한국항행학회 논문지 제16권 제5호 2012년 10월

다양한 환경에 적용 가능한 블록 암호 Piccolo-128에 대한 안전성 분석

Security Analysis of Block Cipher Piccolo-128 Applicable to Various Environments

정기태*

Ki-Tae Jeong*

요 약

64-비트 블록 암호 Piccolo-128은 무선 센서 네트워크 환경과 같이 제한된 환경에 적합하도록 설계된 경량 블록 암호이다. 본 논문에서는 Piccolo-128에 대한 biclique 공격을 제안한다. 본 논문에서 제안하는 공격은 2^{24}개의 선택 평문과 약 $2^{127.35}$의 계산 복잡도를 이용하여 Piccolo-128의 비밀키를 복구한다. 본 논문의 공격 결과는 Piccolo-128의 전체 라운드에 대한 첫 번째 이론적인 분석 결과이다.

Abstract

Piccolo-128 is a 64-bit ultra-light block cipher suitable for the constrained environments such as wireless sensor network environments. In this paper, we propose biclique cryptanalysis on the full Piccolo-128. To recover the secret key of Piccolo-128, the proposed attack requires 2^{24} chosen plaintexts and the computational complexity of about $2^{127.35}$. This result is the first known theoretical attack result on the full Piccolo-128.

Key words : Block Cipher(블록 암호), Piccolo, Biclique Cryptanalysis

I. 서 론

CHES 2011에 제안된 경량 블록 암호 Piccolo는 80/128-비트 비밀키를 사용하는 64-비트 블록 암호로서, 무선 센서 네트워크 환경과 같이 제한된 환경에 적합하도록 설계되었다 [1]. 이 알고리즘은 일반화된 Feistel 구조를 가지며, 비밀키의 길이에 따라, 각각 Piccolo-80, Piccolo-128로 표기된다. Piccolo-80의 전체 라운드 수는 25이고, Piccolo-128의 전체 라운드 수는 31이다. 현재까지 제안된 Piccolo에 대한 분석 결과로 biclique 공격과 차분 오류 공격이 제안되었다 [2, 3, 4]. 이 중 차분 오류 공격은 부채널 공격이기 때문에, 본 논문에서는 [3, 4]에서 소개된 결과를 고려하지 않는다. ISPEC 2012에서 소개된 결과는 Piccolo-80의 전체 라운드(postwhitening key 부분 제외)와 Piccolo -128의 28 라운드 축소 버전에 대한 결과이다 [2]. 따라서 Piccolo-128의 전체 라운드에 대한 이론적인 분석 결과는 제안되지 않았다.

Biclique 공격은 중간 일치 공격에 biclique 개념을 적용한 개념으로서, Asiacrypt 2011에서 처음 소개되

* 고려대학교 정보보호연구원(Center for Information Security Technologies(CIST), Korea University)
· 제1저자 (First Author) : 정기태
· 투고일자 : 2012년 8월 14일
· 심사(수정)일자 : 2012년 8월 17일 (수정일자 : 2012년 10월 9일)
· 게재일자 : 2012년 10월 30일

었다 [5]. [5]에서 저자들은 중간 일치 공격에 biclique 개념을 적용하여 AES-128/192/256의 전체 라운드에 대해 전수조사 보다 적은 계산 복잡도로 비밀키를 복구할 수 있음을 보였다. 이후, biclique 공격은 Piccolo [2], HIGHT [6], TWINE [7] 등과 같은 다양한 블록 암호 알고리즘에 적용되었다.

본 논문에서는 Piccolo-128의 전체 라운드에 대한 biclique 공격을 제안한다. 본 논문에서 제안하는 공격은 7 라운드에 대한 8-dimensional biclique에 기반을 둔다. Piccolo-128의 비밀키를 복구하기 위해, 이 공격은 2^{24}개의 선택 평문과 $2^{127.35}$의 계산 복잡도를 필요로 한다. 이 공격 결과는 Piccolo-128의 전체 라운드에 대한 첫 번째 이론적인 분석 결과이다.

본 논문은 다음과 같이 구성되어 있다. 먼저, 2절에서는 블록 암호 Piccolo-128의 구조를 소개한다. 3절에서는 Piccolo-128의 전체 라운드에 대한 biclique 공격을 소개한 후, 마지막으로 4절에서 결론을 맺는다.

