SN Computer Science (2020) 1:94
https://doi.org/10.1007/s42979-020-0103-z

ORIGINAL RESEARCH

New Biclique Cryptanalysis on Full-Round PRESENT-80 Block Cipher

K. B. Jithendra[1] · **T. K. Shahana**[2]

Published online: 19 March 2020

Abstract
Biclique cryptanalysis is a recent technique developed for key retrieval of block ciphers. In this paper, biclique attack is carried out on full-round, PRESENT-80 block cipher. Here, the biclique is constructed using independent related key differential cryptanalysis. Matching with precomputation is used for the analysis for other rounds. The computational complexity for the successful implementation of the proposed attack is found less than that of attacks published so far. The data complexity and time complexity of the proposed attack are calculated as 2^{23} and $2^{79.63}$, respectively.

Keywords Block cipher · Cryptanalysis · PRESENT · Biclique · Computational complexity

Introduction

Lightweight block ciphers [1–6] are developed for satisfying the security needs for resource-constrained devices like RFID tags. Lightweight block ciphers work efficiently in hardware and consume less amount of memory. Since there is a direct relation between hardware complexity and security, lightweight block ciphers may lead to poor security, if proper attention is not paid in the design phase. Conventional cryptographic strength evaluation is mainly based on linear and differential cryptanalysis, which is efficient, but fails in full-round cryptanalysis of most of the ciphers. Khovratovich et al. introduced biclique cryptanalysis [7], in 2011 as part of the preimage attack on SHA-2, which is an extension of splice and cut method [8]. The splice and cut framework is derived from the concept of meet in the middle attacks [9].

The biclique technique was introduced for the analysis of block cipher-based hash functions, but later cryptologists started using the same for block cipher key recovery. In 2011, for the first time, cryptanalysis of full-round AES [10] was successfully done by Bogdanov et al. using bicliques. Later, key recovery attacks were successfully proposed against various block ciphers including HIGHT [11], Piccolo [12], TWINE [13], SQUARE [14], AREA-256 [15], etc.

In 2007, Bogdanov et al. proposed the lightweight block cipher PRESENT [16], which supports the key lengths 80 and 128 denoted by PRESENT-80 and PRESENT-128, respectively. Because of its simple structure and security, PRESENT attracted the attention of the cryptographic society to a great extent. A differential cryptanalysis was proposed in 2008 [16] by Wang on 16-round PRESENT with a time complexity 2^{65}. In 2009, Kenji Ohkuma has proved linear cryptanalysis can be used to attack 24 rounds of PRESENT [17]. Using linear cryptanalysis, 25 rounds of PRESENT 80 were attacked by Cho [18] in 2010. Farzaneh et al. used biclique technique on full-round PRESENT in 2012 [19]. The keys of PRESENT 80 and PRESENT-128 are retrieved with time complexities $2^{79.46}$ and $2^{127.37}$, and data complexity 2^{60} and 2^{44} chosen plain texts, respectively. Jeong et al. published a different biclique analysis on both key variants of PRESENT [20] with time complexities $2^{79.76}$ and $2^{127.91.}$

In this paper, a new biclique attack on full-round PRESENT is proposed. A four-dimensional biclique is developed for rounds 1–3. For the remaining rounds, matching with precomputation is applied. The comparison of the results

This article is part of the topical collection "Advances in Computational Intelligence, Paradigms and Applications" guest edited by Young Lee and S. Meenakshi Sundaram.

✉ K. B. Jithendra
jithendrakb@yahoo.com

T. K. Shahana
shahanatk@cusat.ac.in

[1] College of Engineering, Thalassery, Kannur (Dt), Kerala, India

[2] School of Engineering, Cochin University of Science and Technology, Cochin, Kerala, India

of the cryptanalysis proposed in this paper with the results achieved so far by other researchers is given in Table 1.

The paper is organized as follows. Section 2 gives an overall idea about PRESENT block cipher operation and functioning. This section focuses on the 80-bit key variant of PRESENT. Section 3 gives the theory behind biclique cryptanalysis. In Sect. 4 the proposed attack is presented with all computational details. Section 5 concludes the paper.

