Advanced Science and Technology Letters
Vol.51 (ITCS 2014), pp.250-255
http://dx.doi.org/10.14257/astl.2014.51.57

Meet-in-the-middle Attack on the 6-round Variant of the Block Cipher PRINCE

Yasutaka Igarashi[1], Toshinobu Kaneko[2], Satoshi Setoguchi[1], Seiji Fukushima[1], and Tomohiro Hachino[1]

[1] Kagoshima University, 1-21-40 Korimoto, Kagoshima, 890-0065 Japan
{igarashi,fukushima,hachino}@eee.kagoshima-u.ac.jp
[2] Tokyo University of Science, 2641 Yamazaki, Noda, Chiba, 278-8510 Japan
kaneko@ee.noda.tus.ac.jp

Abstract. We show a meet-in-the-middle (MITM) attack on the 6-round variant of the block cipher PRINCE. PRINCE is proposed by Borghoff et al. in 2012, which applies substitution-permutation network (SPN) with 64-bit data block and 128-bit secret key. MITM attack was proposed by Diffie and Hellman in 1977 as a generic method to analyze symmetric-key cryptographic algorithms. In this paper we show that PRINCE can be attacked with 2^{32} words of memory, 19 pairs of known plain and cipher texts, and $2^{99.3}$ times of an encryption operation.

Keywords: meet-in-the-middle attack, block cipher PRINCE

1 Introduction

We show a meet-in-the-middle (MITM) attack on the 6-round variant of the block cipher PRINCE. PRINCE is proposed by Borghoff et al. in 2012, which applies substitution-permutation network (SPN) with 64-bit data block and 128-bit secret key [1]. PRINCE is optimized with respect to latency when implemented in hardware for many future pervasive applications with real-time security needs. The designers assessed the resistance of PRINCE core function against MITM attack at up to 4 rounds.

MITM attack was proposed by Diffie and Hellman in 1977 as a generic method to analyze symmetric-key cryptographic algorithms [2], [3], [4]. Its basic idea is that if a target algorithm can be decomposed into two small consecutive segments and the computation of each segment only involves portions of a master key, then we can check the consistence of the intermediate data of each segment. Because separately analyzing two small segments does not require much effort, the overall time complexity to analyze the whole algorithm could decrease significantly compared to a brute force attack.

In this article we decompose 6-round PRINCE into a 3-round forward segment and a 3-round backward segment, and show that PRINCE can be attacked with 2^{32} words of memory, 19 pairs of known plain and cipher texts, and $2^{99.3}$ times of an encryption operation.

2 Overview of PRINCE

Figure 1 shows the encryption process of PRINCE, which consists of XOR ($\oplus$), R and R^{-1} functions, S and S^{-1} layers, and M' layer. RC_i ($i = 0,1, \cdots, 11$) denotes a 64-bit

ISSN: 2287-1233 ASTL

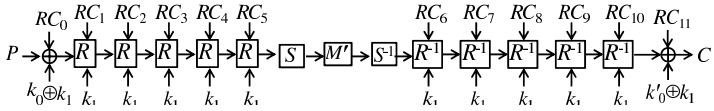

Fig. 1. Encryption process of PRINCE [1].

round constant given by [1]. k_0 and k_1 denote 64-bit secret keys being independent from each other. k_0' is also a 64-bit key given by

$$k_0' = (k_0 >>> 1) \oplus (k_0 >> 63) \tag{1}$$

where $(x >>> i)$ and $(x >> i)$ represent i-bit cyclic and non-cyclic right shifts of a bit string x, respectively. Because (1) is a bijective linear transformation, we can inversely derive k_0 from k_0'. S layer consists of 16 pieces of parallel 4-bit S-box that is a bijective nonlinear function, and S^{-1} layer is the inverse function of S layer. M' layer is a 64×64 involutional matrix given by

$$M' = M'^{-1} = \begin{pmatrix} \hat{M}^{(0)} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{M}^{(1)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \hat{M}^{(1)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \hat{M}^{(0)} \end{pmatrix}, \tag{2}$$

$$\hat{M}^{(0)} = \hat{M}^{(0)-1} = \begin{pmatrix} M_0 & M_1 & M_2 & M_3 \\ M_1 & M_2 & M_3 & M_0 \\ M_2 & M_3 & M_0 & M_1 \\ M_3 & M_0 & M_1 & M_2 \end{pmatrix}, \quad \hat{M}^{(1)} = \hat{M}^{(1)-1} = \begin{pmatrix} M_1 & M_2 & M_3 & M_0 \\ M_2 & M_3 & M_0 & M_1 \\ M_3 & M_0 & M_1 & M_2 \\ M_0 & M_1 & M_2 & M_3 \end{pmatrix}, \tag{3}$$

$$M_0 = \begin{pmatrix} 0&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 0&0&0&1 \end{pmatrix}, \quad M_1 = \begin{pmatrix} 1&0&0&0 \\ 0&0&0&0 \\ 0&0&1&0 \\ 0&0&0&1 \end{pmatrix}, \quad M_2 = \begin{pmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&0&0 \\ 0&0&0&1 \end{pmatrix}, \quad M_3 = \begin{pmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 0&0&0&0 \end{pmatrix}. \tag{4}$$

$\mathbf{0}$ in (2) is a 16×16 zero matrix.

