[i]=\beta[i]=\gamma[i]\\ 0 & others\end{cases}$.

It is possible to express and calculate the differenctials by adding legally to the two arrangements proposed by Lipmaa. Also, it can be expressed as an objective function of the MILP problem using an $eq(\alpha[i],\beta[i],\gamma[i])$.

In order to construct an ARX-based block cipher with a linear constraint expression, it is sufficient to express modulus, bit circular movement, XOR, input, and output value which are operations constituting ARX. The way of expression is as follows.

Modular addition)

If vectors $(\alpha[i],\beta[i],\gamma[i],\alpha[i+1],\beta[i+1],\gamma[i+1],\neg eq(\alpha[i],\beta[i],\gamma[i]))$ are expressed using the points satisfying the arrangement proposed by Lipmaa, 56 vectors are generated. The nature of modulo addition is expressed in the form of 13 lines which can express only 56 vectors, and the form of 13 lines is as shown in table 1.

Rotation)

Correct the index of the bit used in the addition of the modular and use it.

XOR)

In order to express $a \oplus b \oplus c = 0$ which is a 3-bit XOR state, it is expressed in the form of the next one line.

$$a+b+c=2d_{\oplus}$$

Input, output value)

It expresses in the form of a line to set bits of input and output values to the bit you want to do.

By setting the objective function and the linear constraint equation in the above method, it is possible to search for MILP-based impossible differential characteristics against ARX-based block cipher.

Cui et al. [3] proposed an algorithm for finding impossible differential characteristics by solving the MILP problem by using Gurobi, an optimization program, to create a cryptosystem in lp file format, which is an input form of Gurobi. The algorithm corrected and used in the paper which saw this is as in [Algorithm 1].

In the case of the *mycallback* function in [algorithm 1], it is a function that interrupts optimization of the cryptosystem, not impossible differential characteristics. The input and output specified by discovering the differential characteristics satisfied with the input and output values set in case of finding any year in the set cryptosystem are not impossible differential characteristics, so optimization finish.

III. New impossible differential characteristic of SPECK 64

In this chapter, we introduce the new impossible differencetial characteristics of SPECK64 using the method described in Chapter 2.

3.1. SPECK

SPECK is an ARX-based lightweight block cipher published by NSA in 2013 [4]. The shape of the round function is shown in Fig. It is as follows.

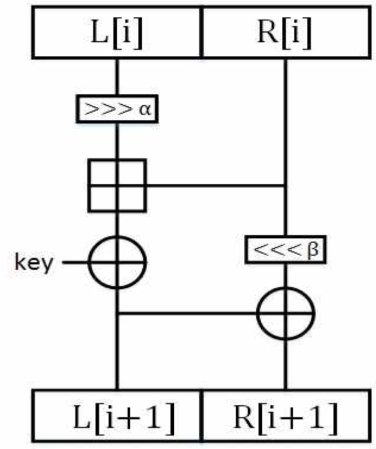

Fig. 1. One Round of SPECK

α and β values mean the bitwise circular movement of left and right words respectively, and the round of the SPECK family is the same as table 2.

table 2. Number of rounds and α,β of SPECK Family

type	size of key	α	β	# of rounds
SPECK32	64	7	2	22
SPECK48	72	8	3	22
	96			23
SPECK64	96			26
	128			27

SPECK96	96			28
	144			29
SPECK128	128			32
	192			33
	256			34

For details about the key schedule, etc., see [4].

In the case of SPECK64, analysis was made up to 19 rounds by differential analysis and 14 rounds by rectangle attack. As far as SPECK family has not been attacked by impossible differential attacks, this paper introduces the new impossible differential characteristics of SPECK64.

