Lightweight Block Cipher Algorithms: Review Paper

Sufyan Salim Mahmood Aldabbagh[1], Imad Fakhri Taha Al Shaikhli[2], Alyaa Ghanim Sulaiman[3]

[1,2]Dept. of computer science, [3]Dept. of software engineering,[1,3]University of Mosul, [1,3]Iraq
[2]IIUM, [2]Malaysia

ABSTRACT

Lightweight block cipher algorithms are very important for constraint environment. This paper analyses most of the known lightweight block cipher algorithms such as, HISEC, OLBCA, PRINCE, TWINE, KLEIN, LED, LBLOCK, PRINT, PRESENT through four factors (Algorithm specifications, S-boxes, Cost, Cryptanalysis). All the results have been presented.

Index terms: Lightweight block cipher algorithm, Cryptography, Symmetric key, constraint environment security.

I. INTRODUCTION

No doubt that the life is changing tremendously, especially in information technology and the needs of security system to protect data is becoming crucial [1]. Generally, it is difficult to suggest a cryptographic algorithm that can suit all types of target devices. However, it is not suitable to use common cryptographic algorithms in specific devices with extremely constrained resources [2].

The fundamental principles and trends to design algorithms proposed for implementation in devices with extremely low resources are to some extent different from the design aspect of commonly used cryptographic algorithms. In this specific field is supported by a branch of the modern cryptography lightweight cryptography [2].

Many lightweight block cipher algorithms are proposed [3] like HISEC[4], OLBCA[5], PRINCE [6], PRINT [7], PRESENT [8], KLEIN [9], Lblock [10], TWINE [11] and LED [12].

This paper analyses most of the known lightweight block cipher algorithms through following four factors

- First factor: Algorithm specifications.
- Second factor: S-boxes.
- Third factor: Cost.
- Fourth factor: Cryptanalysis.

II. FIRST FACTOR: ALGORITHM SPECIFICATIONS

This section describes lightweight block cipher algorithms from the side of algorithm specifications.

HISEC

Aldabbagh and his colleagues proposed this algorithm in 2014. This proposal was focus on the security factor which is one of the important factors in designing lightweight algorithms. Also, there are many important applications which need a high security lightweight block cipher algorithm such as a credit card, an E-passport, and so on. HISEC is 64-bit plaintext and 80-bit key size. There are 16 rounds and in each round there are operations like: Substitution box, Bit permutation, XOR, Rotation and key update. Moreover, there is XOR between the cipher text and key in the last round [4].

OLBCA

Aldabbagh and Alshaikhli proposed this algorithm in 2014. This proposal was optimized three factors (security, cost, and performance). OLBCA is 64-bit plaintext and 80-bit key size. There are 22 rounds in the proposed algorithm. In each round, there are 12 4-bit S-boxes, bit permutation, rotations, XOR, word permutation and key schedule [5].

TWINE

Tomoyasu Suzaki and his colleagues proposed this algorithm in 2013**.** This proposal is a 64-bit lightweight block cipher, supporting two key lengths, 80 and 128-bits. TWINE enables quite small hardware implementation similar to the previous proposals, yet it enables efficient implementation on embedded software. Moreover, it allows for compact implementation of unified encryption and decryption. This characteristic mainly originates from the use of a generalised Feistel with many sub blocks combined with recent improvement on the diffusion layer. In the event the key length is needed to be specified, by writing TWINE-80 or TWINE-128 it denotes the corresponding version. The global structure of TWINE is a variant of Type-2 GFS with 16 4-bit sub-blocks [11].

PRINCE

Julia Borgho and his colleagues proposed this algorithm in 2012. Their proposal introduced a block cipher that is optimised with respect to latency when implemented in hardware. PRINCE is a 64-bit block cipher with a 128-bit key. The key is split into two parts of 64 bits each and extended to 192 bits by the mapping. It is based on the so-called FX construction. The first two sub keys k0 and k0' are used as whitening keys while the key k1 is the 64-bit key used for 12-round block cipher, referred to as PRINCE $_{\text{core}}$.

