Related-Key Differential Cryptanalysis of Reduced-Round ITUBee

Xiaoming Tang[1], Weidong Qiu[1], Zheng Gong[2], Zheng Huang[1] and Jie Guo[1]
School of Information Security Engineering & Shanghai Jiao Tong University [1]
School of Computer Science & South China Normal University[2]

December 24, 2014

Abstract

ITUBEE is a software oriented lightweight block cipher, which is first proposed at LightSec 2013. The cipher is especially suitable for limited resource application, such as sensor nodes in wireless sensor networks. To evaluate the security level of the cipher, we perform differential attacks on ITUBEE reduced to 10 rounds and 11 rounds with the time complexities $2^{65.97}$ and $2^{79.03}$, respectively. To our best knowledge, our analysis is the first related-key differential cryptanalysis on the security of 10-round ITUBEE.

Keywords : lightweight cryptography, block cipher, cryptanalysis, truncated differential, related-key attack

1 Introduction

Lightweight cryptography is concerned for the security and privacy demands of the applications used in the limited resource devices, such as sensor nodes and RFID tags , or low-end embedded software systems. In the past decades, lots of new lightweight primitives are published for the different applications, including the block cipher PRESENT [1], KATAN and KTANTAN [2], PRINTcipher [3], Klein [4], Hummingbird-2 [5], LED [6], Piccolo [7]. Most of the algorithms are hardware oriented, with using a combination of a small S-box and simplistic linear layer, or using shift register based construction.

ITUBEE [8] is a new software oriented lightweight block cipher, designed for application of resource-constrained devices which include a micro-controller and use a limited battery power such as sensor nodes in wireless sensor networks. To reduce the energy consumption of cipher, the designer of ITUBEE use a Feistel structure without a key schedule. However, the nonexistence of a key schedule or the usage of alternating keys compromise the security of the cipher especially for related-key attack. Trying to make ITUBEE more secure, designers use a new approach in round function, which is named AKF (A Key alternating Feistel

scheme) [9]. Using AKF structure, ITUBEE has a better performance than most of lightweight cipher and strong security margin against the linear and differential cryptanalysis. In [10], the author make a security analysis of key-alternating Feistel ciphers by considering the round function as public random function where the adversary is allowed to query in black-box way, however, we did not follow this method in this paper.

The original security analysis of ITUBEE shows that, the upper bound of differential trail probability for 8-round ITUBEE is 2^{-85} in ordinary differential attacks [8]. Also the author of ITUBEE claims that the related-key attack is not applicable to the 10-round ITUBEE cipher [8, 9]. In [9], the author make more precise calculation on the 5-round differential trail, the probability of this trail is $2^{-107.58}$, which can not be used in a differential attack. More details are listed in Table 1.

In [11], the author claims that they find a deterministic related-key differential distinguisher up to 8 rounds of ITUBEE, the complexity data of the attacks is not available as we failed to download the full version of the paper, thus we use a question mark for the data which is not available for reference in the Table 1.

Contribution. By using the related-key techniques, we construct a 10-round truncated differential trail with the probability $2^{-69.65}$. Based on this related-key differential trail, we mount attacks on 10-round and 11-round ITUBEE with the time complexities $2^{65.97}$ and $2^{79.03}$, respectively. Detail results are given in Table 1. To best of our knowledge, this is the best cryptanalysis result on the security of reduced-round ITUBEE.

