Biclique Cryptanalysis of the Full-Round KLEIN Block Cipher

Zahra Ahmadian, Mahmoud Salmasizadeh, and Mohammad Reza Aref

Sharif University of Technology, Tehran, Iran.
ahmadian@ee.sharif.edu,{salmasi,aref}@sharif.edu

Abstract. In this paper we present a biclique attack on the newly proposed block cipher KLEIN-64. We first introduce some weaknesses of the diffusion layer and key schedule of this algorithm. Then we exploit them to present a full round attack on KLEIN-64 using an asymmetric biclique. The (worst case) computations and data complexity of this attack are $2^{62.84}$ and 2^{39}, respectively. A modified version of this attack is also presented which is slightly faster at the expense of the data required.

Keywords: lightweight cryptography, biclique attack, KLEIN family

1 Introduction

Biclique cryptanalysis, first introduced in cryptanalysis of AES [1], is the most recent technique for security evaluation of block ciphers. Soon after publishing the seminal paper of AES cryptanalysis, lots of cryptanalytical results on the other block ciphers were proposed [2–5]. Biclique cryptanalysis often breaks the full version of the cipher with a reasonable data and memory complexities but marginal computations.

This technique enjoys a biclique structure in the cipher in which one set of vertices is composed of 2^{d_1} plaintexts (or ciphertexts) while the other set contains 2^{d_2} intermediate states of the cipher (about 2-3 rounds apart). If $d_1 = d_2 = d$ the biclique is said to have dimension d as is often the case with the published cryptanalytical results.

Each edge of this graph is actually a master key under which the encryption of one vertex yields another one. In the straightforward manner, constructing a biclique with dimension d for a block cipher requires about 2^{2d} partial cryptanalysis. Bogdanove et. al. [1] presented an efficient tool for constructing bicliques with the complexity of about 2^{d+1} partial encryption/decryption that makes use of two related key characteristics in two directions with non-overlapping nonlinear parts (the so called independent bicliques).

Lightweight block ciphers are those that are specifically designed for constrained environments such as RFID tags or sensor nodes. Due to the implementation considerations in such environments the key size of the cipher is typically 64 or 80 bits. A number of biclique attacks on some lightweight block cipher such as Piccolo [3, 7], LBlock [4], LED and PRESENT [6, 7] and TWINE [8]

have been published thus far. Apart from LED, the diffusion layers in all are purely permutation transformations.

Although the computations of biclique attack is close to the exhaustive search, a successful one on a full lightweight block cipher may result in more interesting consequences. As an example, assume a lightweight block cipher with a short master key, say 64 bits, is targeted by an attacker with a typical realistic budget who can not afford the brute force attack. A successful biclique attack on such an algorithm, reducing the key space even about 1-2 bits, may convert the practically impossible brute force attack into a practically feasible attack under a reasonable assumption on the attacker's computational budget. Whereas in the case of large key sizes (e.g. 128 or 256 bits), the practical security of the cipher is not so sensitive to such a slight reduction in the key space.

KLEIN family of lightweight block ciphers is proposed by Gong et al. in RFIDsec 2011 [9]. It supports three key sizes of 64, 80 and 96 bits. Software and hardware implementation 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 [9], its real security level is not determined without further external analysis. In fact, the role of cryptanalysis is very vital in the case of lightweight ciphers where some conventional design constrains (large Sbox, complicated key schedule, high diffusion, ...) are relaxed in order to achieve more efficient implementation.

KLEIN makes use of the combination of 4-bit Sboxes with AES MixColumn in a SPN structure. Although such a combination allows compact and low memory implementation in software and hardware, it poses serious security risks to KLEIN, as part of which was discovered and exploited by Aumasson et al. in Idocrypt 2011 [10] to break 8 rounds out of 12 rounds of KLEIN-64 and by Yu et al. in Inscrypt 2011 [11] to cryptanalyse 8 out of 12 rounds and 8 out of 16 rounds of KLEIN-64 and KLEIN-80, respectively (see Table 1).

In this paper, we report more observations on this cipher that along with some of those observed in [10, 11], are used to apply a biclique attack on the full version of KLEIN-64. To minimize the number of active Sboxes, we construct a biclique with $d_1 \neq d_2$, and call it *asymmetric biclique*. This approach has been used only in the biclique cryptanalysis of IDEA [2] whereas the biclique cryptanalyses of all the SPN-based ciphers [1, 3–7] are symmetric. In the case of KLEIN, the asymmetric approach significantly works more efficient than the symmetric one. We have also provided a precise evaluation of the number of Sbox computations. Therefore, the proposed complexity of the attack is calculated in a rigorous and conservative way.