PRESENT-80 Block Cipher

PRESENT is an SPN-type block cipher with 64 data bits. The cipher has two key lengths, 80 bits and 128 bits, and the variants are denoted as PRESENT-80 and PRESENT-128, respectively. Except the key schedule, both invariants function similarly. The cipher takes 31 rounds to complete the encryption process. The operations performed in each round of the cipher are round transformation and key scheduling.

Round Transformation

In each round, partial data encryption is done in three steps

1. Key addition: The most significant 64 bits of the key is exclusively ORed with the 64-bit data.
2. Substitution: 4-bit bijective sboxes are used in PRESENT. Sixteen parallel sboxes are used, which substitute the entire 64 bits data in a clock cycle.
3. Permutation: The substituted data in each round are permutated using the pLayer. The pictorial representation

Table 1 Comparison on attack complexities on PRESENT-80

References	Attack type	Number of rounds	Time complexity	Data complexity
[16]	Differential	16	2^{65}	2^{64} CP
[21]	Differential + algebraic	19	2^{113}	NA
[22]	Saturation	24	2^{57}	2^{57} CP
[17]	Linear	24	NA	$2^{63.5}$ KP
[18]	Linear	25	2^{65}	$2^{62.4}$ KP
[18]	Linear	26	2^{72}	2^{64} KP
[19]	Biclique	21	$2^{79.03}$	2^{60} CP
[19]	Biclique	31 (Full)	$2^{79.46}$	2^{60} CP
[20]	Biclique	31 (Full)	$2^{79.86}$	2^{23} CP
This paper	Biclique	31 (Full)	$2^{79.63}$	2^{23} CP

Fig. 1 PRESENT round function

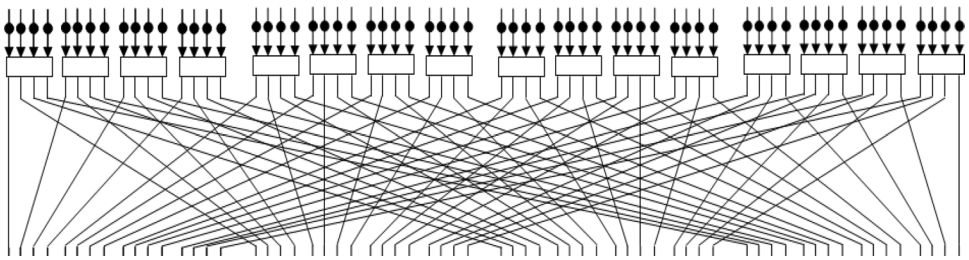

Table 2 Permutation table of PRESENT-80

ip	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
op	0	16	32	48	1	17	33	49	2	18	34	50	3	19	35	51
ip	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*	24	25	26	27	28	29	30	31
op	4	20	36	52	5	21	37	53	6	22	38	54	7	23	39	55
ip	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*	*40*	*41*	*42*	*43*	*44*	*45*	*46*	*47*
op	8	24	40	56	9	25	41	57	10	26	42	58	11	27	43	59
ip	*48*	*49*	*50*	*51*	*52*	*53*	*54*	*55*	*56*	*57*	*58*	*59*	*60*	*61*	*62*	*63*
op	12	28	44	60	13	29	45	61	14	30	46	62	15	31	47	63

of data permutation is shown in Fig. 1. The positions of input and output of the pLayer are shown in Table 2.

Key Scheduling

Since this paper is focused on the biclique cryptanalysis on PRESENT-80, key scheduling of 80-bit key variant is given here. In every round, the key is scheduled to get the key bits for the next round. Since the encryption is done on 64 bits of data, only 64 bits of key is used in each round. Let the 80-bit initial key be

$K = k_{79}k_{78}k_{77}\ldots k_2k_1k_0$. Then, ith round key is denoted as $K_i = K_{64}K_{63}K_{62}\ldots..K_2K_1K_0 = k_{79}k_{78}k_{77}\ldots k_{18}k_{17}k_{16}$, where $1 \le i \le 32$, where K_{32} is used for post-whitening.

