An Improved Truncated Differential Cryptanalysis of KLEIN

Shahram Rasoolzadeh[1,2], Zahra Ahmadian[1,2], Mahmood Salmasizadeh[2], and Mohammad Reza Aref[1]

[1] Information Systems and Security Lab (ISSL), Department of Electrical Engineering,
[2] Electronic Research Institute,
Sharif University of Technology, Tehran, Iran
{sh_rasoolzadeh,ahmadian}@ee.sharif.edu,{salmasi,aref}@sharif.edu

Abstract. KLEIN is a family of lightweight block ciphers which proposed at RFIDSec 2011 by Gong et al. It has a 64-bit state and 64, 80 or 96-bit key size which introduce its version. It uses a 4-bit S-Box combined with Rijndael's byte-oriented MixColumn transformation for each round. This approach allows compact implementations of KLEIN in both low-end software and hardware. Its simplicity attracts the attention of cryptanalysts, and several security analyses have been published. The best of these was represented by Lallemand et al. which was a truncated differential attack. They could attack up to 12, 13 and 14 rounds out of total number of 12, 16 and 20 rounds for KLEIN-64, -80 and -96, respectively. In this paper, by finding new truncated differential paths and improving key recovery method we present two new truncated differential attacks on KLEIN, which recover the full secret key with better time and data complexities for 12, 14 and 15 rounds for KLEIN-64, -80 and -96, respectively.

Keywords: KLEIN, Lightweight Block Cipher, Truncated Differential Cryptanlysis.

1 Introduction

Lightweight block ciphers are those that are specially designed for constrained environments. Due to the implementation considerations in such environments the key size of the cipher is typically 64 or 80 bits. In order to answer the requirements of a large number of applications, like RFID and wireless sensor networks. Through these last years an enormous amount of promising such primitives has been proposed that KLEIN [1] is one of them. Correctly evaluating the security of these proposals has become a primordial task that merits all the attention from the community.

KLEIN family of lightweight block ciphers is proposed by Gong et al. in RFIDSec 2011 [1]. It supports three key sizes of 64, 80 and 96 bits, with 12, 16 and 20 rounds respectively. KLEIN makes use of the combination of 4-bit S-Boxes with AES MixColumn in an SPN structure. Such a combination allows compact and low memory implementation in software and hardware, which results show that this cipher is utilizable in constrained-resource environments in the viewpoint of the performance.

But from the security point of view, although some basic evaluations have been carried out on KLEIN in [1], its real security level is not determined without further external analysis. In fact, this type of combination poses serious security risks to KLEIN, since its publication, several cryptanalysts have been interested in its analysis and some results on round-reduced versions have been published [2–7]. Biclique analyses [4, 5] can be remarked that this analyses require to perform an exhaustive search on the whole key and that the acceleration factors are very small. So far, except biclique analyses the highest number of attacked rounds is full 12-round in the 64-bit version, 13 out of 16 in the 80-bit version and 14 out of 20 in the 96-bit version, which was discovered and exploited by Lallemand et al. in FSE 2014 [7].

In this paper, by finding new truncated differential paths and improving key recovery method we present two new truncated differential attacks, which the first attack can attacks up to 14 rounds, and the second one up to 15 rounds. They can be applied to the full 12-round KLEIN-64 with a (time, data) complexities of ($2^{55.7}$, $2^{48.6}$) or ($2^{58.8}$, $2^{45.5}$), respectively. To 14-round KLEIN-80 time and data complexities are ($2^{75.9}$, $2^{60.6}$) or ($2^{78.9}$, $2^{57.5}$), respectively. Also for 15-round KLEIN-80 time and data complexities are ($2^{92.9}$, $2^{63.5}$). The complexities of existing attacks and ours are summarized in Table 1.

This paper is organized as follows: Section 2 presents a brief description of KLEIN. In Section 3, new truncated differential paths, the outline of the key recovery attack on KLEIN with all details and its complexities evaluations are represented. Finally, Section 4 concludes this paper.