The computational complexity of our attack is $2^{62.84}$ and the data required is upper bounded by 2^{39} chosen plaintexts. Using the data and complexity trade off, another version of this attack is also presented that works with $2^{62.81}$ computations and requires 2^{43} chosen plaintexts. The required memory for both of them is less than $2^{4.5}$. The reader should be noted that the mentioned computational complexities are the worst case ones and the average complexities are half

Table 1. Summary of cryptanalytic results on KLEIN

version	Rounds	Computations	Data	Memory	Attack Type	Reference
KLEIN-64	7/12	$2^{45.5}$	$2^{34.3}$	-	Integral	[11]
	8/12	$2^{46.8}$	2^{32}	-	Truncated Differential	[11]
	8/12	2^{35}	2^{35}	-	Differential	[10]
	12 (full)	$2^{62.84}$	2^{39}	$2^{4.5}$	Biclique	Sec. 3
	12 (full)	$2^{62.81}$	2^{43}	$2^{4.5}$	Biclique	Sec. 4
KLEIN-80	8/16	$2^{77.5}$	$2^{34.3}$	-	Differential	[11]

of these values. A Summary of cryptanalytical results on KLEIN-64 is presented in Table 1.

The outline of the paper is as follows: In Section 2 we briefly describe KLEIN-64. In Section 3, first some observations on KLEIN-64 is stated then, a biclique attack on the cipher is presented. A modified version of this attack is presented in Section 4. Finally we conclude our work in Section 5.

2 KLEIN-64 Specifications

Notations. In this paper for 64-bit variable $X = X_7, X_6, \cdots, X_0$, the most significant byte is X_7 and the least significant one is X_0. Similarly, bit's position is counted from the right (the least significant) side, advancing left. Each byte involves two nibbles: the rightmost four bits are called the lower nibble and the leftmost ones are the higher nibble. The input state and the subkey of round r are denoted by $X^{(r)}$ and $K^{(r)}$ respectively.

KLEIN cipher. KLEIN supports 64, 80 and 96-bit key sizes along with 12,16 and 20 rounds respectively. These versions differ in the key schedule and the number of rounds but the block size in all of them is 64-bit. In this paper we consider only KLEIN-64.

KLEIN-64 round function is composed of the following steps:

1. AddRoundKey (AK), which XORs a round key to the 64-bit state.
2. SubNibbles (SN), which applies a 4-bit Sbox to each nibble.
3. RotateNibbles (RN), which left-rotates the state by 2 bytes.
4. MixNibbles (MN), which applies two AES MixColumn's in parallel.

After 12 rounds, an additional AddRoundKey operation is run. Thus, $P = X^{(1)}$ and $C = X^{(12)} \oplus K^{(13)}$. Fig. 1 shows one round of the cipher.

Recall that AES MixColumn works according to the following matrix multiplication in $GF(2^8)$ with the irreducible polynomial $x^8 + x^4 + x^3 + x + 1$:

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

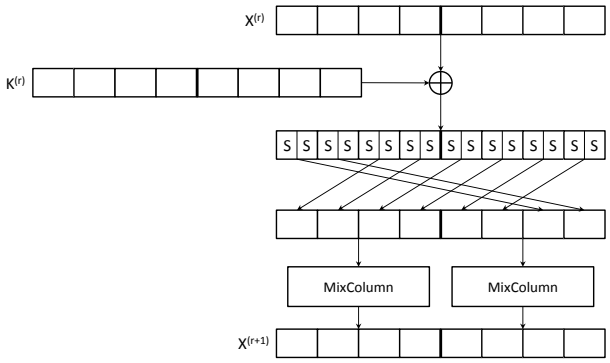

Fig. 1. One round of KLEIN block cipher

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

$$02 \times x = \begin{cases} x \lll 1 & \text{if MSB}(x) = 0 \\ x \lll 1 \oplus 0\text{x}1\text{b} & \text{if MSB}(x) = 1 \end{cases} \tag{2}$$

Key schedule. For KLEIN-64, the round keys $K^{(r)}$, $r = 1, \cdots 12$, and the final whitening key $K^{(13)}$ is generated as follows. First, the 64-bit master key $K = K_7, K_6, \ldots, K_0$ is stored in a key register as $K^{(1)}$. Then the following steps are iteratively applied to K to generate 12 more subkeys:

1. Left-rotate the two halves of the key state of 1 byte each.
2. Swap the two halves by a Feistel-like structure.
3. XOR byte 5 with round counter r and substitute bytes 1 and 2 using four KLEIN Sboxes.

At the end of round r, the content of the key register is $K^{(r)}$. Fig. 2 shows one round of the key schedule.

