블록암호 ABCD에 대한 포화 공격*

이제상**, 이창훈**, 홍석희**, 성재철***, 이상진**
**고려대학교 정보보호대학원
***서울시립대학교

Saturation Attack against Full-Round ABCD

Jesang LEE**, Changhoon LEE**, Seokhie Hong**, Jaechul Sung*** and Snagjin LEE**
**Graduate School of Information Security Korea University
***University of Seoul

요 약

ABCD는 FGCS'2004에 이장두 등에 의하여 제안된 블록 암호이다. ABCD는 256비트의 평문을 입력받아 128비트의 키를 사용하여 256비트의 암호문을 출력하는 블록암호이다. ABCD는 지금까지 분석 결과가 알려져 있지 않으며, 본 논문에서는 7×2^8의 선택 평문을 이용하여, 공격복잡도 2^{54}을 갖는 전체 라운드 포화공격을 소개한다.

I. 서 론

ABCD[7]는 FGCS'2004에 이장두 등에 의하여 제안된 블록 암호로서 256 비트의 평문을 입력받아 128 비트의 키를 사용하여 256 비트의 암호문을 출력하는 블록암호이다. ABCD는 XOR와 S박스($S_1, S_2, \cdots, S_8$)와 같은 단순한 연산으로 구성된다.

지금까지 ABCD 알고리즘은 처음 제안할 때 제시된 공격 이외의 분석결과가 알려져 있지 않으며 본 논문에서는 ABCD에 대한 포화 공격을 소개한다.

본 논문의 구성은 다음과 같다. 2 절에서는 포화공격에 들어가기에 앞서 ABCD에 대한 간략한 알고리즘 설명과 포화공격에 대한 기본적인 개념 그리고 본 논문에서 사용할 표기법에 대하여 살펴 볼 것이다. 3 절에서는 7 라운드 포화 특성을 구성하고, 공격 복잡도 2^{54}을 갖는 전체 라운드 공격을 소개할 것이다. 마지막으로 4 절은 본 논문의 결론이다.

II. ABCD 알고리즘 소개

ABCD는 FGCS'2004에 이장두 등에 의하여 제안된 블록 암호로서 256 비트의 블록 메시지를 8 라운드를 거쳐 128 비트 키로 암호화하는 알고리즘이며 암호화 과정은 다음과 같다. 본 논문에서는 ABCD에 대한 포화공격시 키 스케줄은 사용되지 않으므로 키 스케줄은 소개하지 않는다.

2.1 ABCD 알고리즘

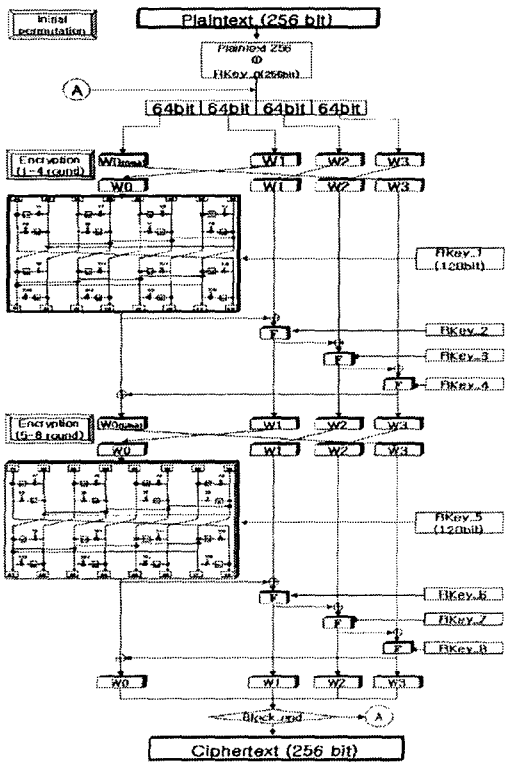
ABCD 알고리즘은 (그림1)과 같이 8 라운드 암호화 과정을 거치며, 라운드 함수 F 는 64 비트 입력을 받아 64 비트 출력을 내는 함수로서 (그림2)와 같다. 라운드 함수 F의 내부 함수 $S_1, S_2, \cdots, S_8$은 8비트 입력을 받아 8 비트를 출력하는 서로 다른 S 박스이다.