Table 1. Summary of cryptanalytic results on KLEIN

Vrsion	Rounds	Time	Data	Memory	Attack Type	Ref.
KLEIN-64	7	$2^{45.5}$	$2^{34.3}$	2^{32}	Integral	[2]
	8	$2^{46.8}$	2^{32}	2^{16}	Truncated	[2]
	8	2^{35}	2^{35}	-	Truncated	[3]
	10	2^{62}	1	2^{60}	PC MitM	[6]
	12	$2^{62.8}$	2^{39}	$2^{4.5}$	Biclique	[4]
	12	2^{57}	$2^{54.5}$	2^{16}	Truncated	[7]
	12	$\mathbf{2^{55.7}}$	$\mathbf{2^{48.6}}$	$\mathbf{2^{32}}$	**Truncated**	**Sec. 4**
KLEIN-80	8	$2^{77.5}$	$2^{34.3}$	2^{32}	Integral	[2]
	11	2^{74}	2	2^{74}	PC MitM	[6]
	13	2^{76}	2^{52}	2^{16}	Truncated	[7]
	14	$\mathbf{2^{75.9}}$	$\mathbf{2^{60.6}}$	$\mathbf{2^{32}}$	**Truncated**	**Sec. 4**
	16	2^{79}	2^{48}	2^{60}	Biclique	[5]
KLEIN-96	13	2^{94}	2	2^{82}	PC MitM	[6]
	14	$2^{89.2}$	$2^{58.4}$	2^{16}	Truncated	[7]
	15	$\mathbf{2^{92.9}}$	$\mathbf{2^{63.5}}$	$\mathbf{2^{32}}$	**Truncated**	**Sec. 4**
	20	$2^{95.18}$	2^{32}	2^{60}	Biclique	[5]

2 Description of KLEIN

The block cipher KLEIN use a Substitution-Permutation Network that operates on 64-bit blocks. It has 3 versions, denoted by KLEIN-64, KLEIN-80 and KLEIN-96 that introduce its key size and they have 12, 16 and 20 rounds, respectively. Each round is composed of 4 layers: AddRoundKey, SubNibbles, RotateNibbles and MixNibbles. In AddRoundKey layer, the entering state to a round is xored with the round-key. The output of AddRoundKey layer is divided to 16 nibbles, and each nibble passed through a same 4×4 S-Box, it isSubNibbles layer. Reason for this choice by designers is that a byte-wise S-Box needs more implementation costs and memory than a nibble-wise S-Box. KLEIN's S-Box is represented in Table 2.

Table 2. Summary of cryptanalytic results on KLEIN

Input	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
Output	7	4	a	9	1	f	b	0	c	3	2	6	8	e	d	5

After SubNibbles layer in RotateNibbles layer, state rotates two bytes to the left and finally MixNibbles applies AES's MixColumn transformation to each half of the state. Unlike to AES, last round's MixNibbles layer is not omitted. After last round, an additional AddRoundKey layer is proceed, so the encryption routine requires one more key than the number of rounds. Structure of a round of KLEIN is shown in Figure 1. $X^{(r)}$ and $K^{(r)}$ are the input state and the subkey of round r, respectively.

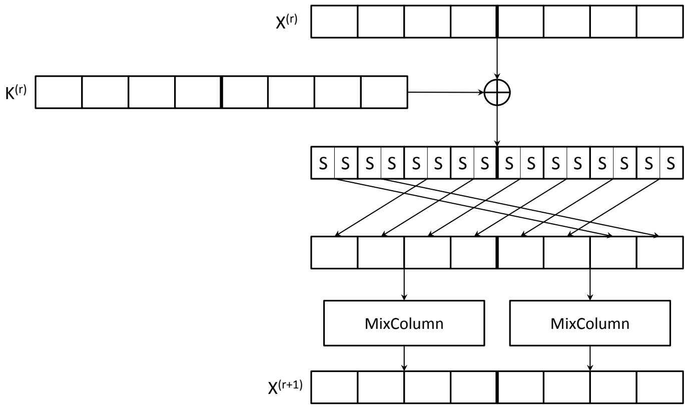

Fig. 1. Structure of one round of KLEIN

Let us focus on AES MixColumn, which works according to the following matrix multiplication in $GF(2^8)$ with the irreducible polynomial $x^8+x^4+x^3+x+1$:

