Journal of Computing and Security

January 2016, Volume 3, Number 1 (pp. 13–26)

http://www.jcomsec.org

A Novel Block Cipher Algorithm with Feistel-Like Structure

Mahmood Deypir [a,*], Yosef Pourebrahim [b]

[a] *Shahid Sattari University of Science & Technology, Tehran, Iran.*

[b] *Islamic Azad University, Meshkin Shahr Branch, Meshkin Shahr, Iran.*

ARTICLE INFO.

Article history:

Received: 20 September 2015

Revised: 09 April 2016

Accepted: 03 October 2016

Published Online: 25 January 2017

Keywords:

Cryptography, Cryptanalysis, Practical Security, Provable Security, Cipher Algorithm.

ABSTRACT

Block ciphers have wide applications for hardware and software implementations. In this paper, a new block cipher algorithm with provable security is proposed. The whole structure of the algorithm is novel and has a good encryption and decryption performance. Additionally, it has good security with few number of rounds. The structure of the proposed algorithm consists of 4-rounds Feistel-Like which uses 3-rounds Feistel type functions. Moreover, a new method for MDS (Maximal Distance Separable) Matrix construction is proposed and used in the round function as a linear layer. Furthermore, some considerations in S-Boxes of the algorithm lead to obtaining better algebraic expression than AES S-boxes. The Algorithm has a high margin of security against various cryptanalysis methods due to using specific functions in its round functions. Our theoretical evaluations show that the devised cipher algorithm has provable security against attacks based on linear and differential cryptanalysis and it is robust against differential, truncated differential, boomerang, and integral cryptanalysis in terms of practical security.

1 Introduction

Block cipher algorithms are used in a large number of secure communication systems for several different purposes including confidentiality, message integrity, pseudo-random generation, etc. In cryptography, a block cipher is a deterministic algorithm working on fixed-length groups of bits, named blocks, with an unvarying transformation which is specified by a symmetric key. Block ciphers are important components in the design of many cryptographic protocols and are widely exploited to implement the encryption of bulk data. Recently, lightweight block cipher algorithms have attracted a great deal of interest due to their application in hardware and software systems. LED [1], Speck, and Simon [2] are examples of block cipher algorithms. A Goal behind designing such algorithms is a hardware or software implementation with low energy consumption. Therefore, these algorithms can be embedded in devices like a sensor network and an RFID [1].

Although these algorithms somewhat reach their aims, but from a security perspective, most of them could not provide provable security against different types of attacks. This is due to more than enough consideration of their lightweight property. Therefore, in their designs, certain structures for hardware implementation were used which lead to difficulty or even impossibility of their security analysis. For example, in Simon algorithm [2] AND operator was used which was not analyzed well and has some weaknesses. As another example, for Speck algorithms, a number of effective attacks were found [3]. Among different attacks against block based cryptographic algorithms,

* Corresponding author.

Email addresses: mdeypir@ssau.ac.ir (M. Deypir), y.pourebrahim@gmail.com (Y. Pourebrahim)

ISSN: 2322-4460

differential attack [4] and linear attack [5] are known as the best attacks for analyzing these types of algorithms. Therefore, the necessity of designing robust block based algorithms against these attacks is revealed. Since the introduction of differential and linear cryptanalysis, a great deal of effort has been made for designing robust block based algorithms against these cryptanalysis. In this way, one of the key points is provable security against these attacks. In order to prove the security of a block cipher structure against these attacks, an upper bound for most attacks of these types must be determined. In this paper, a new cipher algorithm is proposed which is suitable for software implementation and is robust against the well-known attacks. Moreover, it has provable security against differential and linear cryptanalysis.

The organization of this paper is as follows. In the next section, some related works are reviewed. In Section 3, the structure of the proposed cipher algorithm is introduced. In Section 4, the security of the algorithm is analyzed. Subsequently, statistical analysis is presented in Section 5. Finally, Section 6 concludes the paper.

2 Related Works

There are a large number of researches regarding the design and analysis of block ciphers in the last three decades which led to a significant progress. An example of a lightweight cryptographic algorithm proposed in the literature is KLEIN [6]. This algorithm has some potential weaknesses since it does not present provable security [7]. In most cases, the security of block based algorithms is evaluated by analyzing their resistance against well-known attacks. Designing block ciphers with provable and practical security against popular attacks such as differential and linear cryptanalysis is an interesting research area due to the importance of security in critical systems. The first sample of a block cipher having provable security against differential and linear cryptanalysis has been proposed by Nyberg et al. [8, 9]. Matsui introduced an approach for designing block ciphers with provable security against the above-mentioned attacks [10]. This approach that was based on principles presented in [8, 9], reduced the size of substitution box by using an iterative structure for the round function. Thereafter, this approach was used in some block ciphers [11, 12]. In 1997, a group of Japanese researchers discussed the provable security against differential and linear cryptanalysis of generalized Feistel ciphers with multiple functions [13]. In 2000, a group of Korean cryptography researchers, illustrated provable security against differential and linear cryptanalysis for a structure substitution and permutation graph [14]. As an example of a newer structure of block cipher having provable security against the above types of attacks, the pseudo RC6 structure introduced in 2006 can be mentioned [15]. There are other researches regarding cryptanalysis and provable security in the literature [16, 17] in which security is the most important thing.

In [16] seven new block cipher structures, including Feistel-variant structure, were introduced and evaluated. It was shown that these structures have provable security against differential cryptanalysis attack and upper bounds for average differential probabilities over at least 5 rounds were obtained. Moreover, under the assumption that all of the used components are bijective, these structures are provably resistant against linear cryptanalysis. Upper bounds for maximum differential and linear probabilities of an SMS4-like block cipher in order to evaluate practical security against differential and linear cryptanalysis were investigated in [18]. This block cipher which employs a special kind of unbalanced Feistel structure has been accepted as the Chinese National Standard for securing Wireless *Local Area Networks*. Classification, security, and efficiency of Generalized Feistel Networks (GFNs) were investigated in [19]. Generally, any combination of several keyed non-linear functions using XOR operations and permutations in an invertible manner can be named GFN. More precisely, such a combination of functions can be regarded as a GFN only when the rounds are connected by a rotation of one line instead of a generic line permutation. Simpira [20] has been proposed as a Generalized Feistel Structure (GFS) with an F-function that contains two rounds of AES. The analysis performed by designers of this structure is based on two basic bounds including full bit diffusion, and a minimum number of active S-boxes. The number of rounds for each variant was selected as three times the number of rounds required to prove full bit diffusion and a minimum number of 25 (differentially or linearly) active S-boxes. In SAC 2013 Berger et al. [21] devised a unified variation of GFNs based on a matrix representation and used it to further study the diffusion properties of GFNs. They also extended this matrix representation and introduced a broader class of Feistel networks called Extended Generalized Feistel Networks (EGFNs). Differential cryptanalysis of this structure was investigated in [22].