After extracting the round key K_i, the key register is updated in three steps:

1. The key register bits are shifted 19 bit positions in the right circular direction.
2. The most significant 4 bits of key register is applied to the substitution box and substituted by its output.
3. Five key bits $k_{19}k_{18}k_{17}k_{16}k_{15}$ is exclusively ORed with 5-bit counter output with its MSB at the left end.

Biclique Cryptanalysis

The concept of Biclique cryptanalysis is derived from splice and cut at-tacks [23]. For proposing a biclique attack, the cipher is divided into subciphers and bicliques are constructed on the target subcipher, in such a way that the computational efficiency is increased.

Let $\boldsymbol{S}$ be the starting state of a set of elements, and $\boldsymbol{C}$ be the ending state of a set of another elements. Let the elements of $\boldsymbol{S}$ are denoted by S_j and the elements of $\boldsymbol{C}$ are denoted by C_i. If every element of $\boldsymbol{S}$ is connected with every element of $\boldsymbol{C}$, using some key $K[i, j]$, then 3-tuple of sets $[\{S_j\},\{C_i\},\{K[i, j]\}]$ forms a *d-dimensional biclique*, if.

For all $i,j \in \{0,\ldots 2^d - 1\}$, $C_i = f_{K<i,j>}(S_j)$.

A schematic view of biclique is shown in Fig. 2.

A group of keys $K[i, j]$ is defined by a biclique and can be calculated using the base key $K[0, 0]$ as well as the differences Δ_i and ∇_j.

$$K\left[i, j\right] = K[0, 0] \cdot \Delta_i \cdot \nabla_j$$

Let k be the key length and d be the dimension of the biclique. Initially, the total key space is divided into 2^{k-2d} subspaces of 2^{2d} keys. Now the cipher E is divided into three subciphers E_1, E_2 and B in such a way that E_1 converts a plain text P to the intermediate state V, E_2 converts V to another state S, and B converts S to cipher text C

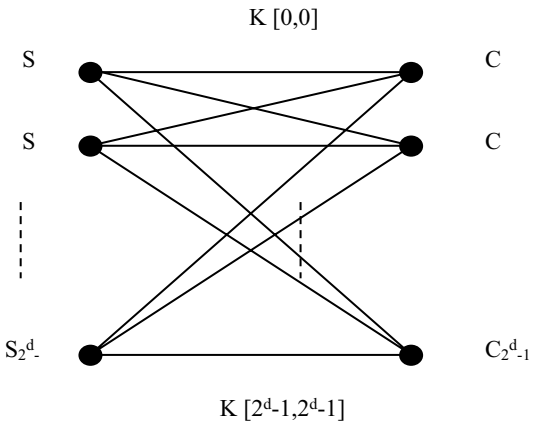

Fig. 2 Schematic view of bicliques

$$P \xrightarrow{E1} V \xrightarrow{E2} S \xrightarrow{B} C$$

The adversary creates biclique on an arbitrary part of the cipher, and for the remaining part, meet-in-the-middle attack matching with precomputation is carried out. In this paper, independent biclique attack is applied against PRESENT-80 cipher

Independent Bicliques

Bicliques can be constructed over any subcipher B from two differentials, one in the forward direction and another in the reverse direction. Here, we take a base computation and two differentials in opposite directions in such a way that the differentials do not overlap, of d bits. Let a 3-tuple $\{S_0, C_0, K[0, 0]\}$ be the base computation, where S_0 is converted to the cipher text C_0 by the key $K[0,0]$, i.e.

$$S_0 \xrightarrow{K[0,0]} C_0$$

Now we choose 2^d forward differentials Δ_i which produce cipher texts C_i from S_0

$$S_0 \xrightarrow{K[0,0]\oplus\Delta_i^K} C_0 \cdot \Delta_i = C_i \,(\text{through subcipher } B)$$