$$M = \begin{pmatrix} \mathsf{02\ 03\ 01\ 01} \\ \mathsf{01\ 02\ 03\ 01} \\ \mathsf{01\ 01\ 02\ 03} \\ \mathsf{03\ 01\ 01\ 02} \end{pmatrix}. \tag{1}$$

For reminding, multiplication of 02 by $x \in GF(2^8)$ can be performed as follows:

$$\mathsf{02} \times x = \begin{cases} x \ll 1 & \text{if MSB}(x) = 0 \\ x \ll 1 \oplus \mathsf{0x1b} & \text{if MSB}(x) = 1 \end{cases}, \tag{2}$$

where $x \ll n$ means shifting x, n bits to left and MSB is the most significant bit. Also the multiplication by 03 is equal to:

$$\mathsf{03} \times x = x \oplus \mathsf{02} \times x \tag{3}$$

These descriptions of finite field multiplications will be more useful in explaining the MixNibbles properties in the next section. It is better to note that only last layer (MixNibbles) is byte-wise while the others can be seen as nibble-wise.

The *Key Schedule* of KLEIN is designed under implementation considerations. The round-keys are computed from the MasterKey with the KeySchedule algorithm that follows a Feistel scheme. The round keys $K^{(r)}$, $r = 1, \cdots, Nbr$ (Nbr is number of rounds), and the final whitening key $K^{(Nbr+1)}$ is generated as follows. First, the master key MK is stored in a key register as $K^{(1)}$. Then the following steps are iteratively applied to MK to generate Nbr more subkeys:

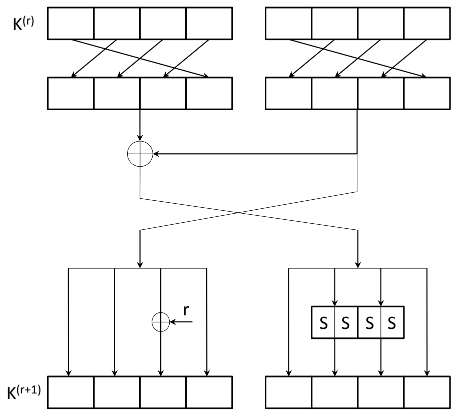

Fig. 2. Key schedule of one round of KLEIN-64

1. Rotate the two halves of the key state to left one byte each.
2. Swap the two halves by a Feistel-like structure.
3. In left half of key state, xor 3rd byte from left with round counter r
4. In Right half of key state, substitute 2nd and 3rd bytes using four KLEIN S-Boxes.

At the end of round r, the content of the key register is $K^{(r)}$. Figure 2 shows one round of the key schedule for KLEIN-64.

3 Truncated Differential Cryptanalysis of KLEIN

In this section, we will introduce two new truncated differential paths. Then we will introduce key recovery method which is improved version of key recovery method used in [7]. Finally, the complexities of our attacks will be represented.

Proposition 1. [2, 3, 7] If the eight nibbles entering MixNibbles are of the form $0X0X0X0X$, where the wild-card X represents any 4-bit value, then the output is of the same form if and only if the MSB of the 4 lower nibbles all have the same value. This case occurs with probability 2^{-3}.

Proposition 2. If the eight nibbles entering MixNibbles are of the form $0X0X0X$ $0X$, then the output is the form of $00000X0X$ or $0X0X0000$ with probability of 31×2^{-15}.