3 The Proposed Cipher Algorithm

In this section, the structure of the proposed cipher is described. The structure is designed so that the resulting cipher algorithm becomes secure enough to be used for critical applications. Moreover, its security can be measured and evaluated before its actual usage. For this purpose, required components are designed

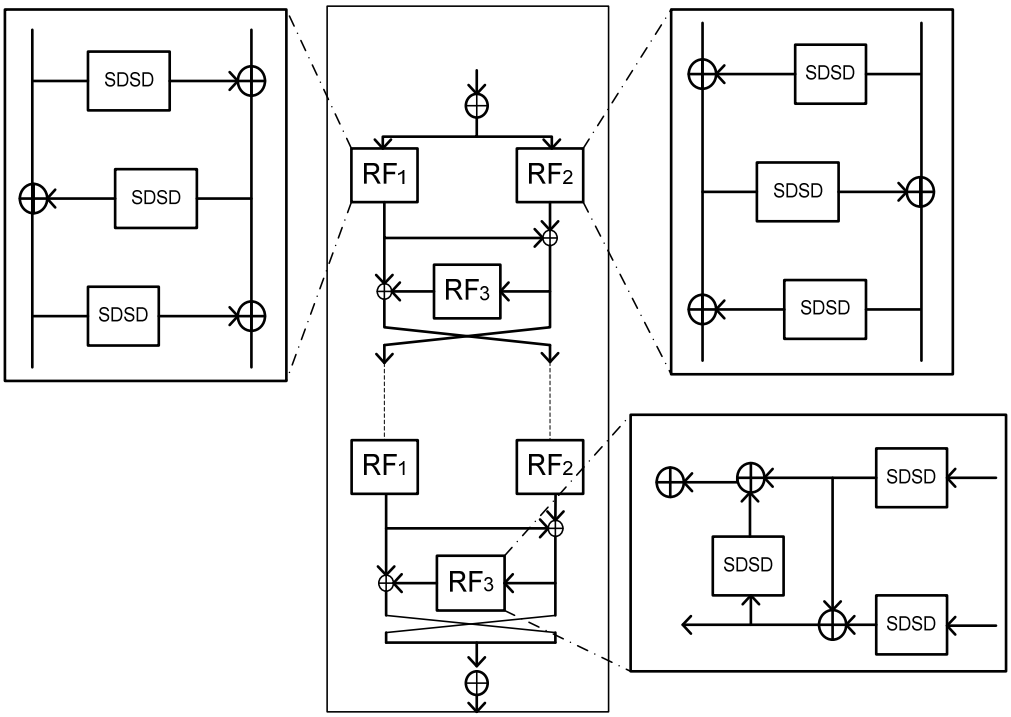

Figure 1. The Proposed Cipher Algorithm

and exploited in the structure. Although it is similar to the Feistel structure, our approach to employ the structure and its round functions are designed so that cryptanalysis can be performed using various types of cryptanalysis attacks. Therefore, the block cipher algorithm introduced in this paper is a Feistel-like algorithm and it benefits from some advantages of the SPN structure. In this algorithm, the block size and the key sizes are 128 bits and 256 bits, respectively. The algorithm has a completely straightforward implementation since it is implemented based on selection table. On the other hand, it has a smaller number of loops with respect to other algorithms. Therefore, the algorithm is semi-lightweight and thus, it is more suitable for software applications but it is not an ideal choice for hardware implementation since providing provable security requires considerable hardware resources. The structures of encryption and decryption of the algorithm are described in the following sections.

3.1 The Cipher Structure

The encryption block diagram is partly shown in Figure 1. This figure shows two rounds of the algorithm. This figure illustrates how SDSD (Substitution Diffusion Substitution Diffusion) functions are exploited in the first, second, and third round functions (RFs). The size of the input block is 128 bits which is organized as four 32-bit words denoted by $m[i], i = 0, 1, 2, 3$ where $m[0]$ is the most significant word (32 bits of the left-hand side of the 128-bit block). For encryption, the input text is added to 128-bit key in modulo 2.

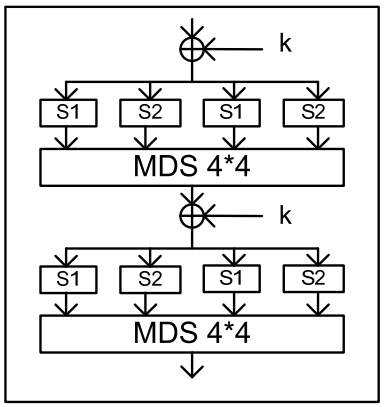

Figure 2. SDSD Round function

The number of rounds is equal to 4 and each round has three functions namely $RF_i, i = 1, 2, 3$. The round functions (RFs) are Feistel with three rounds using SDSD (Substitution Diffusion Substitution Diffusion) function. The structure of round function is depicted in Figure 2.

Each SDSD gets a 32-bit text as the input and produces the result using two 32-bit keys. Therefore, each round function requires $3 \times (32 + 32) = 192$ bits of subkeys since each SDSD function requires a 64-bit subkey and each round function include three SDSD functions. As shown in Figure 1, since each round is

composed of three RF_i functions, there are bits of subkeys in each round.

3.2 SDSD Round Function

The input for SDSD function is 32 bits of text and 64 bits of round subkey. The input is fed into four S-boxes, in 4-byte format. The output is entered into the diffusion layer after performing XOR operation with the least significant 32 bits of the subkey of the round. The diffusion layer is a MDS (Maximal Distance Separable) matrix. The output of this layer is entered into 4 S-boxes in the form of 4 separate bytes after performing XOR operation with 32 most significant bits of the round subkey. Subsequently, the result is moved again through the diffusion layer and produces 32 bits of output.

4 The Principles of the Algorithm

The overall structure of the proposed algorithm is designed to provide provable security against linear and differential cryptanalysis. Moreover, it must be secure against the other attacks like the boomerang and the integral attacks. Furthermore, it must be suitable for software implementation and can be embedded in other systems.

4.1 Round Function

To prove the upper bound for differential probability and linear mask, it is required to determine the differential and linear probability of the round function. Therefore, in the proposed algorithm, a three-round simple Feistel is used whose provable linear and differential probability can be expressed.

4.2 SDSD Function

To reduce the differential and linear probabilities of the structure, the differential and linear probabilities of RF_i functions must be as small as possible. For this purpose, an SDS (Substitution Diffusion Substitution) structure can be used. The advantage of this structure is its provable linear and differential security. In this structure, the linear and differential probabilities are the least, especially when appropriate substitution functions and diffusion layers with the highest branching factor are used. However, SDS structure does not have suitable software implementation. Therefore, here, the SDSD structure is used in which each SD function is implemented as a lookup table. For an implementation based on a lookup table, the SDSD requires smaller memory rather than the SDS structure. In fact, the SDSD can be implemented using one lookup table but the SDS requires two lookup tables, *i.e.*, one for SD and the another one for S. Moreover, using the SDSD structure in the proposed cipher leads to achieving acceptable security (at least in terms of linear and differential) in a smaller number of rounds. Therefore, it has better resistance against cryptanalysis with respect to the SDS structure.

4.3 The Linear Layer and the Round Function

Provable high security for the SDS based structure is achieved when it uses an MDS-type linear layer in the SDS round function. Therefore, in this algorithm, regarding the implementation consideration and the above-mentioned reason, a novel method for constructing MDS matrices is proposed and used. In this proposed approach, for creating an $n \times n$ matrix, firstly, a closed set under XOR operation with distinct members is produced. For example, if $n = 4$, this set is equal to $\{0, \alpha, \beta, \alpha \oplus \beta\}$ in which α, β are distinct nonzero members. Subsequently, each of $\{x_0, \ldots, x_{n-1}\}$ is assigned to one member of the above set and the value of Δ is computed so that it does not belong to the set of x_i . Finally, the values of $y_i = x_i \oplus \Delta$ are computed and $B = (b_{ij})$ matrix is produced so that $b_{ij} = \frac{1}{\lambda(x_i+y_i)}$. In this matrix, λ has two possible values according to existing circumstances. For the self-inverse matrix, we choose $\lambda = \oplus_{k=0}^{n-1} \frac{1}{y_k}$, otherwise, we choose $\lambda = 1$. Since the structure of the algorithm is Feistel-Like, the self-inverse matrix is not required. According to this approach, a large number of MDS matrices can be produced. An example with initial polynomial of $x^8 + x^4 + x^2 + 1$ is as follows:

$$D = \begin{bmatrix} 225 & 1 & 213 & 205 \\ 1 & 225 & 205 & 213 \\ 213 & 205 & 225 & 1 \\ 205 & 213 & 1 & 225 \end{bmatrix} \quad (1)$$

For implementing SD function in a table format, 4 kilobytes of memory is required for storing two S-boxes. Compared to AES, the above MDS matrix has a higher cost. In fact, AES is faster in encryption step because its elements have smaller hamming weights. However, for decryption, it is slower compared to encryption step since the used inverse matrix has elements containing larger hamming weights. Here, the aim is to use new elements in the proposed cipher and to propose a self-inverse matrix construction process.