Each round of the PRINCE $_{\text{core}}$ consists of a key addition, an S-box-layer, a linear layer and the addition of a round constant . This algorithm uses one 4-bit S-box and is repeated 16 times [6].

KLEIN

Zheng Gong and his colleagues proposed this algorithm in 2012. It was designed to work with constrained devices such as wireless sensors and RFID tags. Compared to the related proposals, KLEIN has an advantage in relation to software performance and also its hardware implementation is compact.

KLEIN is a family of block ciphers, with a 64-bit block size and a variable key length - 64, 80 or 96-bits. According to the different key length, it is denoted as KLEIN-64/80/96, respectively. It is well-known that the key length and the block size are two important factors for a block cipher in the trade-off between security and performance. The structure of the KLEIN algorithm is a SPN as shown in Figure 4.6. This structure has been used by many block ciphers like the AES and PRESENT. The KLEIN has three sets for its number of rounds which is dependent on the key size. For example, KLEIN-64 has 12 rounds while KLEIN-80 has 16 rounds and KLEIN-96 has 20 rounds [9].

LED

Guo and his colleagues proposed this algorithm in 2011. The LED is a 64-bit block cipher and the key size ranges from 64 bits up to 128 bits. The block cipher comprising 64-bits is arranged into 16 four-bit nibbles m0||m1||….||m14||m15. The key is viewed nibble-wise and loaded nibble-by-nibble into one or two arrays (K1and K2) depending on the key length. This cipher has 8 steps or 6 steps depending on the size of key as shown in Figure 4.7. Each step has four rounds and each round has four operations AddConstants, SubCells, ShiftRows, and MixColumnsSerial [12].

LBLOCK

Wu and his colleagues proposed this algorithm in 2011. This algorithm is like many other lightweight block ciphers and the structure of the LBlock is a Feistel network and the authors considered security and efficiency when they designed this algorithm. The LBlock has a 64-bit plaintext and the key size is 80-bits. It employs a variant Feistel structure and consists of 32 rounds. The block cipher 64-bit is divided into two parts, a left side 32-bit and right side 32-bit. In the left side, there is a copy of 32-bit to make the right hand side. Another copy of 32-bit is XOR'ed with a 32-bit key and then as input to 8 S-boxes 4-bit. While in the right side, there is one operation called rotate left 8-bit for 32-bits and then XOR'ed with the output of 8 S-boxes in the left hand side to make the left hand side. The same operation is repeated for 32 rounds [10].

PRINT

Lars Knudsen and his colleagues proposed this algorithm in 2010. The authors considered the cryptographic implications of integrated circuit (IC) printing. They presented two block ciphers PRINTcipher-48 and PRINTcipher-96. These algorithms are designed to exploit the properties of IC-printing technology.

The PRINT cipher is a block cipher with b-bit blocks, b ∈ {48, 96}, and an effective key length of 5 /3 × b bits. The essential structure of the PRINT cipher is SP-network with r = b rounds as shown in Figure 4.10. It follows that the PRINTcipher-48 comprises 48-bit plaintext and 80-bit key size while it has 48 rounds. The version PRINTcipher-96 has 96-bit plaintext and 160-bit key size while it has 96 rounds [7] .

PRESENT

A. Bogdanov and his colleagues proposed this algorithm in 2007. They proposed an ultra-lightweight block cipher to offer a level of security. Simplicity was the goal when they designed PRESENT in addition to the requirement for security and an efficient implementation. PRESENT is a SPN and has 31 rounds. The plaintext is 64 bits and there are two key sizes; 80 and 128 bits. The authors recommended the version with 80-bit keys as it provides more than adequate security for low-security applications. Each of the 31 rounds has an XOR operation of the plaintext with the sub key Ki for $1 \leq i \leq 32$, where K32 is used for post-whitening, substitution layer and permutation layer. The non-linear layer uses one 4-bit S-box and is repeated 16 times in parallel in each round [8].

III. SECOND FACTOR: S-BOXES

This section presents the number of S-boxes, the values of S-boxes for each algorithm.