Ⅱ. Piccolo-128

64-비트 블록 암호 Piccolo-128은 그림 1과 같이 128-비트 비밀키를 사용하고 31-라운드 일반화된 Feistel 구조를 갖는다. 본 논문에서는 다음과 같은 표기법을 사용한다.

- $P=(P_0,P_1,P_2,P_3)$: 64-비트 평문.
- $C=(C_0,C_1,C_2,C_3)$: 64-비트 암호문.
- $I_i=(I_{i,0},I_{i,1},I_{i,2},I_{i,3})$: 라운드 i의 64-비트 입력값 $(i=0,1,\cdots,30)$.
- (rk_{2i},rk_{2i+1}): 라운드 i의 라운드 키.
- (wk_0,wk_1,wk_2,wk_3): 화이트닝 키.

64-비트 평문 $P=(P_0,P_1,P_2,P_3)$는 먼저 다음과 같은 화이트닝 키 단계를 거쳐 라운드 0의 입력값 $I_0=(I_{0,0},I_{0,1},I_{0,2},I_{0,3})$이 된다.

$$I_{0,0}=P_0\oplus wk_0,\ I_{0,1}=P_1,$$

$$I_{0,2}=P_2\oplus wk_1,\ I_{0,3}=P_3.$$

$I_0=(I_{0,0},I_{0,1},I_{0,2},I_{0,3})$는 라운드 함수 F와 라운드 치환 RP를 30번 반복 적용하여 라운드 30의 입력값 $I_{30}=(I_{30,0},I_{30,1},I_{30,2},I_{30,3})$이 된다. 이 값으로부터 암호문 $C=(C_0,C_1,C_2,C_3)$는 다음과 같이 계산된다.

$$C_0=I_{30,0}\oplus wk_2,\ C_1=F(I_{30,0})\oplus I_{30,1}\oplus rk_{60},$$
$$C_2=I_{30,2}\oplus wk_3,\ C_3=F(I_{30,2})\oplus I_{30,3}\oplus rk_{61}.$$

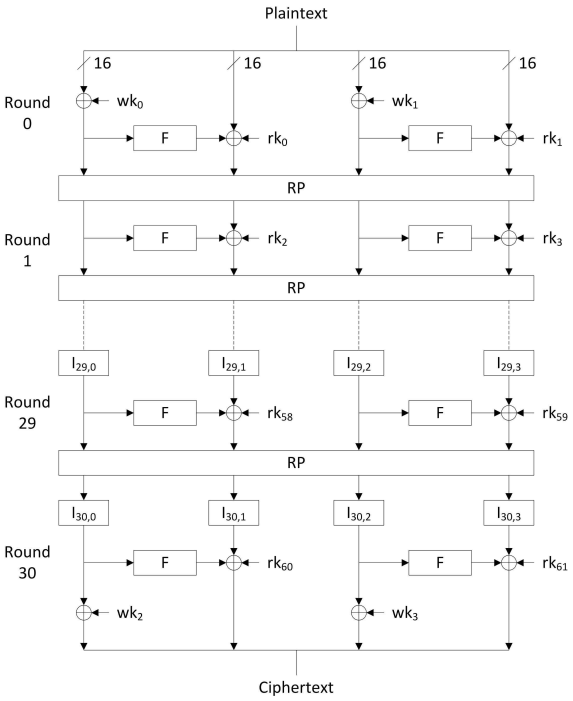

그림 1. Piccolo-128의 전체 구조
Fig. 1. The structure of Piccolo-128.

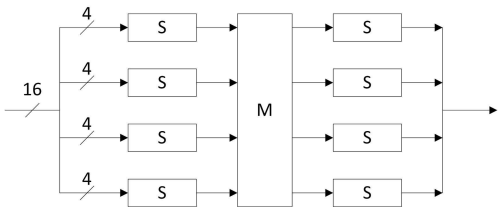

그림 2. 라운드 함수 F
Fig. 2. Round function F of Piccolo-128.

라운드 함수 F는 그림 2와 같이 16-비트 입출력을 갖는 함수로 SMS 구조이다. 본 논문에서 제안하

는 공격은 4×4 S-box S와 4×4 행렬 M의 구체적인 성질을 이용하지 않으므로 생략하기로 한다. 라운드 치환 RP는 그림 3과 같이 64-비트 입력값 $X=(x_0,x_1,x_2,x_3)$을 입력받아 64-비트 출력값 $Y=(y_0,y_1,y_2,y_3)$을 생성하는 바이트 단위 치환이다. 여기서 $x_i=(x_i^L,x_i^R)$이다.