4.4 Substitution Box

In order to enhance the security of the algorithm against cryptanalysis such as differential, boomerang,

integral, etc., as well as providing provable security, two new substitution functions are used in the SDSD round function. S-boxes of the algorithm are similar to those in the AES algorithm. However, using the proposed approach described in the paper, the number of terms is much larger than those in AES. Therefore, the devised algorithm becomes more robust against algebraic attacks. Mathematical relations between input and output of S-boxes are according to the following equations:

$$Sbox1(x) = [A \times ((x + 128) \oplus 99)^{-1}] \oplus 59 \quad (2)$$

$$Sbox2(x) = [B \times (((B \times x) + 192) \oplus 74)^{-1}] \oplus 135 \quad (3)$$

In these equations the Boolean functions A and B are defined as follows:

$$A = \begin{bmatrix} 1&1&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0 \\ 0&0&1&1&0&0&0&0 \\ 0&0&0&1&1&0&0&0 \\ 0&0&0&0&1&1&0&0 \\ 0&0&0&0&0&1&1&0 \\ 0&0&0&0&0&0&1&1 \\ 0&0&0&0&0&0&0&1 \end{bmatrix} B = \begin{bmatrix} 0&1&1&1&1&1&1&0 \\ 1&0&1&1&0&1&1&1 \\ 1&1&0&1&1&0&1&1 \\ 1&1&1&0&1&1&0&1 \\ 1&1&0&1&1&1&0&0 \\ 1&1&1&0&0&1&1&0 \\ 0&1&1&1&0&0&1&1 \\ 1&0&1&1&1&0&0&1 \end{bmatrix} \quad (4)$$

The S-box1 and S-box2 are selected with the following properties. 1) They don't have fixed and reversed points. 2) They have at least 112 non-linearity degrees. 3) They have at most 4 differential values in the approximate differential matrix. 4) Their biases in linear approximation are at most 16. 5) They have the maximum number (255 terms) of coefficients in the algebraic expression (the AES S-boxes have 9 terms [23]). 6) Their maximum algebraic degree is equal to 7.

4.5 Subkey Generation Algorithm

The number of subkeys used in the algorithm is large. Therefore, using ordinary subkey generation of block based ciphers is not affordable regarding security and run-time factors and can lead to having a correlation among the keys.

One of the approaches to avoid correlation among generated subkeys is using the well-known secure stream ciphers to produce the subkeys. This approach has some advantages including rapid subkey generation while all bits of the main key affect all generated subkeys. Therefore, in the proposed algorithm, SNOW3G stream cipher algorithm is used to design the subkey generation method. The SNOW3G is based on SNOW2 and is widely analyzed from a security perspective and it has not any known security breach [24–26].

4.6 Implementation Issues

Since each SD function has a 32-bit input, for its implementation, the input is considered as $a_4a_3a_2a_1$ in which a_1 and a_2 are the least significant and the most significant eight bits of the input, respectively. The input must firstly pass through the S-boxes. The obtained result can be represented as:

$$S_2(a_4)\ S_1(a_3)\ S_2(a_2)\ S_1(a_1) \quad (5)$$

The resulting 32 bits then pass through MDS matrix. For this purpose, a multiplication in Galois field $GF(2^8)$ must be carried out as:

$$\begin{bmatrix} x4 \\ x3 \\ x2 \\ x1 \end{bmatrix} = \begin{bmatrix} S_2(a_4)\ S_1(a_3)\ S_2(a_2)\ S_1(a_1) \end{bmatrix} \bullet \begin{bmatrix} A&B&C&D \\ B&C&D&A \\ C&D&A&B \\ D&A&B&C \end{bmatrix} \quad (6)$$

The resulting values can be formulated as:

$$x4 = S_2(a_4) \bullet A \oplus S_1(a_3) \bullet B \oplus S_2(a_2) \bullet C \oplus S_1(a_1) \bullet D$$
$$x3 = S_2(a_4) \bullet B \oplus S_1(a_3) \bullet C \oplus S_2(a_2) \bullet D \oplus S_1(a_1) \bullet A$$
$$x2 = S_2(a_4) \bullet C \oplus S_1(a_3) \bullet D \oplus S_2(a_2) \bullet A \oplus S_1(a_1) \bullet B$$
$$x2 = S_2(a_4) \bullet D \oplus S_1(a_3) \bullet A \oplus S_2(a_2) \bullet B \oplus S_1(a_1) \bullet C$$

It is obvious that for obtaining the output of SD function, four lookup operations on the lookup table, 12 XORs, and 16 multiplications in Galois field are required which lead to a heavy computation. In order to enhance the implementation for faster computation, four lookup tables can be constructed using S-boxes, each of which contains 256 units of 32-bit elements as:

$$L4 = S_2 \bullet A | S_2 \bullet B | S_2 \bullet C | S_2 \bullet D$$
$$L3 = S_1 \bullet B | S_1 \bullet C | S_1 \bullet D | S_1 \bullet A$$
$$L2 = S_2 \bullet C | S_2 \bullet D | S_2 \bullet A | S_2 \bullet B$$
$$L1 = S_1 \bullet D | S_1 \bullet A | S_1 \bullet B | S_1 \bullet C$$

In the above equations, "|" symbol is used for better representation and separation of bytes. Using these lookup tables, the output of each SD function is obtained as:

$$X = x4x3x2x1 = L_4(a_4) \oplus L_3(a_3) \oplus L_2(a_2) \oplus L_1(a_1) \quad (7)$$

For faster implementation, paralleling techniques can be exploited in the implementation since the left and right branches of the algorithm operate independently because of its pseudo-Feistel structure.

5 Practical and Provable Security against Well-known Cryptanalysis

Suppose $w_h(a)$ as hamming weight of "a" which shows the number of non-zero elements. Elements can be bits or components from $GF(2^m)$. Therefore branch number is defined according to Equation (8) [14]:

$$\beta(D) = \min_{a \neq 0}(w_h(a) + w_h(D(a))) \tag{8}$$

In this equation β shows a measurement of the worst case of diffusion. Since a cryptanalysis searches to find the worst states, the branching factor is a suitable measure for diffusion characteristic. In equation 5, if '**a**' shows the input difference as a linear layer, then the output difference is in the form of '$D(a)$'. Therefore, the branching factor β_d is the least number of active S-boxes in the form of differential in two rounds of SPN. Similarly, β_1 is the least number of active S-boxes in two rounds. A differentially active S-box is defined as an S-box given a non-zero input difference. On the other hand, a linearly active S-box is an S-box given a non-zero output mask value [27].

Diffusion layer matrix used in the algorithm is MDS (Maximum Distance Separable) type since it has not any square sub-matrix with zero determinants. The above discussions can be summarized using the following theorems. The results of these theorems are used for various cryptanalysis of the algorithm.

