Two Kinds of Biclique Attacks on Lightweight Block Cipher PRINCE

Zheng Yuan[1,2],✉ Zhen Peng[1,2],✉ Haiwen Ou[1,2]

1. Beijing Electronic Science &Technology Institute, Beijing ,100070,China
2. Xidian University, Xi'an, 710071, China

zyuan@tsinghua.edu.cn pengzhen0822@126.com

Abstract: PRINCE is a modern involutive lightweight block cipher proposed by Rechberger in Asiacrypt 2012[6], then PRINCE has been widely used in many constrained devices. PRINCE uses the FX construction, in which one part of the cipher is considered as core cipher and remaining parts are used for whitenings before and after the core. Farzaech et al. gave the security evaluations of PRINCEcore against biclique and differential cryptanalysis, respectively[10]. They presented an independent-biclique attack on the full version with computational complexity $2^{62.72}$ and data complexity 2^{40}. Inspired from their work, by better selections of differential characteristics in the biclique construction, we give another balanced biclique attack on PRINCEcore with lower computation complexity and data complexity than previous results in [10]. The computational complexity and data complexity of our attack is $2^{62.67}$ and 2^{32} , respectively. Then, we first illustrate a star-based biclique attack on PRINCEcore. The computational complexity of star-based biclique attack is $2^{63.02}$ and the required data is only a single plaintext-ciphertext pair. This is the optimal data complexity among the existing results of full round attack on PRINCEcore.

Keywords: biclique cryptanalysis, star-based biclique, computational complexity , data complexity, PRINCE

1 Introduction

Lightweight Block Cipher PRINCE. Lightweight block cipher with short block length and key length is suitable for extremely constrained environment. The best studied lightweight block ciphers are mCrypton[1],CLEFIA[2],Piccolo[3],PRESENT[4],KLEIN[5],PRINCE[6] and LED[9]. PRINCE was proposed by Rechberger et al. in Asiacrypt 2012. It has 64-bit block and 128-bit key, and uses so-called FX-construction. PRINCE is consisted of a 64-bit core cipher PRINCEcore and two whitenings before and after the PRINCEcore. PRINCEcore holds the major encryption logic of PRINCE, so the security of PRINCE mainly depends on the properties of PRINCEcore. There are some cryptanalysis such as differential cryptanalysis[10], algebraic cryptanalysis[11], and biclique cryptanalysis[10] on PRINCE. Farzaneh et al. gave the biclique attack on PRINCEcore with computational complexity $2^{62.72}$ and data complexity 2^{40} [10]. Farzaneh also presented upon 2-round attack of differential cryptanalysis of PRINCEcore with computational complexity $2^{32.44}$ and data complexity 2^{32}, and upon 4-round attack of differential cryptanalysis with computation complexity $2^{56.26}$ and data complexity 2^{48}. Lilang gave algebraic attack on PRINCE[11], in which all the key bits of 5-round PRINCEcore could be obtained based on the known plaintexts and all the key bits of 6-round PRINCE can be successful recovered under the chosen plaintexts. Anne used multiple differentials and exploited some properties of PRINCE for recovering the whole key (10 rounds)[12]. The attack could be extended up to 11 rounds with a data complexity of $2^{59.81}$ and a time complexity of $2^{62.43}$.

Biclique Cryptanalysis. Biclique cryptanalysis was first proposed by Khovratovich et al. in 2011[7], they demonstrated the first single-key attacks on full rounds of three variants of the AES with a significant advantage over exhaustive search. In ICISC2014, Bogdanov proposed a star-based biclique[8] with just one state in one vertex set and 2^{2d} states in the other ones. They implemented a star-based independent biclique attack on AES-128/192/256 and achieved the theroetically minimal data complexity. In 2015, Zheng Yuan gave a star-based independent biclique attack on full rounds SQUARE [13], which is the second application of star-based biclique attack to a block cipher.

