Krzysztof BUCHOLC

Poznań University of Technology

doi:10.15199/48.2017.01.20

An approach to evaluation of S-boxes

Abstract. *The paper presents an approach to analysis of substitution boxes (S-boxes) used in block ciphers. The proposed method can serve as an additional criterion for evaluation of S-box quality. In some cases the method may reveal that the S-box design is based on some simple mathematic formula.*

Streszczenie. *W pracy przedstawiono pewne podejście do analizy własności bloków podstawieniowych (s-bloków) wykorzystywanych w szyfrach blokowych. Zaproponowana metoda może służyć jako dodatkowe kryterium do oceny kryptograficznych własności s-bloków. W niektórych przypadkach możliwe jest wykrycie, że s-blok został zaprojektowany z wykorzystaniem prostego przekształcenia matematycznego. (**Pewne podejście do ewaluacji s-bloków**).*

Keywords: Block cipher, S-box.
Słowa kluczowe: Szyfr blokowy, S-box

Introduction

The Substitution Block (S-box) is the fundamental cryptographic component which plays an important role in fulfilling the Shannon's property of confusion in block ciphers. In the two major design strategies for block ciphers, Feistel networks and Substitution/Permutation networks, the S-boxes form the only non-linear part of a block cipher. The strength of the cipher to a great extent depends on the quality of the used s-box (s-boxes).

An S-box takes some number of input bits, *m*, and transforms them into some number of output bits, *n*, where *n* is not necessarily equal to *m*. An *m*×*n* S-box can be implemented as a lookup table with 2^m words of *n* bits each.

The problem to find optimal S-boxes is hard due to the fact that the number of permutations mapping *m*-bits to *n*-bits is very big even for small values of *m.* Therefore exhaustively checking all permutations to find good S-boxes is not practical for *m* > 4 [1].

In practice S-boxes are either designed or generated randomly. In each case it is important to evaluate the S-box quality.

Previous work

The most important property of the S-box is robustness against known attacks (eg. differential and linear cryptanalysis) and attacks which may be invented in the future.

A number of properties have been proposed to measure cryptographic quality of S-boxes. For example: nonlinearity [2], correlation immunity [2], algebraic degree [3], resilience [4], robustness to differential cryptography [5], fixed points and opposite fixed points [6].

Over 20 parameters have been proposed so far and because of the problem complexity it is still possible to develop new measures. For example the open source tool for S-boxes analysis "S-box, SET, Match" [7] computes 17 parameters.

In some cases the design criteria and the process of the S-box crafting remain undisclosed. In such case we may try some sort of reverse engineering [8].

Motivation

Let us consider a substitution/permutation network shown in Fig.1. It is a 64-bit subblock of the PP-1 cipher [9].

In each round a 64-bit subblock is processed as eight 8-bit subblocks by four types of transformations, 8 × 8 S-boxes *S*, XOR [⊕], addition [+] and subtraction [-]. Two round keys k' and k" are applied in each round.

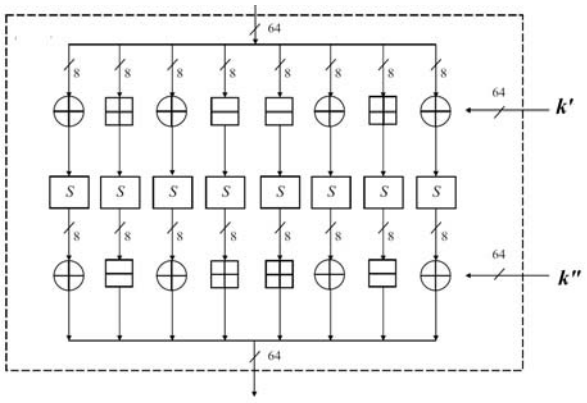

Fig.1. Considered encryption network

It is well known that one round block cipher can be easily broken (e.g., only 2 pairs of plaintext-ciphertext are required to break one round AES [10]). Let us assume that by inserting fault (faults) we have changed the circuit functionality in such a way that it produces a result which corresponds to a one round version of the cipher with the same key.

To evaluate to what extent this can be useful for breaking the cipher we need to find answers to two questions: "what is the probability of obtaining such a result?", and "how many results are required to break the cipher?"