Theorem 1. *For each $n \times n$ MDS matrix, linear and differential branching factor is equal to $n+1$ [14]. This theorem states that, if MDS matrices are used as a linear layer in the SPN structure, the number of active S-boxes will be maximized.*

Theorem 2. *if β_d and β_l are the numbers of active S-boxes in two rounds of SPN (one round SDS), in the forms of differential and linear, respectively, then the probabilities of differential characteristic p and linear characteristic q are as follows [14]:*

$$p \leq p_s^{\beta_d}, \quad q \leq q_s^{\beta_l} \tag{9}$$

where p_s and q_s are the Maximum Differential Probability (MDP) and Maximum Linear Probability (MLP) of an S-box, respectively. This theorem is used in practical security evaluation of the algorithm. In this evaluation, only the number of active S-boxes is required to obtain the probability of differential or linear characteristics.

Theorem 3. *if β_d and β_l are the numbers of active S-boxes in two rounds of SPN (one round SDS), in the form of differential and linear, respectively, then the probabilities of differential p and linear q are as follows:*

$$p \leq p_s^{\beta_d - 1}, \quad q \leq q_s^{\beta_l - 1} \tag{10}$$

This theorem is used for computing provable security in which obtaining differential or linear probability values which are equal to the summation of probabilities for differential or linear characteristics are desirable.

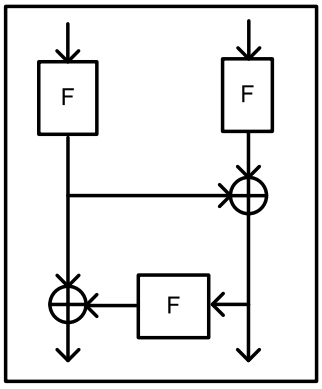

Figure 3. Feistel-Like Structure

Theorem 4. *For the structure depicted in Figure 3 with bijective round function F, the provable differential and linear probabilities p and q are [16]:*

$$p \leq p_F^2, \quad q \leq q_F^2 \tag{11}$$

5.1 Practical Security

In each round of the algorithm, at least two round functions (RF_i) are active for each input element. For each active round function, at least two active SDS functions are created. Consequently, for each round of the algorithm, there exist at least four active SDS functions. For each active SDS function, at least five substitution functions (S-box) are activated due to using MDS diffusion layer. Therefore, in each round, twenty active substitution functions are created which will produce practical linear and differential security from the following order by applying Theorems 1 and 2:

$$\begin{aligned} p \leq p_{RF_i}^2 &= (p_{SDS}^2)^2 = ((p_{Sbox}^5)^2)^2 \\ &= (2^{-6})^{5\times 2\times 2} = 2^{-120} \end{aligned} \tag{12}$$

$$\begin{aligned} q \leq q_{RF_i}^2 &= (q_{SDS}^2)^2 = ((q_{Sbox}^5)^2)^2 \\ &= (2^{-6})^{5\times 2\times 2} = 2^{-120} \end{aligned} \tag{13}$$

For the resistance of the algorithm against differential and linear cryptanalysis, the probabilities of differential and linear characteristics must be less than 2^{-128}. To provide this condition, the number of rounds must be large enough to have twenty two active S-boxes. One and a half rounds are enough to reach this number. For this number of rounds, there are thirty active S-boxes. Therefore, the probability of differential and linear characteristics of 1.5 rounds is equal to $2^{-6\times 30} = 2^{-180} < 2^{-128}$. In fact, before the end of the second round, the number of active S-boxes is enough for resilience against linear and differential cryptanalysis. For complete two rounds of the algorithm, it is more resistant against these attacks.

5.2 The New Approach of Practical Security

In this approach, instead of counting the active S-boxes, the active functions that have provable security are counted. In the proposed algorithm, in each round, there exist at least two active round functions. Based on Theorems 3 and 4, each active round function is three rounds Feistel with SDS round function having provable linear and differential security as follows:

$$p < p_{SDS}^2 = (p_{Sbox}^4)^2 = (2^{-6})^{4\times 2} = 2^{-48} \quad (14)$$

$$q < q_{SDS}^2 = (q_{Sbox}^4)^2 = (2^{-6})^{4\times 2} = 2^{-48} \quad (15)$$

For computing the above probabilities, previously mentioned theorems are used. Therefore, practical security is presented in Table 1.

5.3 Provable Security

According to the theorems described in the previous section, one round of the algorithm has the following provable security:

$$p \le p_F^2, \quad q \le q_F^2 \quad (16)$$

p_F and q_F are obtained regarding Feistel round function with three loops having probabilities $p_F \le p_{SDS}^2, q_F \le q_{SDS}^2$. Therefore, provable differential and linear securities are equal to:

$$p \le p_{SDS}^2 = (p_{Sbox}^4)^2 = (2^{-6})^{4\times 2} = 2^{-48} \quad (17)$$

$$q \le q_{SDS}^2 = (q_{Sbox}^4)^2 = (2^{-6})^{4\times 2} = 2^{-48} \quad (18)$$

The above probabilities are computed assuming that the linear and differential probabilities of S-boxes are at most 2^{-6}. In fact, for the S-boxes used in this algorithm there is only one characteristic with 2^{-6} probability and there are 126 characteristics with 2^{-7} probabilities in the differential characteristic distribution table. As a result, the differential probabilities can be smaller than the value used in Equation (19).

Theorem 5. *For SDS with X^{-1} type substitution layer of size $m \times m$ and diffusion layer with branching factor of β_d, the maximum differential probability is computed as [28]:*

$$\max\Big(\max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} DP^{Sbox}(u \to j)^{B_d}\Big) \quad (19)$$

According to the above theorem, maximum differential probability of SDS function used in the algorithm is equal to:

$$p_{SDS} = 2^{(1-m)(\beta_d-1)} - 2^{(1-m)\beta_d+1} + 2^{(2-m)\beta_d} = 2^{-28} - 2^{-34} + 2^{-30} = 2^{-27.696} \quad (20)$$

On the other hand, the differential probability for RF_i function is:

$$p_F \le p_{SDS}^2 = 2^{-55.39} \quad (21)$$

Therefore, the security of one round of the algorithm is in order of:

$$p \le p_F^2 = 2^{-110.78} \quad (22)$$

The linear probability can also be computed by this method and using a linear characteristic table.

$$q_{SDS} = 2^{-26.477} \quad (23)$$

$$q \le (q_F^2)^2 = 2^{-26.477\times 4} = 2^{-105.91} \quad (24)$$

5.4 Security of the Algorithm Against Boomerang Cryptanalysis

In this section, the security of the algorithm against the boomerang cryptanalysis is evaluated and it is shown that the proposed algorithm with more than two rounds is secured against this attack. The boomerang cryptanalysis is a stronger type of differential attack. It is an adaptively chosen plaintext-ciphertext attack which was suggested by Wagner in 1999. This attack is based on the two characteristics of the short differential which are exploited in a quad structure. The main idea of this attack is to use two high probability short differentials instead of one differential and numerous rounds with lower probability. The complexity of this cryptanalysis is $(p_1p_2)^{-2}$ in which p_1 and p_2 are differential probabilities of forward and returning paths. Suppose that n is the block length of the algorithm. Based on this assumption, the boomerang is effective, provided that $(p_1p_2)^{-2} < 2^n$ [17].

The security of $r+1$ rounds of a block algorithm is not smaller than its security for r rounds. Therefore, if we prove the security of the algorithm for the least number of rounds, which is two, it can be also true for a higher number of rounds.

For applying the boomerang cryptanalysis to two rounds of the algorithm, we decompose the algorithm into r_1 rounds and r_2 rounds components where $r_1 + r_2 = 2$. In Table 2, all probable states for r_1, r_2 and the corresponding probabilities p_1 and p_2 are listed. In the last column, for each state, the upper bound of the boomerang complexity is noted.