HISEC

The OLBCA S-box is a 4-bit as defined in Table-1.

Table 1 HISEC S-box [4]

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(*x*)	3	8	F	1	A	6	5	B	E	D	4	2	7	0	9	**C**

OLBCA

The OLBCA S-box used HISEC S-box.

TWIN Algorithm

The TWIN S-box is a 4-bit as defined in Table-2.

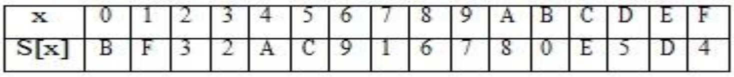
Table-2 TWIN S-box [11]

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(*x*)	C	0	F	A	2	B	9	5	8	3	D	7	1	E	6	4

PRINCE Algorithm

The PRINCE algorithm uses a one 4-bit S-box and it is repeated 16 times. The action of the S-box in hexadecimal notation is given by Table-3.

Table-3 PRINCE S-box [6]



x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S[x]	B	F	3	2	A	C	9	1	6	7	8	0	E	5	D	4

KLEIN Algorithm

This algorithm uses a one 4-bit S-box and it is repeated 16 times. The KLEIN S-box is an involutive S-box as described in Table -4. By choosing an involutive S-box, it can save the implementation costs for its inverse. The KLEIN S-box fulfils the following conditions:

- The S-box satisfies $S(S(x)) = x, x \in F^4 2$, thus it can be used both in the encryption and in the decryption.
- The S-box has no fixed points, i.e. $S(x) = x, x \in F^4 2$.
- For any non-zero input difference $\Delta_I \in F^4 2$ and output difference $\Delta O \in F2$, it holds that $\#\{x \in F^4 2|S(x) +S(x+\Delta I) =\Delta O\} \leq 4$. Furthermore, if $wt\ (\Delta_I) = wt\ (\Delta_O) = 1$, we have $\#\{x \in F^4 2|S(x) + S(x+\Delta I) = \Delta O\} \leq 2$.

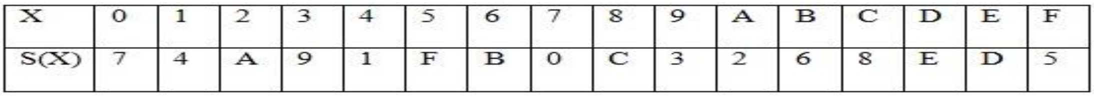
- For any non-zero *a*, *b* ∈F^{4}2,it holds that Furthermore, if *w t*(*a*) = *wt*(*b*) = 1, we have
- $$|S_b^{\mathcal{W}}(a)| = |\sum_{x\in\mathbb{F}_2^4}(-1)^{b\cdot S(x)+a\cdot x}| \leq 8. \qquad 1$$

$$|S_b^{\mathcal{W}}(a)| = |\sum_{x\in\mathbb{F}_2^4}(-1)^{b\cdot S(x)+a\cdot x}| \leq 4 \qquad 2$$

The 4-bit S-box used in PRESENT satisfies ♯{x ∈ F^{4}2|S(x) +S(x+ΔI) =ΔO} =0 if *wt* (Δ_I) = *wt* (Δ_O) = 1, which assures a better avalanche effect but the PRESENT

S-box is not an involution. According to their exhaustive search result, there is no such an involutive 4-bit S-box that can satisfy this additional property [9].

Table-4 KLEIN S-box [9]



X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(X)	7	4	A	9	1	F	B	0	C	3	2	6	8	E	D	5

The LED Algorithm

This algorithm uses the PRESENT S-box.

LBlock Algorithm

This algorithm used 10 S-boxes as shown in Table-5. There are 8 S-boxes out of 10 S-boxes used for encryption and decryption algorithm while 2 S-boxes only are used for key scheduling.

