情報保護學會論文誌
第19卷 第3號, 2009. 6

축소된 SMS4 블록 암호에 대한 향상된 안전성 분석*

김 태 현,[1†] 김 종 성,[2‡] 홍 석 희,[2] 성 재 철,[3] 이 창 훈[4]
[1]LG 전자, [2]고려대학교 정보보호연구원, [3]서울시립대학교 수학과, [4]한신대학교 컴퓨터공학부

Improved Security Analysis of Reduced SMS4 Block Cipher*

Taehyun Kim,[1†] Jongsung Kim,[2‡] Seokhie Hong,[2]
Jaechul Sung,[3] Changhoon Lee[4]

[1]LG Electronics, [2]CIST, Korea University,
[3]Dept. of Mathematics, University of Seoul,
[4]School of Computer Engineering, Hanshin University

요 약

본 논문에서는 중국 무선 네트워크 표준 WAPI에 사용되고 있는 블록 암호 SMS4에 대한 향상된 차분 공격 및 선형 공격을 소개한다: 먼저, SMS4의 전체 32 라운드 중 20 또는 21 라운드에 적용된 기존 차분 공격을 22 라운드 차분 공격으로 확장하는 방법을 소개한다. 또한, 22 라운드로 축소된 SMS4에 적용된 기존 선형 공격의 복잡도, 2^{119} 기지 평문, 2^{109} 메모리 바이트, 2^{117}의 암호와 과정을 새로운 선형 근사식을 이용하여 2^{117} 기지 평문, 2^{109} 메모리 바이트, $2^{112.24}$의 암호와 과정으로 향상시킨다. 본 논문의 분석 결과는 SMS4에 대해 알려진 공격 중 최상의 공격이다.

ABSTRACT

In this paper, we introduce improved differential and linear attacks on the SMS4 block cipher which is used in the Chinese national standard WAPI (WLAN Authentication and Privacy Infrastructure, WLAN - Wireless Local Area Network): First, we introduce how to extend previously known differential attacks on SMS4 from 20 or 21 to 22 out of the full 32 rounds. Second, we improve a previously known linear attack on 22-round reduced SMS4 from 2^{119} known plaintexts, 2^{109} memory bytes, 2^{117}encryptions to 2^{117} known plaintexts, 2^{109} memory bytes, $2^{112.24}$ encryptions, by using a new linear approximation.
Keywords: Side-channel attacks, Meet-in-the-middle attacks, AES

I. 서 론

SMS4는 중국 무선 네트워크 표준 WAPI (WLAN Authentication and Privacy Infrastructure, WLAN-Wireless Local Area Network)에 사용되고 있는 32-라운드 128-비트 블록 암호이다. 현재까지 SMS4에 대한 분석 결과로는 20, 21-라운드 SMS4에 대한 차분 공격[1,2], 22-라운드 SMS4에 대한 선형 공격[3], 16-라운드 SMS4에 대한 불능 차분 공격[4,5], 14-라운드 SMS4에 대한 렉탱글 공격[4,5]과 13-라운드 SMS4에 대한 포화 공격[6]이 소개되었다.

본 논문에서는 블록 암호 SMS4에 대한 기존 최상의 공격인 차분 공격과 선형 공격을 향상시킨다. 먼저, 키 복구 라운드에 최적화된 filtering 기법을 이용하여, 기존의 20, 21 라운드 SMS4 차분 공격을 22 라운드 차분 공격으로 향상시킨다. 또한, 새로운 18-라운드 선형 근사식을 이용하여, 기존의 22 라운드 SMS4 선형 공격에 대한 복잡도를 향상시키는 방

접수일(2008년 10월 23일), 게재확정일(2009년 4월 27일)
* 이 연구에 참여한 연구자는 '2단계 BK21사업'의 지원비를 받았음.
† 주저자, kimth@lge.com
‡ 교신저자, joshep@cist.korea.ac.kr

〔표 1〕 SMS4에 대한 분석 결과 비교

공격 유형	라운드	데이터 복잡도	메모리 복잡도	시간 복잡도
선형 공격 〔3〕	22	2^{119}	2^{109}	2^{117}
차분 공격 〔2〕	21	2^{118}	2^{123}	$2^{126.6}$
차분 공격 〔1〕	20	2^{126}	2^{73}	$2^{105.85}$
불능 차분 공격 〔5〕	16	$2^{117.06}$	$2^{121.06}$	$2^{132.06}$
렉탱글 공격 〔5〕	14	$2^{107.89}$	$2^{111.89}$	$2^{107.89}$
포화 공격 〔6〕	13	2^{16}	2^{20}	2114
차분 공격 〔본 논문〕	22	2^{118}	2^{123}	$2^{125.71}$
선형 공격 〔본 논문〕	22	2^{117}	2^{109}	$2^{112.24}$

법을 소개한다. 〔표 1〕은 SMS4에 대한 기존의 분석 결과와 본 논문의 향상된 분석 결과를 나타낸 것이다.

본 논문의 구성은 다음과 같다. 2절에서는 본 논문에서 사용하는 표기법을 정리하고 SMS4에 대한 간략한 알고리즘을 설명한다. 3절에서는 SMS4에 대한 향상된 차분 공격을 소개하고, 4절에서는 SMS4에 대한 향상된 선형 공격을 소개한다. 마지막으로 5절은 본 논문의 결론으로 구성된다.

II. 표기법 및 SMS4

본 절에서는 본 논문에 전반적으로 사용되는 표기법을 정리하고, SMS4 블록 암호를 간략히 소개한다.

2.1 표기법

워드의 비트(바이트) 위치는 가장 오른쪽 최하위 비트(바이트)부터 0으로 시작하며, 왼쪽으로 갈수록 커진다.