2^d backward differentials ∇_j are also chosen which produce intermediate data S_i from cipher text C_0.

$$S_j = S_0 \cdot \nabla_j \xleftarrow{K[0,0]\oplus\Delta_j^K} C_0\left(\text{through subcipher } B^{-1}\right)$$

If no active linear operations are overlapped on the trails of forward differential Δ_i and reverse differential ∇_j, then each of the input difference ∇_j and output difference Δ_i can be connected to each other. So, 2^{2d} independent trails (Δ_i, ∇_j) are obtained here.

$$S_0 \cdot \nabla_j \overset{K[0,0]\oplus\Delta_i^K\oplus\Delta_j^K}{\longleftrightarrow} C_0 \cdot \Delta_i \quad i,j \in \{0 \ldots 2^d - 1\}$$

From the above discussion, it is clear that, for a biclique of dimension d, 2^{2d} keys can be tested using 2×2^d computations. If the complete key space is divided into $22d$ keys, then the attack complexity comes down to 2^{k-2d} times of computation of E. The complexity of biclique construction and matching is to be taken in addition to this, for calculating the full complexity.

Matching with Precomputation

For the remaining part of the cipher, other than biclique, matching with precomputation is an efficient method to perform matching [10, 20] of selected bits. Between E_1 and E_2, an internal state V is selected by the adversary. The steps for the matching are given below

1. Using $K[i, 0]$, the adversary calculates the value of least significant 4 bits of round 18 output, in the forward direction, for all $i = 0$ to $2^4 - 1$. These values are stored as $\underset{v_i}{\rightarrow}$ in memory, along with the intermediate values. Mathematically,

   $\text{Pi} \overset{K[i,0]}{\rightarrow} \overrightarrow{v_{i,0}}$ (using the sub cipher E_1)

2. Calculation of 2^d values in the backward direction, from the state S_j, is also done

   $\overleftarrow{v_{0,j}} \overset{K[0,j]}{\longleftarrow} \text{S}_\text{j}$ (using the sub cipher E_2)

3. For the remaining $2^{2d} - 2^d$ computations, the above calculated values are reused.

   $\text{Pi} \overset{K[i,j]}{\rightarrow} \overrightarrow{v_{i,j}}$ and $\overleftarrow{v_{i,j}} \overset{K[i,j]}{\longleftarrow} S_j$ (using both E_1 and E_2)

Only part of the key schedule and round transformations which differ from the stored values needs to be recomputed. So, the computational complexity gets reduced significantly.

Complexity Calculations

The advantage of biclique cryptanalysis is the fact that 2^{2d} keys can be tested with $2 \times 2d$ computations. To cover the full key space, the adversary constructs 2^{k-2d} biclique. The full complexity of the analysis is

$$C_{\text{full}} = 2^{k-2d}\left(C_{\text{biclique}} + C_{\text{decrypt}} + C_{\text{precompute}} + C_{\text{recompute}} + C_{\text{falsepos}}\right),$$

where C_{biclique} is the cost for the construction of biclique, C_{decrypt} is the complexity of the oracle for decryption of 2^d cipher text, $C_{\text{precompute}}$ is the cost for computation of v, $C_{\text{recompute}}$ is the complexity of recomputation of 2^{2d} values $v_{i,j}$, C_{falsepos} is the complexity for elimination of false positives.

Proposed Attack

Construction of Biclique

Here, we propose an attack on PRESENT-80. First, we need to construct biclique over certain number of rounds. Here, rounds 1–3 are selected to construct a four-dimensional biclique. In rounds 0–3, partial secret keys used are

RoundKey 1: $k_{79}k_{78}\ldots k_{17}k_{16}$
RoundKey 2: $k_{18}k_{17}\ldots k_1k_0k_{79}\ldots k_{36}k_{35}$
RoundKey 3: $k_{37}k_{36}\ldots k_1k_0k_{79}\ldots k_{55}k_{54}$
RoundKey 4: $k_{56}k_{55}\ldots k_1k_0k_{79}\ldots k_{74}k_{73}$

