情報保護學會論文誌
第15券 第3號, 2005. 6

축소 라운드 SHACAL-2의 연관키 공격*

김 종 성,† 김 구 일, 이 상 진, 임 종 인‡
고려대학교 정보보호기술연구센터

Related-Key Attacks on Reduced Rounds of SHACAL-2*

Jongsung Kim,† Guil Kim, Sangjin Lee, Jongin Lim‡
Center for Information of Security Technologies, Korea University

요 약

SHACAL-2는 해쉬 알고리즘 SHA-2의 압축 함수에 기반을 둔 최대 512 비트 키 크기를 가지는 256 비트 블록 암호이다. 최근에 SHACAL-2는 NESSIE 프로젝트의 256 비트 블록 암호에 선정되었다. 본 논문에서는 연관키를 이용한 두 가지 형태의 연관키 차분-비선형 공격과 연관키 Rectangle 공격에 대한 SHACAL-2의 안전성을 논의한다. 연관키 차분-비선형 공격 기법을 통하여 512 비트 키를 사용하는 35-라운드 SHACAL-2를 분석하고, 연관키 렉탱글 공격 기법을 통하여 512 비트 키를 사용하는 37-라운드 SHACAL-2를 분석한다. 본 논문에서 소개하는 512 비트 키를 가지는 37-라운드 SHACAL-2 연관키 렉탱글 공격은 SHACAL-2 블록 암호에 알려진 분석 결과 중 가장 효과적이다.

ABSTRACT

SHACAL-2 is a 256-bit block cipher with up to 512 bits of key length based on the hash function SHA-2. It was submitted to the the NESSIE project and was recommended as one of the NESSIE selections. In this paper, we present two types of related-key attacks called the related-key differential-(non)linear and the related-key rectangle attacks, and we discuss the security of SHACAL-2 against these two types of attacks. Using the related-key differential-nonlinear attack, we can break SHACAL-2 with 512-bit keys up to 35 out of its 64 rounds, and using the related-key rectangle attack, we can break SHACAL-2 with 512-bit keys up to 37 rounds.

Keywords : *SHACAL-2, Related-Key Differential-Nonlinear Attacks, Related-Key Rectangle Attacks*

Ⅰ. 서 론

SHACAL-2[5]는 해쉬 알고리즘 SHA-2의 압축 함수에 기반한 256 비트 블록 암호이다. 64 라운드로 구성된 SHACAL-2는 최대 512 비트 키 크기를 가지며, 최근 NESSIE(New European Schemes for Signatures, Integrity, and Encryption) 프로젝트에 256 비트 블록 암호로 선정되었다.

현재까지 SHACAL-2 블록 암호에 대한 가장 효과적인 암호 분석 결과는 차분-비선형 특성을 이용한 32-라운드 SHACAL-2 공격이다[8]. [8]에 의해 소개된 공격은 14-라운드 부정 차분 특성과 마지막 3 라운드의 비선형 관계식을 이용한 17-라운드 차분-비선형 특성을 기반으로 하고 있다. 본 논문에서 소개하는 공격 또한 [8]의 마지막 3 라운드의 비선형 관계식을 이용한다.

본 논문에서는 연관키 공격의 두 가지 형태를 소

접수일 : 2005년 3월 30일 ; 채택일 : 2005년 6월 10일
* 본 연구는 고려대학교 특별연구비에 의하여 수행 되었습니다.
† 주저자 : joshep@cist.korea.ac.kr
‡ 교신저자 : jilim@korea.ac.kr

개한다. 하나는 연관키 차분-비선형 특성을 이용한 공격이며, 다른 하나는 연관키 렉탱글 특성을 이용한 공격이다. 두 가지 공격에 대한 일반적인 형태를 살펴본 후 SHACAL-2의 분석 결과를 소개한다.

연관키 차분-선형 공격[6]은 확률 1을 가지는 연관키 차분-선형 특성을 이용한다. 하지만 본 논문에서는 특성 확률이 1보다 작은 경우로 일반화하여 공격 적용 가능성을 향상시킨다. 또한 선형 특성 부분을 비선형 특성으로 일반화한 연관키 차분-비선형 특성을 이용한 연관키 차분-비선형 공격 가능성을 소개하며, 공격 적용 가능성을 더욱 향상시킨다. 본 공격을 SHACAL-2에 적용 결과 연관키 차분-비선형 특성을 사용하여 $2^{42.32}$ 연관키 선택 평문과 $2^{451.1}$ 시간 복잡도를 가지고 전수조사 보다 빠르게 35-라운드 SHACAL-2를 공격할 수 있다.

연관키 렉탱글 공격[9]은 두 가지 형태의 연관키 렉탱글 특성을 이용할 수 있다. 본 논문에서는 두 가지 형태의 연관키 렉탱글 특성 중 SHACAL-2에 더욱 쉽게 적용할 수 있는 한 가지 특성을 적용한다. 본 공격을 SHACAL-2에 적용 결과 연관키 렉탱글 특성을 사용하여 $2^{233.16}$ 연관키 선택 평문과 $2^{484.95}$ 시간 복잡도를 가지고 전수조사 보다 빠르게 37-라운드 SHACAL-2를 공격할 수 있다. 본 논문의 분석과 기존 SHACAL-2 분석 결과의 비교는 표 1과 같다.

표 1. SHACAL-2의 분석 결과 비교

공격 유형	라운드	복잡도 데이터 / 시간/ 메모리
불능 차분 공격	30	744 CP / $2^{495.1}$ / $2^{14.5}$ [7]
차분-비선형 공격	32	$2^{43.4}$ CP / $2^{504.2}$ / $2^{48.4}$ [8]
포화-비선형 공격	28	$463 \cdot 2^{32}$ CP / $2^{494.1}$ / $2^{45.9}$ [8]
연관키 차분 -비선형 공격	35	$2^{42.32}$ RK-CP / $2^{452.10}$ / $2^{47.32}$ [본 논문]
연관키 렉탱글 공격	37	$2^{233.16}$ RK-CP / $2^{484.95}$ / $2^{238.16}$ [본 논문]

II. 표기법 및 SHACAL-2

본 절에서는 본 논문에 전반적으로 사용하는 표기법을 정리하고, SHACAL-2 블록 암호를 간략하게 소개한다. 또한 본 논문의 연관키 차분-비선형 공격에 사용하는 SHACAL-2의 3-라운드 비선형 관계식[7,8]을 요약한다.

2.1. 표기법

본 소절에서는 표기법을 정의한다. 단, 워드의 비트 위치는 가장 오른쪽 최하위 비트부터 0으로 시작하며, 왼쪽으로 갈수록 커진다.

- P : 256 비트 평문, $P=(A,B,\cdots,H)$ 또는 $P=(A^0,B^0,\cdots,H^0)=P^0c$.
- P^r : r번째 라운드의 256 비트 입력 값, $P^r=(A^r,B^r,\cdots,H^r)$.
- x_i^r : 32 비트 워드 X^r의 i번째 비트, $X^r \in \{A^r,B^r,\cdots,H^r,W^r,K^r,T_1^r\}$.
- ? : 알 수 없는 값
- e_i : i번째 비트를 제외한 모든 비트가 0인 32 비트 워드.
- $e_{i_1,\cdots,i_k}$: $e_{i_1}\oplus \cdots \oplus e_{i_k}$
- $e_{i_1,\cdots,i_k,\sim}$: $i_1,\ldots,i_k$번째 자리의 값은 1, $(i_k+1)\sim31$번째 자리의 값은 0, 1, 또는 알 수 없는 값이고, $i_1<\cdots<i_k$의 $i_1,\ldots,i_k$를 제외한 자리의 값은 0인 32 비트 워드.

2.2. SHACAL-2 블록 암호 소개

H. Handschuch와 D. Naccache에 의해 제안된 SHACAL-2[5]는 해쉬 함수 알고리즘 SHA-2[13]의 압축 함수에 기반을 두었으며, 다양한 키 길이(최대 512 비트)를 가지는 256 비트 블록 암호이다. SHACAL-2 암호화 과정은 다음과 같다.