3 Biclique Attack on KLEIN-64

3.1 Some Observations on KLEIN

Aumasson et al. [10] reported some observations on KLEIN-64 and exploited them for their differential attack on the 8-round cipher. One of those properties was independently observed and used for the truncated differential attack on 8-round cipher by Yu et al. [11], as well. These two attacks make use of the same differential truncated characteristic, though, the differential probability is calculated in [10] with more accuracy, resulting in less computational complexity.

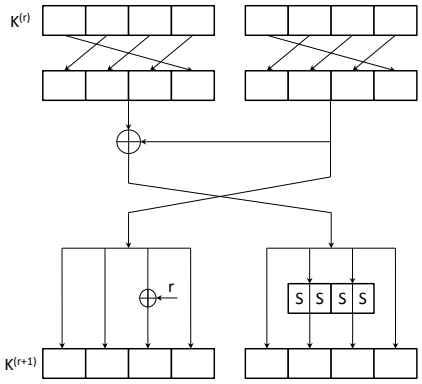

Fig. 2. One round of KLEIN-64 key schedule

In this subsection, we cover those properties from [10, 11] that are exploited in our attack. Furthermore, we introduce more weaknesses of KLEIN-64 in both the key schedule and the diffusion layer. Properties 1 and 2 are associated with KLEIN-64's key schedule, while the other ones go with the diffusion layer of the cipher, hence hold true for all versions of KLEIN.

Property 1. The higher and lower nibbles of KLEIN-64 subkeys are never mixed [10].

Property 2. Each subkey byte of KLEIN-64 depends exactly on the one of the following subsets of the master key bytes:

$$\{K_0, K_4\}\ \{K_1, K_5\}\ \{K_2, K_6\}\ \{K_3, K_7\} \tag{3}$$

That along with Property 1, each subkey nibble depends only on two nibbles of the master key and consequently, changing a single nibble of the master key, affects at most on two nibbles of each subkey.

For the next three properties assume the input difference of the MixColumn matrix is $X = X_3, X_2, X_1, X_0$ and the output difference is $Y = Y_3, Y_2, Y_1, Y_0$.

Property 3. If the higher nibbles of all X_i's are null and the lower nibbles contain a difference in $\{0, \cdots, 7\}$, the higher nibbles of all Y_i's remains inactive. The same property holds when the active input nibbles are in $\{8, \cdots, f\}$ [10, 11].

Property 4. If the lower nibbles of all X_i's are null and the higher nibbles contain a difference in $\{0, \cdots, 7\}$, then the lower nibbles of all Y_i's remain inactive. This property also holds when the active input nibbles are in $\{8, \cdots, f\}$.

Property 5. The leftmost three bits of each Y_i does not depend on the lower nibbles of X_i's.

Properties 3-5 can be investigated considering Eq. 2.

3.2 Attack Description

The principles of the biclique attack can be found in [1] and with a simpler language in [3] and [4]. To avoid repetition, we do not explain the attack basis here but three phases of the attack on KLEIN-64 is presented in detail.

Phase 1. Key Partitioning We define the key groups with respect to $K^{(3)}$ and enumerate the groups of keys by 2^{57} base keys. The base key $K^{(3)}[0,0]$ of each group is defined as follows:

$$K^{(3)}[0,0] = [\mathsf{X},\mathsf{X},\mathsf{X},\mathsf{Y},\mathsf{X},\mathsf{X},\mathsf{X},\mathsf{X}], \tag{4}$$

where the bytes with the wildcard X take all 2^8 possible values (not necessarily equally) and the byte with Y takes only two values $\mathsf{0x00}$ and $\mathsf{0x80}$; Hence 2^{57} base keys.

$K^{(3)}[0,j], j \in \{\mathsf{0},\cdots,\mathsf{15}\}$ is defined as follows:

$$K^{(3)}[0,j] = K^{(3)}[0,0] \oplus [0,0,0,J,0,0,0,J], \tag{5}$$

where $J = \mathsf{0x0}j$.

$K^{(3)}[i,0], i \in \{\mathsf{0},\cdots,\mathsf{7}\}$ is defined as follows:

$$K^{(3)}[i,0] = K^{(3)}[0,0] \oplus [2I,3I,I,I,0,0,0,0], \tag{6}$$

where $I = \mathsf{0x}i\mathsf{0}$ and multiplications are performed in $GF(2^8)$. Note that since the MSB of I is zero, the lower nibbles of all bytes of $K^{(3)}[i,0]$ remains inactive. Finally, $K^{(3)}[i,j]$ is defined as:

$$K^{(3)}[i,j] = K^{(3)}[0,0] \oplus K^{(3)}[0,j] \oplus K^{(3)}[i,0] \tag{7}$$

$$= K^{(3)}[0,0] \oplus [2I,3I,I,I \oplus J,0,0,0,J] \tag{8}$$

Thus, the key space of $K^{(3)}$ is partitioned into 2^{57} groups of 2^7 keys each. Since there is a bijective relationship between the master key and each of the round subkeys, this partitioning is valid for the master key space as well. For more convenience, we denote $K^{(3)}[i,j]$ by $K[i,j]$ in the rest of the paper.