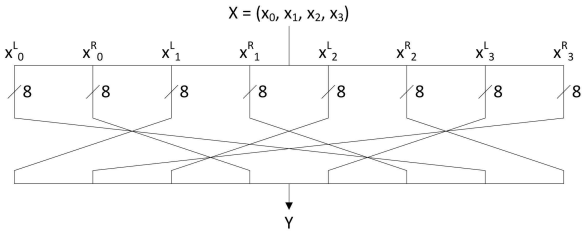

그림 3. 라운드 치환 RP
Fig. 3. Round permutation RP of Piccolo-128.

Piccolo-128의 키스케줄은 매우 단순하다. 먼저 128-비트 비밀키 K를 다음과 같이 8개의 k_j로 나눈다 $(j=0,\cdots,7)$. 여기서 $k_j=(k_j^L,k_j^R)$이다.

$$K=(k_0,k_1,k_2,k_3,k_4,k_5,k_6,k_7).$$

4개의 화이트닝 키 (wk_0,wk_1,wk_2,wk_3)와 31개의 라운드 키 (rk_{2i},rk_{2i+1})는 다음과 같이 생성된다 $(i=0,1,\cdots,30)$. 여기서 $(con_{2i}^{128},con_{2i+1}^{128})$은 각각 16-비트 라운드 상수이다.

○ 화이트닝 키

$$wk_0=k_0^L\parallel k_1^R,\ wk_1=k_1^L\parallel k_0^R,$$
$$wk_2=k_4^L\parallel k_7^R,\ wk_3=k_7^L\parallel k_4^R.$$

○ 라운드 키

For $i\leftarrow0$ to $(2r-1)$ do
  if $(i+2)\bmod8=0$ then
  $(k_0,k_1,k_2,k_3,k_4,k_5,k_6,k_7)\rightarrow$
  $(k_2,k_1,k_6,k_7,k_0,k_3,k_4,k_5)$
  $rk_i\leftarrow k_{(i+2)\bmod8}\oplus con_i^{128}$

표 1. 라운드 키에 사용된 비밀키
Table 1. Secret key used in round keys.

라운드 i	비밀키 정보
whitening	$(k_0^L\parallel k_1^R,\ k_1^L\parallel k_0^R)$
0	(k_2,k_3)
1	(k_4,k_5)
2	(k_6,k_7)
3	(k_2,k_1)
4	(k_6,k_7)
5	(k_0,k_3)
6	(k_4,k_5)
7	(k_6,k_1)
8	(k_4,k_5)
9	(k_2,k_7)
10	(k_0,k_3)
11	(k_4,k_1)
12	(k_0,k_3)
13	(k_6,k_5)
14	(k_2,k_7)
15	(k_0,k_1)
16	(k_2,k_7)
17	(k_4,k_3)
18	(k_6,k_5)
19	(k_2,k_1)
20	(k_6,k_5)
21	(k_0,k_7)
22	(k_4,k_3)
23	(k_6,k_1)
24	(k_4,k_3)
25	(k_2,k_5)
26	(k_0,k_7)
27	(k_4,k_1)
28	(k_0,k_7)
29	(k_6,k_3)
30	(k_2,k_5)
whitening	$(k_4^L\parallel k_7^R,\ k_7^L\parallel k_4^R)$

표 1은 각각의 라운드 키에 사용된 비밀키 정보를 나타낸 것이다. 예를 들어, 라운드 30의 라운드 키 (rk_{60},rk_{61})에 비밀키 정보 (k_2,k_5)이 각각 적용되었다.

Ⅲ. Piccolo-128에 대한 biclique 공격

본 절에서는 Piccolo-128에 대한 biclique 공격을 제안한다. 먼저 전체 비밀키 공간을 2^{16}개의 비밀키를 각각 포함하는 2^{112}개의 부분 공간으로 나눈다. 각각의 비밀키 부분 공간에 대해, 7 라운드에 대한 8-dimensional biclique를 구성한 후, 이를 이용하여 Piccolo-128의 전체 라운드에 대한 비밀키를 복구한다.