256 비트 평문은 여덟 개의 32 비트 워드 $A,B,C,$

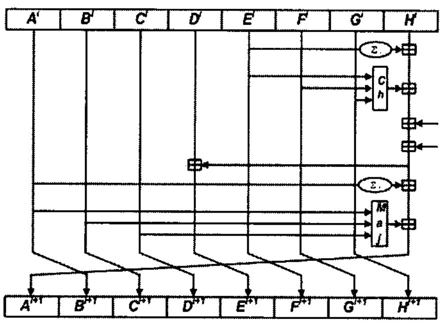


그림 1. SHACAL-2의 i번째 라운드 암호화 과정

D,E,F,G,H로 분할된다. 32 비트 워드 X^i를 i번째 라운드 입력 값이라 하면, 평문 P는 $A^0,B^0,C^0,D^0,E^0,F^0,G^0,H^0$으로 표현되며, 64 라운드 과정을 거친 후 암호문은 $A^{64},B^{64},C^{64},D^{64},E^{64},F^{64},G^{64},H^{64}$이다. $i(=0,...,63)$번째 라운드 암호화 과정은 다음과 같다(그림 1 참조).

$$
\begin{aligned}
T_1^{i+1} &= H^i + \Sigma_1(E^i) + Ch(E^i,F^i,G^i) \\
&\quad + K^i + W^i \\
T_2^{i+1} &= \Sigma_0(A^i) + Maj(A^i,B^i,C^i) \\
H^{i+1} &= G^i \\
G^{i+1} &= F^i \\
F^{i+1} &= E^i \\
E^{i+1} &= D^i + T_1^{i+1} \\
D^{i+1} &= C^i \\
C^{i+1} &= B^i \\
B^{i+1} &= A^i \\
A^{i+1} &= T_1^{i+1} + T_2^{i+1}
\end{aligned}
$$

+는 법 2^{32} 덧셈을 의미하며, W^i는 32 비트 라운드 키, K^i는 32 비트 라운드 상수 값이다. 위에 정의된 i번째 라운드 암호화 과정에 사용하는 함수는 다음과 같다.

$$
\begin{aligned}
Ch(X,Y,Z) &= (X\&Y)\oplus(\neg X\&Z) \\
Maj(X,Y,Z) &= (X\&Y)\oplus(X\&Z)\oplus(Y\&Z) \\
\Sigma_0(X) &= S_2(X)\oplus S_{13}(X)\oplus S_{22}(X) \\
\Sigma_1(X) &= S_6(X)\oplus S_{11}(X)\oplus S_{25}(X)
\end{aligned}
$$

$\neg X$는 32 비트 워드 X의 보수를 의미하며, $S_i(X)$는 32 비트 워드 X의 i비트 오른쪽 순환을 의미한다(즉, $S_i(X) = X \ggg i$).

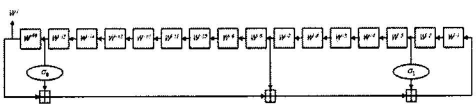


그림 2. SHACAL-2의 라운드 키 생성 과정

SHACAL-2의 키는 최대 512 비트까지 허용되며 512 비트 보다 작은 키에 대해서는 0 스트링을 패딩하여 총 512 비트를 생성한 후 사용한다. 하지만 SHACAL-2는 128 비트 보다 작은 키의 사용은 지양한다. 512 비트 키 스트링을 $W=[W^0\|W^1\|\cdots\|W^{15}]$와 같이 표시하면, 2048 비트 키 확장 과정은 다음과 같다(그림 2 참조).

$$
\begin{aligned}
W^i &= \sigma_1(W^{i-2}) + W^{i-7} + \sigma_0(W^{i-15}) \\
&\quad + W^{i-16}, (16 \le i \le 63), \\
\sigma_0(x) &= S_7(x)\oplus S_{18}(x)\oplus R_3(x) \\
\sigma_1(x) &= S_{17}(x)\oplus S_{19}(x)\oplus R_{10}(x)
\end{aligned}
$$

$R_i(x)$는 32 비트 워드 x의 i비트 오른쪽 쉬프트를 의미한다(즉, $R_i(x) = x \gg i$).

2.3. SHACAL-2의 3-라운드 비선형 관계식

본 소절에서는 SHACAL-2의 3-라운드 비선형 관계식[7,8]을 요약한다.

h_0^r는 비선형 함수 $NF(A^{r+3},B^{r+3},\cdots,H^{r+3},K^r,K^{r+1},K^{r+2},W^r,W^{r+1},W^{r+2})$의 출력 값으로 표현할 수 있다. 이 함수를 간단하게 NF^{r+3}로 표시한다. 단, $0 \le r \le 61$.

$$
\begin{aligned}
h_0^r = {} & c_0^{r+3}\oplus d_2^{r+3}\oplus d_{13}^{r+3}\oplus d_{22}^{r+3}\oplus(d_0^{r+3}\&(e_0^{r+3} \\
& \oplus t_{1,0}^{r+3}))\oplus(d_0^{r+3}\&(f_0^{r+3}\oplus t_{1,0}^{r+2})) \\
& \oplus((e_0^{r+3}\oplus t_{1,0}^{r+3})\&(f_0^{r+3}\oplus t_{1,0}^{r+2})) \\
& \oplus(h_0^{r+3}\&h_0^{r+2})\oplus((\neg h_0^{r+3})\&h_0^{r+1}) \\
& \oplus h_6^{r+2}\oplus h_{11}^{r+3}\oplus h_{25}^{r+3}\oplus k_0^r\oplus w_0^r
\end{aligned}
$$

위 비선형 방정식의 $h_0^{r+1},t_{1,0}^{r+2},h_0^{r+2},t_{1,0}^{r+3}$은 다음과 같이 표현할 수 있다.

$$
\begin{aligned}
h_0^{r+1} &= t_{1,0}^{r+2}\oplus g_6^{r+3}\oplus g_{11}^{r+3}\oplus g_{25}^{r+3}\oplus(g_0^{r+3}\&h_0^{r+3}) \\
&\quad \oplus((\neg g_0^{r+3})\&h_0^{r+2})\oplus k_0^{r+1}\oplus w_0^{r+1} \\
t_{1,0}^{r+2} &= b_0^{r+3}\oplus c_2^{r+2}\oplus c_{13}^{r+3}\oplus c_{22}^{r+3}\oplus(c_0^{r+3}\&d_0^{r+3}) \\
&\quad \oplus(c_0^{r+3}\&(e_0^{r+3}\oplus t_{1,0}^{r+3}))\oplus(d_0^{r+3} \\
&\quad \&(e_0^{r+3}\oplus t_{1,0}^{t+3})) \\
h_0^{r+2} &= t_{1,0}^{r+3}\oplus f_6^{r+3}\oplus f_{11}^{r+3}\oplus f_{25}^{r+3}\oplus(f_0^{r+3}\&g_0^{r+3}) \\
&\quad \oplus((\neg f_0^{r+3})\&h_0^{r+3})\oplus k_0^{r+2}\oplus w_0^{r+2} \\
t_{1,0}^{r+3} &= a_0^{r+3}\oplus b_2^{r+3}\oplus
\end{aligned}
$$

본 논문에서 소개하는 SHACAL-2의 연관키 차분-비선형 공격을 효과적으로 표현하기 위해서 MNF^{r+3} 함수를 사용한다. 단, $MNF^{r+3} = NF^{r+3} \oplus k_0^r \oplus w_o^r$.

Ⅲ. 연관키 공격

본 절에서는 블록 암호를 분석하는데 유용한 도구 중 하나인 연관키 공격[2]의 두 가지 형태를 소개한다. 먼저, 연관키 차분-선형 공격을 소개하고, 본 논문에서 이용하는 연관키 차분-비선형 공격으로의 확장 과정을 설명한다. 다음은 연관키 렉탱글 공격 방법의 일반적인 형태를 소개한다.

3.1. 연관키 차분-(비)선형 공격

1994년 Langford와 Hellman[10]은 차분 공격[1]과 선형 공격[11]을 결합하는 차분-선형 공격 방법을 소개하였다. 이 공격은 확률 1의 차분 특성과 바이어스 q를 갖는 선형 근사식을 이용한다. Biham, Dunkleman, Keller[3]는 차분 특성 확률이 1보다 작은 경우에도 차분-선형 공격을 가능하도록 일반화 한 향상된 차분-선형 공격을 소개하였다. [8]에서 선형 근사식 바이어스 q가 1/2이거나, 1/2에 매우 근접할 때, 차분 특성 대신 포화 특성을 이용할 수 있는 포화-선형 공격을 소개하였다. 또한 [8]에서는 위의 모든 공격은 선형 특성의 이용 대신 비선형 특성을 이용할 수 있음을 보였다. 즉, 1994년에 소개된 차분-선형 공격은 현재까지 다양한 일반화된 형태로 발전되고 있다.