Table-5 Contents of the S-boxes used in Lblock [10]

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S0(x)	E	9	F	0	D	4	A	B	1	2	8	3	7	6	C	5
S1(x)	4	B	E	9	F	D	0	A	7	C	5	6	2	8	1	3
S2(x)	1	E	7	C	F	D	0	6	B	5	9	3	2	4	8	A
S3(x)	7	6	8	B	0	F	3	E	9	A	C	D	5	2	4	1
S4(x)	E	5	F	0	7	2	C	D	1	8	4	9	B	A	6	3
S5(x)	2	D	B	C	F	D	0	9	7	A	6	3	1	8	4	5
S6(x)	B	9	4	E	0	F	A	D	6	C	5	7	3	8	1	2
S7(x)	D	A	F	0	E	4	9	B	2	1	8	3	7	5	C	6
S8(x)	8	7	E	5	F	D	0	6	B	C	9	A	2	4	1	3
S9(x)	B	5	F	0	7	2	9	D	4	8	1	C	E	A	3	6

PRINT Algorithm

The PRINT algorithm involves a single 3-bit S-box which is applied b/3 times in parallel. The values of the S-box are given by the Table-6.

Table-6 PRINT S-box [7]

X	0	1	2	3	4	5	6	7
S(X)	0	1	3	6	7	4	5	2

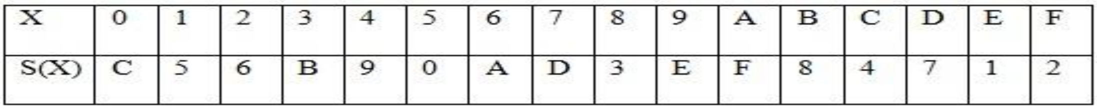
PRESENT Algorithm

The PRESENT algorithm uses an S-box presented as a 4-bit to 4-bit S-box S: F2^4→ F2^4. The values of the PRESENT S-box are given in Table-7 . More precisely, the S-box for PRESENT fulfils the following conditions, whereby the Fourier coefficient of S is denoted by:

- For any fixed non-zero input difference $\Delta_I \in F_2^n$and any fixed non-zero output difference $\Delta o \in F_2^m$ there is: $\#\{S(x) + S(x + \Delta_I) = \Delta o\} \leq 4$;

- For any fixed non-zero input difference $\Delta_I \in F_2^n$and any fixed non-zero output difference $\Delta o \in F_2^m$ such that wt (Δ_I) =wt (Δo) =1 there is:
  $\#\{S(x) + S(x + \Delta_I) = \Delta o\} = 0$;
- For all non-zero a $\in F_2^n$ and all non-zero b $\in F_4$ it holds that $|S_b^W(a)| \leq 8$;
- For all non-zero a $\in F_2^n$ and all non-zero b $\in F_4$ such that wt (a) =wt (b) =1 it holds that $S_b^W(a) \pm 4$ [8].

Table-7 PRESENT S-box [8]



X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(X)	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

IV. THIRD FACTOR: COST OF EXISTING LIGHTWEIGHT ALGORITHMS

The costs of algorithms are listed in Table-8 below.

Table-8 The cost of existing lightweight algorithms

No.	Algorithm	Key Size	Area (GE)	Ref
1.	HISEC	80	1694	[4]
2.	OLBCA	80	1521	[5]
3.	PRINCE	80	3491	[6]
4.	TWINE	80	1503	[11]
		128	1866	
5.	KLEIN	KLEIN-64	1981	[9]
		KLEIN-80	2097	
		KLEIN-96	2213	
6.	LED	LED-64	966	[12]
		LED-80	1,040	
		LED-96	1,116	
		LED-128	1,265	
7.	Lblock	80	1320	[10]
8.	PRINT	PRINTcipher-48	402	[7]
		PRINTcipher-96	726	
9.	PRESENT	80	1570	[8]
		128	1884	

IV. FOURTH FACTOR: CRYPTANALYSIS

There are many attacks applied on lightweight block cipher algorithms such as: linear cryptanalysis[24], differential cryptanalysis [13,14,15], integral cryptanalysis [16], boomerang attack[19] and so on. In each attack, there is a way in which to check the security of the algorithm against the attack. Accordingly, this section presents the attacks on each lightweight block cipher algorithm. The complexity of differential cryptanalysis (DC) and linear cryptanalysis (LC) is completely determined by the number of active S-boxes involved and their characteristic/linear approximation probabilities.