Let us consider in detail the second problem. As we can see in Fig. 1 , there are 2 round keys applied in each round. Therefore we can expect that breaking the one round version of the cipher, using pairs plaintext-ciphertext, will be more complicated than in a cipher with one round key in each round (e.g., the AES).

There are 3 sort of operations in the circuit: xor-substitution-xor, addition-substitution-subtraction, and subtraction- substitution- addition. Where addition and subtraction are performed modulo 2^8.

Let us assume that we have two different pairs plaintext ciphertext (m_1,c_1) (m_2,c_2) for round 1. For xor-S-xor section we have a system of equations:

(1) $$c_1 = S(m_1 \oplus k_1) \oplus k_2$$
$$c_2 = S(m_2 \oplus k_1) \oplus k_2$$

Similarly, for plus-S-minus section we have:

(2) $$c_1 = S(m_1 + k_1) - k_2$$
$$c_2 = S(m_2 + k_1) - k_2$$

For minus-S-plus we obtain

(3) $$c_1 = S(m_1 - k_1) + k_2$$

$$c_2 = S(m_2 - k_1) + k_2$$

We know m_1, c_1, m_2, c_2 and we want to find k_1, k_2. The problem is: how many solutions exist for systems of equations (1), (2), and (3)?

The answer is not obvious because of the nonlinear element S presented in Fig. 2 [9].

```
9E BC C3 82 A2 7E 41 5A 51 36 3F AC E3 68 2D 2A
EB 9B 1B 35 DC 1E 56 A5 B2 74 34 12 D5 64 15 DD
B6 4B 8E FB CE E9 D9 A1 6E DB 0F 2C 2B 0E 91 F1
59 D7 3A F4 1A 13 09 50 A9 63 32 F5 C9 CC AD 0A
5B 06 E6 F7 47 BF BE 44 67 7B B7 21 AF 53 93 FF
37 08 AE 4D C4 D1 16 A4 D6 30 07 40 8B 9D BB 8C
EF 81 A8 39 1D D4 7A 48 0D E2 CA B0 C7 DE 28 DA
97 D2 F2 84 19 B3 B9 87 A7 E4 66 49 95 99 05 A3
EE 61 03 C2 73 F3 B8 77 E0 F8 9C 5C 5F BA 22 FA
F0 2E FE 4E 98 7C D3 70 94 7D EA 11 8A 5D 00 EC
D8 27 04 7F 57 17 E5 78 62 38 AB AA 0B 3E 52 4C
6B CB 18 75 C0 FD 20 4A 86 76 8D 5E 01 ED 46 45
B4 FC 83 02 54 D0 DF 6C CD 3C 6A B1 3D C8 24 E8
C5 55 71 96 65 1C 58 31 A0 26 6F 29 14 1F 6D C6
88 F9 69 0C 79 A6 42 F6 CF 25 9A 10 9F BD 80 60
90 2F 72 85 33 3B E7 43 89 E1 8F 23 C1 B5 92 4F
```

Fig. 2. S-box *S*

Solutions can be found by checking all possible m_1, m_2, c_1, c_2 such that the system of equations is satisfied.

It turns out that for xor-S-xor and *given* m_1, m_2, c_1, c_2 there are 3 possibilities: no solution or 2 solutions or 4 solutions. Examples are presented in Table 1.

Table 1. Examples of solutions of equations system (1)

m_1	c_1	m_2	c_2	Number of solutions	Examples of solutions k_1 and k_2 value
72	cf	cb	09	0	-
36	c2	71	44	2	a0 11 e7 97
4c	db	7c	b0	4	4d 67 7d 0c d9 a7 e9 cc

For plus-S-minus and minus-S-plus we have no solutions or 1 or 2 or 3 or 4 or 5 o 6 or 7 solutions. Examples of solutions are presented in Table 2.