3.2. Application to SPECK64

In order to construct a round of SPECK 64 with a linear constraint expression, it is necessary to satisfy the modular addition and XOR when the input word of SPECK64 round is $L[i]=(l_i^{15},\cdots,l_i^0)$ and $R[i]=(r_i^{15},\cdots,r_i^0)$. Therefore, the following two conditions must be satisfied.

$$\begin{aligned}&(L[i]\gg\alpha)+R[i]=L[i+1]\ (\mathrm{mod}\ 2^{32})\\&(R[i]\ll\beta)\oplus L[i+1]=R[i+1]\end{aligned}$$

In the case of modular addition which is the first condition, 13 pieces of table 1 expressing modular addition from 0th bit to 31st bit of $L[i]$, $R[i]$, and $L[i+1]$ Can be expressed and expressed using the linear constraint equation of. In this case, the third round is expressed in $eq(\alpha[j],\beta[j],\gamma[j])$ and is as follows. $(j\in[0,31])$

$$r_i^j-l_{i+1}^j+e_i^j\geqq 0$$
$$l_i^{j+\alpha\,(\mathrm{mod}\,32)}-r_i^j+e_i^j\geqq 0$$
$$-l_i^{j+\alpha\,(\mathrm{mod}\,32)}+l_{i+1}^j+e_i^j\geqq 0$$
$$-l_i^{j+\alpha\,(\mathrm{mod}\,32)}-r_i^j-l_{i+1}^j-e_i^j\geqq -3$$
$$l_i^{j+\alpha\,(\mathrm{mod}\,32)}+r_i^j+l_{i+1}^j-e_i^j\geqq 0$$
$$-r_i^j+l_i^{j+1+\alpha\,(\mathrm{mod}\,32)}+r_i^{j+1}+l_{i+1}^{j+1}+e_i^j\geqq 0$$
$$r_i^j+l_i^{j+1+\alpha\,(\mathrm{mod}\,32)}+r_i^{j+1}+l_{i+1}^{j+1}+e_i^j\geqq 0$$
$$r_i^j-l_i^{j+1+\alpha\,(\mathrm{mod}\,32)}+r_i^{j+1}+l_{i+1}^{j+1}+e_i^j\geqq 0$$
$$l_i^{j+\alpha(\mathrm{mod}\,32)}+l_i^{j+1+\alpha\,(\mathrm{mod}\,32)}+r_i^{j+1}-l_{i+1}^{j+1}+e_i^j\geqq 0$$
$$l_{i+1}^j-l_i^{j+1+\alpha\,(\mathrm{mod}\,32)}-r_i^{j+1}-l_{i+1}^{j+1}+e_i^j\geqq -2$$
$$-r_i^j+l_i^{j+1+\alpha\,(\mathrm{mod}\,32)}-r_i^{j+1}-l_{i+1}^{j+1}+e_i^j\geqq -2$$
$$-r_i^j-l_i^{j+1+\alpha\,(\mathrm{mod}\,32)}+r_i^{j+1}-l_{i+1}^{j+1}+e_i^j\geqq -2$$
$$-r_i^j-l_i^{j+1+\alpha\,(\mathrm{mod}\,32)}-r_i^{j+1}+l_{i+1}^{j+1}+e_i^j\geqq -2$$

In the above state, it can be represented once to the bit circulation movement by the linear constraint expression of the modular addition to the case $j+\alpha\ (\mathrm{mod}\,32)$. In addition, the condition of LSB of modal addition is represented by XOR as follows.

$$l_i^0+r_i^0+l_{i+1}^0=2d_{\oplus}$$

In the case of XOR which is the second condition, it is necessary to satisfy the XOR between the 0th bit and the 32nd word, so it is as follows.

$$r_i^{0+32-\beta\,(\mathrm{mod}\,32)}+l_{i+1}^0+r_{i+1}^0=2d_{\oplus}$$
$$r_i^{1+32-\beta\,(\mathrm{mod}\,32)}+l_{i+1}^1+r_{i+1}^1=2d_{\oplus}$$
$$\vdots$$
$$r_i^{30+32-\beta\,(\mathrm{mod}\,32)}+l_{i+1}^{30}+r_{i+1}^{30}=2d_{\oplus}$$
$$r_i^{31+32-\beta\,(\mathrm{mod}\,32)}+l_{i+1}^{31}+r_{i+1}^{31}=2d_{\oplus}$$

Also in this case, Bit Cycle Movement was also expressed in $j+32-\beta\,(\mathrm{mod}\,32)$ $(j\in[0,32])$.