Proposition 1 explained enough in previous cryptanalyses, especially in [7], so we don't speak more about that. The proof of Proposition 2 is as follow.

Proof. Consider $0A0B0C0D$ be the eight nibbles entering MixNibbles and $00000E$ $0F$ be the eight output nibbles. Also consider that $X = x_0x_1x_2x_3$, which x_0 is the MSB of X. As two most significant bytes of output is zero, we must have:

$$\begin{cases} B = 7 \times A \oplus 7 \times C \\ D = 3 \times A \oplus 2 \times C \end{cases} \Rightarrow \begin{cases} E = 11 \times A \oplus \ 9 \times C \\ F = 14 \times A \oplus 13 \times C \end{cases} \tag{4}$$

Since B, D, E and F are only four bits (higher nibbles in every byte are zero), it is equal to:

$$\begin{cases} c_0 = a_0 \\ c_1 = a_1 \\ c_2 = a_0 \oplus a_2 \end{cases} \tag{5}$$

Therefore, from 2^{16} case for A, B, C and D only 2^5 of them is acceptable. One of this 32 case is all zeros which it is not acceptable. So the probability for this event is 31×2^{-16}. By purposing second form of MixNibbles's output ($0E0F0000$) the probability will be $31/2^{15}$.

3.1 Truncated Differential Paths

Using *Proposition* 1 an iterated truncated differential path for one round is presented in previous cryptanalyses, which its probability is 2^{-6}. This iterated truncated differential path is shown in Figure 3.

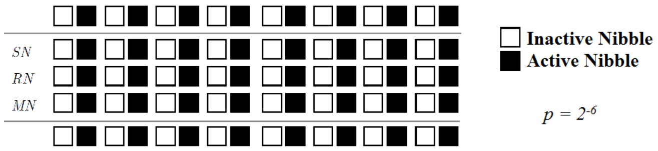

Fig. 3. Iterated truncated differential path for one round of KLEIN

Also using *Propsition* 2, we introduce two new truncated differential paths for four or three rounds that are shown in Figure 4 and 5, respectively. In the first path we consider that the event which was introduced in *Propsition* 2 happens for both of MixColumn with condition of that the output active nibbles be close to each other. Its probability is $p_1 = 1/2 \times (31 \times 2^{-15})^2 \simeq 2^{-21}$. Therefore only one MixColumn is active in round 2 and if the mentioned event happens to this MixColumn again, its probability gets $p_2 = 31 \times 2^{-15} \simeq 2^{-10}$. So there are at most 2 active lower nibbles for input of third round. These lower nibbles will activate only one MixColumn, and only lower nibbles in output of MixColumn will be active with probability of:

$$p_3 = 2/31 \times 7/15 + 29/31 \times (7/15)^2 \simeq 2^{-2.1} \tag{6}$$

About this equation it must be told that for 2 case of 31, only one lower nibble is active, and when a nibble is active with probability one, the probability for that output difference of S-Box has a MSB equal to 0 is 7/15. After this input of each MixColumn in fourth round has at most 2 active lower nibbles. Probability for that output of fourth round have only active lower nibbles is:

$$p_4 \simeq (7/15)^4 \simeq 2^{-4.4} \tag{7}$$

The second path is look like the first one, except that event of second round in first path is omitted. Therefore the probability for that only lower nibbles activated is $p_2 = 2^{-3}$ and $p_3 = 2^{-4.4}$. In both of the paths, we will use introduced iterated truncated path for reminded rounds. The probability for an $(N-1)$-round distinguisher of KLEIN will be $p = 2^{-6\times N-7.6}$ and $p = 2^{-6\times N-4.5}$, respectively using first and second path. As we will see, these two paths will be able to attack up to 14 and 15 rounds, respectively. It must be considered that in Figure 4 or 5 only one side of the probability is shown.