HISEC

For differential cryptanalysis, the number of active S-box as shown in table-8.

Table-8 The number of active S-boxes for HISEC[4]

Round No.	4	8	12	16
Min number of active S-box	17	43	96	124

Regarding to integral attack, the round four is the maximum round no. can reach this attack for HISEC algorithm with complexity 2^{55}.The boomerang attack can reach round five with probability 2^{-48} .

OLBCA

For differential cryptanalysis, the number of active S-box as shown in table-9.

Table-9 The number of active S-boxes for OLBCA[5]

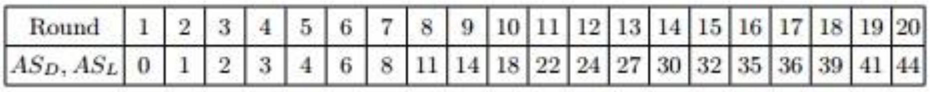
Round No.	**1**	**2**	**3**	**4**	**5**	**10**	**15**	**20**
Min number of active S-box	0	2	3	7	12	29	48	82

Regarding to integral attack, the round eight is the maximum round no. can reach this attack for OLBCA algorithm with complexity 2^{70}. The boomerang attack can reach round seven with probability 2^{-56} .

TWINE Algorithm

In examining the TWINE algorithm, the S-box has 2^{-2} maximum differential and linear probabilities, and the maximum differential and linear characteristic probabilities are both 2^{-64} for 15 rounds as shown in Table-10 below.

Table-10 Active S-box of linear and differential cryptanalysis for TWINE [11]



Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
AS_D, AS_L	0	1	2	3	4	6	8	11	14	18	22	24	27	30	32	35	36	39	41	44

Regarding an integral attack, Table-11 below shows the complexity of this attack on TWINE-80 and TWINE-128.

Table-11 Saturation attack on TWINE-80 and TWINE-128 [11]

No.	**Algorithm**	**Round**	**Data**	**Time**	**Memory**
1.	TWINE-80	22	2^{62}	$2^{68.43}$	2^{67}
2.	TWINE-128	23	$2^{62.81}$	$2^{106.14}$	2^{103}

PRINCE Algorithm

For linear and differential cryptanalysis, the PRINCE algorithm has the following theorem: Any differential characteristic and any linear-trail over 4 consecutive rounds of PRINCE comprise at least 16 active S-boxes. Regarding an integral attack, no one has applied an integral attack on this algorithm [6].

KLEIN Algorithm

The KLEIN algorithm counts the number of active S-boxes for linear and differential cryptanalysis and depends on the following two theorems: For linear and differential cryptanalysis, any four-round has 15 active S-boxes. Regarding an integral attack, the authors have claimed that any integral attack on KLEIN over seven rounds will be more complicated than exhaustive key searches [9].

LED Algorithm

There are two settings of keys in the LED algorithm (single key and related key). For a single key, for every step (4rounds) there are 25 active S-boxes. Accordingly, there are 200 active S-boxes within the 64-bit single key and 300 active S-boxes within a 128-bit single key. With respect to the related key, the number of active S-boxes is different. For two steps (8 rounds), there are 25 active S-boxes. Furthermore, there are 100 active S-boxes within a 64-bit related key and 150 active S-boxes within a 128-bit related key. With respect to an integral attack, the authors claimed that two big LED steps avoid any such observation. Also, they stated that the large number of rounds of LED will make an integral attack very unlikely to be a threat [12].

Lblock Algorithm

In the Lblock algorithm, there is no theorem for counting the active S-box. Accordingly, the number of active S-boxes for linear and differential cryptanalysis is shown in Table-12.

Table-12 Active S-box of Linear and differential for Lblock [10]

Rounds	1	2	3	4	5	6	7	8	9	10
DS	0	1	2	3	4	6	8	11	14	18
LS	0	1	2	3	5	6	8	11	14	18
Rounds	11	12	13	14	15	16	17	18	19	20
DS	22	24	27	30	32	35	36	39	41	44
LS	22	24	27	30	32	35	36	39	41	44

As mentioned previously, DS means differential cryptanalysis and LS means linear cryptanalysis. With respect to an integral attack, Table-13 shows the complexity of this attack on an Lblock algorithm.