Table 2. Examples of solutions of equations system (2)

m_1	c_1	m_2	c_2	Number of solutions	Examples of solutions k_1 and k_2 value
1a	e0	8d	75	0	-
37	76	6c	c8	1	2f fc
37	76	77	80	2	9d 11 9e 5a
3c	f0	8c	a2	3	42 eb 89 20 c8 4e
4b	02	5a	79	4	10 c2 98 f6 b1 41 fc be
53	5c	72	48	5	35 7c 3e 2e 40 0e a0 d7 b4 02
0b	2b	4e	8a	6	38 34 5a 57 96 04 a4 df ab 0b af 9e
10	89	ea	3d	7	02 6e 0b 77 3a d2 42 db 81 5b cb 60 fb dd

Frequency distribution of solutions for xor-xor, plus-minus, and minus-plus operations are presented in tables 3, 4 and 5 respectively.

Table 3. Frequency distribution of solutions of equations (1)

Number of solutions	Absolute frequency	Relative frequency
0	1085276160	50,7353%
1		0,0000%
2	1038090240	48,5294%
3		0,0000%
4	15728640	0,7353%

Table 4. Frequency distribution of solutions of equations (2)

Number of solutions	Absolute frequency	Relative frequency
0	781523456	36,5352%
1	785004032	36,6980%
2	409561856	19,1465%
3	125864960	5,8840%
4	29961216	1,4006%
5	5614592	0,2625%
6	1499904	0,0701%
7	65024	0,0030%

Table 5. Frequency distribution of solutions of equations (3)

Number of solutions	Absolute frequency	Relative frequency
0	788168704	36,8459%
1	784203776	36,6605%
2	392986624	18,3716%
3	135462912	6,3327%
4	30998528	1,4491%
5	5505024	0,2574%
6	1376256	0,0643%
7	393216	0,0184%

So far we have considered solutions for each 8-bit section separately. Now we can evaluate the number of solutions which must be checked to break the cipher. As we can see in Fig.1 there are 4 xor-S-xor sections, 2 plus-S-minus sections, and 2 minus-S-plus sections.

Let us suppose that we have found such round keys (concatenated k_1 ad k_2 for each section) that equations (1), (2), and (3) are satisfied. For each xor-S-xor 2 or 4 solutions are possible. For plus-S-minus and minus-S-plus sections 1-7 solutions are possible. The smallest number of k_1 and k_2 combinations which must be checked is:

(4) $$L\min = 2^4$$

The biggest number of k_1 and k_2 combinations which must be checked is:

(5) $$L\max = 4^4 \times 7^4$$

Weighted average is:

(6) $$Laverage = 105.68$$

Frequency distribution of equations (1) to (3) solutions depends on the substitution S. We will address this problem in the next section.

Table 6. Frequency distribution of solutions for xor-S-xor

Number of solutions	AES	Skipjack	Pi
0	1077903360	1281359872	1244528640
1	0	0	0
2	1052835840	673677312	735772672
3	0	0	0
4	8355840	159088640	143425536
5	0	0	0
6	0	22478848	14548992
7	0	0	0
8	0	2260992	819200
9	0	0	0
10	0	163840	0
11	0	0	0
12	0	65536	0

Table 7. Frequency distribution of solutions for plus-S-minus

Number of solutions	AES	Skipjack	Pi
0	786826752	791054336	783888128
1	779616768	778941440	790350848
2	406228224	398291456	394428928
3	127938048	129255424	131347968
4	30849024	33688064	31411968
5	6136320	6199296	6199296
6	1369344	1599488	1142272
7	130560	65536	260608
8			65024

Solutions for different S-boxes

It turns out that the number of solutions which satisfy equations (1) to (3) depends on the substitution S. The same holds for frequency distribution. Frequency distributions for equations (2) and (3) are very similar. Therefore we will focus on equations (1) and(2).

Frequency distribution of solutions for xor-S-xor and plus-S-minus for AES [11], Skipjack [12], and Pi [13] S-boxes are presented in tables 6 and 7.

As we can see, the frequency distribution of solutions of equations (1) and (2) differs significantly for each of the considered s-boxes.