As can be seen in Table2, for all possible states, the upper bound of the boomerang complexity is greater than 2^{128}. Therefore, it can be concluded that using two rounds, the cipher is secure against this attack. Since the security of $r+1$ rounds of this block based algorithm is not smaller than its r rounds version, it can be concluded that this cipher algorithm is secure against the boomerang attack. Consequently, the algorithm has a higher margin of safety against the boomerang since it has three rounds.

Table 1. The Order of Practical Security Based on the New Approach for Different Rounds of the Algorithm

Number of Rounds	Number of Active Round Functions	Probability Order of Differential Characteristic	Probability Order of Linear Characteristic
1	2	$-48 \times 2 = -96$	$-48 \times 2 = -96$
1.5	3	$-48 \times 3 = -144$	$-48 \times 3 = -144$
2	4	$-48 \times 4 = -192$	$-48 \times 4 = -192$
2.5	5	$-48 \times 5 = -240$	$-48 \times 5 = -240$
3	6	$-48 \times 6 = -288$	$-48 \times 6 = -288$

Table 2. Probable States for Applying Boomerang Cryptanalysis Against Two Rounds of the Proposed Algorithm

State	r_1	r_2	p_1	p_2	Upper Bound of Attack Complexity $(p_1 p_2)^{-2}$
1	1	1	2^{-96}	2^{-96}	2^{384}
2	1.5	0.5	2^{-144}	2^{-48}	2^{384}
3	1.5	1.5	2^{-48}	2^{-144}	2^{384}

5.5 Security Against Impossible Differential Attack

Among the cryptanalyses that we have performed on the cipher algorithm, the impossible differential attack based on truncated differential techniques [29] has the best results. Based on this attack, a two-round impossible differential characteristic is achieved for the algorithm. Figure 4 shows the two-round impossible differential characteristic and corresponding differentials. The used symbols in this figure are as follows:

- "0" is 32 bits of zero values
- "δ" is 32 bits of non-zero values
- "a" and "b"are two 32 bits of optionally determined non-zero values
- "?" is 32 bits undetermined value

According to Figure 4, the input difference is equal to $\Delta x = (0, a, 0, 0)$ and the output difference after two rounds is equal to $\Delta y = (b, 0, 0, 0)$. We denote the probability of mapping Δx to Δy after two rounds as $DP^{2r}(\Delta x \rightarrow \Delta y)$. This probability is equal to zero:

$$DP^{2r}(\Delta x \rightarrow \Delta y) = 0 \qquad (25)$$

According to Figure 4, this characteristic is obtained using meet in the middle approach and has two parts, *i.e.* forward and backward, and reach a contradiction at half of the first round (which is the input of the next one and a half rounds) as:

- According to the figure in forward part, if the difference of the input of the structure is equal to $\Delta x = (0, a, 0, 0)$, the output difference after the first half of round becomes $\Delta x = (\delta, \delta + a, 0, 0)$.
- Similarly, the backward part forces that if the difference of the output for the second round is equal to $\Delta y = (b, 0, 0, 0)$, the input difference of the remaining 1.5 rounds becomes $\Delta y = (\delta + b, \delta, \delta + b, \delta)$ and this is in contradiction to the previous state.

From the fact that impossible differential characteristic is obtained for two rounds and the algorithm consists of three rounds, it can be concluded that the algorithm is robust against impossible differential attack.

5.6 The Security Against Integral Cryptanalysis

The square or integral attack was originally proposed as a dedicated attack on square block cipher algorithm [30]. In this section, the results of integral cryptanalysis based on 32-bit ports on the proposed algorithm are illustrated. Using reversible property of the round function, one round integral characteristic is obtained for the algorithm. In Figure 5, one round integral differentiator as well as the status of each 32-bit port is shown. The symbols used in this figure are as follows [31]:

- "C" is an inactive 32-bit port (with a 32-bit constant value)
- "A" is an active 32-bit port (with all probable 32-bit values)
- "B" is a balanced 32-bit port (with all 232 32-bit values that the results of performing XOR operation on them is equal to a 32-bit zero value).
- "G" is a garbled 32-bit port (not inactive, not active and not balanced).

Table3 shows the impact of XOR operation on the integral attack.

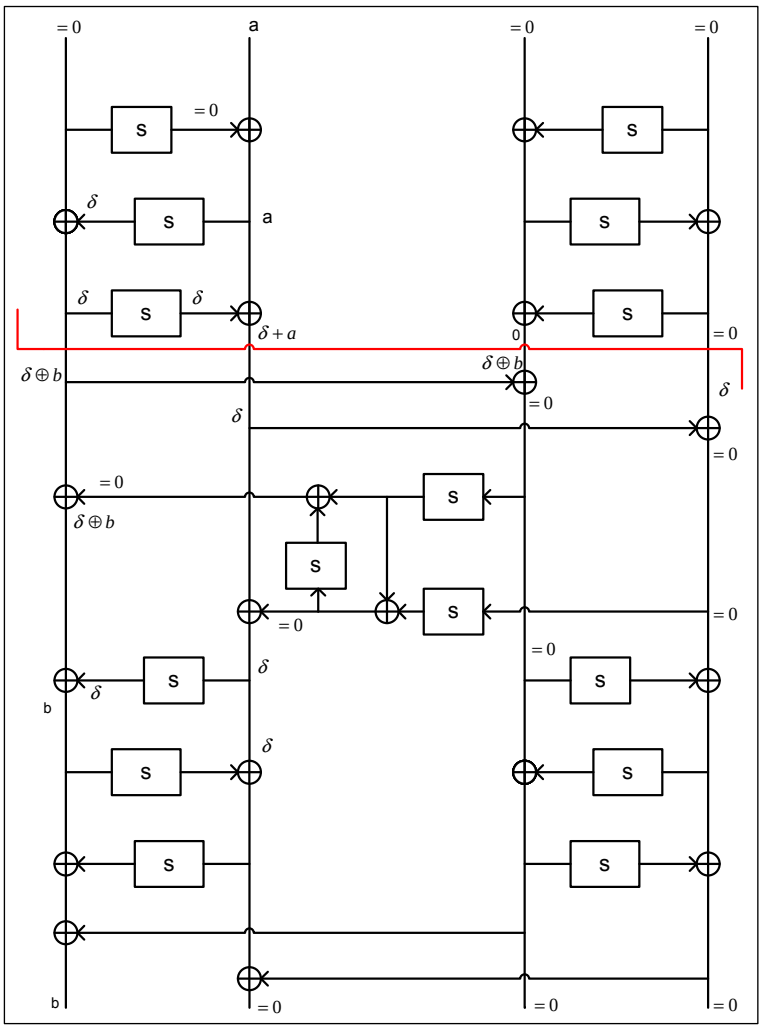

Figure 4. Two-round Impossible Differential Specification

Table 3. The Impact of XOR Operation in Integral Attack

XOR	G	B	A	C
C	G	B	A	C
A	G	B	B	A
B	G	G	B	B
G	G	G	G	G

According to Figure 5, the attacker starts with (C, A, C, C) input. That is, the second port (from the left-hand side) gets all of the probable 32-bit values and the values of the other ports are kept constant. In this situation, according to the figure, the output of the first round will be (G, G, A, B). That is, from the right-hand side, the first port is balanced; the second port is active; the third and the fourth ports are garbled. Using this differentiator, at most two rounds of this algorithm can be attacked by guessing some sub keys.

Since the best integral differentiator was obtained for one round and the algorithm consists of three rounds, it is expected that the algorithm is secure enough against the integral cryptanalysis.

5.7 Truncated Differential Attack

For applying truncated differential cryptanalysis on block ciphers, the following definitions are used [32, 33].