Phase 2. Constructing the Biclique Let's call the first three rounds of KLEIN-64 f. In order to construct a biclique for each key group, we combine two related key differential characteristics for f. This biclique is asymmetric i.e. the number of vertices in two sides are not equal which is consistent with the pre-defined key partitioning. In each group, the biclique connects 2^{d_2} plaintexts to 2^{d_1} states $S = X^{(4)}$ under one of the keys in the group, where $d_1 = 4$ and $d_2 = 3$. The procedure of constructing the biclique is as follows:

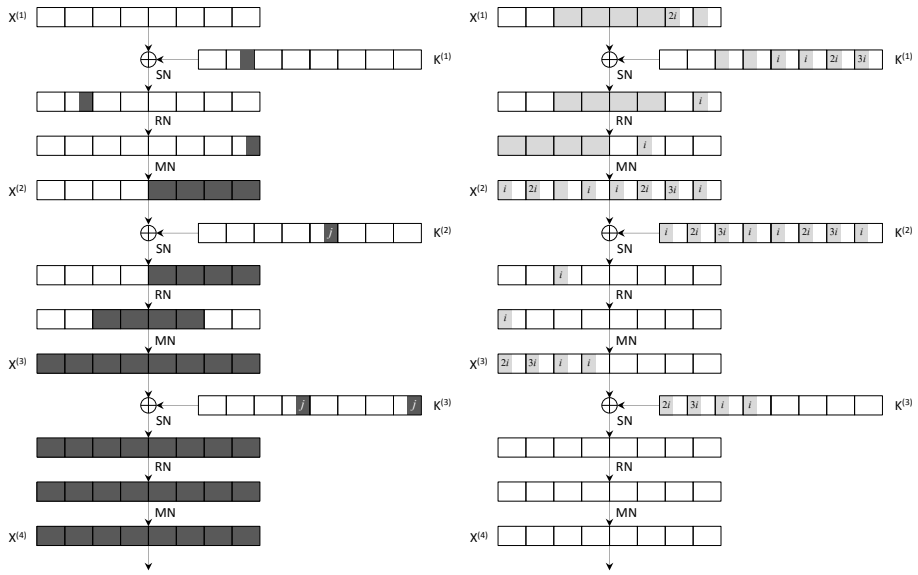

Fig. 3. Three round biclique

Step 1. Fix $P_0 = 0$ and drive $S_0 = f_{K[0,0]}(P_0)$.

Step 2. Encrypt P_0 under different keys $K[0, j]$, $j \in \{1, \cdots, 2^{d_1} - 1\}$ to get the corresponding state S_j. The active nibbles in the key schedule and the cipher are shown in black in Fig. 3, left. These nibbles should be calculated $2^{d_1} - 1$ times, while the other ones are computed just once that have been already done in Step. 1. This step constructs $P_0 \xrightarrow[f]{K[0,j]} S_j$.

Step 3. Decrypt S_0 under different keys $K[i, 0]$, $i \in \{1, \cdots, 2^{d_2} - 1\}$ to get the corresponding plaintext P_i. In Fig. 3 right, the active nibbles are in gray. Note that $2i$ and $3i$ do not activate the lower nibbles since the MSB of i is zero. $K[i, 0]$ is defined in such a way that after MN^{-1} it activates only a single nibble in round 2 and thanks to the key schedule of KLEIN-64, this property is still preserved in the left half of the state after MN^{-1} of round 1. Here, the active nibbles should be calculated $2^{d_2} - 1$ times, while the other ones have been already computed. At the end of this step, we have constructed $P_i \xleftarrow[f^{-1}]{K[i,0]} S_0$.

Finally, in order to complete the $2^{d_2} \times 2^{d_1}$ biclique structure, we can combine these two characteristics in such a way that $P_i \xrightarrow[f]{K[i,j]} S_j$. We are allowed to do

this since the two characteristics do not overlap on any active nonlinear elements in the cipher or the key schedule (independent bicliques [1]).

Hint. Compute and save all other subkeys (rounds 4-13) for $K[i,0]$ and $K[j,0]$ each for later use in Phase 3. Due to the asymmetry of the biclique ($d_2 = 3$ v.s $d_1 = 4$) along with Property 2, these two characteristics are never overlapped on an Sbox in the key schedule, hence the relation $K^{(r)}[i,j] = K^{(r)}[0,0] \oplus K^{(r)}[i,0] \oplus K^{(r)}[0,j]$ is valid for all $r = 1, \cdots 13$.