Figure 2 shows R function that consists of S layer, M' layer, and SR layer. In the case of R^{-1} the input (output) becomes the output (input).

3 Outline of Meet-in-the-middle Attack and its application to the 6-round variant of PRINCE

MITM attack is based on the primary idea that we decompose a cipher algorithm into two consecutive parts. Each part of them only involves partial information of a secret key. We encrypt/decrypt each part separately and check whether the intermediate data from each part correspond to each other. Because separately cryptanalyzing each part requires low computational complexity, the overall complexity to cryptanalyze the whole algorithm could decrease significantly.

Figure 3 shows the general model of MITM attack. We suppose an encryption algorithm $E(k, P) = C$ can be decomposed into two consecutive parts $E_f(k_f, P)$ and $E_b(k_b, C)$

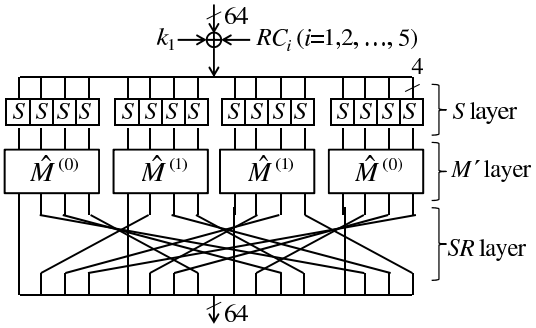

Fig. 2. R function.

where P and C are plain/cipher texts, and k_f and k_b are subkeys used in E_f and E_b. f and b denote forward and backward processes. k_f and k_b are given by $k_f = (k'_f, k_c)$ and $k_b = (k'_b, k_c)$, respectively where k_c is the dependent (common) key on both k_f and k_b. k'_f (k'_b) is the independent key from k_b (k_f). The number of bits of k_f and k_b are given by $|k_f| = |k'_f| + |k_c|$ and $|k_b| = |k'_b| + |k_c|$, respectively where $|x|$ denotes the number of bits of data x. v and v' are intermediate data of an encryption process. r is the total number of S-boxes in the whole algorithm. r_f and r_b represent the total numbers of S-boxes that must be calculated to derive v and v' through $E_f(k_f, P)$ and $E_b(k_b, C)$, respectively. The detailed steps of MITM attack are as follows [3], [4]:

1. Prepare one known pair of P and C.
2. For each guess of the common key k_c
   (a) Compute all possible values of v through $E_f(k_f, P)$ for all possible key k'_f. And then collect all k'_f in a set T indexed by v.
   (b) Compute v' through $E_b(k_b, C)$ for all values of k'_b and check whether $v' \in T$. If so output the corresponding key pair (k_f, k_b) as a possible key.

The time complexity for the step 2(a) in terms of complete encryption/decryption is given by $2^{|k_f|} \times r_f/r$. Similarly the time complexity of the step 2(b) is given by $2^{|k_b|} \times r_b/r$. The memory complexity of T is given by $2^{k'_f}$. The memory complexity to store possible keys is given by $2^{|k_c|+|k'_f|+|k'_b|-|v|}$ because the total number of possible keys $2^{|k_c|+|k'_f|+|k'_b|}$ is reduced to $1/2^{|v|}$ through the steps 1 and 2. When we perform these steps n times with different n pairs of P and C, the total number of possible keys is reduced to $2^{|k_c|+|k'_f|+|k'_b|-n\times|v|}$. If the following equation (5) is satisfied, we check the remaining possible keys by exhaustive search with a few different m (described later) pairs of P and C.

$$2^{|k_c|+|k'_f|+|k'_b|-n\times|v|} \leq (2^{|k_f|} \times r_f/r + 2^{|k_b|} \times r_b/r). \quad (5)$$

Figure 4 shows the 6-round variant of PRINCE, to which we apply MITM attack. Bold data lines are necessary for the attack. According to the designers' recommendation we omit 3 pieces of R with RC_i (i = 3, 4, 5) and 3 pieces of R^{-1} with RC_i (i = 6, 7, 8) from the original in order to keep the symmetry around the middle. $E_f(k_f, P)$