Definition 1. χ of $GF(2)^4 \rightarrow GF(2)$ transformation is defined as:

$$\chi(x) = \begin{cases} 0 & if \quad x = 0 \\ 1 & if \quad x \neq 0 \end{cases} \tag{26}$$

$$\chi(x_1, x_2, x_3, x_4) = (\chi(x_1), \chi(x_2), \chi(x_3), \chi(x_4)) \tag{27}$$

Definition 2. $\delta x \in GF(2)^4$ and $\delta y \in GF(2)^4$ are defined as:

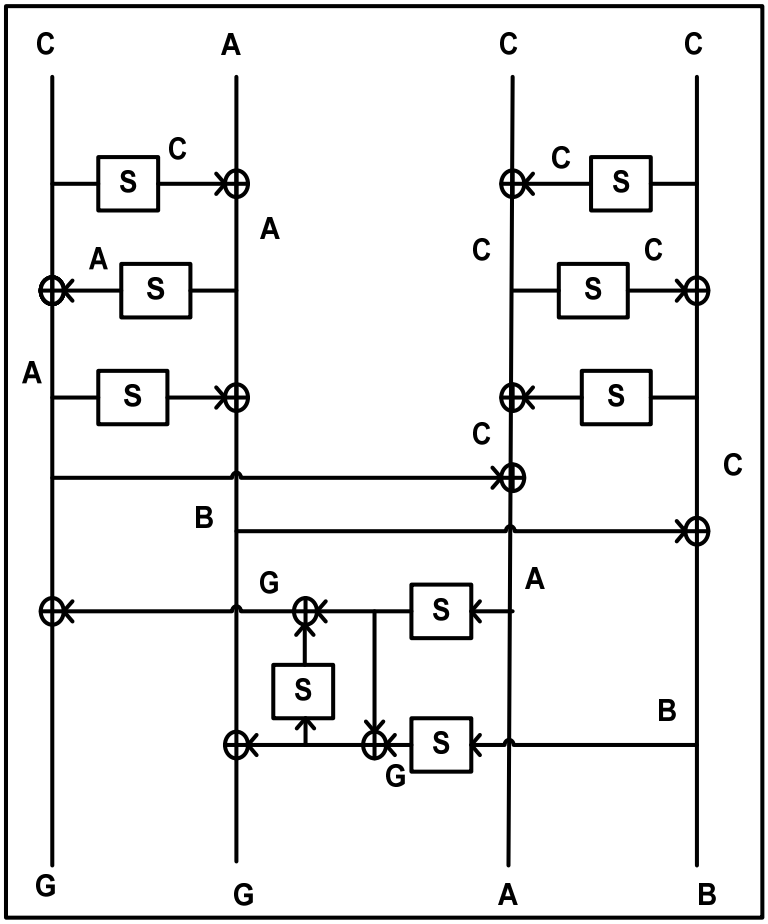

Figure 5. Applying Integral Cryptanalysis on One Round of the Proposed Algorithm

$$\delta x = (\delta x_1, \delta x_2, \delta x_3, \delta x_4), \quad \delta x_i \in GF(2) \tag{28}$$
$$\delta y = (\delta y_1, \delta y_2, \delta y_3, \delta y_4), \quad \delta y_i \in GF(2) \tag{29}$$
$$\delta x_i = \chi(\Delta x_i), \quad \Delta x_i \in GF(2)^8 \tag{30}$$
$$\delta y_i = \chi(\Delta y_i), \quad \Delta y_i \in GF(2)^8 \tag{31}$$

Definition 3. The probability of the byte characteristic of function F is defined as:

$$p = \underset{x \in GF(2)^{32}}{Pr}[\chi(F(x)) \oplus F(x \oplus \Delta x)) = \delta y \mid \chi(\delta x) = \delta x] \tag{32}$$

In the above relation, it is assumed that the output difference of uniform distribution for each input difference is not equal to zero. Therefore, byte characteristic δx goes to δy with probability of p.

Theorem 6. *If $\delta x = (0,0,0,0)$ then $\delta y = (0,0,0,0)$ with probability of 1, otherwise the probability is computed using the complete search of all byte characteristics of SDSD function. Table 4 show the byte characteristics for SD function.*

For example, if the input characteristic is (0011), then 255 output characteristics equal to (0111) are created. Therefore, the probability of occurring this characteristic in the output is computed as follows:

$$Pr(0011 \rightarrow 0111) = \frac{255}{255 \times 255} = 2^{-7.994} \tag{33}$$

Similarly, the probability of the output characteristic (1111) for the input characteristic (0011) is computed as follows:

$$Pr(0011 \rightarrow 1111) = \frac{64005}{255 \times 255} = 2^{-0.0228} \tag{34}$$

Since the branching factor of the diffusion layer is the most probable value ($\beta_d = 5$), therefore, for each input byte characteristic with hamming weight equal to 1, a characteristic with hamming weight of 4 (1111) is produced in the output with probability of 1.

If the input byte characteristic is equal to (1111), the following characteristics with corresponding probabilities are occurred in the output. It is useful to note that, for a specific input characteristic, hamming weight is used as the representative output characteristic due to the equal probability of occurring the characteristics with the same hamming weights:

Table 4. Byte Specification of Two Inputs SD Function

Input Characteristic	Output Specification/Number of Iterations for Output Characteristic														
	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0011	0	0	0	0	0	0	255	0	0	0	255	0	255	255	64005
0101	0	0	0	0	0	0	255	0	0	0	255	0	255	255	64005
1001	0	0	0	0	0	0	255	0	0	0	255	0	255	255	64005
0110	0	0	0	0	0	0	255	0	0	0	255	0	255	255	64005
1010	0	0	0	0	0	0	255	0	0	0	255	0	255	255	64005
1100	0	0	0	0	0	0	255	0	0	0	255	0	255	255	64005
p	0	0	0	0	0	0	7.944								0.0228

Table 5. Byte Specification of Three Inputs SD Function

Input Characteristic	Hamming Weight of Output Specification/Number of Iterations for Characteristic														
	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0111	0	0	255	0	255	255	64005	0	255	255	64005	255	64005	64005	16323825
1011	0	0	255	0	255	255	64005	0	255	255	64005	255	64005	64005	16323825
1101	0	0	255	0	255	255	64005	0	255	255	64005	255	64005	64005	16323825
1110	0	0	255	0	255	255	64005	0	255	255	64005	255	64005	64005	16323825
p					15.98		8.01								0.0225

$$Pr(1111 \rightarrow wh(1)) = \frac{255}{255^4} = 2^{-23.98} \quad (35)$$

$$Pr(1111 \rightarrow wh(2)) = \frac{64005}{255^4} = 2^{-16.01} \quad (36)$$

$$Pr(1111 \rightarrow wh(3)) = \frac{16323825}{255^4} = 2^{-8.01} \quad (37)$$

$$Pr(1111 \rightarrow wh(4)) = 1 - \sum_{i=1}^{3} Pr(1111 \rightarrow wh(i)) = 2^{-0.0056} \quad (38)$$

5.7.1 Evaluating Truncated Differential Attack Against the Algorithm

For evaluation, two-byte characteristics are used in the input. In Figure 6, the approach for propagating this characteristic within one round is depicted. Additionally, the probability of occurring each byte characteristic after passing through the SDSD function can be inferred from the figure. It is useful to note that, each input characteristic after passing through two SDS functions, produces the output. For example, (0001) characteristic after passing through SD function produces (1111) characteristic with probability of 1 and, subsequently, the produced characteristic generates (1111) characteristic with probability of $2^{-0.0056}$. Consequently, the total probability is equal to:

$$Pr[(0001) \rightarrow (1111) \rightarrow (1111)] = 1 \times 2^{-0.0056} = 2^{-0.0056} \quad (39)$$

The probability of the characteristic depicted in Figure 6 is also:

$$Pr([0000000100010000] \rightarrow [0011011100000000]) \cong 2^{-23.9-15.988-16.3-8.01-16.01-8.03} \cong 2^{-88.238} \quad (40)$$

The probability of failure is:

$$Pr([0000000100010000] \not\rightarrow [0011011100000000]) = 2^{-8\times 11} = 2^{-88} \quad (41)$$

By comparing success and failure probabilities and owing to the fact that some characteristics are impossible, it can be concluded that the probability of truncated differential attack for one round of the algorithm, according to Figure 6 is not unexpected. However, the complete round of the algorithm is resistant to this attack. The reason for this phenomenon can be found in SDSD structure. If instead of this structure, SDS structure is used, the success probability of this attack will become high.