3-1 비밀키 분할

각각의 비밀키 부분 공간에서 기본 비밀키는 다음과 같은 형태를 갖는다. 즉, k_1^L과 k_6^L을 0으로 고정시키고 나머지 값들은 임의의 값이다.

$$K_{0,0} = (?||?, 0||?, ?||?, ?||?, ?||?, ?||?, 0||?, ?||?).$$

따라서 각각의 비밀키 부분 공간에 포함되는 2^{16}개의 비밀키는 다음과 같이 계산된다.

$$K_{i,j} = K_{0,0} \oplus (0||0, j||0, 0||0, 0||0, 0||0, 0||0, i||0, 0||0).$$

3-2 7 라운드에 대한 biclique 구성

먼저 다음과 같은 두 개의 집합을 고려한다.

- $A = \{K_{i,0} |\ 0 \le i \le 2^8 - 1\}$.
- $B = \{K_{0,j} |\ 0 \le j \le 2^8 - 1\}$

각각의 집합에서, $K_{i,0}$와 $K_{0,j}$는 다음과 같이 계산된다.

$$K_{i,0} = K_{0,0} \oplus \Delta_i^K,\ K_{0,j} = K_{0,0} \oplus \nabla_j^K.$$

여기서 Δ_i^K와 ∇_j^K는 다음과 같다.

$$\Delta_i^K = (0||0, 0||0, 0||0, 0||0, 0||0, 0||0, i||0, 0||0),$$

$$\nabla_j^K = (0||0, j||0, 0||0, 0||0, 0||0, 0||0, 0||0, 0||0).$$

생성된 $K_{i,0}$와 $K_{0,j}$를 이용하여 집합 $\{C_i\}$, $\{S_j\}$는 다음과 같이 계산된다. 여기서 f는 Piccolo-128의 마지막 7 라운드(라운드 24 ~ 라운드 30)를 의미한다.

$$S_0 \xrightarrow[f]{K_{i,0}} C_i \quad , \quad S_j \xleftarrow[f^{-1}]{K_{0,j}} C_0 \quad .$$

여기서 $C_0 = 0$이고 $S_0 = f_{K_{0,0}}^{-1}(C_0)$이다.

그러면 7-라운드 biclique는 다음과 같이 구성된다. 여기서 $\Delta_i = C_0 \oplus C_i$, $\Delta_i^K = K_{0,0} \oplus K_{i,0}$, $\nabla_j = S_0 \oplus S_j$, $\nabla_j^K = K_{0,0} \oplus K_{0,j}$.

(1) f에 대한 Δ_i 차분: $0 \xrightarrow[f]{\Delta_i^K} \Delta_i$.

(2) f^{-1}에 대한 ∇_j 차분: $\nabla_j \xrightarrow[f]{\nabla_j^K} 0$.

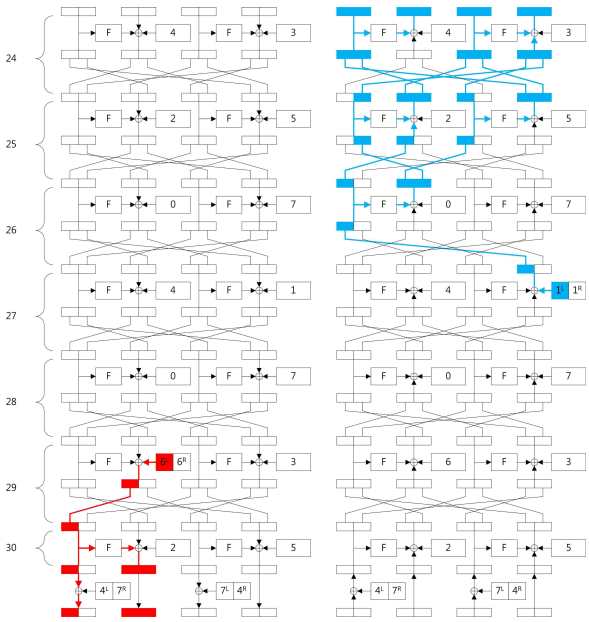

그림 4. 7 라운드 biclique
Fig. 4. 7-round biclique.