Phase 3. Meeting in the Middle The matching variable V is the leftmost three bits of bytes 6 and 7 of $X^{(11)}$ (6 bits in total, see Fig. 4) which is selected considering the total number of Sboxes that it depends on (Property 5) and an effective filtering of the wrong keys. We call rounds 4-10 of the cipher, h_1 and the two last rounds h_2.

We calculate the value of matching variable in both directions to find the correct key that meets this matching.

Forward direction. Each state value S_j is encrypted by the function h_1 under the key $K[0,j]$, once to get $S_j \xrightarrow[h_1]{K[0,j]} \overrightarrow{V}_{0,j}$. Then, S_j is encrypted by the function h_1 under all the $2^{d_2} - 1$ keys $K[i,j]$ to get $S_j \xrightarrow[h_1]{K[i,j]} \overrightarrow{V}_{i,j}$. This procedure does not need to be performed exhaustively. In fact, one can determine $\overrightarrow{V}_{i,j}$ by the influence of the differences between keys $K[0,j]$ and $K[i,j]$. This process is shown in the upper part of Fig. 4, where the active nibbles are in gray, the gridded ones are those that are calculated only once and the white nibbles do not need to be calculated because they do not affect the value of matching variable $\overrightarrow{V}_{i,j}$. Note that the lower nibbles of all subkeys remain unaffected (Property 1).

Backward direction. Call the encryption oracle to obtain the ciphertext C_i of each plaintext P_i. Each ciphertext C_i is decrypted under key $K[i,0]$ to get $\overleftarrow{V_{i,0}} \xleftarrow[h_2^{-1}]{K[i,0]} C_i$. After that, C_i is decrypted under all the $2^{d_1} - 1$ keys $K[i,j]$ to get $\overleftarrow{V_{i,j}} \xleftarrow[h_2^{-1}]{K[i,j]} C_i$. Again, this procedure does not need to be performed exhaustively. One can determine $\overleftarrow{V_{i,j}}$ by the influence of the difference between keys $K[i,0]$ and $K[i,j]$. In the lower part of Fig.4, the active nibbles are in gray and the white nibbles do not need to be calculated.

In each group, check the equality of $\overrightarrow{V_{i,j}}$ and $\overleftarrow{V_{i,j}}$ for all $i \in \{1, \cdots, 2^{d_2} - 1\}$ and $j \in \{1, \cdots, 2^{d_1} - 1\}$ to find the correct key.

3.3 Complexities

Data Complexity. The data complexity is determined by the number of plaintexts P_i to be encrypted. Fig. 3 right, shows that all the plaintexts share the