Linear transformation of S-box (e.g. XORing with a constant) does not change the frequency distributions of considered systems of equations.

```
CD EF 90 D1 F1 2D 12 09 02 65 6C FF B0 3B 7E 79
B8 C8 48 66 8F 4D 05 F6 E1 27 67 41 86 37 46 8E
E5 18 DD A8 9D BA 8A F2 3D 88 5C 7F 78 5D C2 A2
0A 84 69 A7 49 40 5A 03 FA 30 61 A6 9A 9F FE 59
08 55 B5 A4 14 EC ED 17 34 28 E4 72 FC 00 C0 AC
64 5B FD 1E 97 82 45 F7 85 63 54 13 D8 CE E8 DF
BC D2 FB 6A 4E 87 29 1B 5E B1 99 E3 94 8D 7B 89
C4 81 A1 D7 4A E0 EA D4 F4 B7 35 1A C6 CA 56 F0
BD 32 50 91 20 A0 EB 24 B3 AB CF 0F 0C E9 71 A9
A3 7D AD 1D CB 2F 80 23 C7 2E B9 42 D9 0E 53 BF
8B 74 57 2C 04 44 B6 2B 31 6B F8 F9 58 6D 01 1F
38 98 4B 26 93 AE 73 19 D5 25 DE 0D 52 BE 15 16
E7 AF D0 51 07 83 8C 3F 9E 6F 39 E2 6E 9B 77 BB
96 06 22 C5 36 4F 0B 62 F3 75 3C 7A 47 4C 3E 95
DB AA 3A 5F 2A F5 11 A5 9C 76 C9 43 CC EE D3 33
C3 7C 21 D6 60 68 B4 10 DA B2 DC 70 92 E6 C1 1C
```

Fig. 3. Modified S-box *S*

For example the S-box shown in Fig.3. was obtained by xoring each element of S-box S (Fig. 2) with 0x53. Frequency distribution of solutions for modified S-box is the same as for the original S-box (presented in tables 3, 4, and 5).

An approach to S-box evaluation

Let us consider a more general version of encryption network - Fig. 4. We take into account only one section with a single S-box. The whole network consists of many sections - similarly to those shown in Fig. 1. Operations *OP1* and *OP2* are two-argument operations such as xor, addition modulo 2^n, subtraction modulo 2^n, etc. Any reversible two-argument operation may be used. *S* is a *n*×*n* substitution box (S-box); k_1 and k_2 are n-bit fragments of round keys *k′* and *k″*.

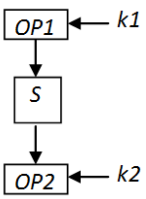

Fig.4. More general version of encryption network

In the next part we will limit ourselves to xor-xor operations. We will use 8×8 S-boxes as an example.

Let us take a set of randomly generated 8×8 S-boxes as a reference point. For a sample of 100 random s-boxes the observed number of solutions was 10 or 12 or 14. The frequency distribution of the number of solutions is presented in Fig. 5.

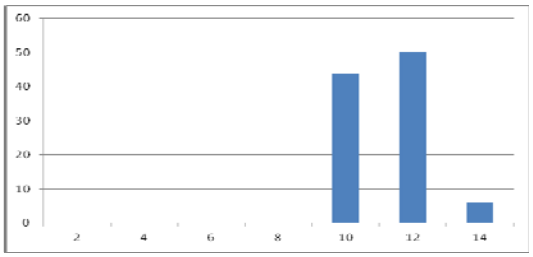

Fig.5. Frequency distribution of solutions for randomly generated S-boxes; xor-xor operations.

As we can see, in considered sample we don't observe less than 10 solutions and more than 14 solutions.

Frequency distribution of solutions for randomly generated S-boxes is presented in Table 8.

Table 8. Frequency distribution of solutions for randomly generated S-boxes.

No of solutions	Min	Max	Average
0	1289388032	1306361856	1298523095
1	0	0	0
2	635142144	660045824	646883901
3	0	0	0
4	157024256	166297600	162617754
5	0	0	0
6	24313856	29753344	27252817,9
7	0	0	0
8	2457600	4685824	3444572,16
9	0	0	0
10	131072	655360	348651,52
11	0	0	0
12	0	98304	21626,88
13	0	0	0
14	0	65536	2293,76

Let us compute for each number of solutions

$$d = \frac{Max - Min}{Average} \tag{7}$$

Results are presented in Fig. 6.