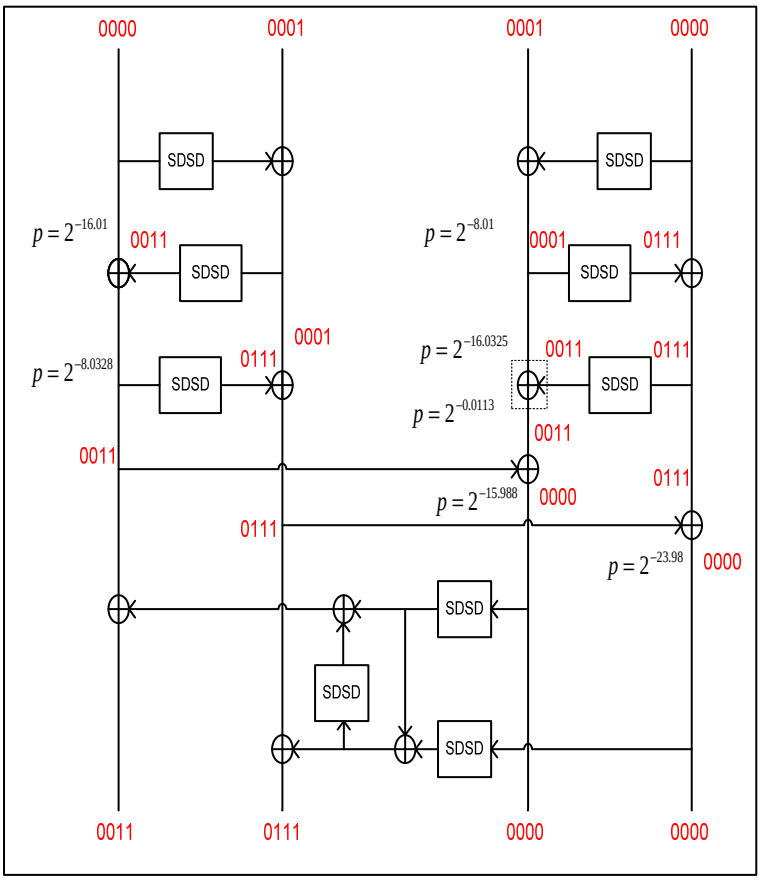

Figure 6. Truncated Differential Diffusion

6 Conclusion and Discussion

In this study, a new cryptographic algorithm is proposed which has practical and provable security and can be used in different applications for telecommunication. In order to trust the algorithm and it's resistance to different types of attacks, it was analyzed against various cryptanalyses. The algorithm structure is Feistel-Like in which Feistel functions were used in each round. The non-linear function used in its round functions is SDSD (Substitution Diffusion Substitution Diffusion) in which the substitution functions are quite novel. On the other hand, it has an overall new structure with respect to other block-based algorithms. In order to enhance the speed of encryption and decryption using the lowest number of rounds, MDS (Maximal Distance Separable) matrices are used which have suitable software implementation. In the proposed cipher algorithm and corresponding cryptanalysis, four rounds of executions are enough. However, adding another round to the algorithm to obtain better security does not significantly degrade the performance due to using special functions in its round structure. To decrease the runtime, the structure of the proposed cipher is selected so that it can be implemented using well-known parallel processing approaches. It was shown that using SDSD structure in round functions produces appropriate security against various attacks such as truncated differential cryptanalysis. Moreover, the proposed algorithm has both practical and provable security against other well-known cryptanalyses. In order to reduce the complexity of the algorithm for hardware implementation, it is possible to use smaller substitution functions. In this case, it is required to use more rounds. However, the proposed algorithm is not completely lightweight since it has linear and differential cryptanalysis which requires specific components in its round functions.

References

[1] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matt Robshaw. *The LED Block Cipher*, pages 326–341. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-23951-9. doi: 10.1007/978-3-642-23951-9_22. URL `http://dx.doi.org/10.1007/978-3-642-23951-9_22`.

[2] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The simon and speck lightweight block ciphers. In *Proceedings of the 52Nd Annual De-*

sign Automation Conference, DAC '15, pages 175:1–175:6, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3520-1. doi: 10.1145/2744769.2747946. URL http://doi.acm.org/10.1145/2744769.2747946.

[3] Farzaneh Abed, Eik List, Stefan Lucks, and Jakob Wenzel. *Differential Cryptanalysis of Round-Reduced Simon and Speck*, pages 525–545. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015. ISBN 978-3-662-46706-0. doi: 10.1007/978-3-662-46706-0_27. URL http://dx.doi.org/10.1007/978-3-662-46706-0_27.

[4] Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. *Journal of Cryptology*, 4(1):3–72, 1991. ISSN 1432-1378. doi: 10.1007/BF00630563. URL http://dx.doi.org/10.1007/BF00630563.

[5] Mitsuru Matsui. *Linear Cryptanalysis Method for DES Cipher*, pages 386–397. Springer Berlin Heidelberg, Berlin, Heidelberg, 1994. ISBN 978-3-540-48285-7. doi: 10.1007/3-540-48285-7_33. URL http://dx.doi.org/10.1007/3-540-48285-7_33.

[6] Zheng Gong, Svetla Nikova, and Yee Wei Law. *KLEIN: A New Family of Lightweight Block Ciphers*, pages 1–18. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-25286-0. doi: 10.1007/978-3-642-25286-0_1. URL http://dx.doi.org/10.1007/978-3-642-25286-0_1.

[7] Jean-Philippe Aumasson, María Naya-Plasencia, and Markku-Juhani O. Saarinen. *Practical Attack on 8 Rounds of the Lightweight Block Cipher KLEIN*, pages 134–145. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-25578-6. doi: 10.1007/978-3-642-25578-6_11. URL http://dx.doi.org/10.1007/978-3-642-25578-6_11.

[8] Kaisa Nyberg. *Linear approximation of block ciphers*, pages 439–444. Springer Berlin Heidelberg, Berlin, Heidelberg, 1995. ISBN 978-3-540-44717-7. doi: 10.1007/BFb0053460. URL http://dx.doi.org/10.1007/BFb0053460.

[9] Kaisa Nyberg and Lars Ramkilde Knudsen. *Provable Security Against Differential Cryptanalysis*, pages 566–574. Springer Berlin Heidelberg, Berlin, Heidelberg, 1993. ISBN 978-3-540-48071-6. doi: 10.1007/3-540-48071-4_41. URL http://dx.doi.org/10.1007/3-540-48071-4_41.

[10] Mitsuru Matsui. *New structure of block ciphers with provable security against differential and linear cryptanalysis*, pages 205–218. Springer Berlin Heidelberg, Berlin, Heidelberg, 1996. ISBN 978-3-540-49652-6. doi: 10.1007/3-540-60865-6_54. URL http://dx.doi.org/10.1007/3-540-60865-6_54.

[11] Mitsuru Matsui. *New block encryption algorithm MISTY*, pages 54–68. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997. ISBN 978-3-540-69243-0. doi: 10.1007/BFb0052334. URL http://dx.doi.org/10.1007/BFb0052334.