ABCD는 크게 초기 치환과정, 4 라운드 암호화 과정, 스왑, 4라운드 암호화 과정을 수행함으로 암호문을 출력한다.

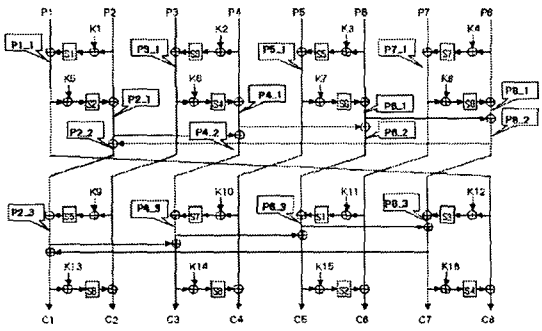
초기치환 과정은 256 비트 평문을 입력 받아 256 비트의 초기 화이트닝 키와 XOR한뒤 스왑과정을 수행한다.

암호화 과정은 64 비트 워드 단위로 수행되며, 라운드 키 128 비트를 이용하여 라운드 함수 F에 의하여 갱신된 워드는 이웃한 워드에 영향을 주도록 설계되었다. 4 개의 64 비트 워드가 모두 갱신되면 스왑과정을 거치며, 위와 동일한 과정에 의하여 남은 4 라운드 암호화 과정이 수행한뒤 암호문을 출력한다.

* 본 연구는 정보통신부 대학 IT연구센터 육성·지원사업의 연구결과로 수행되었습니다.



(그림 1) ABCD 전체 구조



(그림 2) ABCD 라운드 함수

2.3 표기법

본 사용될 표기법은 다음과 같다.

- P : 256 비트 평문
- Y : 256 비트 암호문
- C : 64 비트 고정된 상수 값
- $A'=(a,c,c,c,c,c,c,c)$: 8 비트 포화집합
- B' : $(b,?,b,?,a,a,a,b)$을 만족하는 64 비트 집합
- a : 8 비트 포화집합
- b : 8 비트 균일집합
- ? : 8 비트 임의의 집합
- Z^i : i 번째 라운드의 256 비트 입력 값
- Z_j : j 번째 워드의 64 비트 입력 값
- Z_j^i : i 라운드의 j 번째 워드의 입력 값
- W_j^i : 7 라운드의 i 번째 입력 값 중 j 번째 8 비트 값

Ⅲ. ABCD 포화공격

포화공격은 주어진 라운드 함수의 일대일 대응 성질을 이용하여 선택된 평문에 대하여 몇 라운드 후의 출력 모양이 포화 집합이 되거나 균일 집합이 되는 성질을 유도하여 올바른 키를 찾아내는 공격방법이다. 포화집합과 균일집합의 정의는 다음과 같다.

- 포화집합(A) : 집합 A을 n 비트로 이루어진 집합이라고 하자. 모든 n 비트 수열들이 집합 A에 정확하게 한 번씩 나타나면, 이때 M 을 포화집합 이라고 한다.
- 균일집합(B) : 집합 B을 n 비트 수열들로 구성되어 있다고 하자. 만약 B의 모든 원소들을 XOR한 값이 0이 된다면, 즉

$$\bigoplus_{x_i \in B} x_i = 0$$

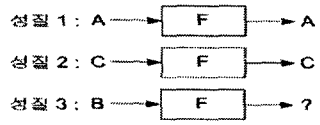
하면, 이 때 집합 B를 균일집합이라고 한다.