Table 1: Comparison of attacks on ITUBEE

rounds	Prob.	data	time	memory	attack method	ref.e
8	$< 2^{-85}$	N/A	N/A	N/A	differential	[8]
8	$< 2^{-85}$	N/A	N/A	N/A	Related-key	[8, 9]
2	?	?	$2^{79.678}$	?	biclique	[8, 9]
5	$2^{-107.58}$	N/A	N/A	N/A	differential	[9]
8	?	?	?	?	related-key	[11]
10	$2^{-69.65}$	$2^{73.97}$	$2^{73.97}$	negligible	related-key	this paper
	$2^{-69.65}$	$2^{65.97}$	$2^{65.97}$	2^{10}	related-key	this paper
11	$2^{-69.65}$	$2^{71.65}$	$2^{79.04}$	negligible	related-key	this paper
	$2^{-69.65}$	$2^{63.65}$	$2^{79.03}$	2^{10}	related-key	this paper

The remainder of this paper is organized as follows. Part 2 starts with a brief description of ITUBEE. Part 3 first present a high-probability truncated differential trail with related-key techniques, then build a key-recovery attack by using this differential trail, and present its analysis results of computational complexities. And we conclude this paper in part 4.

2 Brief introduction of ITUBEE

2.1 Notation

In this paper, we use following notations:

$A\|\|B$	: Concatenation of two bit strings A and B.
P	: 80-bit plaintext.
P_L	: The left half of the plaintext.
P_R	: The right half of the plaintext.
C	: 80-bit ciphertext.
C_L	: The left half of the ciphertext.
C_R	: The right half of the ciphertext.
K	: 80-bit master key.
K_L	: The left half of the master key.
K_R	: The right half of the master key.
RC_i	: The round constant used in the i-th round.

2.2 Definition of ITUBEE

ITUBEE is a lightweight cipher with AKF structure. The cipher is consisting of 20 iterative rounds with two alternating round keys, block size and key length of the cipher are both 80 bits [8, 9]. The encryption process is given in Algorithm 1 and pictured in Figure 1.

For the purpose of less memory usage and energy consumption requirement, the whitening and round keys of ITUBEE are derived from the master key directly. In the encryption algorithm, $(K_L||K_R)$ and $(K_R||K_L)$ are used as whitening keys at the first round and the final round, respectively. K_R is used as round keys for odd rounds while K_L is used for even rounds. [8]

Algorithm 1 presents the encryption process of the cipher.

Algorithm 1 ITUBEE encryption process

1: $X_1 \leftarrow P_L \oplus K_L$ and $X_0 \leftarrow P_R \oplus K_R$
2: for $i = 1 \ldots 20$ do
   (a) if $i \in 1, 2, 3, \ldots, 19$
   $RK \leftarrow K_R$
   (b) else
   $RK \leftarrow K_L$
   (c) $X_{i+1} \leftarrow X_{i-1} \oplus F(L(RK \oplus RC_i \oplus F(X_i)))$
3: $C_L \leftarrow X_{20} \oplus K_R$ and $C_R \leftarrow X_{21} \oplus K_L$

The definitions of the function used in the algorithm are:

- $F(X) = S(L(S(X)))$.