Table-13 Integral attack on an Lblock Algorithm

No.	Algorithm	Round	Data	Time	Memory
1.	First paper [20]	20	$2^{63.6}$	$2^{39.6}$	2^{35}
2.	Second paper [21]	22	2^{61}	2^{70}	2^{63}

PRINT Algorithm

The number of active S-boxes is equal to one for each round for both linear and differential cryptanalysis [7]. From integral attack perspective, no one has applied an integral attack on this algorithm.

PRESENT Algorithm

Any five-round differential characteristic of PRESENT has a minimum of 10 active S-boxes. The case of the linear cryptanalysis of PRESENT is handled by the following theorem where the analysis of the best linear approximation to four rounds of PRESENT is made:

Let 4R be the maximal bias of a linear approximation of four rounds of PRESENT. Then $4R = 2^{-7}$. So, the maximal bias of a 28-round linear approximation by $2^6 \times 4R = 2^6 \times (2^{-7})7 = 2^{-43}$.

Regarding an integral attack, the design of PRESENT is almost exclusively bitwise and this attack does not work with bit permutation [8]. Z’aba and his colleagues found that the integral attack can reach round six for key size 80bits [22] while Shengbao and his colleagues found that the integral attack can reach round nine for key size 80 bits [23].

CONCLUSION

In this paper, an analysis has been made of the majority of lightweight block cipher algorithms in terms of their cost and security. An intensive analysis has been performed in this paper which has provided a detailed picture concerning the design of encryption algorithms. As discussed in this paper, some of lightweight block cipher algorithms use a Feistel network while the others used the SPN and each one has their own properties. Moreover, in researching and analysing the existing lightweight block cipher algorithms it was found that the algorithms with many S-boxes meant that the security is good but the cost is high. Furthermore, in the event the algorithm has enough number of S-boxes and also has well designed linear operations, then the security is high and the cost is dependent upon the design.

REFERENCES

[1]. F Alshaikhli, Imad, and Mohammad A AlAhmad. "Security Threats of Finger Print Biometric in Network System Environment." Journal of Advanced Computer Science and Technology Research 1.1 (2011).

[2]. Panasenko, S., & Smagin, S., “Lightweight Cryptography: Underlying Principles and Approaches”, International Journal of Computer Theory and Engineering, Vol 3 No.4, (2011).

[3]. Sufyan Salim Mahmood AlDabbagh and Imad Al Shaikhli,” Lightweight Block Ciphers: a Comparative Study”, in Journal of Advanced Computer Science and Technology Research Vol.2 No.4, November 2012, 159-165.

[4]. S. S. M. AlDabbagh, et al., "Hisec: A new lightweight block cipher algorithm," in Proceedings of the 7th International Conference on Security of Information and Networks, 2014, p. 151.

[5]. S. S. M. Aldabbagh, et al., "OLBCA: A New Lightweight Block Cipher Algorithm," in Advanced Computer Science Applications and Technologies (ACSAT), 2014 3rd International Conference on, 2014, pp. 15-20.

[6]. J. Borghoff, et al., "PRINCE – A Low-Latency Block Cipher for Pervasive Computing Applications," in Advances in Cryptology – ASIACRYPT 2012. vol. 7658, Springer Berlin Heidelberg, 2012, pp. 208-225.

[7]. L. Knudsen, et al., "PRINTcipher: A Block Cipher for IC-Printing," in Cryptographic Hardware and Embedded Systems, CHES 2010. vol. 6225, Springer Berlin Heidelberg, 2010, pp. 16-32.

[8]. A. Bogdanov, L. Knudsen, G. Leander, C. Paar, A. Poschmann, M. Robshaw, Y. Seurin, and C. Vikkelsoe, "PRESENT: An Ultra-Lightweight Block Cipher Cryptographic Hardware and Embedded Systems - CHES 2007." Vol. 4727, Springer Berlin / Heidelberg, 2007, pp. 450-466.