Therefore, since SPECK64 has a word size of 32 in the case of modular addition in one round to construct SPECK64 in an cryptosystem, 31 × 13 = 403 pieces and 1 line in LSB condition, a total of 417 line types are required . Even in XOR, 32 × 1 = 32 line types are required. In addition, 128 ×

2 = 258 line types are required to set the input and output differential values. Therefore, to construct an n-round linear system, a format of 449 n + 258 lines in total is required.

3.3. New impossible differential characteristic of SPECK64

In the case of SPECK64, the number of settable input and output differential values is $(2^{64}-1)(2^{64}-1) \approx 2^{128}$. Due to the limitation of computing power, we only searched for differentials input and output, each with only one bit difference value. In other words, I examined one case. In the experimental environment, Gurobi 6.5.2 was executed on Inter (R) Core (TM) i 5 - 4690 (3.50 Ghz, 9.9 GB RAM, Windows 10) using one thread and searched for about 10 minutes. I visited 150 six-round impossible differential characteristics. Some of them are as follows.

$$
\begin{gathered}
(00\,00\,00\,00\,00\,04\,00\,00) \not\rightarrow_6 (00\,00\,00\,00\,00\,00\,00\,02) \\
(00\,00\,00\,00\,00\,08\,00\,00) \not\rightarrow_6 (00\,00\,00\,00\,00\,00\,00\,02) \\
\vdots \\
(80\,00\,00\,00\,00\,00\,00\,00) \not\rightarrow_6 (00\,00\,00\,00\,00\,00\,00\,02) \\
(80\,00\,00\,00\,00\,00\,00\,00) \not\rightarrow_6 (00\,00\,00\,00\,00\,00\,00\,04)
\end{gathered}
$$

In the input and output from the 7th round of SPECK64, when there was only one bit difference value, the impossible differential characteristic was not found.

IV. Conclusion

In this thesis, using the method proposed by [3] such as Cui, we found the new SPECK64 6-round impossible differential characteristics. Until now SPECK64 impossible differential characteristics have not been introduced. It is expected that it will be able to find a impossible differential characteristic of a longer round when inputting the differential and searching with widening the range of the output value, and this makes it possible to use impossible differential attack on the SPECK family using this. It is possible to search the impossible differential characteristics of ARX-based block cipher such as LEA in future MILP base impossible differential search method. Also, it can be one of the ways to check the security against impossible differential attacks when designing an ARX-based block cipher.

[References]

[1] Lipmaa, Helger, and Shiho Moriai. "Efficient algorithms for computing differential properties of addition." International Workshop on Fast Software Encryption. Springer Berlin Heidelberg, 2001.

[2] Fu, Kai, et al. "MILP-Based Automatic Search Algorithms for Diff erential and Linear Trails for SPECK." representations 21 (2016): 27.

[3] Cui, Cui, et al. "New Automatic Search Tool for Impossible Differentials and Zero-Correlation Linear Approximations."

[4] Beaulieu, Ray, et al. "The SIMON and SPECK lightweight block ciphers." Proceedings of the 52nd Annual Design Automation Conference. ACM, 2015.

[5] Biryukov, Alex, Vesselin Velichkov, and Yann Le Corre. "Automatic search for the best trails in arx: Application to block cipher SPECK." Fast Software Encryption - FSE. 2016.

[6] Knudsen, Lars. "DEAL-a 128-bit block cipher." complexity 258.2 (1998): 216.

[7] Biham, Eli, Alex Biryukov, and Adi

Shamir. "Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials." International Conference on the Theory and Applications of Cryptographic Techniques. Springer Berlin Heidelberg, 1999.

[8] Mouha, Nicky, et al. "Differential and linear cryptanalysis using mixed-integer linear programming." International Conference on Information Security and Cryptology. Springer Berlin Heidelberg, 2011.