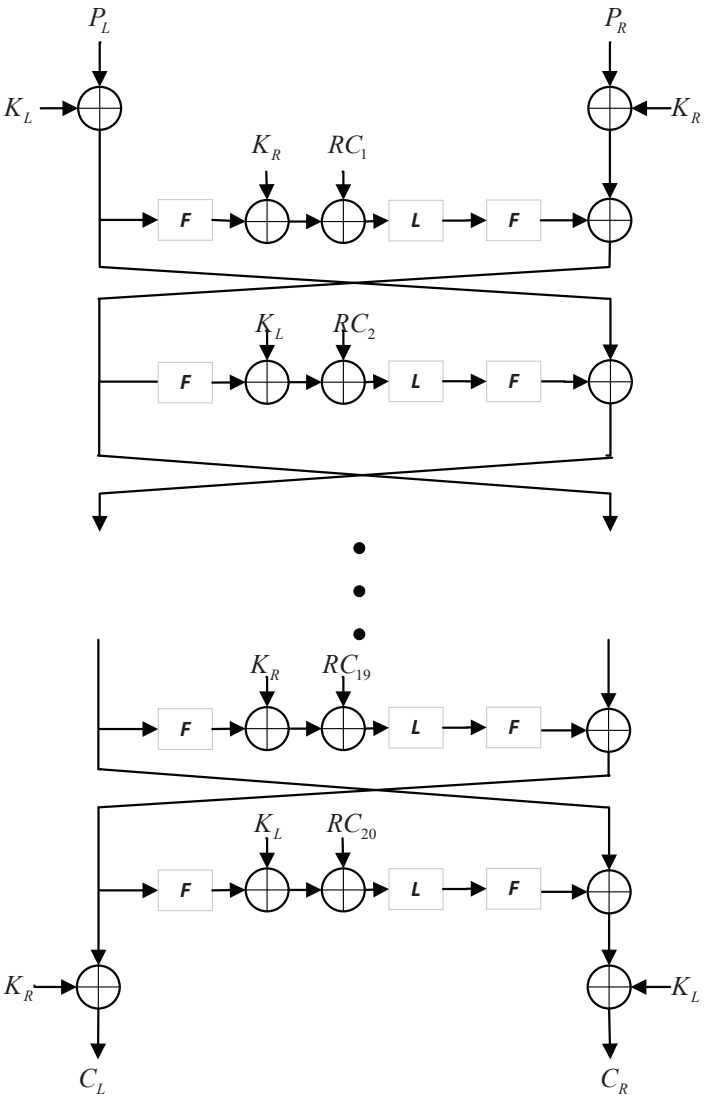

Figure 1: ITUBEE encryption algorithm

- $S(a||b||c||d||e) = s[a]||s[b]||s[c]||s[d]||s[e]$, where a, b, c, d, e are 8-bit values and s is the S-box used in AES [12].
- $L(a||b||c||d||e) = (e \oplus a \oplus b)||(a \oplus b \oplus c)||(b \oplus c \oplus d)\ ||(c \oplus d \oplus e)||(d \oplus e \oplus a)$.

For more details of ITUBEE, the readers can refer to the original paper [8, 9].

3 Truncated Differential Attacks on ITUBEE with Related-Key techniques

Since most of ciphers use S-box as non-linear part of the encryption, it is natural to use truncated differential attack to analysis the security level of the cipher. The truncated differential attack [13, 14] is a powerful tool for security evaluation of block cipher, which is firstly introduced by Knudsen in 1994. The main ideal of this tool is to consider the sets of difference instead of the concrete values of difference. The related-key attack [15] provides more flexible attack scenario to adversary. The attackers can obtain the encryption of certain plaintext not only under the original session key which we try to recover, but also under some associated keys. Under this circumstance, key alternating structure used in ITUBEE cipher give us an opportunity to get a desired attack result by using truncated differential attack with related-key techniques.

ITUBEE is a byte-oriented primitive, in which both confusion and diffusion layer are provided only by inter-byte operations. Hence we can find the high-probability differential trail by applying byte-based automatic search method. The search result shows there exists a 10-round iterative related-key differential trail. Using this trail, we perform 10-round and 11-round related-key attack on ITUBEE cipher.

3.1 Construction of 10-round differential trail

In this section, we present some propositions that help us to construct high-probability differential trail for ITUBEE. Since the addition of whitening key and round constant will not affect the probability of differential trail, we ignore those operation during the differential trail construction and consider its effect later.

Proposition 1. The minimal number of active S-boxes is 4 in one round with related-key.

Proof: The byte-based exhaust search proved that the minimal number of active S-boxes is 4 in round function under related-key scenario. Let ζ, η α, β, γ donate 8-bit differences and 0 donate 8 bits which don't have any difference, one of the active difference trail with four active S-boxes can be as following:

$$(\zeta 0000) \xrightarrow{S} (\eta 0000) \xrightarrow{L} (\eta\eta 00\eta) \xrightarrow{S} (\alpha\beta 00\gamma)$$
$$\xrightarrow{ARK} (00000) \xrightarrow{S} (00000) \xrightarrow{L} (00000) \xrightarrow{S} (00000)$$

In this differential trail, we use related-key techniques with the key difference form $(\alpha\beta 00\gamma)$ to cancel the output difference of first F function in the add-round-key (ARK) operation. Thus in this case, we have only 4 active S-boxes active for one round.