Our contribution. Stimulated by the balanced independent biclique cryptanalysis of PRINCEcore[10] and star-based independent biclique crypanalysis of AES [8], in this paper we first present another balanced independent biclique attack on PRINCEcore and then give the first star-based independent biclique attack on PRINCEcore. Both of them are full rounds attack of PRINCEcore. Our balanced independent biclique attack is superior to the previous balanced biclique attack with computational complexity $2^{62.67}$ and data complexity 2^{32}; Our star-based independent biclique attack, first proposed on PRINCEcore, has computation complexity $2^{63.02}$ and required data is only a single plaintext-ciphertext pair. To be the best of our knowledge, this is the optimal data complexity among the existing results of full rounds attack. We can note that the computational complexity and data complexity are influenced by the biclique construction.

Outline. This paper is organized as follows: Section 2 describes the lightweight block cipher PRINCE. Section 3 and section 4 presents the balanced independent biclique attack and star-based independent biclique attack on PRINCEcore, respectively, and section 5 summarizes the whole paper.

2 Description of Lightweight Block Cipher PRINCE

PRICNE is a 64-bit block cipher with a 128-bit key and the construction of it is decipted in Fig1:

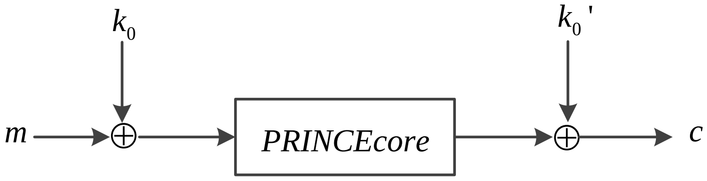

Fig1. The construction of PRINCE

2.1 The Key Schedule

The key schedule of PRINCE is not so complicated. Firstly, the 128-bit key k is split into two 64-bit words

$$k = k_0 \| k_1$$

Then, it is extended to192-bit by a linear mapping of the form

$$k = (k_0 \| k_1) \longrightarrow (k_0 \| k_0' \| k_1) := (k_0 \| (k_0 >>> 1) \oplus (k_0 >>> 63) \| k_1)$$

where the 64-bit k_1 is used for the PRINCEcore, the 64-bit k_0 and k_0' are used to wrap the core with two additions, the pre- and post- whitening.

2.2 Round Transformation

As is described in Fig2, we can see that PRINCEcore consists of the first five rounds, the middle rounds, the last five rounds and two additional XORs with the key and a different round constant. Every round in PRINCEcore constains five steps. Round operation in the last five rounds is the inverse of the first five rounds.

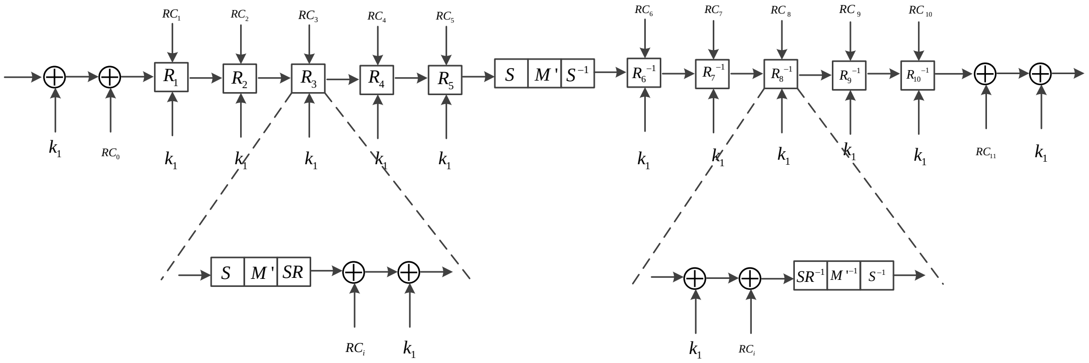

Fig2. PRINCEcore

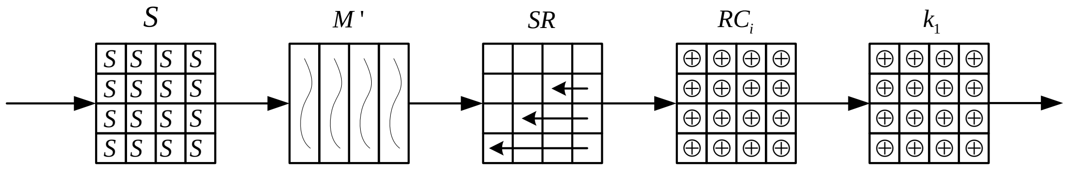

Fig3. Single round of PRINCEcore

S **-layer.** PRINCEcore uses one 4-bit Sbox. Every nibble in the state is replaced by the nibble generated after using S-box.