그림 4는 위에서 구성된 Δ_i 차분(그림 4에서 왼

쪽)과 ∇_j 차분(그림 4에서 오른쪽)을 나타낸 것이다. 그림에서 알 수 있듯이, 두 개의 차분은 active F 함수를 공유하지 않는다. 따라서 다음 식이 성립함을 쉽게 알 수 있다 $(i, j \in \{0,1,\cdots,2^8-1\})$.

$$S_0 \oplus \nabla_j \xrightarrow{K_{0,0} \oplus \Delta_i^K \oplus \nabla_j^K} f \; C_0 \oplus \Delta_i \quad .$$

그림 4에서 알 수 있듯이, Δ_i 차분은 암호문의 3 바이트만 영향을 준다. 따라서 본 공격의 데이터 복잡도는 2^{24}을 초과하지 않는다.

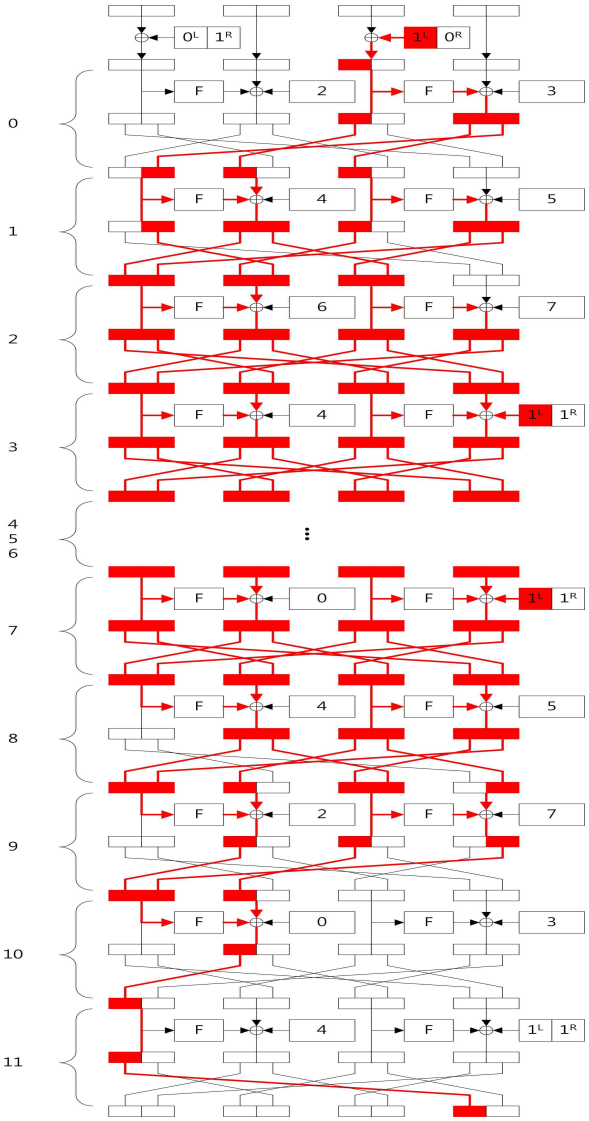

그림 5. Forward 방향에서의 recomputation
Fig. 5. Recomputation in forward direction.

3-3 중간 일치 단계

중간 일치 단계는 Piccolo-128의 첫 24 라운드(라운드 0 ~ 라운드 23)에서 수행된다. 먼저, 앞에서 구성한 7-라운드 biclique에 대응되는 평문 집합 $\{P_i\}$이 decryption oracle로부터 생성되었다고 가정한다. 그러면 다음 식을 만족하는 후보 비밀키를 계산한다. 여기서 g_1은 Piccolo-128의 라운드 0 ~ 라운드 11이고 g_2는 라운드 12 ~ 라운드 23을 의미한다. Matching point는 라운드 12의 7번째 바이트 값 $I_{12,3}^L$ 이다.

$$P_i \xrightarrow[g_1]{K_{i,j}} \rightarrow v \overset{?}{=} \rightarrow v \xleftarrow[g_2^{-1}]{K_{i,j}} S_j \quad .$$

위의 식을 모든 i와 j에 대해 $I_{12,3}^L$을 계산해야 하지만, 각각의 경우마다 동일한 부분에 대해서는 반복하여 계산할 필요는 없다. 따라서 추가적으로 계산해야 할 부분만 고려하면 된다. 그림 5와 6은 forward 방향(g_1)과 backward 방향(g_2)에서 추가적으로 계산해야 할 부분을 각각 나타낸 것이다.