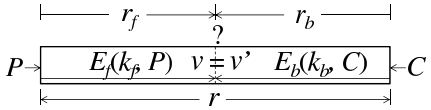

Fig. 3. General model of MITM attack [3], [4].

consists of the rounds 1, 2, and 3. $E_b(k_b, C)$ consists of the rounds 4, 5, and 6. X and Y represent the 64-bit intermediate data at the output of the round 3 and at the input of the round 4, respectively. We set variables k_x and k_y as $k_x = k_0 \oplus k_1$ at the input part and $k_y = k'_0 \oplus k_1$ at the output part of Fig. 4. And we decompose k_x as

$$k_x = k_x^{(15)} \| k_x^{(14)} \| k_x^{(13)} \| \cdots \| k_x^{(1)} \| k_x^{(0)} \tag{6}$$

where $x \| y$ represents concatenation of two data x and y. $k_x^{(i)}$ ($i = 0, 1, \cdots, 15$) is 4-bit data that is further decomposed as

$$k_x^{(i)} = k_x^{(i,3)} \| k_x^{(i,2)} \| k_x^{(i,1)} \| k_x^{(i,0)} \tag{7}$$

where $k_x^{(i,j)}$ ($j = 0, 1, 2, 3$) represents 1-bit data. We also apply these decompositions and notations to k_y, k_1, X, and Y.

From Fig. 4 we can derive that $|k_f| = |k_b| = 96$, $|k'_f| = |k'_b| = 32$, $|k_c| = 64$, and $|v| = |v'| = 2$. These reasons are as follows. k_f has 96 bits on the bold line in $E_f(k_f, P)$ given by

$$k_f = (k_x, k_1^{(15)}, k_1^{(14)}, k_1^{(13)}, k_1^{(12)}, k_1^{(3)}, k_1^{(2)}, k_1^{(1)}, k_1^{(0)}). \tag{8}$$

k_b also has 96 bits on the bold line in $E_b(k_b, C)$ given by

$$k_b = (k_y, k_1^{(15)}, k_1^{(14)}, k_1^{(13)}, k_1^{(12)}, k_1^{(3)}, k_1^{(2)}, k_1^{(1)}, k_1^{(0)}). \tag{9}$$

$k_1^{(i)}$ ($i = 15, 14, 13, 12, 3, 2, 1, 0$) is common to both k_f and k_b. Furthermore there are 32-bit common keys, which are

$$(k_c \text{ of } k_f) \ni (k_x^{(15)}, k_x^{(14)}, k_x^{(13)}, k_x^{(12)}, k_x^{(3)}, k_x^{(2)}, k_x^{(1)}, k_x^{(0)}) \tag{10}$$

and

$$(k_c \text{ of } k_b) \ni (k_y^{(15)}, k_y^{(14)}, k_y^{(13)}, k_y^{(12)}, k_y^{(2)}, k_y^{(1)}, k_y^{(0)}, k_y^{(11,3)}, k_y^{(3,2)}, k_y^{(3,1)}, k_y^{(3,0)}). \tag{11}$$

The reason why (10) and (11) hold is that k_c of k_y (k_c of k_x) is uniquely determined by (1) with k_c of k_x (k_c of k_y) and the common key $k_1^{(i)}$ ($i = 15, 14, 13, 12, 3, 2, 1, 0$). Therefore the independent keys k'_f and k'_b have the 32 bits given by

$$k'_f = (k_x^{(11)}, k_x^{(10)}, k_x^{(9)}, k_x^{(8)}, k_x^{(7)}, k_x^{(6)}, k_x^{(5)}, k_x^{(4)}) \tag{12}$$

and

$$k'_b = (k_x^{(10)}, k_x^{(9)}, k_x^{(8)}, k_x^{(7)}, k_x^{(6)}, k_x^{(5)}, k_x^{(4)}, k_x^{(11,2)}, k_x^{(11,1)}, k_x^{(11,0)}, k_x^{(3,3)}). \tag{13}$$

v and v' have the 2 bits given by

$$v = (X^{(15,0)} \oplus X^{(12,0)}, X^{(3,2)} \oplus X^{(2,2)}) \text{ and } v' = (Y^{(15,0)} \oplus Y^{(12,0)}, Y^{(3,2)} \oplus Y^{(2,2)}). \quad (14)$$