1998년 Hawkes[6]는 연관키와 차분-선형 공격을 결합한 연관키 차분-선형 공격을 소개하였다. 여기서 사용된 공격은 확률 1의 연관키 차분 특성과 바이어스 1/2의 선형 근사식을 이용한다. 하지만 연관키 차분-선형 공격은 차분-선형 공격의 발전과 유사하게 확률 1보다 작은 차분 특성과 바이어스가 1/2보다 작은 선형 근사식을 이용할 수 있도록 일반화가 가능하다. 더구나 연관키 차분-선형 공격은 비선형 관계식을 이용할 수 있는 경우로 확장할 수 있으며, 이를 연관키 차분-비선형 공격이라 부른다.

연관키 차분-선형 공격은 키 k에 대한 평문 P와 키 k^*에 대한 평문 P^*의 암호문 쌍을 요구한다. 단, k와 k^*는 서로 다르지만, 연관된 키이다. 차분 특성을 표현하기 위하여 본 논문에서는 입력 차분을 Ω_P으로, 출력 차분을 Ω_T으로 표기한다. 또한 선형 근사식을 표현하기 위하여 입력 비트 마스크, 출력 비트 마스크, 부분 키 비트 마스크를 각각 λ_P, λ_T, λ_K으로 표기한다.

블록 암호 $E_k : 0,1^n \rightarrow 0,1^n$가 두 개의 부분 암호 E_k^0, E_k^1의 합성 형태 $E_k = E_k^1 \circ E_k^0$로 표현된다고 가정하자. E_k를 함수$E: \{0,1\}^{|k|} \times \{0,1\}^n \rightarrow \{0,1\}^n$로 나타낸다면, 위의 가정 하에 E는 E^0, E^1의 합성 형태 $E = E^1 \circ E^0$으로 표현할 수 있다. E^0에 확률 $p^* \leq 1$의 연관키 차분 특성 $\Omega_P \rightarrow \Omega_T$ (즉, $\Pr_X[E_k^0(X) \oplus E_{k^*}^0(X^*) = \Omega_T | X \oplus X^* = \Omega_P] = p^*$)이 존재한다고 가정한다. 그리고 E^1에 확률 $\frac{1}{2}+q$ 또는 바이어스 q의 선형 특성 $\lambda_p \rightarrow \lambda_T$ (즉, $\Pr_X[\lambda_P \cdot X \oplus \lambda_T \cdot E^1(X) \oplus \lambda_K \cdot K = 0] = \frac{1}{2}+q$, 단, K는 부분 키)이 존재한다고 가정한다. P와 P^*를 차분 조건 $P \oplus P^* = \Omega_P$을 만족하는 평문 쌍이라 가정할 때, 평문쌍 P와 P^*가 확률 p^*의 차분 특성 $\Omega_P \rightarrow \Omega_T$을 만족하는 경우에 확률 1의 한 비트 방정식 $\lambda_P \cdot (E_k^0(P) \oplus E_{k^*}^0(P^*)) = a$을 얻을 수 있다. 단, a의 값은 0 또는 1이며, $a = \lambda_P \cdot \Omega_T$이다.

만약 평문쌍 P와 P^*가 확률 $1-p^*$의 차분 특성 $\Omega_P \rightarrow \Omega_T$을 만족하지 않는 경우에 한 비트 식 $\lambda_P \cdot (E_k^0(P) \oplus E_{k^*}^0(P^*))$은 랜덤한 값을 갖는다. 위의 두 경우를 고려하여 다음과 같은 확률 $\frac{1}{2}+\frac{p^*}{2}$ $(=p^* \cdot 1 + (1-p^*) \cdot \frac{1}{2})$의 한 비트 방정식을 얻을 수 있다.

$$\lambda_P \cdot (E_k^0(P) \oplus E_{k^*}^0(P^*)) = a \qquad (1)$$

위의 E^1의 선형 근사식의 존재 가정에 따라 다음과 같은 확률 $\frac{1}{2}+q$ 또는 바이어스 q의 갖는 선형 근사식 두 개를 얻을 수 있다.

$$\lambda_P \cdot E_k^0(P) \oplus \lambda_T \cdot E_k^1(E_k^0(P)) \oplus \lambda_K \cdot K = 0 \qquad (2)$$
$$\lambda_P \cdot E_{k^*}^0(P^*) \oplus \lambda_T \cdot E_{k^*}^1(E_{k^*}^0(P^*)) \oplus \lambda_K \cdot K^* = 0 \qquad (3)$$

따라서, [11]에 소개된 piling up lemma를 사용하여 확률 $\frac{1}{2}+2p^*q^2 (= \frac{1}{2}+2^{3-1} \cdot \frac{p^*}{2} \cdot q^2)$을 갖는 선형 근사식을 다음과 같이 얻을 수 있다.

$$\lambda_T \cdot E_k^1(E_k^0(P)) \oplus \lambda_T \cdot E_{k^*}^1(E_{k^*}^0(P^*)) \oplus \lambda_K \cdot K \oplus \lambda_K \cdot K^* = a \qquad (4)$$

즉, 바이어스 $2p^*q^2$의 다음 선형 근사식을 얻을 수 있다.

$$\lambda_T \cdot E_k(P) \oplus \lambda_T \cdot E_{k^*}(P^*) = 0 \qquad (5)$$

따라서 (5)을 이용한 선형 공격을 수행하기 위해서 $O(p^{*-2}q^{-4})$ 개의 연관키 선택 평문쌍을 요구한다.

위에서 언급하였다시피 연관키 차분-선형 공격은 선형 근사식 대신에 비선형 관계식을 이용할 수 있다. 본 논문에서는 이를 연관키 차분-비선형 특성이라 명한다.

3.2. 연관키 렉탱글 공격

2004년 연관키[2]와 렉탱글[4] 공격을 결합한 연관키 렉탱글 공격이 소개되었다[9]. 연관키 렉탱글 공격의 주는 연속된 두 개의 차분 특성을 사용하는 것이다. 단, 하나는 연관키 차분 특성이고, 다른 하나는 차분 특성을 이용한다. 따라서 본 공격은 확률이 높은 연관키 차분 특성에 높은 확률을 갖는 차분 특성을 연결하여 구성할 수 있을 때 매우 유용하다.

위와 같이 블록 암호 $E_k : 0,1^n \rightarrow 0,1^n$가 두 개의 부분 암호 E_k^0, E_k^1의 합성 형태 $E_k = E_k^1 \circ E_k^0$로 표현된다고 가정하자. E^0에 확률 p_β^*의 연관키 차분 특성 $\alpha \rightarrow \beta$이 존재한다고 가정하자. 즉, $\Pr_X[E_k^0(X) \oplus E_{k^*}^0(X^*) = \beta \,|\, X \oplus X^* = \alpha] = p_\beta^*$이며, k, k^*는 서로 다르지만, 연관된 키이다. 그리고 E^1에 확률 q_γ의 차분 특성 $\gamma \rightarrow \delta$이 존재한다고 가정하자. 즉, $\Pr_X[E_k^1(X) \oplus E_{k^*}^1(X') = \delta \,|\, X \oplus X' = \gamma] = q_\gamma$.

연관키 렉탱글 특성은 몇 가지의 차분 조건을 만족하는 4개의 평문 쌍 (P_i, P_i^*, P_j, P_j^*)에 의해 생성된다. P_i, P_j는 E_k에 의해, P_i^*, P_j^*는 E_{k^*}에 의해 암호화 될 때, 차분 조건 $P_i \oplus P_i^* = P_j \oplus P_j^* = \alpha$이 성립한다고 가정하자. 평문 P_i, P_i^*, P_j, P_j^*에 대한 E^0의 암호문을 각각 X_i, X_i^*, X_j, X_j^*라 하고, X_i, X_i^*, X_j, X_j^*에 대한 E^1의 암호문을 각각 C_i, C_i^*, C_j, C_j^*라 하자. 위의 가정 하에 차분 조건 $X_i \oplus X_i^* = X_j \oplus X_j^* = \beta$와 $X_i \oplus X_j = \gamma$을 만족한다면, $X_i^* \oplus X_j^* = (X_i \oplus \beta) \oplus (X_j \oplus \beta) = \gamma$이 성립한다. 만약 위의 차분 조건 하에 암호문 쌍 C_i, C_j와 C_i^*, C_j^*이 차분 δ를 만족한다면(즉, $C_i \oplus C_j = C_i^* \oplus C_j^* = \delta$), 위의 4개의 평문 쌍 (P_i, P_i^*, P_j, P_j^*)을 올바른 quartet이라 부른다. 또한, 차분 조건 $X_i \oplus X_i^* = X_j \oplus X_j^* = \beta$와 $X_i \oplus X_j^* = \gamma$을 만족한다면, $X_i^* \oplus X_j = (X_i \oplus \beta) \oplus (X_j^* \oplus \beta) = \gamma$이 성립한다. 만약 암호문 쌍 C_i, C_j^*와 C_i^*, C_j이 차분 δ를 만족한다면(즉, $C_i \oplus C_j^* = C_i^* \oplus C_j = \delta$), 이러한 모든 차분 조건을 만족하는 4개의 평문 쌍 (P_i, P_i^*, P_j, P_j^*) 또한 올바른 quartet이라 부른다. 올바른 quartet은 그림 3에 나타나 있다. 더욱 일반적으로 올바른 quartet은 주어진 차분 값 α와 δ에 대해 임의의 β와 γ 차분 값에 따라 만족한다.