3-4 계산 복잡도

본 논문에서 소개하는 Biclique 공격의 계산 복잡도는 다음과 같이 계산된다.

$$C_{total} = 2^{n-2d}\{C_{biclique} + C_{precomp} + C_{recomp} + C_{falsepos}\}.$$

- $n = 128$이고 $d = 8$이다.
- $C_{biclique}$는 한 개의 biclique를 구성하는데 필요한 계산 복잡도를 의미한다. 일반적으로, 이 복잡도는 2^{d+1} f computation이다. 따라서 본 공격에서는 다음과 같이 계산된다.

$$2^{6.85}\left(\approx 2^{8+1} \cdot \frac{7}{31}\right) \text{ Piccolo-128 computations.}$$

- $C_{precomp}$는 중간 일치 단계에서 $\vec{v}$를 선계산하는데 필요한 계산 복잡도를 의미한다. 일반적으로, 이 복잡도는 2^d $(g_1 + g_2)$ computation이다.

따라서 본 공격에서는 다음과 같이 계산된다.

$$2^{7.63}\left(\approx 2^8 \cdot \frac{24}{31}\right) \text{ Piccolo-128 computations.}$$

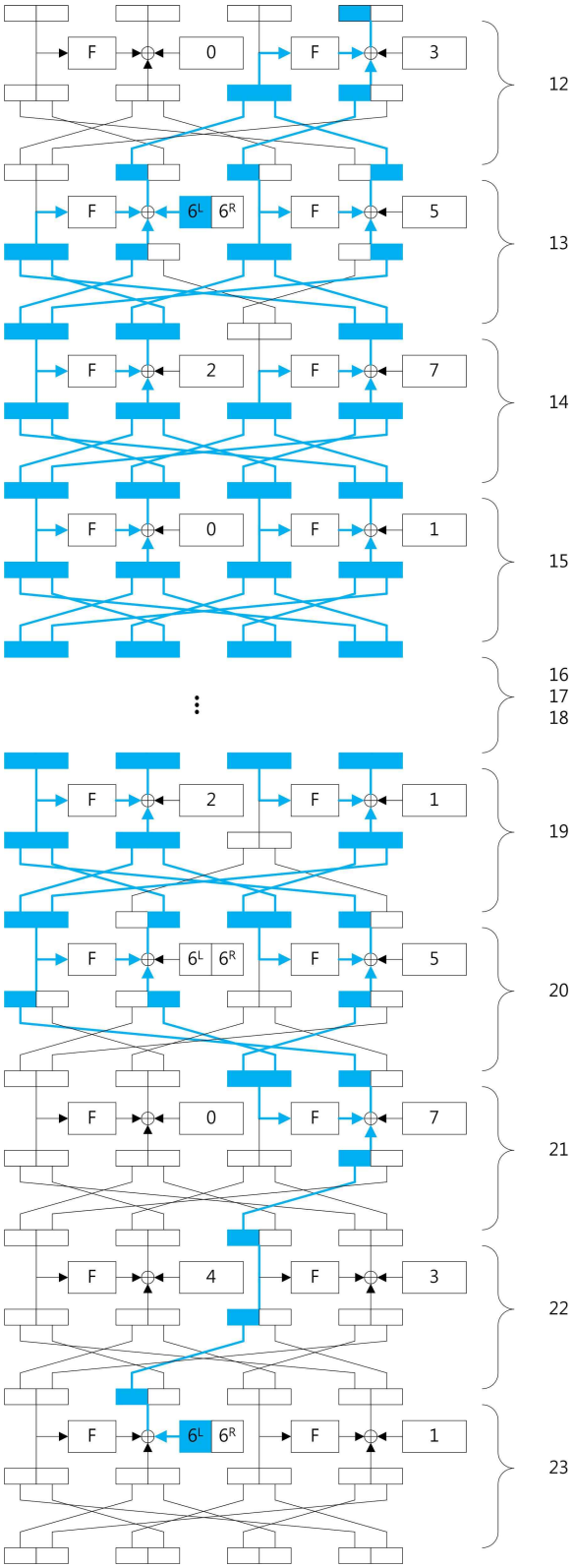

그림 6. Backward 방향에서의 recomputation
Fig. 6. Recomputation in backward direction.