[9]. Z. Gong, S. Nikova, and Y. Law, "KLEIN: A New Family of Lightweight Block Ciphers RFID. Security and Privacy." Vol. 7055, Springer Berlin / Heidelberg, 2012, pp. 1-18.

[10]. W. Wu and L. Zhang, "LBlock: A Lightweight Block Cipher Applied Cryptography and Network Security." Vol. 6715, Springer Berlin / Heidelberg, 2011, pp. 327-344.

[11]. T. Suzaki, et al., "TWINE: A Lightweight Block Cipher for Multiple Platforms," in Selected Areas in Cryptography. vol. 7707, Springer Berlin Heidelberg, 2013, pp. 339-354.

[12]. J. Guo, T. Peyrin, A. Poschmann, and M. Robshaw, "The LED Block Cipher Cryptographic Hardware and Embedded Systems – CHES 2011." Vol. 6917, Springer Berlin / Heidelberg, 2011, pp. 326-341.

[13]. E. Biham and A. Shamir, "Differential Cryptanalysis of DES Variants," in Differential Cryptanalysis of the Data Encryption Standard, Springer New York, 1993, pp. 33-77.

[14]. J.-S. Kang, et al., "Practical and Provable Security against Differential and Linear Cryptanalysis for Substitution-Permutation Networks," ETRI Journal, vol. 23, pp. 158-167, 2001.

[15]. Biham, E., Shamir, A.: Differential cryptanalysis of des-like cryptosystems. In: Menezes, A., Vanstone, S.A. CRYPTO 1990. LNCS, vol. 537, pp. 2–21. Springer, Heidelberg (1990).

[16]. L. Knudsen and D. Wagner, "Integral Cryptanalysis," in Fast Software Encryption. vol. 2365, J. Daemen and V. Rijmen, Springer Berlin Heidelberg, 2002, pp. 112-127.

[17]. W. Shengbao and W. Mingsheng, "Integral Attacks on Reduced-Round PRESENT," in Information and Communications Security. vol. 8233, Springer International Publishing, 2013, pp. 331-345.

[18]. Y. Sasaki and L. Wang, "Comprehensive Study of Integral Analysis on 22-Round LBlock," in Information Security and Cryptology – ICISC 2012. vol. 7839, Springer Berlin Heidelberg, 2013, pp. 156-169.

[19]. D. Wagner, "The Boomerang Attack," in Fast Software Encryption. vol. 1636, Springer Berlin Heidelberg, 1999, pp. 156-170.

[20]. Sasaki, Y., & Wang, L.. Comprehensive Study of Integral Analysis on 22-Round LBlock. In T. Kwon, M.-K. Lee & D. Kwon (Eds.), Information Security and Cryptology – ICISC 2012: Springer Berlin Heidelberg. 2013, Vol. 7839, pp. 156-169.

[21]. Sasaki, Y., & Wang, L. Meet-in-the-Middle Technique for Integral Attacks against Feistel Ciphers. In L. Knudsen & H. Wu (Eds.), Selected Areas in Cryptography: Springer Berlin Heidelberg. 2013, Vol. 7707, pp. 234-251.

[22]. Z'aba, M., Raddum, H., Henricksen, M., & Dawson, E.. Bit-Pattern Based Integral Attack. In K. Nyberg (Ed.), Fast Software Encryption: Springer Berlin Heidelberg. 2008, Vol. 5086, pp. 363-381.

[23]. Shengbao, W., & Mingsheng, W.. Integral Attacks on Reduced-Round PRESENT. In S. Qing, J. Zhou & D. Liu (Eds.), Information and Communications Security: Springer International Publishing. 2013, Vol. 8233, pp. 331-345.

[24]. Matsui, M.. Linear Cryptanalysis Method for DES Cipher. In T. Helleseth (Ed.), Advances in Cryptology —EUROCRYPT '93: Springer Berlin Heidelberg. (1994), Vol. 765, pp. 386-397.