차분 α를 만족하는 m개의 평문 쌍(키 k를 사용하는 한 평문과 키 k^*를 사용하는 다른 한 평문)이 있다고 가정하자. 부분 암호 E^0을 통해 연관키 차분 특성 $\alpha \rightarrow \beta$를 만족하는 쌍은 약 mp_β^*개 이며, mp_β^*개의 평

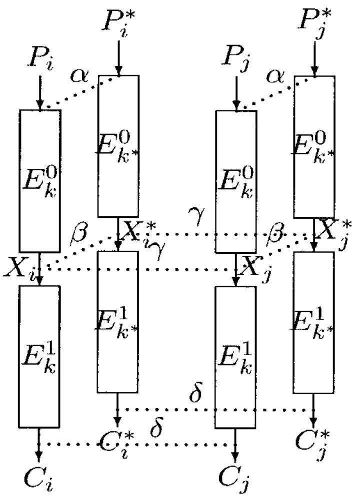


그림 3. 연관키 Rectangle Distinguisher-1

문 쌍은 약 $\frac{(mp_\beta^*)^2}{2}$개의 quartet을 생성한다. 블록 암호의 암호화 과정에서 중간 과정의 값이 랜덤하게 분포한다고 가정하면, 확률 2^{-n}으로 차분 $X_i \oplus X_j = \gamma$를 얻는다. 이 경우에 올바른 quartet이 되기 위해

X_i, X_j와 X_i^*, X_j^*는 확률 q_γ를 갖는 차분 특성 $\gamma \rightarrow \delta$을 만족해야 한다. 따라서 차분 α를 만족하는 m개의 평문 쌍에 대한 올바른 quartet의 평균 기대값은 다음과 같다. 단, $\hat{p}^* = (\sum_\beta (p_\beta^*)^2)^{\frac{1}{2}}$, $\hat{q} = (\sum_\gamma (q_\gamma)^2)^{\frac{1}{2}}$.

$$\sum_{\beta,\gamma} \frac{(mp_\beta^*)^2}{2} \cdot 2^{-n} \cdot (q_\gamma)^2 = m^2 \cdot 2^{-n-1} \cdot (\hat{p}^*)^2 \cdot (\hat{q})^2$$

랜덤 치환 함수인 경우에 올바른 quartet의 평균 기대값은 약 $m^2 \cdot 2^{-2n-1} (= \binom{m}{2} \cdot 2^{-2n})$이므로, 만약 $\hat{p}^* \cdot \hat{q} > 2^{-n/2}$을 만족하고, m이 충분히 크다면, 연관키 렉탱글 특성을 갖는 E와 랜덤 치환 함수는 구별 가능하다.

Ⅳ. 35-라운드 SHACAL-2에 대한 연관키 차분-비선형 공격

본 절에서는 SHACAL-2의 28-라운드 연관키 차분-비선형 특성을 구성하는 방법을 소개한 후, 이 특성을 이용한 35-라운드 SHACAL-2의 연관키 차분-비선형 공격을 설명한다.

2.2절에서 설명하였다시피 SHACAL-2의 키 스케줄 알고리즘은 선형 귀환 쉬프트 레지스터를 기반으로 수행한다. 하지만, 키 스케줄 알고리즘은 처음 몇 라운드에 대해 차분 확산 효과가 작다. 즉, 6 라운드 키 W^6를 제외하고 모두 같은 연관키를 고려할 때, 확장 키 $W^{16}, W^{17}, ..., W^{20}$은 모두 같으며, W^{21}은 $e_{13,\sim}$ 차분 값을 가지며, W^{22}는 e_{31} 차분 값을 갖는다. 이러한 연관키 차분 특성은 높은 확률로 성립하는 25-라운드 연관키 부정 차분 특성을 구성할 수 있게 해준다. 다시 말해서, 확률 2^{-16}으로 만족하는 라운드 $0 \sim 24(E^0)$에 대한 25-라운드 부정 차분 특성 $\Omega_P \rightarrow \Omega_T$을 구성할 수 있다. 단,

$$\Omega_P = (0, e_{31}, 0, 0, e_{6,20,25}, 0, 0, e_{9,13,19}),$$
$$\Omega_T = (?, ?, ?, e_{13,\sim}, ?, ?, ?, e_{13,\sim})$$

A, A^*	C, C^*	F, F^*	G, G^*
$a_{31} = a_{31}^* = 0$	$c_{31} = c_{31}^* = 0$	$f_6 = f_6^* = 0$, $f_{20} = f_{20}^* = 0$, $f_{25} = f_{25}^* = 0$	$g_6 = g_6^* = 0$, $g_{20} = g_{20}^* = 0$, $g_{25} = g_{25}^* = 0$

표 2. 평문 쌍 P, P^*의 고정 비트

라운드 (i)	ΔA^i	ΔB^i	ΔC^i	ΔD^i	ΔE^i	ΔF^i	ΔG^i	ΔH^i	ΔW^i	확률
입력 ($i=0$)	0	e_{31}	0	0	$e_{6\cdot20\cdot25}$	0	0	$e_{9\cdot13\cdot19}$	0	2^{-13}
1	0	0	e_{31}	0	0	$e_{6,20,25}$	0	0	0	2^{-4}
2	0	0	0	e_{31}	0	0	$e_{6\cdot20\cdot25}$	0	0	2^{-3}
3	0	0	0	0	e_{31}	0	0	$e_{6\cdot20\cdot25}$	0	2^{-4}
4	0	0	0	0	0	e_{31}	0	0	0	2^{-1}
5	0	0	0	0	0	0	e_{31}	0	0	2^{-1}
6	0	0	0	0	0	0	0	e_{31}	e_{31}	1
7	0	0	0	0	0	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
20	0	0	0	0	0	0	0	0	0	1
21	0	0	0	0	0	0	0	0	$e_{13\cdot\sim}$	1
22	$e_{13\cdot\sim}$	0	0	0	$e_{13\cdot\sim}$	0	0	0	e_{31}	1
23	?	$e_{13\cdot\sim}$	0	0	?	$e_{13\cdot\sim}$	0	0	?	1
24	?	?	$e_{13\cdot\sim}$	0	?	?	$e_{13\cdot\sim}$	0	?	1
출력 ($i=25$)	?	?	?	$e_{13\cdot\sim}$	?	?	?	$e_{13\cdot\sim}$		

표 3. SHACAL-2의 25-라운드(E^0) 연관키 부정 차분 특성

이며, 평문 쌍 P, P^*은 표 2과 같이 고정된 비트 값을 갖는다. 연관키 부정 차분 특성 경로에 대한 자세한 내용은 표 3를 참고하라.

위에 소개한 25-라운드 연관키 부정 차분 특성은 확률 $\frac{1}{2}+2^{-17} (= 2^{-16} + \frac{1}{2} \cdot (1-2^{-16}))$을 만족하는 선형 특성으로 바꾸어 적용할 수 있다. 즉, 만약 평문쌍 P, P^*가 차분 $(0, e_{31}, 0, 0, e_{6,20,25}, 0, 0, e_{9,13,19})$을 만족하고, [표 2]와 같은 고정 값을 갖는다면, 확률 $\frac{1}{2}+2^{-17}$으로 $h_0^{25} = h_0^{*25}$을 만족한다. 확률 $\frac{1}{2}+2^{-17}$을 확인하기 위하여 2^{34} 평문 쌍을 이용하여 다섯 번의 시뮬레이션을 수행하였다.