The key bit groups selected to construct biclique on full PRESENT-80 over rounds 1–3 are $\{k_{10}, k_{11}, k_{12}, k_{13}\}$ and $\{k_{69}, k_{70}, k_{71}, k_{72}\}$. That means the forward differential Δ_i and the reverse differential ∇_j are created using $\{k_{10}, k_{11}, k_{12}, k_{13}\}$ and $\{k_{69}, k_{70}, k_{71}, k_{72}\}$, respectively. We take a subcipher f with rounds 1–3. Here, C_0 is fixed first and S_0 is derived from C_0, i.e. $S_0 = f^{-1}_{K\langle 0,0\rangle}(C_0)$. In each round, the $\{k_{10}, k_{11}, k_{12}, k_{13}\}$ has a difference i and $\{k_{69}, k_{70}, k_{71}, k_{72}\}$ has a difference j. All other bits will have a zero difference.

Let us see how the forward differential key bits propagate in 1–4 rounds. Corresponding data bit position also is given

1st round: $\{k_{10}, k_{11}, k_{12}, k_{13}\}$ – {-, -, -, -}
2nd round: $\{k_{71}, k_{72}, k_{73}, k_{74}\}$ – {55, 56, 57, 58}
3rd round: $\{k_{52}, k_{53}, k_{54}, k_{55}\}$ – {36, 37, 38, 39}
4th round: $\{k_{33}, k_{34}, k_{35}, k_{36}\}$ – {17, 18, 19, 20}

The backward differential key bits propagate in 1–4 rounds

1st round: $\{k_{69}, k_{70}, k_{71}, k_{72}\}$ – {53, 54, 55, 56}
2nd round: $\{k_{50}, k_{51}, k_{52}, k_{53}\}$ – {34, 35,36, 37}
3rd round: $\{k_{31}, k_{32}, k_{33}, k_{34}\}$ – {15, 16, 17, 18}
4th round: $\{k_4, k_5, k_6, k_7\}$ – {-, -, -, -}

where '-' denotes that the key bits do not appear in the operation. Figure 3 shows the four-dimensional biclique drawn from the above given calculations

From Fig. 3, it can be seen that only 23 bits are affected by the propagation of Δ_i. Thus, the data complexity can not exceed 2^{23}. Here, PRESENT-80 is divided into three

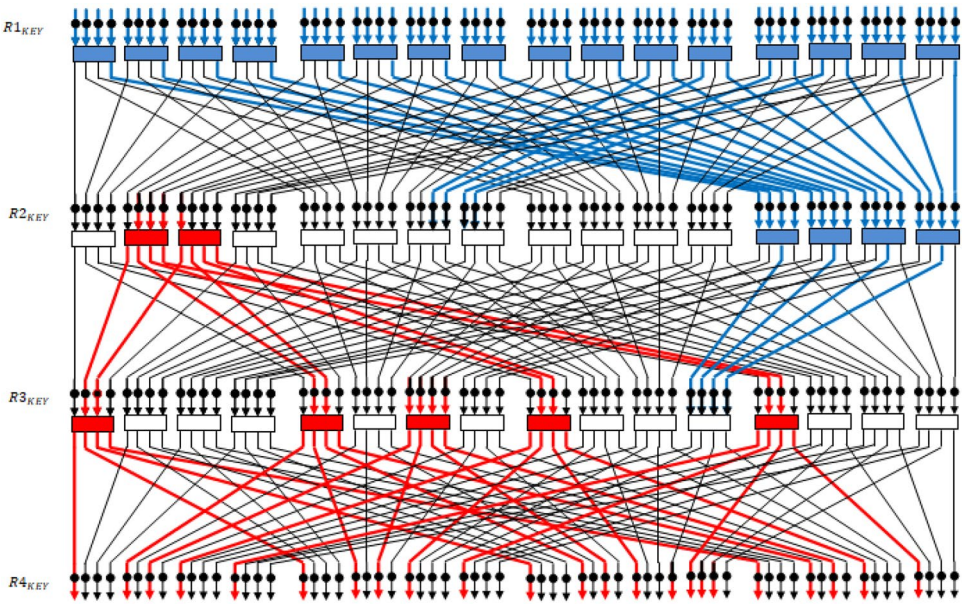

Fig. 3 Four-dimensional biclique on PRESENT-80

subciphers B, E_1 and E_2 as explained in Sect. 3. Encryption of rounds 1–3, 4–17 and 18–31 is carried out by B, E_1 and E_2, respectively.