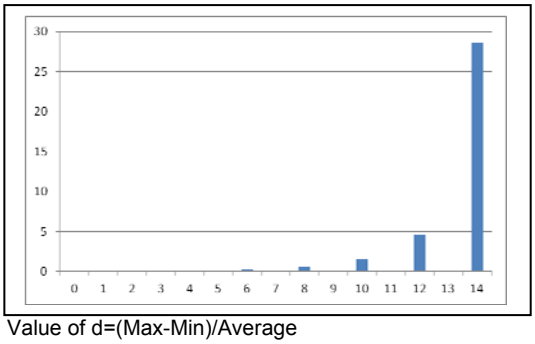

Fig.6. Value of d=(Max-Min)/Average

Let us now apply our approach to a few S-boxes. First let us consider a S-box presented in Fig. 7.

```
A4 EC D1 D7 D2 2E E4 58 67 66 98 65 07 76 DA 87
23 D9 C6 BE CD 2D 18 CA EE CC E9 85 88 F0 94 81
DE 4B A5 24 B6 C4 7B 21 83 4E 4C 54 0C BC 93 06
B8 AF 13 45 5C 4F 80 C3 75 84 A1 00 6C 7D 3D 5E
95 08 29 46 FE 41 FB 53 EF 22 49 89 1F 0B D5 6B
F3 F2 E2 33 0F 82 1E 14 BA C0 6D A9 99 8E 56 35
5F 3F 77 78 E8 ED D3 F8 B1 DF AA 36 F7 FA D0 9C
7C 47 32 6E 4D F5 8B 28 D4 16 52 31 8D 7A E0 03
2B AD 9B 19 B5 2C CE 2A 71 70 3E 55 E1 12 42 72
60 01 59 A6 92 09 CB AC DC E6 96 7F D8 1A 1D 6F
B3 34 9E 27 3C AB E5 DD A7 F9 A8 D6 CF C8 79 3B
25 4A 05 7E C5 44 C9 1C E3 8C B0 69 86 EA 6A 2F
BF 37 9D EB F4 F6 B2 0E DB B4 38 5B B7 0A 5A 20
57 5D 48 51 63 FD 43 9F 91 C2 C7 40 AE F1 97 0D
B9 A3 90 26 74 9A 64 04 61 17 8A 62 73 FF 50 A0
1B 11 02 39 3A 15 30 68 FC A2 8F BB C1 E7 10 BD
```

Fig. 7. The first considered S-box

To check the cryptographic properties of the S-box we may apply a set of tests [7]. Selected results are presented below:

```
S-box is balanced.
Nonlinearity is 94.
Algebraic degree is 7.
Algebraic immunity is 4.
Transparency order is 7.813.
Number of fixed points is 0.
Number of opposite fixed points is 0.
```

From these results w cannot deduce whether the S-box was generated randomly or constructed in some other way.

Now let us find the frequency distribution of solutions for xor-xor operations. For each number of solutions we compute:

$$e = \frac{N - Average}{Average} \tag{8}$$

where N is measured absolute frequency of solutions and *Average* is average absolute frequency of solutions for random S-boxes.

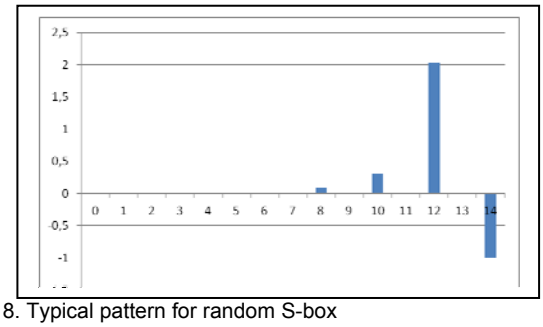

Fig. 8. Typical pattern for random S-box

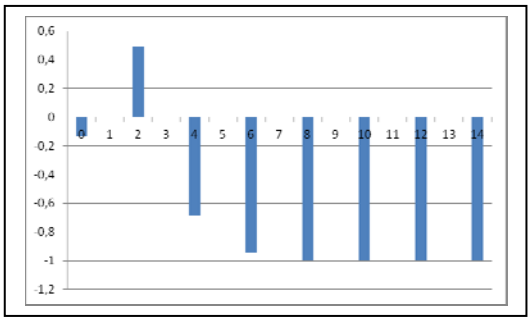

Fig.9. Non-random S-box (AES)

Results are presented in Fig. 8. As we can see, there are small values of e for small number of solutions and much greater for bigger number of solutions. It is a typical pattern for randomly generated S-boxes. Conclusion: with high probability the S-box is random. In fact this S-box was generated randomly.