Proposition 2. The maximal Probability of truncated differential with related-key techniques for one round is $2^{-13.93}$.
Proof: As we know, less number of active S-boxes leads to a higher probability of differential trail. From proposition 1, we exhibit a related-key differential trail with minimal number of active S-boxes. To find the maximal probability of this truncated differential trail, we use the difference distribution table to calculate the concrete probability value of the trail. We define this truncated differential probability as a collection of differential trails from input of the form $(\zeta 0000)$ to output of the form $(\alpha\beta 00\gamma)$, which ζ takes all possible values, consequently the probability of this truncated differential trail can be calculated as follows.

$$Pr((\zeta 0000) \xrightarrow{F} (\alpha\beta 00\gamma))$$
$$= \sum_{\zeta} \sum_{\eta} Pr(\zeta \xrightarrow{S} \eta) \times Pr(\eta \xrightarrow{S} \alpha) \times Pr(\eta \xrightarrow{S} \beta) \times Pr(\eta \xrightarrow{S} \gamma)$$

In the differential trail, $(\eta\eta 00\eta)$ is an intermediate state of differential trails, which takes all possible values. In this case, we can summate probabilities of all trails from$(\zeta 0000)$ to $(\alpha\beta 00\gamma)$, where ζ represent all possible differential values and $(\alpha\beta 00\gamma)$ is a fixed differential value. The calculation result leads to a maximal probability of $2^{-13.93}$ when the differential output α, β and γ take the same value.

Proposition 3. The maximal probability of 10-round related-key differential trail is $2^{-69.65}$.
Proof: The 10-round related-key differential trail with maximal probability is depicted in Figure 2, the trail relies on the differential characteristic which is presented in proposition 2. In this case, the differential trail is totally iterative, consequently it is composed of one round differential characteristic with probability $2^{-13.93}$ and one round with non-active byte, its probability is 1. Thus the maximal probability of 10-round related-key differential trail is $(2^{-13.93})^5 = 2^{-69.65}$.

3.2 Attack 10-round reduced ITUBEE

The procedure of our related-key differential attack is as follows:

Step 1 In this step, we are trying to find the right ciphertext pair which conform the related-key differential trail. Randomly select plaintext pair such that the plaintext difference satisfy the input of truncated differential trail. Perform the following step for each plaintext pair until we find sufficient right ciphertext pairs.

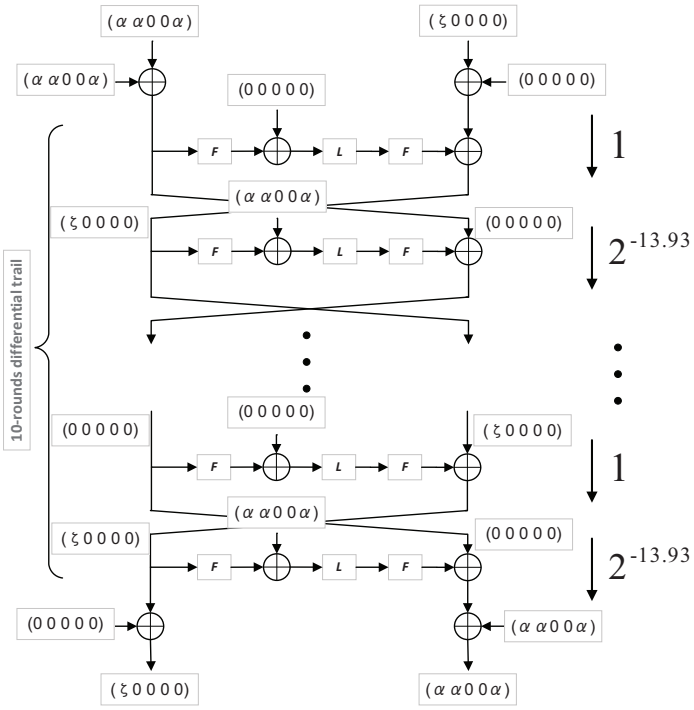

Figure 2: 10-round truncated differential trail with related-key techniques

- ask for the encryption of the plaintext pair under the related-key scenaro where key difference is of the form $(\alpha\alpha 00\alpha 00000)$.
- check the difference of corresponding ciphertext pair, if the difference is of the form $(\zeta 0000\alpha\alpha 00\alpha)$, we consider it is a right ciphertext pair.