Matching for the Remaining Rounds

It can be calculated that recomputation in the forward direction starts from k_{57}, k_{58}, k_{59} and k_{60}, from round 5 as shown in blue trails in Fig. 4. The 4th round is not considered because the key bits selected to do the matching do not appear in that

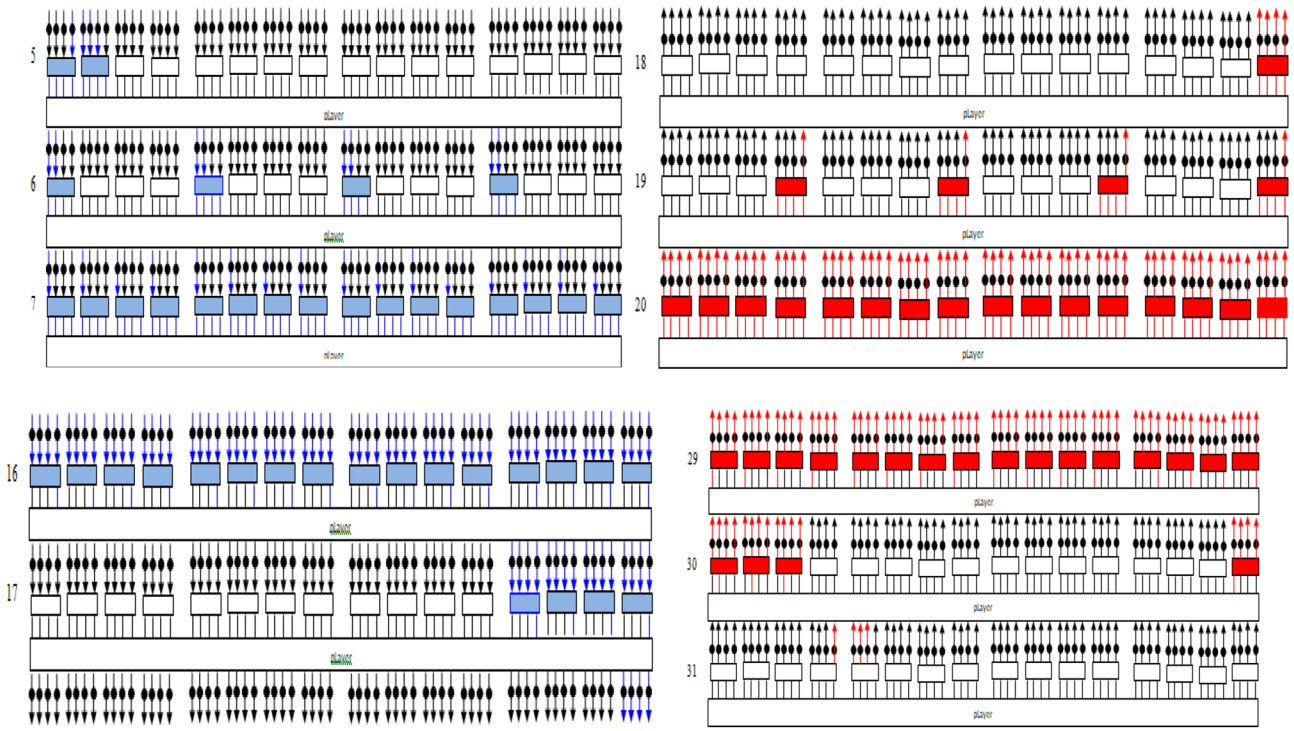

Fig. 4 Recomputation in the forward and reverse directions

round. Recomputation in the backward direction starts from $k_{45}, k_{46}, k_{47}, k_{48}$ from round 31 as shown in red trails in Fig. 4

As explained in Sect. 3, matching with precomputation is applied on the remaining number of rounds. Figure 3 shows the forward and reverse computations for the rounds 4–17 and 18–31, respectively. Here, matching is done on the bits v_3, v_2, v_1, v_0 of round 17 where the intermediate state v is given by $v = \{ v_{63}, v_{62}, \ldots v_1, v_0 \}$.