Indeed the S-box in Fig. 10 is a modified AES S-box proposed by Fuller and Millan [14].

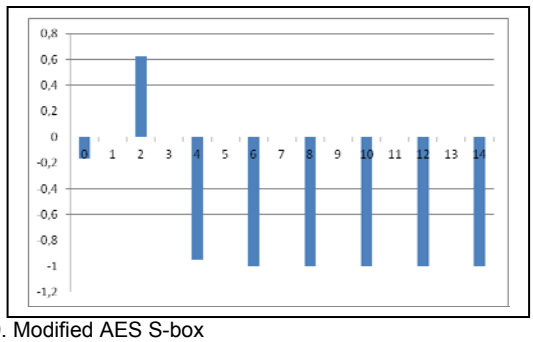

Fig. 10. Modified AES S-box

Finally, let us consider a S-box presented in Fig.11.

```
01 2D E2 93 BE 45 15 AE 78 03 87 A4 B8 38 CF 3F
08 67 09 94 EB 26 A8 6B BD 18 34 1B BB BF 72 F7
40 35 48 9C 51 2F 3B 55 E3 C0 9F D8 D3 F3 8D B1
FF A7 3E DC 86 77 D7 A6 11 FB F4 BA 92 91 64 83
F1 33 EF DA 2C B5 B2 2B 88 D1 99 CB 8C 84 1D 14
81 97 71 CA 5F A3 8B 57 3C 82 C4 52 5C 1C E8 A0
04 B4 85 4A F6 13 54 B6 DF 0C 1A 8E DE E0 39 FC
20 9B 24 4E A9 98 9E AB F2 60 D0 6C EA FA C7 D9
00 D4 1F 6E 43 BC EC 53 89 FE 7A 5D 49 C9 32 C2
F9 9A F8 6D 16 DB 59 96 44 E9 CD E6 46 42 8F 0A
C1 CC B9 65 B0 D2 C6 AC 1E 41 62 29 2E 0E 74 50
02 5A C3 25 7B 8A 2A 5B F0 06 0D 47 6F 70 9D 7E
10 CE 12 27 D5 4C 4F D6 79 30 68 36 75 7D E4 ED
80 6A 90 37 A2 5E 76 AA C5 7F 3D AF A5 E5 19 61
FD 4D 7C B7 0B EE AD 4B 22 F5 E7 73 23 21 C8 05
E1 66 DD B3 58 69 63 56 0F A1 31 95 17 07 3A 28
```

Fig.11. The last considered S-box.

To start with, let us analyze this S-box using S-box, SET, Match tool [7]. We obtain the following results:

```
Nonlinearity is 82.
Algebraic degree is 7.
Number of fixed points is 3.
Composite algebraic immunity is 4.
Robustness to differential cryptanalysis
is 0.500.
```

However is not obvious whether the S-box is generated randomly or obtained in some other way. Let us find the frequency distribution of solutions for xor-xor operations for this S-box. Results are presented in Table 9.

Table 9. Frequency distribution of solutions for the last S-box

Number of solutions	Absolute frequency
0	1403355136
2	470220800
4	221511680
6	27066368
8	14385152
10	1671168
12	458752
14	65536
16	163840
18	32768
22	65536
32	32768
64	32768
128	32768

As we can see in Table 9 the distribution is very unusual. For some *m* and *c* values we can find as much as 128 solutions for k_1 and k_2. With high probability the S-box is based on some simple mathematic formula.

Indeed this is one of two Safer-64 [15] S-boxes denoted as $\log_{45}$ S-box. It is computed as follows [15]:

```
logtab[1]:= 0; exptab[0]:= 1;
FOR i:= 1 TO 255 DO
BEGIN
exptab[i]:= (45 * exptab[i - 1]) mod 257;
logtab[exptab[i]]:= i;
END;
exptab[128]:=    0;    logtab[0]:=    128;
exptab[0]:= 1;
```

The *logtab* table is our S-box.