Step 2. Once sufficient right ciphertext pair satisfying the differential trail appears, we can identify the correct partial key using these right ciphertext pairs.

- Initialize a counter array of the key candidates for K_R
- Guess 40-bit value of K_R, and calculate the first F function output of the last round using these partial key candidates under related-key scenario.
- If the difference of the first F function output conform the form $(\alpha\alpha 00\alpha)$, we increment the counter of the key candidates. Assuming we verify the key candidates using two right cipher pairs, the correct partial key will be counted at least twice.

Step 3. After finding the K_R, the rest 40 key bits K_L then can be brute-forced in 2^{40} encryptions.

We notice that position changes of ζ is not important due to the symmetric of L layer. That means we can use these multiple differential trails for key recovery. All these 5 differential trails will provide the information of 40-bit value of K_R. We can exploit this characteristic to improve successful rate.

As the probability of this related-key differential trail is $2^{-69.65}$, it takes $2^{70.65}$ chosen-plaintext pairs to drive two right ciphertext pairs on average. Thus the required number of chosen-plaintext pairs here is $2^{70.65} \times 5 \approx 2^{72.97}$. That means the data complexity required for completed 80-bit key recovery is $2^{73.97}$ plaintexts.

We need $2^{73.97}$ encryptions in the right pairs searching stage. Further more, in the key recovery stage we need to calculate transformation of 8 S-boxes out of 20 S-boxes in the final round. That is equivalent to around $\frac{8}{10 \times 20} \times 2 \times 2^{40} \times 2 \approx 2^{37.36}$ encryptions. Therefore time complexity of attack is $2^{73.97} + 2^{37.36} \times 5 + 2^{40} \approx 2^{73.97}$ encryptions.

We can use a structure to reduce the data and time complexities. On plaintext side, we submit a structure such that the position with active S-boxes in the first round get sufficiently values. Let byte 0 of P_R be active with unknown difference ζ, byte 0,1,4 of P_L be active with fixed difference α, such structure contains 2^9 plaintexts, which can provide 2^{16} plaintext pairs for truncated differential attack. We expect to obtain two right ciphertext pairs if we use $\frac{2^{70.65}}{2^{16}} \approx 2^{54.65}$ structure, i.e., $2^{63.65}$ plaintexts, with the memory complexity increasing to 2^{10} plaintexts. Therefore data complexity for full key recovery is $2^{63.65} \times 5 \approx 2^{65.97}$, and time complexity of attack is $2^{65.97} + 2^{37.36} \times 5 + 2^{40} \approx 2^{65.97}$ encryptions.

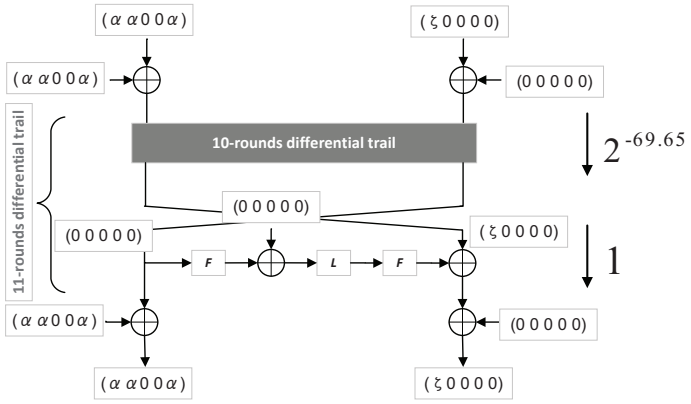

Figure 3: 11-round truncated differential trail with related-key

3.3 Extend to 11-round attack

We extend 10-round differential trail to 11 rounds by adding an extra trivial differential trail, see Figure 3. The form of extra differential trail is (00000ζ0000) to (00000ζ0000) with the probability 1. It is expected to take $2^{70.65}$ plaintext pairs, i.e., $2^{71.65}$ plaintexts to find two right ciphertext pairs on average for key recovery.