〔표 1〕 XOR 연산의 특성의 특성

XOR(⊕)	포화집합 (A)	상수 (C)	균일집합 (B)
포화집합 (A)	균일집합 (B)	포화집합 (A)	균일집합 (B)
상수 (C)	포화집합 (A)	상수 (C)	균일집합 (B)
균일집합 (B)	균일집합 (B)	균일집합 (B)	균일집합 (B)

어떤 집합 A가 포화집합이면 A는 균일집합이 된다는 사실은 정의로부터 쉽게 알 수 있다. 또한 포화집합과 균일집합에 대한 XOR 연산의 특성은 [표 1]과 같다.[10]

함수 F가 일대일 대응 함수라고 할 때 다음 성질을 만족한다.



[그림 3] 일대일 대응함수의 함수 성질

[성질 1, 2, 3]과 XOR 연산의 특성을 통하여 11 라운드 ABCD 알고리즘의 포화특성을 구성할 수 있다.

3.1 ABCD 포화특성

본 소절에서는 전체 라운드 포화공격에 사용 할 7 라운드 포화 특성을 구성할 것이다. 위에서 언급한 포화 성질을 이용하면, 다음과 같은 포화 특성을 간단하게 이끌어 낼 수 있다.

먼저 평문 집합으로 $P=(A',C,C,C)$를 선택한다. 여기서 C는 고정된 임의의 64 비트 상수 값이고, $A'=(a,c,c,c,c,c,c,c)$는 8 비트 포화집합이다.(단 a는 8 비트 포화집합이고, c는 8 비트 고정된 상수값이다.) ABCD의 라운드 함수 F 의 성질에 의하여, [표 2]와 같이, 선택 평문 집합 $P=(A',C,C,C)$ 에 대하여 7 라운드 입력 부분에서 포화특성 $Z^7=(C,C,B',B')$을 얻을 수 있다. 여기서 $B'=(b,?,b,?,a,a,a,b)$로서 a는 8 비트 포화집합이고, b는 8 비트 균일집합이며, ?는 8 비트 임의의 집합이다. 즉, 7 라운드 세 번째 입력 값 Z_3^7 에서 균일 성질

$$\bigoplus_{w_i \in W_1^3} w_i = 0$$

을 확률 1로 만족한다.

[표 2]ABCD 포화특성 테이블

	Z^1	Z^2	Z^3	Z^4
P	A'	C	C	C
Z^1	C	C	C	A'
Z^2	C	C	C	A'
Z^3	C	C	C	A'
Z^4	C	C	C	B'
Z^5	C	C	B'	B'
Z^6	C	C	B'	B'
Z^7	C	C	B'	B'

3.2 ABCD 전체 라운드 포화공격

본 소절에서는 앞 소절에서 구성한 7 라운드 포화 특성을 이용하여 ABCD 전체 라운드 포화공격을 소개한다. ABCD 전체 라운드 포화공격 시나리오는 다음과 같다.

[준비과정] 공격자는 $P=(A',C,C,C)$을 만족하는 선택평문 집합에 대하여 암호문 집합 $Y=(Y_1, Y_2, Y_3, Y_4)$을 얻는다.

[여과과정] 공격자는 키를 추측하여 암호문 집합을 7 라운드 세 번째 워드 입력 값중 첫 번째 8 비트 값을 복호화한 뒤 균일성질

$$\bigoplus_{w_i \in W_1^3} w_i = 0$$

만족여부를 판단한다. 포화 특성을 만족하면 추측한 키는 옳은 키 후보로 간주하고, 만족하지 않는 키는 옳지 않은 키로 간주하여 버린다.

[반복과정] 하나의 키가 남을 때까지 준비과정과 여과과정을 반복한다.