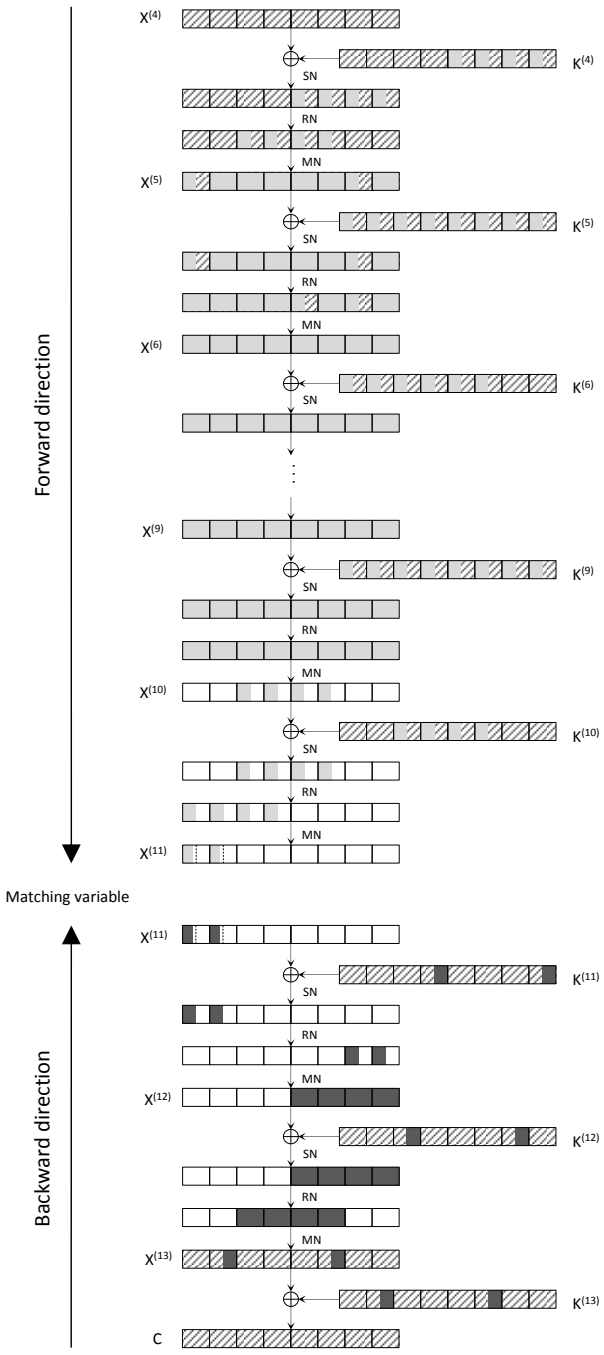

Fig. 4. Computing the matching variable

same values in 6 nibbles (white nibbles). In addition, the higher nibble of byte 1 contains $2i$, $i \in \{0, \cdots, 7\}$, hence a null difference in the LSB of this nibble, too. So all plaintexts share 25 bits with $P_0 = 0$ and data complexity does not exceed 2^{39} chosen plaintexts.

Computational Complexity. Since the attack workload is dominated by the number of Sboxes to be calculated, it is conventional in the biclique attack to take only these operations into account. Each round of KLEIN-64, together with one round of key schedule, takes 20 Sbox computations. So, the complexity of a single encryption equals calculation of $12 \times 20 = 420$ Sboxes.

For each $2^{64-7} = 2^{57}$ group of keys, the following computations should be performed:

Biclique complexity. For the first characteristic, 26 Sboxes should be calculated 2^{d_1} times and for the second one, 10 Sboxes should be calculated 2^{d_2} times. The remaining 14 Sboxes are calculated once. Thus, this phase demands $14 + 26 \times 2^{d_1} + 10 \times 2^{d_2} = 510$ Sbox calculations in total.

Matching complexity. In forward direction, 14 Sboxes are calculated once and 86 ones are calculated 2^{d_2} times. This process is repeated for all S_j's. So, the total complexity of this step is $2^{d_1} \times (14 + 2^{d_2} \times 86) = 11232$.

In backward direction, Only 10 Sboxes should be calculated 2^{d_1} times. Since this procedure should be done for all P_i's, the overall complexity is $2^{d_2} \times 2^{d_1} \times 10 = 1280$ Sbox calculations.

Key schedule complexity In the key schedule, 5 Sboxes should be calculated 2^{d_1} times and 20 Sboxes should be calculated 2^{d_2} times. The remaining 23 Sboxes are calculated once, hence the total complexity of $23 + 5 \times 2^{d_1} + 20 \times 2^{d_2} = 263$ Sbox calculations.

Finally, in each group 2^7 key candidates are tested by a 6-bit matching variable, resulting in an average of $2^{7-6} = 2$ remaining key candidates to be re-checked. Thus, the computational complexity of the attack is:

$$2^{57} \times (\frac{510 + 11232 + 1280 + 263}{20 \times 12} + 2) = 2^{62.84} \tag{9}$$

Memory complexity. The required memory for this attack does not exceed $2^{d_1} + 2^{d_2} = 2^{4.5}$ that is used for saving all the keys in a group as well as the P_i's and S_j's.

4 A modified Version of the Attack

In this Subsection, we present a modified version of the attack presented in Section 3.2 with a slightly less computational complexity at the expense of the

data required. Roughly speaking, the new key partitioning causes less active nibble diffusion in the meet in the middle part whereas it imposes more diffusion in the biclique part, hence less workload but more data complexity.

4.1 Modifications

Key Partitioning. Here we use the following key partitioning:

$$K^{(3)}[0,0] = [\mathsf{W},\mathsf{X},\mathsf{X},\mathsf{Y},\mathsf{X},\mathsf{X},\mathsf{X},\mathsf{X}], \tag{10}$$

where the bytes with the wildcard X take all 2^8 possible values. The lower nibble of byte with Y is null, while its higher nibble takes all 2^4 values. The higher nibble of byte with W takes only two values $\mathsf{0x00}$ and $\mathsf{0x80}$, while its lower nibble takes all 2^4 values. Hence 2^{57} base keys.