To attack the 11-round cipher, we need to guess full 80-bit keys instead of partial keys and the verification will happen in the second round from bottom. To identify the correct key, it is necessary to verify the difference output of F function of the second round from the bottom under related-key scenario. It leads to calculation of 28 active Sboxes, thus we know it takes $\frac{28}{11 \times 20} \times 2 \times 2^{80} \times 2 \approx 2^{79.03}$ encryptions. Therefore the time complexity is $2^{71.65} + 2^{79.03} \approx 2^{79.04}$ encryptions.

Similarly, if we use a structure specified in 10-round attack, the data complexity can be reduced to $2^{63.65}$ plaintexts. And time complexity is aproximately $2^{63.65} + 2^{79.03} \approx 2^{79.03}$ encryptions, with memory complexity increasing to 2^{10} plaintexts.

3.4 The sucessful rate of the proposed attacks

An important measure for the success rate of differential attack is S/N ratio, i.e., signal-to-noise ratio. Biham and shamir observed a strong relationship between the S/N ratio and the success rate, and they observed that high values of S/N ratio(i.e., significantly larger than one) lead to a small need in right pairs. [16] S/N is in fact is a ratio for number of times the right key is counted and number of times an average key is counted.

In our 10-round attack, we assume the cipher is a random permutation

ideally. The number of key candidates is 2^{40}, the average number of target key candidates suggested by each pair is 2^{16}, then the S/N ratio of 10-round attack becomes:

$$S/N = \frac{2^{40} \times 2^{-69.65}}{2^{16} \times 2^{-72}} = 2^{26.35} > 1$$

For our 11-round related-key attack, the successful rate is also large enough for key recovery. The number of key candidates of 11-round attack is 2^{80}, while the S/N ratio will be $2^{66.35}$.

4 Conclusion

We applied the related-key differential cryptanalysis to lightweight cipher ITUBEE reduced to 10 rounds and 11 rounds respectively. In 10-round related-key attack, the numbers of required chosen plaintexts is $2^{65.97}$ with 2^8 plaintexts memory, and time complexity is about $2^{65.97}$. For 11-round ITUBEE, the complexity of attack is about $2^{79.03}$ encryptions and it requires $2^{63.65}$ chosen plaintexts, with 2^8 plaintexts memory.

In practical, our attack will not create a real threat for ITUBEE, but our result show a potential weakness of this cipher. And this weakness might be useful for further cryptanalysis of this new lightweight block cipher. Since the security of a new cipher must be analyzed thoroughly before practical implementation, our result opens an insight on ITUBEE on its simple key schedule design.

References

[1] A. Bogdanov, L. Knudsen, G. Leander, C. Paar, A. Poschmann, M. Robshaw, Y. Seurin, and C. Vikkelsoe, "Present: An ultra-lightweight block cipher," in *Cryptographic Hardware and Embedded Systems - CHES 2007*, ser. Lecture Notes in Computer Science, P. Paillier and I. Verbauwhede, Eds. Springer Berlin Heidelberg, 2007, vol. 4727, pp. 450–466.

[2] C. De Canniere, O. Dunkelman, and M. Knežević, "KATAN and KTANTAN A Family of Small and Efficient Hardware-Oriented Block Ciphers," in *Cryptographic Hardware and Embedded Systems - CHES 2009*, ser. Lecture Notes in Computer Science, C. Clavier and K. Gaj, Eds. Springer Berlin Heidelberg, 2009, vol. 5747, pp. 272–288.