각 시뮬레이션은 무작위 연관키 쌍과 평문 쌍을 가지고 수행하였다. 25-라운드 특성은 확률 $\frac{1}{2}+2^{-17}$을 만족하기 때문에, 2^{34} 평문 쌍 중 $h_0^{25} = h_0^{*25}$을 만족하는 평문 쌍의 개수의 기댓값은 약 $2^{33}+131072$ $(= 2^{34} \cdot (\frac{1}{2}+2^{-17}))$이다. 다섯 번의 시뮬레이션 결과는 $2^{33}+128629$, $2^{33}+130921$, $2^{33}+1138897$, $2^{33}+143916$,

위의 25-라운드 특성 식에 앞서 설명한 3 라운드 비선형 관계식을 연결함으로써 더욱 강력한 특성 식을 얻을 수 있다. 주어진 평문 쌍 P, P^*는 $\frac{1}{2}+2^{-17}$의 근사 확률을 가지고 $h_0^{25}=h_0^{*25}$을 만족한다. 이에 3 라운드 비선형 관계식을 이용하여 확률 $\frac{1}{2}+2^{-17}$으로 $NF^{28}=NF^{*28}$을 만족하는 28-라운드 특성식을 구성할 수 있다. 같은 의미로 선형 근사 바이어스 2^{-17}을 만족하는 다음 식으로 나타낼 수 있다.

$$MNF^{28}=MNF^{*28} \tag{6}$$

따라서 확률 $\frac{1}{2}+2^{-17}$을 만족하는 28-라운드 연관키 차분-비선형 특성을 구성할 수 있다.

다음 과정은 28-라운드 연관키 차분-비선형 특성을 이용하여 35-라운드 SHACAL-2의 연관키를 찾는 방법을 소개한다.

알고리즘 1

1) 차분 Ω_P을 만족하고 8비트 고정 값을 갖는 2^{39}개의 평문쌍 $(P_{i,j}, P^*_{i,j}), i=0,1,...,4, j=0,1,..., 2^{39}-1$으로 이루어진 5 개의 집합을 선택한다. 단, 임의의 두 집합 i에 대해서 평문 쌍은 모두 다르게 선택한다. 여기서 $P_{i,j}$는 키 k를 사용하여 암호화 하며, $P^*_{i,j}$는 키 k와 차분 $(0,0,0,0,0,0,e_{31},0,0,0,0,\ 0,0,0,0,0,0)$을 갖는 키 k^*를 사용하여 암호화 한다. 평문 쌍의 선택 후 2^{39}개의 평문 쌍으로 이루어진 5개의 집합을 키 k, k^*를 사용하여 암호문 쌍 $(C_{i,j}, C^*_{i,j}), i=0,1,...,4, j=0,1,...,2^{39}-1$을 요구한다.
2) 207 비트 부분 키 쌍 (sk, sk^*)를 추측한다. 부분 키 sk는 $W^{34}, W^{33}, W^{32}, W^{31}, w_0^{30}, w_1^{30}, ..., w_{25}^{30}, w_0^{29}, w_1^{29}, ..., w_{25}^{29}, w_0^{28}, w_1^{28}, ..., w_{24}^{28}, w_0^{27}, w_0^{26}$을 다른 부분 키 sk^*는 $W^{*34}, W^{*33}, W^{*32}, W^{*31}$ $w_0^{*30}, w_1^{*30}, ..., w_{25}^{*30}, w_0^{*29}, w_1^{*29}, ..., w_{25}^{*29}, w_0^{*28}, w_1^{*28}, ..., w_{24}^{*28}, w_0^{*27}, w_0^{*26}$ 을 표현한다. 주어진 35-라운드 SHACAL-2의 연관키 암호문 쌍으로 부터$\triangle MNF^{28}$ 을 계산하기 위해서 207비트 키의 추측만으로 충분하다.
3) $i=0,1,...,4$에 대해서 다음을 수행한다.
추측한 부분 키 sk를 사용하여 모든 2^{39}개의 암호문 $C_{i,j}$를 부분적으로 복호화 하고 추측한 부분 키 sk^*를 사용하여 모든 2^{39}개의 암호문 $C^*_{i,j}$를 부분적으로 복호화 하여 식 (6)의 성립여부를 판단한다. 만약 식 (6)을 만족하는 암호문 쌍의 개수가 $2^{38}-2^{21.6}$보다 크고 $2^{38}+2^{21.6}$보다 작다면 단계 2로 돌아간다.
단계 3까지 통과한 부분 키 sk에 대해 나머지 305 비트 키에 대한 전수조사를 수행한다. 이 과정을 통과한다면, 512 비트 키 k'를 35-라운드 SHACAL-2의 마스터 키로 $k' \oplus (0,0,0,0,0,0,e_{31},0,0,0,0,0,0,0,0,0)$를 연관된 512 비트 마스터 키로 출력한다. 그렇지 않다면, 단계 2로 돌아간다. 207 비트 부분키 쌍 (sk, sk^*) 을 모두 테스트 했음에도 불구하고 k'가 출력 되지 않는다면, 알고리즘 1은 출력 값 없이 종료한다.

알고리즘 1의 데이터 복잡도는 $2^{42.32}(\approx 5 \cdot 2 \cdot 2^{39})$ 연관키 선택 평문을 요구하며, 공격에 사용되는 메모리는 암호문 쌍 $(C_{i,j}, C^*_{i,j})$의 저장 공간에 의존하므로 약 $2^{47.32}(=5 \cdot 2 \cdot 2^{39} \cdot \frac{256}{8})$ 바이트를 요구한다.

또한 알고리즘 1의 계산 복잡도는 다음과 같이 평가할 수 있다. 단계 1의 계산 복잡도는 $2^{42.32}$번의 35-라운드 SHACAL-2 암호화 연산이다(데이터 수집 단계). 단계 3의 계산 복잡도를 평가하기 위해서 각각의 i에 대해서 생존하는 부분 키 쌍 (sk, sk^*)의 비율을 계산하여야 한다. 랜덤 치환 함수인 경우 각 쌍은 랜덤하게 보이기 때문에, 올바르지 않게 추측한 부분 키 쌍에 대해서는 전체 암호문 쌍 중 평균 절반이 식 (6)을 만족한다. 그러므로 식 (6)을 만족하는 평문 쌍의 개수는 이항 분포 $X \sim Bin(2^{39}, \frac{1}{2})$를 따른다. 이항분포는 쉽게 정규 분포로 근사할 수 있다. 즉, $X \sim N(\mu, \sigma^2)$. 단, $\mu=2^{38}$, $\sigma^2=2^{37}$. 따라서 $Z(=\frac{X-\mu}{\sigma}) \sim N(0,1)$이고,

$$Pr[X \geq 2^{38}+2^{21.6}\ or\ X \leq 2^{38}-2^{21.6}] =$$
$$Pr[Z \geq 8.5742\ or\ Z \leq -8.5742] \approx 2^{-53.27}$$

을 확인할 수 있다. 각각의 i에 대해 생존하는 부분 키 쌍의 비율은 약 $2^{-53.27}$이며, 단계 3의 i번째에 대해 생존하는 부분 키 쌍의 개수는 약 $(2^{207})^2 \cdot 2^{-53.27*(i+1)}$이다. 그러므로 단계 3의 계산 복잡도는

$$2^{451.64}(\approx \sum_{i=0}^{4} 2^{39} \cdot 2 \cdot (2^{207})^2 \cdot 2^{-53.27*i} \cdot \frac{7}{35})$$

번의 35-라운드 SHACAL-2 암호화 연산이다. 단계 3을 통과하는 부분 키 쌍의 개수는 약 $2^{147.65}(\approx (2^{207})^2 \cdot 2^{-53.27*5})$이기 때문에, 단계 4의 계산 복잡도는 약 $2^{450.65}(\approx 2^{147.65} \cdot 2^{305} \cdot \frac{7}{35})$ 35-라운드 SHACAL-2 암호화 연산이다. 따라서 알고리즘 1의 전체 계산 복잡도는 약 $2^{452.1}(\approx 2^{42.32}+2^{451.64}+2^{450.21})$ 번의 35-라운드 SHACAL-2 암호화 연산이다.

알고리즘 1의 성공확률을 계산하기 위해 올바른 부분 키 쌍이 단계 3을 통과하는 확률을 계산한다. 올바른 부분 키 쌍에 대해 선형 근사식 확률 $\frac{1}{2}+2^{-17}$ 또는 $\frac{1}{2}-2^{-17}$으로 식 (6)을 만족한다. $\frac{1}{2}+2^{-17}$인 경우에 이항 분포 $X \sim Bin(2^{39}, \frac{1}{2}+2^{-17})$을 따른다. 즉, $X \sim N(\mu, \sigma^2)$이며, 단, $\mu = 2^{38}+2^{22}$, $\sigma^2 = \mu \cdot (\frac{1}{2}-2^{-17})$이다. X는 올바른 부분 키 쌍에 대해 식 (6)을 만족하는 평문 쌍의 개수이다. 위의 공격 과정을 사용하여 단계 3의 각 i에 대해 올바른 부분 키 쌍이 통과할 확률은 약 $1-2^{-8.34}(\approx \Pr[X \geq 2^{38}++2^{21.6}] = \Pr[Z \leq -2.7395])$이다. 따라서 단계 3에서 올바른 부분 키 쌍이 통과할 확률은 약 $0.98(\approx (1-2^{-8.34})^5)$ 이다. $\frac{1}{2}-2^{-17}$인 경우에도 같은 결과를 얻을 수 있다. 그러므로 알고리즘 1의 성공 확률은 98%이다. 또한 알고리즘 1은 같은 복잡도와 성공 확률로 [12]에 소개된 키 랭킹 알고리즘으로 바꾸어 생각할 수 있다. 하지만 키 랭킹 알고리즘은 모든 가능한 2^{414} 부분 키 쌍 (sk, sk^*)에 대한 더 많은 메모리를 요구한다.