Concluding remarks

The proposed method may be used to get some insight in the S-box which design criteria are unknown. It may also be used to reason about the S-box properties. S-boxes with similar frequency distribution of solutions may have similar cryptographic properties.

Finding all the solutions is time consuming. For example for 8×8 S-box solving all the xor-xor equations requires 350 seconds of computation on 4-core Xeon 3.4 GHz processor. Approximately 310 seconds takes finding solutions for plus-minus operations.

In many cases we can consider some limited number of randomly selected data. From tables 6, 7 and 9 we can deduce that even for only 10^6 solutions we can still obtain valuable results. In such a case computations require a few seconds.

The randomly selected data is the only practical approach for bigger e.g. 16×16 S-boxes.

***Acknowledgment**: This work is partially supported by research project 04/45/DSPB/0136.*

***Author**: dr inż. Krzysztof Bucholc, Politechnika Poznańska, Instytut Automatyki i Inżynierii Informatycznej ul. Piotrowo 3a, 60-965 Poznań, E-mail: Krzysztof.Bucholc@put.poznan.pl*

REFERENCES

[1] Saarinen, M.-J.O.: Cryptographic Analysis of All 4 x 4-Bit S-Boxes. In: Miri, A.,Vaudenay, S. (eds.) SAC 2011. LNCS, vol. 7118 (2012),. Springer, Heidelberg pp. 118–133

[2] Braeken, A.: Cryptographic Properties of Boolean Functions and S-Boxes. PhD thesis. Katholieke Universiteit Leuven (2006)

[3] Knudsen, L.R., Robshaw, M.: The Block Cipher Companion. Information Security and Cryptography. Springer (2011)

[4] Burnett, L.D.: Heuristic Optimization of Boolean Functions and Substitution Boxes for Cryptography. PhD thesis. Queensland University of Technology (2005)

[5] Biham, E., Shamir, A.: Differential Cryptanalysis of DES-like Cryptosystems. In: Menezes, A., Vanstone, S.A. (eds.) Advances in Cryptology - CRYPTO 1990. LNCS, vol. 537, Springer, Heidelberg (1991) pp. 2–21.

[6] Daemen, J., Rijmen, V.: The Design of Rijndael. Springer-Verlag New York, Inc.,Secaucus (2002)

[7] Picek, S., Batina, L., Jakobović, D., Ege, B., Golub, M, S-box, SET, Match: A Toolbox for S-box Analysis Volume of the series LNCS Vol. 8501, (2014), pp 140-149.

[8] Biryukov, A., Perrin L., On Reverse-Engineering S-Boxes with Hidden Design Criteria or Structure, Advances in Cryptology -- CRYPTO 2015, Springer-Verlag, (2015), pp. 116-140.

[9] Bucholc K., Chmiel K., Grocholewska-Czuryło A., Idzikowska E., Janicka-Lipska I., Stokłosa J., Scalable PP-1 block cipher. International Journal of Applied Mathematics and Computer Science, vol. 20, No. 2, (2010), 401−411.

[10] Piret G. and Quisquater J.-J., A differential fault attack technique against SPN structures, with application to the AES and Khazad, Proc. of CHES 2003, pp. 77-88.

[11] Daemen, J., Rijmen, V.: The Design of Rijndael. Springer-Verlag New York, Inc.,Secaucus (2002).

[12] SKIPJACK and KEA Algorithm Specifications (1998),http://csrc.nist.gov/groups/ST/toolkit/documents/skipjack/skipjack.pdf

[13] Federal Agency on Technical Regulation and Metrology (GOST). Block ciphers, 2015. http://www.tc26.ru/en/standard/draft/ENG_GOST_R_bsh.pdf.

[14] Fuller J.,Millan W.,Linear Redundancy in S-Boxes, LNCS vol. 2887, (2003), pp.74-86.

[15] Massey J.L..SAFER K-64: A byte-oriented block-ciphering algorithm, LNCS Vol. 809, (2005), pp. 1-17.