여과과정에서 7 라운드 세 번째 입력 값 중 첫 번째 8 비트 값까지 복호화하는 과정은 7 라운드 세 번째 입력 값 중 첫 번째 8 비트 값의 특성이 균일성질을 만족하는지의 여부를 판단하므로, 첫 번째 8 비트 값을 제외한 나머지 56 비트의 특성은 고려하지 않아도 된다. 따라서 복호화 할 경우 첫 번째 8 비트값에 영향을 주는 키워드만을 추측하면 된다. 알고리즘에 의하여 7 라운드 키중 추측해야 하는 부분 키는 다음과 같이 48 비트이다.

$$K=\{K_1, K_5, K_9, K_{12}, K_{13}, K_{16}\}$$

만을 추측하면 7 라운드 세 번째 입력 값 중 첫 번째 8 비트 값을 복호화 할 수 있음을 쉽게 확인 할 수 있다.

ABCD의 전체라운드 포화공격은 선택 평문 집합 $P=(A',C,C,C)$에 대하여, 옳지 않은 48 비트 부분키가 위의 포화특성을 만족할 확률은 2^{-8}이다. 따라서 부분키 공간의 크기가 2^{48}이라고 할 때, 확률적으로 옳은 키를 찾아내기 위해서는 적어도 7 개의 선택 평문 집합이 필요하다.

따라서 ABCD 전체 라운드 포화 공격 복잡도는 선택 평문 집합 7 개에 대하여 48 비트의 부분키를 여과과정(1 라운드 복호화 연산 필요)을 거쳐 찾아내므로 약

$$2^8 \times (2^{48}+2^{40}+2^{32}+2^{24}+2^{16}+2^8+2^0) \times \frac{1}{8} \approx 2^{54}$$

라 할 수 있다.

위와 같은 방법에 의하여 7 라운드에 쓰이는 모든 라운드 키 128 비트를 분할-정복에 의하여 찾을 수 있다.

Ⅲ. 결론

ABCD는 지금까지 알고리즘을 제안할 때 제시된 공격 이외의 분석결과가 알려져 있지 않았다. 본 논문에서는 7×2^8의 선택 평문을 이용하여, 공격 복잡도 2^{54}을 갖는 포화공격을 성공시켰다. 이를 통하여 ABCD 알고리즘의 확산효과가 좋지 않음을 알수 있다.

참 고 문 헌

[1] P. Barreto, V. Rijmen, J. Nakahara Jr., B. Preneel, J. Vandewalle, and H. Y. Kim. "Improved SQUARE attacks against reduced-round HIEROCRYPT", *FSE 2001*, LNCS 2355, Springer-Verlag 2002, pp. 165-173.

[2] Mark Blunden, Adrian Escott, "Related Key Attacks on Reduced Round KASUMI", *FSE 2001*, LNCS 2355, Springer-Verlag 2002, pp. 277-285.

[3] Y. He, S. Qing, "Square Attack on Reduced Camellia Cipher", *ICICS 2001*, LNCS 2229, Springer-Verlag 2002, pp. 89-99.

[4] Y. Hu, Y. Zhang, and G. Xiao, "Integral cryptanalysis of SAFER+", *IEE*, vol 35, no 17, 19. Aug. 1999, pp. 1458-1459.

[5] L.R. Knudsen, D. Wagner, "Integral Cryptanalysis", *FSE 2002*, LNCS 2365, Springer Verlag 2002, pp. 112-127.

[6] Ulrich Kuhn, "Crypanalysis of Reduced-Round MISTY", *EUROCRYPT 2002*, LNCS 2045, Springer-Verlag 2001, pp. 325-339.

[7] Chang-Doo Lee, Bong-Jun Choi, Kyoo-Seok Park, "Design and evalution of a block encryption algorithm using dynamic-key mechanism", FGCS 2004, pp. 327-338

[8] S. Lucks, "The Saturation Attack - a Bait for Twofish", *FSE 2001*, LNCS 2355, Springer-Verlag 2002, pp. 1-15.

[9] R. Scott, "Wide Open Encryption Design Offers Flexible Implementations," Cryptologia, v. 9, n. 1, Jan 1985, pp. 75-90.

[10] Yongjin Yeom, Sangwoo Park, Iljun Kim, "On the Security of CAMELLIA against the Square Attack", *FSE 2002*, LNCS 2365, Springer-Verlag 2002, pp. 89-99.