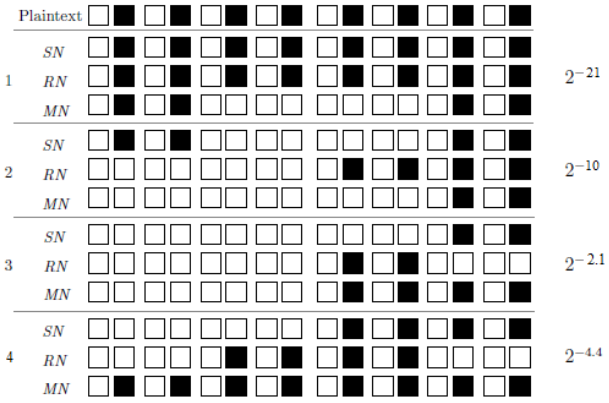

Fig. 4. Truncated differential path for 4 round of KLEIN

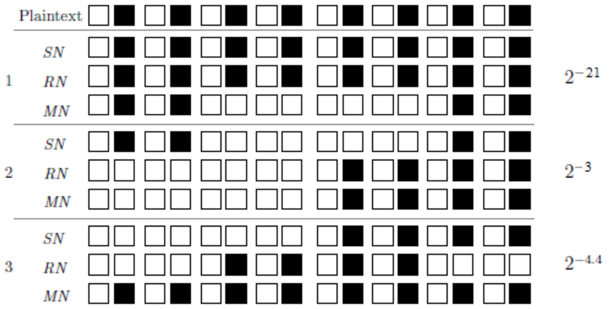

Fig. 5. Truncated differential path for 3 round of KLEIN

3.2 Procedure of key recovery attack

For recovering key's lower nibbles we will improve the used procedure of key recovery in [7] and by some differences with this procedure we will attack on KLIEN. First we will bring two propositions that introduced in previous cryptanalysis. Using these propositions we will be able to decrypt the lower or nibbles in each round.

Proposition 3. [2, 4, 6, 7] In the *Key Schedule* algorithm, lower nibbles and higher nibbles are not mixed: the lower/higher nibbles of any round-key can be computed directly from the lower/higher nibbles of the master key.

Proposition 4. [7] The values of the lower nibbles outputting MixNibbles depend on the values of the lower nibbles at the input and on 3 quantities computed from the higher nibbles that we will call 3 information bits. A similar property holds for the computation of the output lower nibbles of $\mathsf{MixNibbles}^{-1}$.

Key recovery method is as follow:

1. With the use of structures, we generate a certain number of ciphertexts such that we obtain enough pairs to be ensured to get one that verify our differential path. The size of the truncated difference entering the first round determines the size of the structures that we can build with the input plaintexts. Then if our truncated have Δ active bits, the size of structures will be 2^{Δ}. For obtaining the required p^{-1} pairs, we must encrypt $p^{-1}/2^{\Delta-1}$ plaintexts then this number is our Data Complexity. All 2^{Δ} plaintexts in a structure will be saved and processed and then be deleted, so we need a memory to save all this plaintexts. As we will see this is our Memory Complexity and other needs to memory is negligible.
2. As detailed in [2, 3], by inverting the output difference through the last MixNibbles we can observe the value of the difference entering this transformation and then discard the ones that do not have the higher nibbles inactive. In practice, we construct a sorted list of all the 8 nibbles values obtained by MixColumn of $0X0X0X0X$, and look for collisions. Such a collision occurs with probability 2^{-32}, so there are $p^{-1} \times 2^{-32}$ remaining pairs of plaintexts.
3. For each pair of plaintexts and their associated ciphertexts that collide at the previous step, we will find possible values of the first 8 lower nibbles of the key in two levels. For event described in *Proposition 2* there are 2×31 possible input differences for MixColumn, so 62×2^{16} pair is possible for half of output SubNibbles. Therefore there are normally 31 pairs which have same differences. By passing these pairs from SubNibbles and saving them and their differences in a table, we can find all 62 passible keys for 4 lower nibbles only by xoring the plaintexts pair with pairs of table that both pairs have a same differences.