M/M' **-layer.** M/M' layer is a linear layer. M' is a 64×64 block diagonal matrix. M_0 and M_1 are two 16×16 submatrices which are placed on the diagonal of M'.

$$M'(x) = M' \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} M_0 & 0 & 0 & 0 \\ 0 & M_1 & 0 & 0 \\ 0 & 0 & M_1 & 0 \\ 0 & 0 & 0 & M_0 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = M_0(x_1) \| M_1(x_2) \| M_1(x_3) \| M_0(x_4)$$

However, M' is only used in the middle round. To ensure the α -reflection property M' need to be an involution. In the first five rounds, linear layer uses matrix M which can be derived from M'. $M = SR \circ M'$, SR is a shift row operation.

SR **-layer.** SR operation in the PRINCEcore is as same as the one in the AES. Row i of the state is rotated i cell positions to the left, $i = 0,1,2,3$.

k_i **-add.** In the k_i -add step, a 64-bit state is Xor with the 64-bit subkey k_1.

RC_i **-add.** In the RC_i -add step, the verying 64-bit round constants RC_i is Xor with the 64-bit state.

3 Balanced Independent Biclique Attack on PRINCEcore

Inspired from [10], we give another balanced biclique attack on the full PRINCEcore using the better differential characteristics in the biclique construction. We construct a biclique over the initial round of PRINCEcore and match with precomputations technique on the remaining rounds.

3.1 Key Partitioning

We divide the 64-bit key space into 2^{48} 16-nibble key groups. The base keys $K[0,0]$ are all 2^{48} 16-nibble values with four nibbles fixed to 0 and all other nibbles in the state take on all possible values. The 2^{16} keys in a set $K[i, j]$ are defined relative to the base key $K[0,0]$,and two difference Δ_i^K and ∇_j^K, where $i, j \in \{0,...,2^8 - 1\}$ and $i = (i_1 \| i_2)$ and $j = (j_1 \| j_2)$.

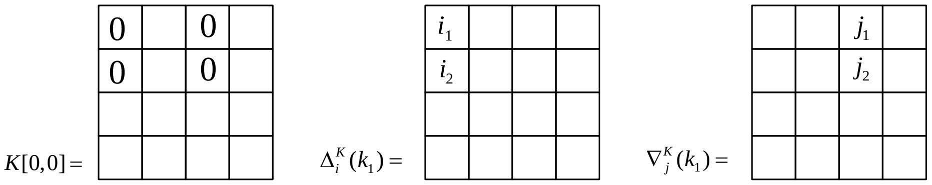

Fig4. Key Partitioning

3.2 Constructing Single Round Independent-Biclique of Dimension 8

Here, we construct an independent Biclique on the first round of PRINCEcore. We consider the block cipher as a composition of three subciphers: $e = g_2 \circ g_1 \circ f$.

$$P \xrightarrow{f} S \xrightarrow{g_1} V \xrightarrow{g_2} C$$

According to section 3.1, Δ_i -trail actives nibble 0 and 1 , ∇_j -trail actives nibble 8 and 9. we determine 2^8 plaintexts P_i and 2^8 internal states S_j that satisfy the definition of the biclique $P_i \xrightarrow[f]{K[i,j]} S_j$ $(i, j \in \{0,...,2^8-1\})$. P_i is the plaintext and S_j is the internal state which refers to the output of round 1 in this paper. Fig5 illustrates the 1-round independent biclique on PRINCEcore, including base computation, Δ_i-differentials and ∇_j-differentials.

Step 1. Fix $P_0 = 0_{(64)}$, and derive $S_0 = f_{K[0,0]}(P_0)$ with key $K[0,0]$. This process is called base computation (Fig5, left).

Step 2. Encrypt P_0 under different keys $K[0,j]$ ($j \in \{0,...,2^8-1\}$) and derive $P_0 \xrightarrow[f]{K[0,j]} S_j$ (Fig5,middle). This process has the same starting point and ending point with base computation, so the computation complexity of this process is determined by the difference between $K[0,j]$ and $K[0,0]$.