Complexity of the Attack

As explained in Sect. 2, the total attack complexity is given by

$$C_{\text{full}} = 2^{k-2d}\left(C_{\text{biclique}} + C_{\text{decrypt}} + C_{\text{precompute}} + C_{\text{recompute}} + C_{\text{falsepos}}\right)$$

C_{biclique} is the cost for constructing the biclique. Since the biclique dimension is 4 and the number of rounds covered by the biclique is 3, $C_{\text{biclique}} = 2.2^4 \times 3/31 = 2^{1.63}$.

C_{decrypt} is computations required for decrypting d bits through a decryption oracle. So, here $C_{\text{decrypt}} = 2^4$.

$C_{\text{precomputation}}$ is the computational cost for the forward and reverse calculation through the subcipher E_1 and E_2, respectively, for the rounds which are not covered by biclique. Since in this attack 28 rounds are covered by E_1 and E_2, $C_{\text{precomputation}} = 2^4 \times 28/31 = 2^{3.85}$.

$C_{\text{recomputation}}$ is the computational cost for recomputing those parts which differ from the stored values due to the injected key differences Δ_i and ∇_j in the forward and reverse direction, respectively. Here, we consider PRESENT cipher as a nibble-based one since the sbox is 4-bit input–4-bit output element. So, the number of sboxes in round transformation and key schedule is calculated here to find the recomputation complexity.

From Fig. 4,

The number of active sboxes in the forward direction is $2+4+9\times16+4=154$

The number of active sboxes in the reverse direction is $1+4+10\times16+4=169$

Since key diffusion is very slow, only one sbox is to be recomputed for the key schedule

So, the total sboxes to be recomputed $=154+169+1=324$

Total number of sboxes in PRESENT-80 $=16\times31+1\times31=527$ (16 sboxes for round transformation and 1 for key schedule in each round

So, the complexity of recomputation $=2^8\times324/527=2^{7.3}$.

C_{falsepos} is the computational complexity to eliminate false positives. Since matching is done over 4 bits, C_{falsepos} will be 2^4 full PRESENT 80 encryptions

Total complexity of biclique attack $=2^{80-2.4}(2^{1.63}+2^4+2^{3.85}+2^{7.3}+2^4)=2^{79.63}$. The data complexity is 2^{23}, and memory required is 2^4 nibbles.

Conclusion

A new biclique attack on PRESENT-80 is proposed in this paper. We used the concept of independent bicliques here. A four-dimensional biclique is constructed over the rounds 1–3, and matching with precomputation is applied for the remaining rounds. The complexity of the attack is the lowest amongst the attacks published so far. The attack is successfully implemented with a data complexity of 2^{23} and a time complexity of $2^{79.63}$ encryption cycles.