[12] Ju-Sung Kang, Sang-Uk Shin, Dowon Hong, and Okyeon Yi. *Provable Security of KASUMI and 3GPP Encryption Mode f8*, pages 255–271. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-540-45682-7. doi: 10.1007/3-540-45682-1_16. URL http://dx.doi.org/10.1007/3-540-45682-1_16.

[13] Yasuyoshi Kaneko, Fumihiko Sano, and Kouichi Sakurai. On provable security against differential and linear cryptanalysis in generalized feistel ciphers with multiple random functions. In *Proceedings of SAC*, volume 97, pages 185–199, 1997.

[14] Seokhie Hong, Sangjin Lee, Jongin Lim, Jaechul Sung, Donghyeon Cheon, and Inho Cho. *Provable Security against Differential and Linear Cryptanalysis for the SPN Structure*, pages 273–283. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-540-44706-1. doi: 10.1007/3-540-44706-7_19. URL http://dx.doi.org/10.1007/3-540-44706-7_19.

[15] Changhoon Lee, Jongsung Kim, Jaechul Sung, Seokhie Hong, and Sangjin Lee. *Provable Security for an RC6-like Structure and a MISTY-FO-like Structure Against Differential Cryptanalysis*, pages 446–455. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. ISBN 978-3-540-34076-8. doi: 10.1007/11751595_48. URL http://dx.doi.org/10.1007/11751595_48.

[16] KIM Jongsung, LEE Changhoon, SUNG Jaechul, HONG Seokhie, LEE Sangjin, and LIM Jongin. Seven new block cipher structures with provable security against differential cryptanalysis. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 91(10):3047–3058, 2008.

[17] Jorge NAKAHARA Júnior. *Cryptanalysis and design of block ciphers*. PhD thesis, Katholieke Universiteit Leuven, June 2003. URL http://www.bybin.narod.ru/nakahara/phd-thesis.pdf.

[18] Qiu-Yan Wang, Bin Zhang, and Chen-Hui Jin. Practical security against differential and linear cryptanalysis for sms4-like cipher. *Journal of Networks*, 8(8):1689–1693, 2013.

[19] Andrey Bogdanov and Kyoji Shibutani. Generalized feistel networks revisited. *Designs, Codes and Cryptography*, 66(1):75–97, 2013. ISSN 1573-7586. doi: 10.1007/s10623-012-9660-z. URL http://dx.doi.org/10.1007/s10623-012-9660-z.

[20] Shay Gueron and Nicky Mouha. Simpira v2: A family of efficient permutations using the aes round function. Cryptology ePrint Archive, Re-

port 2016/122, 2016. URL http://eprint.iacr.org/2016/122.

[21] Thierry P. Berger, Marine Minier, and Gaël Thomas. *Extended Generalized Feistel Networks Using Matrix Representation*, pages 289–305. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. ISBN 978-3-662-43414-7. doi: 10.1007/978-3-662-43414-7_15. URL http://dx.doi.org/10.1007/978-3-662-43414-7_15.

[22] Lei Zhang and Wenling Wu. Differential analysis of the extended generalized feistel networks. *Information Processing Letters*, 114(12):723 – 727, 2014. ISSN 0020-0190. doi: http://dx.doi.org/10.1016/j.ipl.2014.07.001. URL http://www.sciencedirect.com/science/article/pii/S0020019014001318.

[23] Jingmei Liu, Baodian Wei, Xiangguo Cheng, and Xinmei Wang. An aes s-box to increase complexity and cryptographic analysis. In *19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 1 (AINA papers)*, volume 1, pages 724–728 vol.1, March 2005. doi: 10.1109/AINA.2005.84.

[24] Philip Hawkes and Gregory G. Rose. *Guess-and-Determine Attacks on SNOW*, pages 37–46. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. ISBN 978-3-540-36492-4. doi: 10.1007/3-540-36492-7_4. URL http://dx.doi.org/10.1007/3-540-36492-7_4.

[25] Dai Watanabe, Alex Biryukov, and Christophe De Cannière. *A Distinguishing Attack of SNOW2.0 with Linear Masking Method*, pages 222–233. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. ISBN 978-3-540-24654-1. doi: 10.1007/978-3-540-24654-1_16. URL http://dx.doi.org/10.1007/978-3-540-24654-1_16.

[26] Olivier Billet and Henri Gilbert. *Resistance of SNOW 2.0 Against Algebraic Attacks*, pages 19–28. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005. ISBN 978-3-540-30574-3. doi: 10.1007/978-3-540-30574-3_3. URL http://dx.doi.org/10.1007/978-3-540-30574-3_3.

[27] Masayuki Kanda, Youichi Takashima, Tsutomu Matsumoto, Kazumaro Aoki, and Kazuo Ohta. *A Strategy for Constructing Fast Round Functions with Practical Security Against Differential and Linear Cryptanalysis*, pages 264–279. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999. ISBN 978-3-540-48892-7. doi: 10.1007/3-540-48892-8_21. URL http://dx.doi.org/10.1007/3-540-48892-8_21.

[28] Sangwoo Park, Soo Hak Sung, Sangjin Lee, and Jongin Lim. *Improving the Upper Bound on the Maximum Differential and the Maximum Linear Hull Probability for SPN Structures and AES*, pages 247–260. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. ISBN 978-3-540-39887-5. doi: 10.1007/978-3-540-39887-5_19. URL http://dx.doi.org/10.1007/978-3-540-39887-5_19.

[29] Jongsung Kim, Seokhie Hong, Jaechul Sung, Sangjin Lee, Jongin Lim, and Soohak Sung. *Impossible Differential Cryptanalysis for Block Cipher Structures*, pages 82–96. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. ISBN 978-3-540-24582-7. doi: 10.1007/978-3-540-24582-7_6. URL http://dx.doi.org/10.1007/978-3-540-24582-7_6.

[30] Lars Knudsen and David Wagner. *Integral Cryptanalysis*, pages 112–127. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002. ISBN 978-3-540-45661-2. doi: 10.1007/3-540-45661-9_9. URL http://dx.doi.org/10.1007/3-540-45661-9_9.

[31] Yongjin Yeom. Integral cryptanalysis and higher order differential attack. *Trends in Mathematics (ICMS)*, 8(1):101–118, 2005.

[32] Lars R. Knudsen. *Truncated and higher order differentials*, pages 196–211. Springer Berlin Heidelberg, Berlin, Heidelberg, 1995. ISBN 978-3-540-47809-6. doi: 10.1007/3-540-60590-8_16. URL http://dx.doi.org/10.1007/3-540-60590-8_16.

[33] Shiho Moriai, Makoto Sugita, Kazumaro Aoki, and Masayuki Kanda. *Security of E2 against Truncated Differential Cryptanalysis*, pages 106–117. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000. ISBN 978-3-540-46513-3. doi: 10.1007/3-540-46513-8_8. URL http://dx.doi.org/10.1007/3-540-46513-8_8.

Mahmood Deypir received his Ph.D. degree in 2011 and M.Sc degree in 2006 both from Shiraz University. He is currently assistant professor at Computer and Information Technology department at Shahid Sattari University of Science and Technology. His research interests include Data Mining and Cyber space Security. He has published a number of papers in ISI journals and international conferences.

Yosef Pourebrahim received his Bachelor of Electric Engineering from Faculty of Engineering, Islamic Azad University (IAU), Ardabil Brunch, Iran, in 2003. He received his Master of Science degree from Faculty of Information and Communication Technology, Imam Hossein University, Tehran, Iran, in 2005, where he was working on cryptography systems until 2010. Currently, he is a Ph.D. student in IAU, science and research branch, Tehran, Iran. His research interest is in cryptography systems and signal processing.