Step3. Decrypt S_0 under different keys $K[i,0]$ $(i \in \{0,...,2^8-1\})$ and derive $P_i \xleftarrow[f^{-1}]{K[i,0]} S_0$ (Fig5,right). This process are from over the same part of the cipher, so the computation complexity of this process is determined by the difference between $K[i,0]$ and $K[0,0]$.

From Fig5, we can see that both the forward differential trails and the backward differential trails do not share any non-linear components during the first round. Therefore, it is easy to find that $P_i \xrightarrow[f]{K[i,j]} S_j$ $(i, j \in \{0,...,2^8-1\})$ is true. So we construct an independent biclique for every key group. In forward differential, differences are injected into nibble 0 and nibble 1 of key k_1 and it affects five nibbles after one round. In backward differential, differences are injected into nibble 8 and nibble 9 of key k_1 and it influences eight nibbles plaintexts.

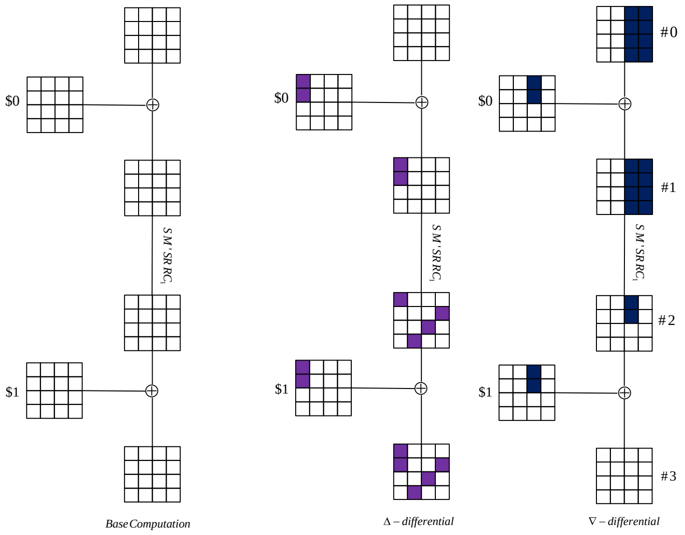

Fig5.Balanced independent biclique over the first round of PRINCEcore

3.3 Matching with Precomputation

In this section, we apply matching with precomputation technique on the remaining rounds to reduce computational complexity. Because PRINCEcore is an involutive structure, we choose the two nibbles of the state after the middle round, i.e, before 6-th round, as the matching values.

Forward computation. Firstly, we encrypt S_j under key $K[0,j]$ and store 2^8 precomputation values $v_{0,j}$, $S_j \xrightarrow[g_1]{K[0,j]} v_{0,j}$. Then, we encrypt the same S_j under keys $K[i,j]$, $S_j \xrightarrow[g_1]{K[i,j]} v_{i,j}$. Because both the processes have the same starting point, the recomputation complexity is determined by the difference between $K[0,j]$ and $K[i,j]$.

Backward computation. Firstly ,we should ask the oracle to encrypt plaintext P_i with the secret key $K_{\sec ret}$ and obtain ciphertext C_i . Then, we decrypt C_i under key $K[i,0]$ and store 2^8 precomputation values $v_{i,0}$, $v_{i,0} \xleftarrow[g_2^{-1}]{K[i,0]} C_i$. Lastly, we decrypt the same C_i under keys $K[i,j]$ $v_{i,j} \xleftarrow[g_2^{-1}]{K[i,j]} C_i$. Because both the processes have the same ending point, the recomputation complexity is determined by the difference between $K[i,0]$ and $K[i,j]$.