$K^{(3)}[0,j]$ is defined as the previous attack:

$$K^{(3)}[0,j] = K^{(3)}[0,0] \oplus [0,0,0,J,0,0,0,J], \tag{11}$$

where $J = \mathsf{0x0}j$, $j \in \{\mathsf{0},\cdots,\mathsf{15}\}$. For $i \in \{\mathsf{0},\cdots,\mathsf{7}\}$, $K^{(3)}[i,0]$ is defined as follows:

$$K^{(3)}[i,0] = K^{(3)}[0,0] \oplus [I,3I,0,2I,0,0,0,0], \tag{12}$$

where $I = \mathsf{0x}i\mathsf{0}$. Finally, $K^{(3)}[i,j]$ is defined as:

$$K^{(3)}[i,j] = K^{(3)}[0,0] \oplus [I,3I,0,2I \oplus J,0,0,0,J]. \tag{13}$$

Thus, the key space of $K^{(3)}$ is partitioned into 2^{57} groups of 2^7 keys each.

Biclique. Half of the biclique changes. The new characteristic is shown in Fig. 5 of Appendix A. $K[i,0]$ is defined in such a way that after MN^{-1} in round 2 it activates only two nibbles both with value i. This property is preserved in the left half of the state after MN^{-1} in round 1, too.

Meeting in the middle. The forward direction of this phase is changed while the backward computations are the same. This process is shown in Fig. 6 of Appendix A.

4.2 Complexities

Data Complexity. Fig. 5 right, illustrates that all the plaintexts share the same values in five nibbles (white nibbles) and the MSB of byte 1 is null. So, the data complexity of this version of the attack does not exceed 2^{43} chosen plaintexts.

Computational Complexity. The computational complexity of this version of the attack is:

$$2^{57} \times (\frac{515 + 11008 + 1280 + 235}{20 \times 12} + 2) = 2^{62.81} \tag{14}$$

Memory Complexity. The require memory for this attack is again less than $2^{4.5}$.

5 Conclusion

We presented two biclique attacks on the full KLEIN-64. The first attack has a workload of $2^{62.84}$ and requires 2^{39} chosen plaintexts. With a small modification in the differential characteristics, we achieved a slightly faster attack at the expense of data required. The computations and the required data for this version of the attack are $2^{62.81}$ and 2^{43} respectively.

This is the first successful cryptanalysis of the full version of this algorithm. As the previous attacks on the reduced-round cipher, the combination of 4-bit Sboxes with AES MixColumn along with some observed key schedule weaknesses made the cipher vulnerable against this attack.

This result is an example illustrating that the design of lightweight ciphers is a very challenging task and the mechanisms adopted for the efficient implementation of the algorithm should not result in great expense in the security of the cipher.

References

1. Bogdanov, A., Khovratovich, D., Rechberger, C.: Biclique Cryptanalysis of the Full AES. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 344-371. Springer, Heidelberg (2011)
2. Khovratovich, D., Leurent, G., Rechberger, C.: Narrow-Bicliques: Cryptanalysis of Full IDEA. In EUROCRYPT 2012, LNCS, pages 392-410. Springer, Heidelberg (2012).
3. Wang, Y., Wu, W., and Yu, X.: Biclique Cryptanalysis of Reduced-Round Piccolo Block Cipher. In Mark Dermot Ryan, Ben Smyth, and Guilin Wang, editors, ISPEC 2012, LNCS, vol. 7232, pp. 337-352. Springer, Heidelberg (2012).
4. Wang, Y., Wu, W., Yu, X., Zhang, L.: Security on LBlock against Biclique Cryptanalysis, In WISA 2012.
5. Mala., H.: Biclique Cryptanalysis of the Block Cipher SQUARE. Cryptology ePrint Archive, Report 2011/500, (2011)
6. Abed, F., Forler, C., List, E., Lucks, S., Wenzel, J.: Biclique Cryptanalysis of the PRESENT and LED Lightweight Ciphers, Cryptology ePrint Archive, Report 2012/591, (2012)
7. Jeong, K., Kang, H., Lee, C., Sung, J., Hong, S.: Biclique Cryptanalysis of Lightweight Block Ciphers PRESENT, Piccolo and LED, Cryptology ePrint Archive, Report 2012/621, (2012)
8. Coban, M., Karakoc, F., and Boztas, O.: Biclique Cryptanalysis of TWINE. Cryptology ePrint Archive, Report 2012/422, 2012. http://eprint.iacr.org/
9. Gong, Z., Nikova, S., Law, Y.W.: KLEIN: A New Family of Lightweight Block Ciphers. In: Juels, A., Paar, C. (eds.) RFIDSec 2011. LNCS, vol. 7055, pp. 118. Springer, Heidelberg (2012)
10. Aumasson, J.P., Naya-Plasencia, M. , Saarinen, M.J.O.: Practical attack on 8 rounds of the lightweight block cipher KLEIN. In: INDOCRYPT 2011, LNCS, vol. 7107, pp.134-145. Springer, Heidelberg (2011)
11. Yu, X. , Wu, W., Li, Y., Zhang, L.: Cryptanlysis of Reduced-Round KLEIN Block Cipher, In: Inscrypt 2011, LNCS, vol. 7537, pp.237-250. Springer, Heidelberg (2012)