References

1. Bogdanov A, Knudsen LR, Leander G, Paar C, Poschmann A, Robshaw MJB, Seurin Y, C Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In: Paillier P, Verbauwhede I (eds) CHES, volume 4727 of lecture notes in computer science. Springer; 2007, p. 450–466.
2. Shibutani K, Isobe T, Hiwatari H, Mitsuda A, Akishita T, Shirai T. Piccolo: an ultra-lightweight blockcipher. In: Preneel and Takagi, p. 342–357.
3. De Canniere C, Dunkelman O, Knezevic M. KATAN and KTANTAN - a family of small and efficient hardware-oriented block ciphers. In: Clavier C, Gaj K (eds) CHES, volume 5747 of Lecture Notes in Computer Science. Springer; 2009, p. 272–288.
4. Gong Z, Nikova S, Law YW. KLEIN: a new family of lightweight block ciphers. In: Juels A, Paar C (eds) RFIDSec, volume 7055 of Lecture Notes in Computer Science. Springer; 2011, p. 1–18.
5. Guo J, Peyrin T, Poschmann A, Robshaw MJB. The LED block cipher. In: Preneel and Takagi, p. 326–341.
6. Hong D, Sung J, Hong S, Lim J, Lee S, Koo B, Lee C, Chang D, Lee J, Jeong K, Kim H, Kim J, Chee S. Hight: a new block cipher suitable for low-resource device. In: Goubin L, Matsui M (eds) CHES, volume 4249 of lecture notes in computer science. Springer, 2006, p. 46–59.
7. Khovratovich D, Rechberger C, Savelieva A. Bicliques for preimages: attacks on skein-512 and the SHA-2 Family. Cryptology ePrint archive, report 2011/286, 2011. http://eprint.iacr.org/.
8. Aoki K, Sasaki Y. Preimage attacks on one-block MD4, 63-step MD5 and more. In: Selected areas in cryptography'08; 2008, p. 103–119.
9. Bogdanov A, Rechberger C (2011) A 3-subset meet-in-the-middle attack: cryptanalysis of the lightweight block cipher KTANTAN. In: Biryukov A, Gong G, Stinson DR (eds) Selected areas in cryptography. SAC 2010. Lecture notes in computer science, vol 6544. Springer, Berlin, Heidelberg.
10. Bogdanov A, Khovratovich D, Rechberger C. Biclique cryptanalysis of the full AES. Cryptology ePrint archive, report 2011/449; 2011. http://eprint.iacr.org/.
11. Hong D, Koo B, Kwon D. Biclique attack on the full HIGHT. In: Kim H (eds) ICISC, volume 7259 of lecture notes in computer science. Springer; 2011, p. 365–374.
12. Wang Y, Wu W, Yu X. Biclique cryptanalysis of reduced-round piccolo block cipher. In: Ryan MD, Smyth B, Wang G (eds) ISPEC, volume 7232 of lecture notes in computer science. Springer, 2012; p. 337–352.
13. Çoban M, Karakoc F, Boztaş Ö. Biclique cryptanalysis of TWINE. Cryptology ePrint archive, report 2012/422, 2012. http://eprint.iacr.org/.

14. Mala H. Biclique cryptanalysis of the block cipher SQUARE. Cryptology ePrint archive, report 2011/500, 2011. http://eprint.iacr.org/.
15. Chen S, Xu T. Biclique attack of the full ARIA-256. IACR cryptology ePrint archive, 2012:11, 2012.
16. Wang M. Differential Cryptanalysis of reduced-round PRESENT. In: Vaudenay S (eds) AFRICACRYPT, volume 5023 of Lecture Notes in Computer Science. Springer; 2008, p. 40–49.
17. Ohkuma K (2009) Weak keys of reduced-round PRESENT for linear cryptanalysis. In: Jacobson MJ, Rijmen V, Safavi-Naini R (eds) Selected areas in cryptography. SAC 2009. lecture notes in computer science, vol 5867. Springer, Berlin, Heidelberg.
18. Cho JY. Linear cryptanalysis of reduced-round PRESENT. In: Pieprzyk J (eds) CT-RSA, volume 5985 of lecture notes in computer science. Springer; 2010, p. 302–317.
19. 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.
20. 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.
21. Albrecht M, Cid C. Algebraic techniques in differential cryptanalysis. Cryptology ePrint archive, report 2008/177, 2008. http://eprint.iacr.org/.
22. Collard B, Standaert F-X. A statistical saturation attack against the block cipher PRESENT. In: Fischlin M (eds) CT-RSA, volume 5473 of lecture notes in computer science. Springer; 2009, p. 195–210.
23. Khovratovich D, Rechberger C. A splice-and-cut cryptanalysis of the AES. IACR cryptology ePrint archive, 2011:274, 2011. http://eprint.iacr.org/2011/274.