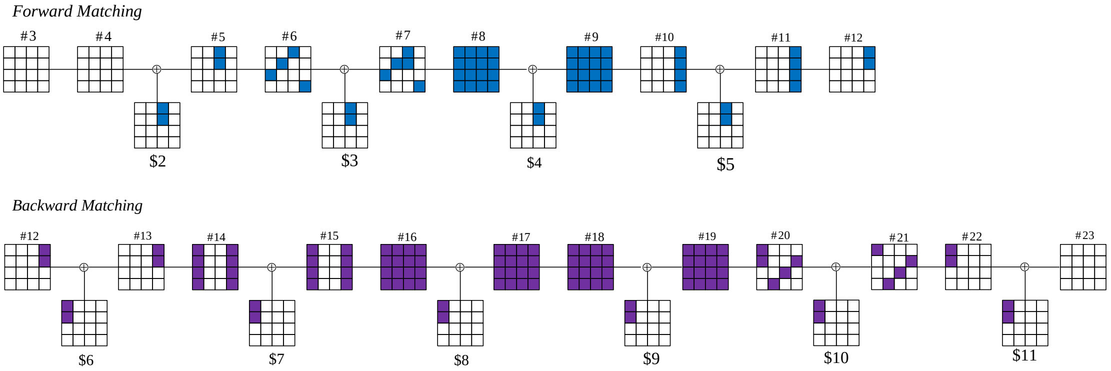

Fig6. Recomputations for PRINCEcoer in forward and backward direction of balanced biclique attack.

3.4 Complexity of Our Attack

There are 192 S/S^{-1} -boxes in total in all round transformation of PRINCEcore. In the forward matching, we only need to compute 29 S-boxes; Fig6 top illustrates the active nibbles in the S operations in the states directly after the key additions during the matching. In the backward matching, we only need to compute 46 S-boxes. Fig6 bottom illustrates the active nibbles in the S operations in the states directly after the key additions during the matching. During matching with precomputation, there are 75 S-boxes to be recomputed.

Computation Complexity. For a key group of 2^{16} keys, the complexity of recomputation is

$C_{recomp} = 2^{16} \cdot \frac{75}{192} \approx 2^{14.64}$. The complexity for constructing a biclique is $C_{biclique} = 2 \cdot 2^{8} \cdot \frac{1}{12} \approx 2^{5.42}$.The complexity of precomputations is $C_{precomp} = 2^{8} \cdot \frac{11}{12} \approx 2^{7.88}$. The complexity to eliminate false positives is $C_{falsepos} = 2^{8}$. Therefore, the total computational complexity is

$$\begin{aligned} C_{full} &= 2^{k-2d}(C_{biclique} + C_{recomp} + C_{precomp} + C_{falsepos}) \\ &= 2^{48} \cdot (2^{5.42} + 2^{14.64} + 2^{7.88} + 2^{8}) \\ &= 2^{62.67} \end{aligned}$$

The data complexity is determined by the encrypted plaintexts. We fix $P_0 = 0_{(64)}$ for every biclique and all the plaintexts P_i ($i \in \{0,...,2^8 - 1\}$) share eight nibbles, so the data complexity does not exceed 2^{32} chosen plaintexts. During the precomputation , we need to store 2^8 values, so the memory complexity is 2^8.

4 Star-Based Independent-Biclique Attack on PRINCEcore

Inspired from their work [8], we first give a star-based biclique attack on full rounds of PRINCEcore. A star-based biclique is different from balanced biclique with just one state in one vertex set and 2^{2d} states in the other ones. In this section, we construct a 1-round star-based biclique over the first round of PRINCEcore.

4.1 Key Partitioning

We divide the 64-bit key space into 2^{48} 16-nibble key groups. The form of partition is as same as section 3.1, so we do not describe key partitioning in this section.

4.2 Constructing Single Round Star-Based Independent-Biclique of Dimension 8

Similar to balanced biclique, stars can be constructed efficiently from independent sets of differentials. Unlike balanced biclique, the necessary form of differentials is different. For PRINCEcore, we place the star at the beginning of the cipher, let x be the plaintext and $y_{i,j}$ be the output of round 1 encryption. Fig7 shows the 1-round star-based independent biclique, including the base computation, Δ_i-differentials and ∇_j-differentials. Both the differentials do not share any non-linear components.

Step1. Let x_0 be the plaintext, and obtain $y_{0,0} = f_{K[0,0]}(x_0)$ with key $K[0,0]$. This process is called base computation (Fig7, left).

Step2. Encrypt x_0 under different keys Δ_i^K ($i \in \{0,...,2^8 - 1\}$) and obtain $x_0 \xrightarrow[f]{\Delta_i^K} y_i$ (Fig7,middle). This process has the same starting point and ending point with base computation.

Step3. Encrypt x_0 from over the same part of the cipher as step 2 under different keys ∇_j^K ($j \in \{0,...,2^8 - 1\}$) and obtain $x_0 \xrightarrow[f]{\nabla_j^K} y_j$ (Fig7,right). This process also has the same starting point and ending point with base computation.