   Using this method again we can find 31 possible keys for other 4 lower nibbles. In other meaning, for each pair of plaintexts and their ciphertexts that pass the previous step, we have 2×31^2 possible key for the 8 first lower nibbles of the master key.

   If the version attacked is KLEIN-64, the 8 lower nibbles correspond to the lower nibbles of the whole key but for KLEIN-80 and KLEIN-96, we have to make additional guesses to obtain all the possible lower nibble values. For KLEIN-64, KLEIN-80 and KLEIN-96 we obtain respectively $p^{-1} \times 2^{-32} \times 2 \times 31^2$, $p^{-1} \times 2^{-32} \times 2 \times 31^2 \times 2^8$ and $p^{-1} \times 2^{-32} \times 2 \times 31^2 \times 2^{16}$ possible candidates (C, C', k_{low}).

   This step requires a negligible time because all used operations are xoring, and allows us to compute the associated pair of states at the input of the first MixNibbles that already satisfies the conditions from round 1. This pair of states will be denoted by $(S, S')_1^*$.
4. For first path in round 2 that the mentioned event of *Proposition 2* happens again, we will use saved table again to know whether possible candidates for lower nibbles of key can pass this round or not. Because we know values of active nibbles after SubNibbles of first round in both plaintexts, we can guess

values of lower nibbles (we will path value of for nibbles through MixColumn and this value is one possible value and other is this value xored with 0xb). So we will search on 2^4 possible differences on the table and examine that value of 4 nibbles xored with corresponding 4 nibbles of subkey of candidate key is equal with the plaintext values saved in table or not. A plaintext pair and a candidate key can pass through this sieve with probability of $31 \times 2^{-15} \times 2^4$. Note that this step will be used only with first path. Like previous step, this step has not such salient Time Complexity.

5. At this point we start inverting the rounds from the candidates that we have obtained, generating possible pairs $(S, S')_r$ for r from N to 1. That step requires 2×2^3 round encryptions per inversion and per triple. During the iterative rounds, the number of possible triples stays the same, contrary to what happens during the non-iterative rounds where the number of candidates is reduced. The attack is performed one triple at a time. Once we have computed $(S, S')_1$, we have to guess the 6 bits needed to invert the first MixNibbles, and next we have to match values and active differences with the already computed values $(S, S')^*_1$.
   If the number of remaining candidates is smaller than $2^{k_{low}}$, as there is one possible value for k_{low} per candidate, the cost of recovering the key is smaller than the one of exhaustive search. In practice, after inverting all the rounds, the number of remaining candidates is currently very small.
   The cost of this step for KLEIN-64 is given by the initial candidate triples multiplied by 2^4 (cost of inverting), multiplied by the number of inverted rounds with probability of 2^{-6} and by the relative cost to one encryption of each inverted round. The cost for inverting non-iterative rounds are so small, because number of candidates will be reduced so much. Because time cost for other steps are not so important, this step will determine this attack's Time Complexity.
6. Finally, the higher nibbles will be recovered with an exhaustive search. Also there is a better process than exhaustie search in [7], but it will not cause to a better time complexity. More detailed attack can be found in [7].

3.3 Results and Complexities

Applying described key recovery attack to paths 1 and 2, we will able to attack up to 14 and 15 rounds KLEIN which cases introduced in [7] could not. Results of our attacks is shown in Table 3. In all of our attacks the Memory Complexity is 2^{32} of 64-bit (size of plaintext).

As it can be seen, using first path makes a good Time Complexity and second path a good Data Complexity. These have a trade-off between time and data. Our attack to full-round KLEIN-64 and reduced 13-round KLEIN-80 can recovery full master key with 2.5 times more speed and 1/64 data better than in [7]. About attack to reduced 14-round KLEIN-96, our attack by using first path have 1/40 time but 2 times data complexities toward previous cryptanalysis, and using second path 1/5 time and 1/2 data. Except biclique attacks, cryptanalyzing reduced 14-round KLEIN-80 and reduced 15-round KLEIN-96 are introduced for first time.