V. 37-라운드 SHACAL-2의 연관키 렉탱글 공격

본 절에서는 SHACAL-2의 33-라운드 연관키 렉탱글 특성을 구성하는 방법을 소개한 후, 이 특성을 이용한 37-라운드 SHACAL-2의 연관키 렉탱글 공격을 설명한다.

앞서 설명하였다시피 SHACAL-2의 키 스케줄 알고리즘은 처음 몇 라운드에 대해 차분 확산 효과가 작다. 8 라운드 키 W^8를 제외하고 모두 같은 연관키를 고려할 때, 확장 키 $W^{16}, W^{17}, \ldots, W^{22}$은 모두 같다. 이러한 연관키 차분 특성은 높은 확률로 성립하는 23-라운드 연관키 차분 특성을 구성할 수 있게 해준다. 다시 말해서, 확률 2^{-31}으로 만족하는 라운드 $0 \sim 22(E^0)$에 대한 23-라운드 연관키 차분 특성 $\alpha \rightarrow \beta$을 구성할 수 있다. 단,

$$\alpha = (0,0,e_{6,9,18,20,25,29,},e_{31},0,e_{9,13,19,},e_{18,29},e_{31}),$$
$$\beta = (0,0,0,0,0,0,0,0)$$

이며, 평문 쌍 P, P^*은 표 4과 같이 고정된 비트 값을 갖는다. 연관키 차분 특성 경로에 대한 자세한 내용은 표 5를 참고하라.

표 4. 평문 쌍 P, P^*의 고정 비트

A,A^*	B,B^*	E,E^*	F,F^*	G,G^*
$a_6 = a_6^* = 0$, $a_9 = a_9^* = 0$, $a_{18} = a_{18}^* = 0$, $a_{20} = a_{20}^* = 0$, $a_{25} = a_{25}^* = 0$, $a_{29} = a_{29}^* = 0$	$b_6 = b_6^* = 0$, $b_9 = b_9^* = 0$, $b_{18} = b_{18}^* = 0$, $b_{20} = b_{20}^* = 0$, $b_{25} = b_{25}^* = 0$, $b_{29} = b_{29}^* = 0$	$e_9 = e_9^* = 1$, $e_{13} = e_{13}^* = 1$, $e_{18} = e_{18}^* = 1$, $e_{19} = e_{19}^* = 1$, $e_{29} = e_{29}^* = 1$	$f_{18} = f_{18}^* = 0$, $f_{29} = f_{29}^* = 0$	$g_9 = g_9^* = 0$, $g_{13} = g_{13}^* = 0$, $g_{19} = g_{19}^* = 0$

표 5. SHACAL-2의 23-라운드(E^0) 연관키 차분 특성 ($M = \{6,9,18,20,25,29\}$)

라운드 (i)	ΔA^i	ΔB^i	ΔC^i	ΔD^i	ΔE^i	ΔF^i	ΔG^i	ΔH^i	ΔW^i	확률
0	0	0	e_M	e_{31}	0	$e_{9 \cdot 13 \cdot 19}$	$e_{18 \cdot 29}$	e_{31}	0	1
1	e_{31}	0	0	e_M	0	0	$e_{9 \cdot 13 \cdot 19}$	$e_{18 \cdot 29}$	0	2^{-11}
2	0	e_{31}	0	0	$e_{6 \cdot 20 \cdot 25}$	0	0	$e_{9 \cdot 13 \cdot 19}$	0	2^{-7}
3	0	0	e_{31}	0	0	$e_{6 \cdot 20 \cdot 25}$	0	0	0	2^{-4}
4	0	0	0	e_{31}	0	0	$e_{6 \cdot 20 \cdot 25}$	0	0	2^{-3}
5	0	0	0	0	e_{31}	0	0	$e_{6 \cdot 20 \cdot 25}$	0	2^{-4}
6	0	0	0	0	0	e_{31}	0	0	0	2^{-1}
7	0	0	0	0	0	0	e_{31}	0	0	2^{-1}
8	0	0	0	0	0	0	0	e_{31}	e_{31}	1
9	0	0	0	0	0	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
22	0	0	0	0	0	0	0	0	0	1
23	0	0	0	0	0	0	0	0		

앞에 설명하였다시피 연관키 Rectangle 특성은 E^1에 대해 연관키 차분 특성이 아닌 차분 특성을 이용한다. 즉, E^1에 대해서는 라운드 키의 차분 확산 효과를 고려할 필요가 없다. 따라서 E^1에 대한 확률 2^{-74}로 10-라운드(23 ~32) 차분 특성 $\gamma \rightarrow \delta$를 구성할 수 있다. 단,

$$\gamma = (0, e_{9,18,29}, 0, 0, e_{31}, e_{6,9,18,20,25,29}, 0, 0),$$

$$\delta = (e_{11,23}, e_{3,14,15,24,25}, e_{5,27}, e_{9,18,29}, e_{31}, 0, 0, 0)$$

이다. 차분 특성 경로에 대한 자세한 내용은 표 6을 참고하라.

표 6. SHACAL-2의 10-라운드(E^1) 차분 특성 ($M = \{3,14,15,24,25\}$)

라운드 (i)	ΔA^i	ΔB^i	ΔC^i	ΔD^i	ΔE^i	ΔF^i	ΔG^i	ΔH^i	확률
23	0	$e_{9\cdot18\cdot29}$	0	0	e_{31}	e_M	0	0	2^{-13}
24	0	0	$e_{9\cdot18\cdot29}$	0	0	e_{31}	e_M	0	2^{-10}
25	e_{31}	0	0	$e_{9\cdot18\cdot29}$	e_{31}	0	e_{31}	e_M	2^{-10}
26	0	e_{31}	0	0	0	e_{31}	0	e_{31}	2^{-2}
27	0	0	e_{31}	0	0	0	e_{31}	0	2^{-2}
28	0	0	0	e_{31}	0	0	0	e_{31}	1
29	e_{31}	0	0	0	0	0	0	e_{31}	2^{-4}
30	$e_{9\cdot18\cdot29}$	e_{31}	0	0	0	0	0	0	2^{-6}
31	$e_{5\cdot27}$	$e_{9\cdot18\cdot29}$	e_{31}	0	0	0	0	0	2^{-12}
32	e_M	$e_{5\cdot27}$	$e_{9\cdot18\cdot29}$	e_{31}	0	0	0	0	2^{-15}
33	$e_{11\cdot23}$	e_M	$e_{5\cdot27}$	$e_{9\cdot18\cdot29}$	e_{31}	0	0	0	

표 7. SHAHCAL-2의 10-라운드(E^1) 차분 특성에 사용한 확률에 따른 차분 특성 개수

확률(p)	2^{-74}	2^{-75}	2^{-76}	2^{-77}	…
차분 특성 개수	15	60	236	440	…

$\widehat{p^*}$를 계산하기 위해 E^0의 입력 차분 α를 갖는 모든 연관키 차분 특성 확률을 계산하여야 한다. 또한 $\hat{q}$를 계산하기 위해 E^1의 출력 차분 δ를 갖는 모든 차분 특성 확률을 계산하여야 한다. 하지만 모든 특성 경로를 고려하는 것은 계산적으로 불가능하기 때문에 $E^0(E^1)$에 대한 $\widehat{p^*}(\hat{q})$의 하한 값을 계산하기 위해 가능한 많은 연관키 차분 특성(차분 특성)을 고려하여야 한다. 하지만 분석 결과 E^0의 입력 차분 α를 갖는 임의의 23-라운드 연관키 차분 특성 확률은 2^{-31}보다 너무 작다. 따라서 E^0의 연관키 차분 특성 확률의 하한 값을 2^{-31}으로 결정할 것이다. $\hat{q}$의 하한 값을 계산하기 위해 E^1의 10-라운드 차분 특성 확률에서 마지막 9-라운드 특성이 같은 다양한 입력 차분 값을 고려하였다. 표 7은 확률에 따른 존재하는 차분 특성의 개수를 나타낸다. 결과적으로 $\widehat{p^*}$의 하한 값은 2^{-31}이며, $\hat{q}$의 하한 값은 $2^{-71.16}$으로 증가시킬 수 있다. 연관키 Rectangle 특성 확률 $\widehat{p^*} \cdot \hat{q}(\approx 2^{-102.16})$은 2^{-128} 보다 크기 때문에, 구성한 33-라운드 연관키 Rectangle 특성은 33-라운드 SHACAL-2와 랜덤 치환 함수를 구별할 수 있는 특성이 될 수 있다. 다음 과정은 33-라운드 연관키 Rectangle 특성을 이용하여 37-라운드 SHACAL-2의 연관키를 찾는 방법을 소개한다.