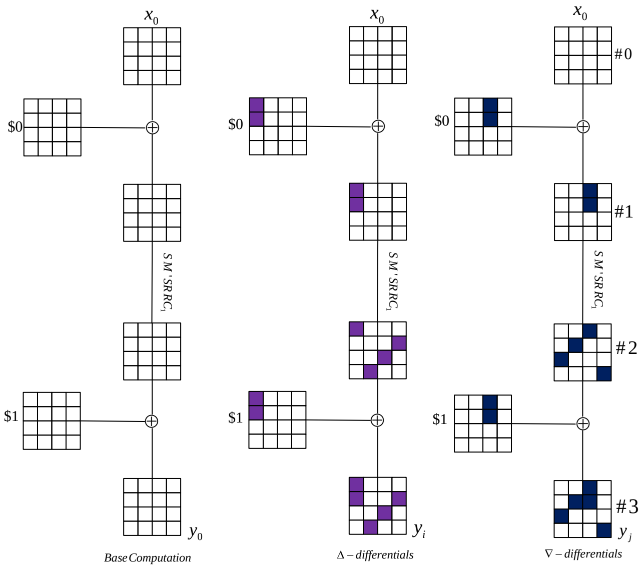

Fig7. Star-based independet biclique over the first round of PRINCEcore

From Fig7, we can observe that both the differentials do not share any non-linear components during the first round. So, it is easy to prove that $x \xrightarrow[f]{K[0,0]\oplus\Delta_i^K\oplus\nabla_j^K} y_{i,j}\ (i, j \in \{0,...,2^8-1\})$ is true. Therefore, we successfully construct a star-based independent biclique for every key group.

4.3 Matching with Precomputation

In this section, we apply matching with precomputation technique on the remaining rounds. We choose the same nibbles of the state after the middle round, i.e, before 6-th round, as the matching values.

Forward matching. In the forward direction of matching, starting in round 2, a part of the state has to be recomputed. Because difference propagation in these differentials over one round is non-overlapping, no S-boxes has to be recomputed in 2-th round. However, starting in 3-th round and forwards, the propagation affects the whole state (Fig8, top). So 54 S-boxes have to be recomputed in the forward direction of matching.

Backward matching. In the backward direction of matching, 42 S-boxes need to be recomputed (Fig8, bottom).

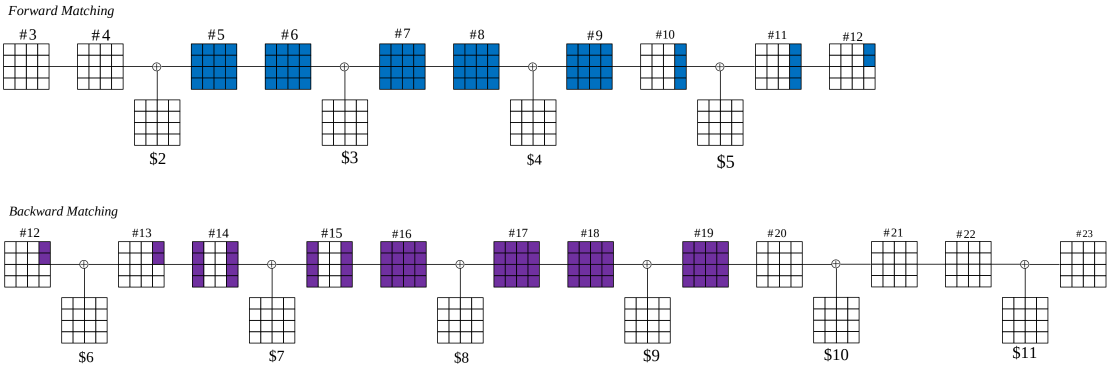

Fig8. Recomputations for PRINCEcore in forward and backward direction of star-based biclique attack.

4.4 Complexity

During the matching with precomputation, 91 S-boxes have to be recomputed.