[3] L. Knudsen, G. Leander, A. Poschmann, and M. Robshaw, "PRINTcipher: A Block Cipher for IC-Printing," in *Cryptographic Hardware and Embedded Systems, CHES 2010*, ser. Lecture Notes in Computer Science, S. Mangard and F.-X. Standaert, Eds. Springer Berlin Heidelberg, 2010, vol. 6225, pp. 16–32.

[4] Z. Gong, S. Nikova, and Y. Law, "KLEIN: A New Family of Lightweight Block Ciphers," in *RFID. Security and Privacy*, ser. Lecture Notes in Computer Science, A. Juels and C. Paar, Eds. Springer Berlin Heidelberg, 2012, vol. 7055, pp. 1–18.

[5] D. Engels, M.-J. Saarinen, P. Schweitzer, and E. Smith, "The Hummingbird-2 Lightweight Authenticated Encryption Algorithm," in *RFID. Security and Privacy*, ser. Lecture Notes in Computer Science, A. Juels and C. Paar, Eds. Springer Berlin Heidelberg, 2012, vol. 7055, pp. 19–31.

[6] J. Guo, T. Peyrin, A. Poschmann, and M. Robshaw, "The LED Block Cipher," in *Cryptographic Hardware and Embedded Systems CHES 2011*, ser. Lecture Notes in Computer Science, B. Preneel and T. Takagi, Eds. Springer Berlin Heidelberg, 2011, vol. 6917, pp. 326–341.

[7] K. Shibutani, T. Isobe, H. Hiwatari, A. Mitsuda, T. Akishita, and T. Shirai, "Piccolo: An Ultra-Lightweight Blockcipher," in *Cryptographic Hardware and Embedded Systems CHES 2011*, ser. Lecture Notes in Computer Science, B. Preneel and T. Takagi, Eds. Springer Berlin Heidelberg, 2011, vol. 6917, pp. 342–357.

[8] F. Karakoç, H. Demirci, and A. E. Harmancı, "ITUbee: A Software Oriented Lightweight Block Cipher," in *Lightweight Cryptography for Security and Privacy*, ser. Lecture Notes in Computer Science, G. Avoine and O. Kara, Eds. Springer Berlin Heidelberg, 2013, vol. 8162, pp. 16–27.

[9] Karakoç, F and Demirci, H and Harmancı, AE, "AKF: A key alternating Feistel scheme for lightweight cipher designs," *Information Processing Letters*, vol. 115, pp. 359–367, 2014.

[10] R. Lampe and Y. Seurin, "Security analysis of key-alternating feistel ciphers." *IACR Cryptology ePrint Archive*, vol. 2014, p. 151, 2014.

[11] H. Soleimany, "Self-similarity cryptanalysis of the block cipher ITUbee," *IET Information Security*, 2014.

[12] J. Daemen and V. Rijmen, *The design of Rijndael: AES-the advanced encryption standard*. Springer, 2002.

[13] L. Knudsen, "Truncated and higher order differentials," in *Fast Software Encryption*, ser. Lecture Notes in Computer Science, B. Preneel, Ed., vol. 1008. Springer Berlin Heidelberg, 1995, pp. 196–211.

[14] S. Lee, S. Hong, S. Lee, J. Lim, and S. Yoon, "Truncated Differential Cryptanalysis of Camellia," in *Information Security and Cryptology ICISC 2001*, ser. Lecture Notes in Computer Science, K. Kim, Ed. Springer Berlin Heidelberg, 2002, vol. 2288, pp. 32–38.

[15] A. Biryukov and D. Khovratovich, "Related-Key Cryptanalysis of the Full AES-192 and AES-256," in *Advances in Cryptology ASIACRYPT 2009*, ser. Lecture Notes in Computer Science, M. Matsui, Ed. Springer Berlin Heidelberg, 2009, vol. 5912, pp. 1–18.

[16] Biham, Eli and Shamir, Adi, *Differential cryptanalysis of the data encryption standard*. Springer, 1993, vol. 28.