We can derive $v = v'$ by (2). Because there are 96 pieces of S-box in the whole algorithm in Figure 4, we can derive that $r = 96$. Similarly we can derive that $r_f = r_b = 28$ because there are respectively 28 pieces of S-box on the bold lines in $E_f(k_f, P)$ and in $E_b(k_b, C)$. To satisfy (5), we set n as $n = 17$. In this case the total number of possible keys is reduced to $2^{64+32+32-17\times2} = 2^{94} < 2^{96} \times 28/96 \times 2 \approx 2^{95.2}$. These 2^{94} pieces of possible key are furthermore reduced to $2^{94-2\times64} = 2^{-34}$ when we check these keys by exhaustive search with different 2 (= m) pairs of P and C because the block size of PRINCE is 64 bits. However the true key definitely survives. Therefore the number of known plain/cipher texts pair required for this MITM attack is derived as 17+2=19. And the time complexity for the attack in terms of complete encryption/decryption is derived as $17\times(2^{96}\times28/96+2^{96}\times28/96)+2^{94}+2^{94-64} \approx 2^{99.3}$. The memory complexity of a set T is derived as 2^{32}.

4 Conclusions

We have shown MITM attack on the 6-round variant of the block cipher PRINCE. We reviewed the general model of MITM attack and applied it to PRINCE. We decomposed the 6-round PRINCE into a 3-round forward segment and a 3-round backward segment. As a result, we showed that PRINCE can be attacked with 2^{32} words of memory, 19 pairs of known plain and cipher texts, and $2^{99.3}$ times of an encryption operation, which is more efficient than 2^{128} times of simple exhaustive key search.

References

1. Borghoff, J., Canteaut, A., Gneysu, T., et al.: PRINCE — A Low-Latency Block Cipher for Pervasive Computing Applications. LNCS, vol. 7658, pp. 208–225. Springer, Heidelberg (2012)
2. Diffie, M.E., Hellman, W.: Special Feature Exhaustive Cryptanalysis of the NBS Data Encryption Standard. Computer 10(6), 77–84 (1977)
3. Zhu, B., Gong, G.: Multidimensional Meet-in-the-Middle Attack and Its Applications to KATAN32/48/64. Cryptology ePrint Archive: Report 2011/619, https://eprint.iacr.org/2011/619
4. Boztaş, Ö., Karakoç, F., Çoban, M.: Multidimensional Meet-in-the-Middle Attacks on Reduced-Round TWINE-128. LNCS, vol. 8162, pp. 55–67. Springer, Heidelberg (2013)

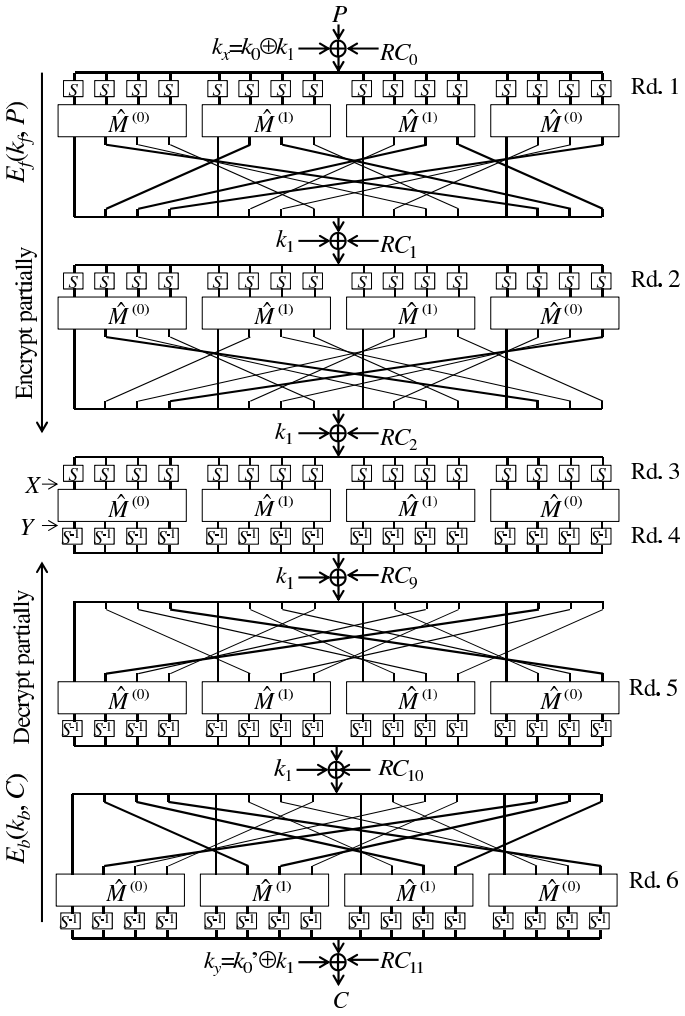

Fig. 4. 6-round variant of PRINCE.