Computation Complexity. For a key group of 2^{16} keys, the complexity of recomputation is $C_{recomp}=2^{16}\cdot\frac{96}{192}\approx 2^{15}$. The effort for constructing one Biclique $C_{biclique}=2\cdot 2^{8}\cdot\frac{1}{12}\approx 2^{5.42}$. The complexity of precomputations is $C_{precomp}=2^{8}\cdot\frac{11}{12}\approx 2^{7.88}$. The complexity to eliminate false positives is $C_{falsepos}=2^{8}$. Therefore, the total computation complexity is

$$\begin{aligned}C_{full}&=2^{k-2d}(C_{bilique}+C_{recomp}+C_{precomp}+C_{falsepos})\\&=2^{48}\cdot(2^{5.42}+2^{15}+2^{7.88}+2^{8})\\&=2^{63.02}\end{aligned}$$

The data complexity is determined by the encrypted plaintexts. We let x be the plaintext, so the data complexity will be 1. One known plaintext-ciphertext pair can sometimes be enough, and two known plaintext-ciphertext pairs yield a success probability of practically1.During the precomputation , we need to store 2^{8} values, so the memory complexity is upper bounded by 2^{8}.

5 Conclusion

In this paper, we concentrate on independent biclique attack on full rounds PRINCEcore. Firstly we give another balanced-biclique attack on PRINCEcore, compared with previous biclique attack, our attack has an advantage of computation complexity and data complexity. Then we first utilize star-based biclique (unbalanced biclique)attack to reduce the data complexity to the theoretically attainable minimum, compared with other existing attack results, the data complexity of our attack is optimal. It's worth mentioning that the structure of biclique is important for the data complexity of the attack whereas the length of the biclique seems to be correlated with the computation complexity.

References

[1] Chae Lim and Tymur Korkishko. mCrypton-A Lightweight Block Cipher for Security of Low-Cost RFID Tags and Sensors. In *Information Security Applications*, volume 3786 of *LNCS*, pages 243–258. Springer, 2006.

[2] Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai, and Tetsu Iwata. The 128-bit Block Cipher CLEFIA (Extended Abstract). In Fast Software Encryption-FSE 2007, volume 4593 of LNCS, pages 181-195. Springer, 2007.

[3] Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai. Piccolo: An Ultra-Lightweight Blockcipher. In Cryptographic Hardware and Embedded Systems - CHES 2011, volume 6917 of LNCS, pages 342-57. Springer, 2011.

[4] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, AxelPoschmann, Matthew J. B. Robshaw, Yannick Seurin, and Charlotte Vikkelso. PRESENT: An Ultra-Lightweight Block Cipher. In Cryptographic Hardware and Embedded Systems-CHES 2007, volume 4727 of Springer LNCS, pages 450-466,2007.

[5] Zheng Gong, Svetla Nikova, and Yee Wei Law. KLEIN: A New Family of Lightweight Block Ciphers. In RFID Security and Privacy-RFIDSec 2011, volume 7055 of LNCS, pages 1-18. Springer, 2011.

[6] Julia Borghoff, Anne Canteaut, Tim G”uneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalcin. PRINCE-A Low-latency Block Cipher for Pervasive Computing Applications. Cryptology ePrint Archive, Report 2012/591, 2012. http://eprint.iacr.org/.

[7] Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique Cryptanalysis of the Full AES. Cryptology ePrint Archive, Report 2011/449, 2011. http://eprint.iacr.org/.

[8] Andrey Bogdanov,Donghoon Chang, Mohona Ghosh et al. Biclqiues with minimal Data and Time Complexity for AES.ICISC2014.

[9] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The LED Block Cipher. In Cryptographic Hardware and Embedded Systems-CHES 2011, volume 6917 of LNCS, pages 326-341. Springer, 2011

[10] Farzaneh Abed, Eik List, Stefan Lucks. On the security of the Core of PRINCE Against Biclique and Differential Cryptanalysis. Lacr Cryptology Eprint Archive.2012.

[11] Li Lang, Du Guo-quan, ZENG Ting,et.al. Research on the PRINCE Algebraic Attack . MATHEMATICS IN PRACTICE AND THEORY, Vol.45, No.5, pp153-159, 2015.

[12] Anne Canteaut, Thomas Tuhr, Henri Gilbert et.al. Multiple differential cryptanalysis of round-reduced PRINCE(Full version),FSE 2014.

[13] Zheng Yuan, Zhen Peng, Ming Mao. A Star-based Independent Biclique Attack on Full

Rounds SQUARE. http://eprint.iacr.org/