알고리즘 2

1) 차분 α을 만족하고 22비트 고정 값을 갖는 $2^{232.16}$개의 평문 쌍 $(P_i, P_i^*), i = 0,1,\ldots,2^{232.16}-1$을 선택한다. 여기서 P_i는 키 k를 사용하여 암호화 하며, P_i^*는 키 k와 차분 $(0,0,0,0$ $0,0,0,0,e_{31},0,0,0,0,0,0,0)$을 갖는 키 k^*를 사용하여 암호화 한다. 평문 쌍의 선택 후 키 k, k^*를 사용하여 암호문 쌍(C_i, C_i^*)을 요구한다.
2) 128 비트 부분 키 쌍 $sk(= W^{33}, W^{34}, W^{35}, W^{36})$, $sk^*(= W^{*33}, W^{*34}, W^{*35}, W^{*36})$를 추측한다.
3) 추측한 부분 키 sk를 사용하여 36~33 라운드에 대한 모든 C_i를 부분 복호화를 수행한다. 또한 추측한 부분 키 sk^*를 사용하여 36 ~33 라운드에 대한 모든 C_i^*를 부분 복호화를 수행한다. 부분 복호화를 수행하여 얻은 복호문 T_i, T_i^*를 해쉬 테이블에 저장한다.
4) 모든 i_1, i_2에 대해 $T_{i_1} \oplus T_{i_2} = T_{i_1}^* \oplus T_{i_2}^* = \delta$을 테스트 한다. 단, $0 \le i_1 < i_2 \le 2^{232.16} - 1$. 만약 δ 값을 테스트 하였을 때, 성립하는 개수가 6보다 크다면, 그때의 모든 $(T_{i_1}, T_{i_2}, T_{i_1}^*, T_{i_2}^*)$와 추측한 128 비트 부분 키 sk를 저장한다. 만약 그렇지 않다면, 단계 2로 돌아간다.
5) 두 개의 32 비트 부분 키 W^{32}, W^{*32}를 추측한다. 추측한 부분 키 W^{32}를 사용하여 32 라운드에 대한 모든 T_{i_1}, T_{i_2}의 부분 복호화를 수

행하여, T'_{i_1}, T'_{i_2}에 저장한다. 또한 추측한 부분 키 W^{*32}를 사용하여 32 라운드에 대한 모든 $T^*_{i_1}, T^*_{i_2}$의 부분 복호화를 수행하여 $T'^*_{i_1}, T'^*_{i_2}$에 저장한다. 만약 모든 $T^*_{i_1}, T^*_{i_2}$와 $T'^*_{i_1}, T'^*_{i_2}$이 차분 $(e_{3,14,15,24,25}, e_{5,27}, e_{9,18,29}, e_{31}, 0,0,0,0)$을 만족한다면, $T^*_{i_1}, T^*_{i_2}, T'^*_{i_1}, T'^*_{i_2}$와 추측한 부분 키 W^{32}를 저장한다. 만약 그렇지 않다면, 단계 5로 돌아간다. 만약 모든 추측한 W^{32}, W^{*32}에 대해서도 주어진 차분을 만족하지 않는다면, 단계 2로 돌아간다.

6) 두 개의 32 비트 부분 키 W^{31}, W^{*31}를 추측한다. 추측한 부분 키 W^{31}를 사용하여 31 라운드에 대한 모든 T'_{i_1}, T'_{i_2}의 부분 복호화를 수행하여, T''_{i_1}, T''_{i_2}에 저장한다. 또한 추측한 부분 키 W^{*31}를 사용하여 31 라운드에 대한 모든 $T'^*_{i_1}, T'^*_{i_2}$의 부분 복호화를 수행하여 $T''^*_{i_1}, T''^*_{i_2}$에 저장한다. 만약 모든 T''_{i_1}, T''_{i_2}와 $T''^*_{i_1}, T''^*_{i_2}$이 차분 $(e_{5,27}, e_{9,18,29}, e_{31}, 0,0,0,0,0)$을 만족한다면, 추측한 부분 키 W^{31}를 저장한다. 만약 그렇지 않다면, 단계 6으로 돌아간다. 만약 모든 추측한 W^{31}, W^{*31}에 대해서도 주어진 차분을 만족하지 않는다면, 단계 5 또는 2로 돌아간다. 즉, 만약 sk, sk^*에 대해 추측하지 않은 W^{32}, W^{*32}가 존재한다면, 단계 5로 아닌 경우에는 단계 2로 돌아간다.

7) 단계 6까지 통과한 부분 키 sk, W^{32}, W^{31}에 대해 나머지 320 비트 키에 대한 전수조사를 수행한다. 이 과정을 통과한다면, 512 비트 키 k'를 37-라운드 SHACAL-2의 마스터 키로 $k' \oplus (0,0,0,0,0,0,0,0,e_{31},0,0,0,0,0,0,0)$를 연관된 512 비트 마스터 키로 출력한다.

알고리즘 2의 데이터 복잡도는 $2^{233.16}$ 연관키 선택 평문을 요구하며, 공격에 사용되는 메모리는 암호문 쌍 저장 공간에 의존하므로 약 $2^{238.16} (=2^{233.16} \cdot 32)$ 바이트를 요구한다.

또한 알고리즘 2의 계산 복잡도는 다음과 같이 평가할 수 있다. 단계 1의 계산 복잡도는 $2^{233.16}$ 37-라운드 SHACAL-2 암호화 연산이다(데이터 수집 단계). 단계 3의 계산 복잡도는 평균 $2^{484.95}$ $(\approx 2^{233.16} \cdot 2^{256} \cdot \frac{1}{2} \cdot \frac{4}{37})$ 번의 37-라운드 SHACAL-2 암호화 연산이다. 단계 4에서 모든 가능한 각각의 $(T_{i_1}, T_{i_2}, T^*_{i_1}, T^*_{i_2})$의 δ 차분 성립 여부를 테스트 한다. 이는 해쉬 테이블을 이용하여 효과적으로 수행할 수 있다. 유사하게 해쉬 테이블은 단계 5,6의 나타난 차분 값을 평가하는데에도 효과적으로 사용할 수 있다. 단계 4에서 올바르지 않은 추측 키가 적어도 6개의 $(T_{i_1}, T_{i_2}, T^*_{i_1}, T^*_{i_2})$에 대해 δ 차분 성립 여부 테스트를 통과하는 확률은 약

$$2^{-59.22} (\approx \sum_{i=6}^{t} ({}_tC_i \cdot (2^{-256.2})^i \cdot (1-2^{-256.2})^{t-i}))$$

이다. 단, t는 $2^{232.15}$개의 평문 쌍으로부터 유도되는 모든 가능한 quartet의 개수를 표현하는 $2^{463.32}$이다. 따라서 단계 4를 통과하는 128 비트 부분 키 쌍의 기댓값은 평균 $2^{195.78} (\approx 2^{256} \cdot \frac{1}{2} \cdot 2^{-59.22})$이다.