○ C_{recomp}는 $I_{12,3}^{L}$을 $2^{2d}(=2^{16})$번 반복 계산하는데 필요한 계산 복잡도를 의미한다. 따라서 이 값은 다음과 같이 계산된다.

$$2^{15.33}\left(\approx 2^{2\cdot 8}\cdot\frac{19.5}{31}\right) \text{ Piccolo-128 computations.}$$

○ $C_{falsepos}$는 중간 일치 단계에서 통과하는 틀린 후보 비밀키를 검사하는데 필요한 계산 복잡도를 의미한다. 따라서 이 값은 다음과 같이 계산된다.

$$2^{8}\left(\approx 2^{2\cdot 8-8}\right) \text{ Piccolo-128 computations.}$$

따라서 본 공격의 전체 계산 복잡도는 $2^{127.35}$ full Piccolo-128 computations이다.

$$2^{127.35}\left\{\approx 2^{112}\left(2^{6.85}+2^{7.63}+2^{15.33}+2^{8}\right)\right\}.$$

Ⅳ. 결 론

본 논문에서는 Piccolo-128의 전체 라운드에 대한 biclique 공격을 제안하였다. 본 논문에서 제안한 공격은 Piccolo-128의 전체 라운드에 대한 첫 번째 이론적인 공격 결과이다. 기제안된 공격 결과와 비교하면 표 2와 같다.

표 2. Piccolo-128에 대한 biclique 공격 결과
Table 1. Attack results on Piccolo-128.

	[2]	본 논문
Piccolo-128	28 라운드	전체 라운드
Biclique	6 라운드	7 라운드
데이터 복잡도	2^{24}	2^{24}
계산 복잡도	$2^{126.79}$	$2^{127.35}$

표에서 알 수 있듯이, [2]에서 제안된 공격은 6-라운드 biclique를 구성하여 28-라운드 축소 버전에 대해 적용된 반면, 본 논문에서 제안된 공격은 7-라운드 biclique를 구성하여 전체 라운드에 대해 적용되었다.

두 공격 모두 계산 복잡도가 $2^{126.79}$와 $2^{127.35}$로 매우 비현실적이라고 할 수 있다. 비록 계산 복잡도가 비현실적이긴 하지만, 이 공격 결과들은 Piccolo- 128의 확산 효과가 좋지 않다는 성질로부터 비롯되었다. 이는 Piccolo-128에 대해 구조적인 취약점이 존재함을 의미한다. 이 취약점을 이용하여 다양한 분석 결과를 얻을 수도 있기 때문에, 본 논문에서 제안한 공격 결과가 의미 있다고 할 수 있다.

본 공격을 확장하여 Piccolo-80에 대한 biclique 공격은 향후 연구 과제이다.

참 고 문 헌

[1] K. Shibutani, T. Isobe, H. Hiwatari, A. Mitsuda, T. Akishita and T. Shirai, “Piccolo: An Ultra-Lightweight Blockcipher”, *CHES 2011, LNCS 6917,* pp. 342-357, Springer-Verlag, 2011.

[2] Y. Wang, W. Wu and X. Yu, “Biclique Cryptanalysis of Reduced-Round Piccolo Block Cipher”, *ISPEC 2012*, LNCS 7232, pp. 337-352, Springer-Verlag, 2012.

[3] K. Jeong, “Differential Fault Analysis on Block Cipher Piccolo”, *ePrint2012/399,* 2012.

[4] 정기태, “블록 암호 Piccolo-80에 대한 차분 오류 공격”, *한국항행학회논문지 제16권 제3호* pp. 510-517, 2012.

[5] A. Bogdanov, D. Khovratovich and C. Rechberger, “Biclique Cryptanalysis of the Full AES”, *Asiacrypt 2011, LNCS 7073,* pp. 344-371, Springer, 2011.

[6] D. Hong, B. Koo and D. Kwon, “Biclique Attack on the Full HIGHT”, *ICISC 2011, LNCS 7259,* pp. 15-25, Springer, 2012.

[7] M. Coban, F. Karakoc and O. Boztas, “Biclique Cryptanalysis of TWINE”, *ePrint2012/422,* 2012.

[8] 정기태, “무선 센서 네트워크 환경에 적합한 블록 암호 LED-64에 대한 안전성 분석”, *한국항행학회 논문지*, 제 16권, 제 1호, pp. 70-75, 2012.

정 기 태 (鄭基台)

2004년 2월 : 고려대학교 수학과 이학사

2006년 2월 : 고려대학교 정보보호대학원 공학석사

2011년 8월 : 고려대학교 정보보호대학원 공학박사

2011년 9월~현재 : 고려대학교 정보보호 연구원 연구교수

관심분야 : 대칭키 암호에 대한 분석 및 설계