Table 3. Summary of the complexities of our attacks

Vrsion/Rounds	Path	Probability	Time	Data
KLEIN-64/12	I	$2^{-79.6}$	$2^{55.7}$	$2^{48.6}$
	II	$2^{-76.5}$	$2^{58.8}$	$2^{45.5}$
KLEIN-80/13	I	$2^{-85.6}$	$2^{69.8}$	$2^{54.6}$
	II	$2^{-82.5}$	$2^{72.9}$	$2^{51.5}$
KLEIN-80/14	I	$2^{-91.6}$	$2^{75.9}$	$2^{60.6}$
	II	$2^{-88.5}$	$2^{78.9}$	$2^{57.5}$
KLEIN-96/14	I	$2^{-91.6}$	$2^{83.9}$	$2^{60.6}$
	II	$2^{-88.5}$	$2^{86.9}$	$2^{57.5}$
KLEIN-96/15	II	$2^{-94.5}$	$2^{92.9}$	$2^{63.5}$

4 Conclusions

In this paper we introduce two new truncated differential paths for KLEIN, and using an improved key recovery method which basic of that was used before by Lallemand et al. Results show that our attacks have a good time and data complexities on full-round KLEIN-64, reduced 13-round KLEIN-80 and reduced 14-round KLEIN-96. Also we introduce two new attacks on reduced 14-round KLEIN-80 and reduced 15-round KLEIN-96 for first time.

The block cipher KLEIN has two main weaknesses: 1. MixNibbles transformation using Rijndael's MixColumn transformation does not correctly mix higher and lower nibbles, as it is the only transform that does so. 2. KeySchedule does not mix higher and lower nibbles. These two helps the cryptanalyst to perform a reduced partial key search, so maybe considering other matrices instead of Rijndael's and a stronger KeySchedule could help to prevent the attacks.

References

1. Zheng Gong, Svetla Nikova, and Yee Wei Law. *KLEIN: A new family of lightweight block ciphers.* In RFIDSec, volume 7055 of Lecture Notes in Computer Science, pages 1-18. Springer, 2011.
2. Xiaoli Yu, Wenling Wu, Yanjun Li, and Lei Zhang. *Cryptanalysis of reduced-round KLEIN block cipher.* In Inscrypt, volume 7537 of Lecture Notes in Computer Science. Springer, 2011.
3. Jean-Philippe Aumasson, Maria Naya-Plasencia, and Markku-Juhani O. Saarinen. *Practical attack on 8 rounds of the lightweight block cipher KLEIN.* In INDOCRYPT, volume 7107 of Lecture Notes in Computer Science, pages 134-145. Springer, 2011.
4. Zahra Ahmadian, Mahmoud Salmasizadeh, and Mohammad Reza Aref. *Biclique Cryptanalysis of the Full-Round KLEIN Block Cipher.* Cryptology ePrint Archive, Report 2013/097, 2013. `http://eprint.iacr.org/`
5. Farzaneh Abed, Christian Forler, Eik List, Stefan Lucks, and Jakob Wenzel. *Biclique Cryptanalysis Of PRESENT, LED, And KLEIN.* Cryptology ePrint Archive, Report 2012/591, 2012. `http://eprint.iacr.org/`
6. Ivica Nikolic, Lei Wang, and Shuang Wu. *The Parallel-Cut Meet-In-The-Middle Attack.* Cryptology ePrint Archive, Report 2013/530, 2013. `http://eprint.iacr.org/`
7. Virginie Lallemand and Maria Naya-Plasencia. *Cryptanalysis of KLEIN.* In FSE 2014 accepted papers, also in Cryptology ePrint Archive, Report 2014/090, 2014. `http://eprint.iacr.org/`