통과하는 $2^{195.78}$개의 부분 키 쌍을 걸러내는 방법 중 하나로, 단계 5,6을 사용한다. 부가적으로 알고리즘 2는 단계 5,6을 수행함으로써 64 비트 부분 키를 얻을 수 있다. 올바르지 않은 키 쌍 W^{32}, W^{*32}이 단계 5를 만족할 확률은 기껏해야 $2^{-180} (=(2^{-15})^{12})$이다. 확률 2^{-15}는 표 6의 32~33 라운드에 표기된 1 라운드 차분 특성을 만족하기 위해 요구되는 수치이며, 단계 5에서 테스트 하는 quartet의 개수가 적어도 6이상이라는 점에서 계산된 확률이다. 따라서 단계 5에서 통과하는 160비트 부분 키 쌍 $((sk, W^{32}), (sk^*, W^{*32}))$의 기댓값은 $2^{79.78} (\approx 2^{195.78} \cdot 2^{64} \cdot 2^{-180})$이고, 단계 5의 계산 복잡도는 약 $2^{256.16} (\approx 2^{195.78} \cdot 2^{64} \cdot 12 \cdot \frac{1}{37})$번의 37-라운드 SHACAL- 2 암호화 연산이다. 유사하게 단계 6은 통과한 $2^{79.78}$ 부분 키 쌍을 걸러내기 위해 표 6의 31 ~ 32 라운드에 표기된 1 라운드 차분 특성을 이용한다. 1 라운드 차분 특성 확률은 2^{-12}이고, 단계 6에서 테스트 하는 quartet의 개수는 적어도 6 이상이기 때문에, 통과하는 192비트 부분 키 쌍 $((sk, W^{32}, W^{31}), (sk^*, W^{*32}, W^{31}))$의 기댓값은 $2^{-0.22} (\approx 2^{79.78} \cdot 2^{64} \cdot (2^{-12})^{12})$이고, 단계 6의 계산 복잡도는 약 $2^{142.16} (\approx 2^{79.78} \cdot 2^{64} \cdot 12 \cdot \frac{1}{37})$ 번의 37-라운드 SHACAL-2 암호

화 연산이다. 단계 7의 계산 복잡도는 2^{320} 37-라운드 SHACAL-2 암호화 과정을 요구하기 때문에, 알고리즘 2의 전체 계산 복잡도는 약 $2^{484.95}(\approx 2^{233.16}+2^{484.95}+2^{256.16}+2^{142.16}+2^{320})$ 번의 35-라운드 SHACAL-2 암호화 연산이다.

알고리즘 2는 앞서 설명한 확률 $(\widehat{p^*}\cdot\hat{q})^2(\approx (2^{-102.16})^2)$으로 만족하는 33-라운드 연관키 Rectangle 특성을 이용하였기 때문에, 올바른 quartet의 기대값은 약 $2^3(={}_{2^{232.16}}C_2\cdot 2^{-256}\cdot(2^{-102.16})^2)$이다. 올바른 부분 키 쌍이 단계 5,6을 만족하는 확률은 1이다. 그러므로 알고리즘 2의 성공 확률, 즉, 올바른 부분 키 쌍을 가지고 δ 차분 성립 여부 테스트를 통과하는 적어도 6개 이상의 quartet이 생성될 확률은 약 $0.80(\approx\sum_{i=6}^{t}(2^{-256}\cdot(2^{-102.16})^2)^i\cdot(1-2^{-256}\cdot(2^{-102.16})^2)^{t-i}))$이다. 단, t는 $2^{463.32}$이다.

Ⅵ. 결 론

본 논문은 향상된 연관키 차분-선형 공격을 소개하고, 이를 연관키 차분-비선형 공격으로 확장 적용하였다. 연관키 차분-비선형 공격을 이용하여 $2^{42.32}$ 연관키 선택 평문의 데이터 복잡도와 $2^{452.1}$의 계산 복잡도를 가지고 35-라운드 SHACAL-2를 공격하였다. 또한 본 논문에서는 연관키 Rectangle 공격을 이용하여 $2^{233.16}$ 연관키 선택 평문의 데이터 복잡도와 $2^{484.95}$의 계산 복잡도를 가지고 37-라운드 SHACAL-2를 공격하였다. 이는 SHACAL-2에 대한 가장 효과적인 분석 결과이다. 본 논문은 연관키 관련 공격을 이용하여 SHACAL-2의 분석 결과를 소개하였다.

본 고에서 알 수 있듯이 비교적 단순한 키 스케줄을 갖는 블록 암호의 경우는 연관키 공격 관점에서 안전성 평가가 이루어질 필요가 있다.

참 고 문 헌

〔1〕 E. Biham and A. Shamir, "Differential cryptanalysis of the full 16-round DES", *Advances in Cryptology - CRYPTO'92*, LNCS 740, pp. 487-496, Springer-Verlag, 1992.

〔2〕 E. Biham, "New Types of Cryptanalytic Attacks Using Related Keys", *Journal of Cryptology*, Vol. 7, No. 4, pp. 229-246, 1994.

〔3〕 E. Biham, O. Dunkelman and N. Keller, "Enhanced Differential-Linear Cryptanalysis," *Advances in Cryptology - ASIACRYPT'02*, LNCS 2501, pp. 254-266, Springer-Verlag, 2002.

〔4〕 E. Biham, O. Dunkelman and N. Keller, "Rectangle Attacks on 49-Round SHACAL-1," *FSE '03*, LNCS 2887, pp. 22-35, Springer-Verlag, 2003.

〔5〕 H. Handschuh and D. Naccache, "SHACAL : A Family of Block Ciphers," Submission to the NESSIE project, 2002.

〔6〕 P. Hawkes, "Differential-Linear Weak-Key Classes of IDEA," *Advances in Cryptology - EUROCRYPT'98*, LNCS 1403, pp. 112-126, Springer-Verlag, 1998.

〔7〕 홍석희, 김종성, 김구일, 이창훈, 성재철, 이상진, "30 라운드 SHACAL-2의 불능 차분 공격", *정보보호학회논문지*, 14(3), pp. 107-115, 2004.

〔8〕 김구일, 김종성, 홍석희, 이상진, 임종인, "축소 라운드 SHACAL-2의 차분-선형 유형 공격", *정보보호학회논문지*, 15(1), pp. 57-66, 2005.

〔9〕 김종성, 김구일, 홍석희, 이상진, "SHACAL-1의 축소 라운드에 대한 연관키 Rectangle 공격", *정보보호학회논문지*, 14(5), pp. 57-68, 2005..

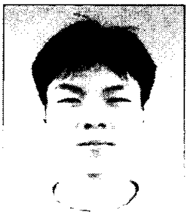
〔10〕 S. K. Langford and M.E. Hellman, "Differential-Linear Cryptanalysis," *Advances in Cryptology - CRYPTO'94*, LNCS 839, pp. 17-25, Springer-Verlag, 1994.

〔11〕 M. Matsui, "Linear Cryptanalysis Method for DES Cipher," *Advances in Cryptology - EUROCRYPT'93*, LNCS 765, pp. 386-397, Springer-Verlag, 1994.

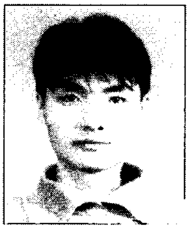
[12] A. A. Selcuk and A. Bicak, "On Probability of Success in Linear and Differential Cryptanalysis," *SCN'00*, LNCS 2576, pp. 174-185, Springer-Verlag, 2002.

[13] U.S. Department of Commerce. *FIPS 180-2*: Secure Hash Standard, Federal Information Processing Standards Publication, N.I.S.T., August 2002.

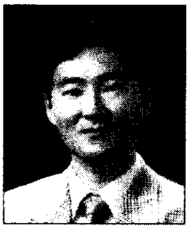
〈著者紹介〉



김 종 성 (Jong-Sung Kim)
2000년 8월: 고려대학교 수학과 학사
2002년 8월: 고려대학교 수학과 석사
2002년 8월~현재: 고려대학교 정보보호대학원 박사 과정
〈관심분야〉 블록 암호 및 스트림 암호의 분석과 설계



김 구 일 (Gu-il Kim)
2002년 2월 : 고려대학교 수학과 학사
2002년 9월 : 고려대학교 정보보호대학원 석사 과정
〈관심분야〉 블록 암호 및 스트림 암호의 분석과 설계



이 상 진 (Samgjin Lee)
1987년 2월: 고려대학교 수학과 학사
1989년 2월: 고려대학교 수학과 석사
1994년 2월: 고려대학교 수학과 박사
1989년 2월~1999년 2월: 한국전자통신연구원 선임 연구원,
1999년 2월~2001년 8월: 고려대학교 자연과학대학 조교수,
2001년 9월~현재: 고려대학교 정보보호대학원 부교수
〈관심분야〉 대칭키 암호의 분석 및 설계, 정보은닉이론, 컴퓨터 포렌식

임 종 인 (Jongin Lim)
1980년 2월: 고려대학교 수학과 학사
1982년 2월: 고려대학교 수학과 석사
1986년 2월: 고려대학교 수학과 박사
1999년 2월~현재: 고려대학교 정보보호대학원 원장, 고려대학교 정보보호기술연구센터 센터장
〈관심분야〉 정보보호이론, 정보보호정책