Appendix A: Figures of the Modified Attack

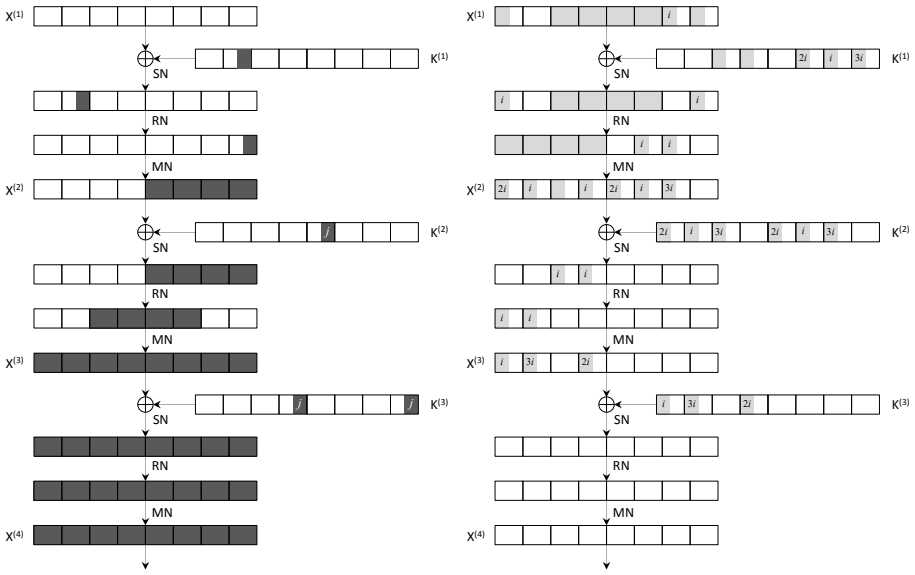

Fig. 5. Three round biclique, modified attack

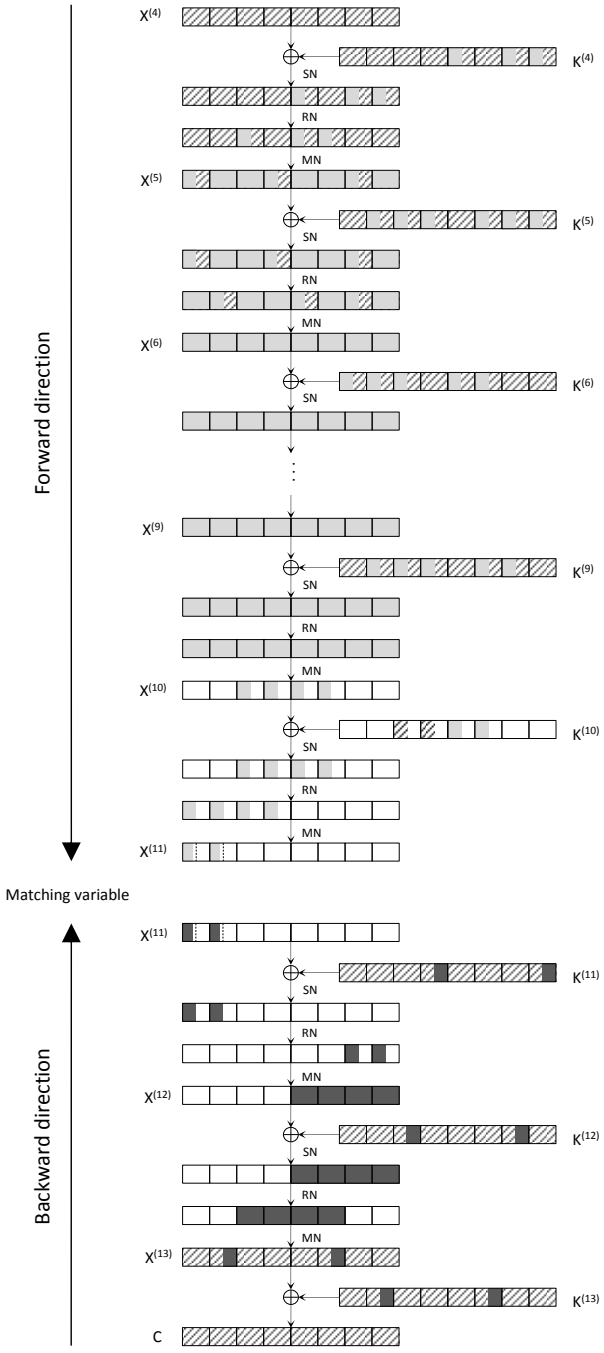

Fig. 6. Computing the matching variable, modified attack