- $\oplus$: 비트별 배타적 논리합.
- $\lll i$: i 비트만큼 왼쪽으로 순환이동.
- ? : 알 수 없는 값.
- $||$: 두 값을 연결하는 연접 연산.
- Sbox : SMS4의 8×8 S-박스.
- X·Y : 두 32-비트 워드 X, Y에 대한 비트별 내적 연산.
- $[X]_{[j]}$ $(j=0,1,2,3)$: 32-비트 워드 X의 j번째 최하위 바이트.

2.2 SMS4 블록 암호 소개

SMS4는 전체 32-라운드 불균형 Feistal network 구조이다. 블록 크기와 비밀키 크기는 128-비트로써 각각 평문과 키 128-비트를 입력받아 128-비트의 암호문을 출력한다. 평문을 4개의 32-비트 워드 $P=(P_0,P_1,P_2,P_3)$로 표기하고 $i(=1,2,\ldots,32)$번째 라운드 암호화 후의 중간값을 X^i로 표기한다. SMS4의 전체 32-라운드 암호화 과정은 다음과 같다.

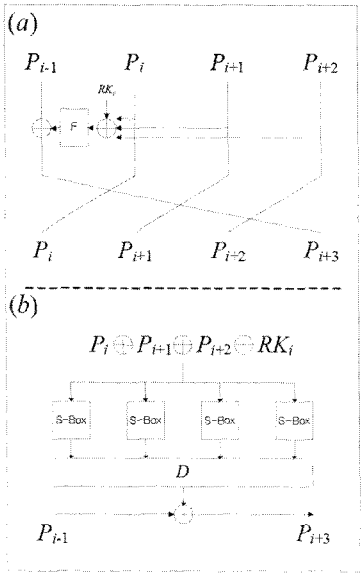
1. 128-비트 평문 $P=(P_0,P_1,P_2,P_3)$을 입력한다.
2. $i(=1,\ldots,32)$번째 라운드 암호화 과정은 다음과 같다.
   - $P_{i+3}=P_{i-1}\oplus D(S(P_i\oplus P_{i+1}\oplus P_{i+2}\oplus RK_i))$,
   - $X^i=(P_i,P_{i+1},P_{i+2},P_{i+3})$,
3. 암호문 $X^{32}=(P_{32},P_{33},P_{34},P_{35})$을 출력한다.

여기서 RK_i는 i 번째 라운드에 대한 32-비트 라운드 키이다. SMS4의 라운드 함수에는 두 가지 함수가 사용된다. 확산의 성질을 주는 선형 변환 함수 $D(x)=x\oplus(x\lll 2)\oplus(x\lll 10)\oplus(x\lll 18)\oplus(x\lll 24)$와 S 박스를 사용하는 비선형 함수 $S(x)$가 사용된다. SMS4의 S 박스는 동일한 네 개의 8×8 S 박스를 사용한다. 초기에 SMS4에 사용되는 S 박스의 설계원리가 공개되지 않았지만 Fen등에 의해 AES의 S 박스를 기반으로 설계됨이 밝혀졌다〔6〕. 네 개의 입력 바이트에 대해 함수 $S(x)$는 다음과 같이 정의된다.

$$\text{입력: } X=(x_0,x_1,x_2,x_3)\in(Z_2^8)^4,$$
$$\text{출력: } S(X)=(s(x_0),s(x_1),s(x_2),s(x_3)).$$

여기서 $s(x)$는 SMS4의 8×8 S 박스이다. 본 논문에서는 합성 함수 $D\circ S$를 비선형 함수 F로 표기한다.

〔그림 1〕은 SMS4의 라운드 함수와 F 함수를 나타낸다.



〔그림 1〕 (a) 라운드 함수, (b) F 함수

SMS4의 복호화 과정은 사용되는 라운드 키가 암호화 과정과 반대로 사용되는 것을 제외하고 암호화 과정과 동일하다. SMS4의 키 생성 과정은 본 논문의 공격 과정에 큰 영향을 미치지 않으므로, 생략한다.

III. SMS4의 22-라운드 차분 공격

3.1 SMS4의 18 라운드 차분 특성〔2〕

Zhang은 〔2〕에서 함수 F의 확률 2^{-21}을 갖는 차분 특성 $\alpha \rightarrow \alpha$에 대해 7905 (약 2^{13})개의 α값이 존재함을 밝혀냈다. 각각의 α의 값은 바이트 헤밍 웨이트 $HWb(\alpha)=3$을 만족하며 첫번째 바이트는 0으로 고정되어 있고 나머지 바이트에 대해서는 각각 확률 2^{-7}를 갖는다. F 함수의 동일한 입출력 차분값의 특성을 이용하여 확률 2^{-42}를 갖는 5-라운드 반복 차분 특성 $(\alpha,\alpha,\alpha,0)\rightarrow(\alpha,\alpha,\alpha,0)$을 구성할 수 있다. 또한, 이 5-라운드 반복 차분 특성을 이용하여 확률 2^{-126}을 갖는 18-라운드 차분 특성을 구성할 수 있다〔2〕. 공격에 사용되는 차분 특성의 정확한 차분값들과 확률은 〔표 2〕와 같다 (〔표 2〕의 Λ_α는 함수 F에 대한 입력 차분이 α일 때, 모든 가능한 출력차분 집합을 의미한다).

〔표 2〕 18-라운드 차분 특성과 키 복구에 사용될 마지막 4 라운드 차분 특성

라운드	입력차분	확률
1	$(\alpha,\alpha,\alpha,0)$	2^{-42}
6	$(\alpha,\alpha,\alpha,0)$	2^{-42}
11	$(\alpha,\alpha,\alpha,0)$	2^{-42}
16	$(\alpha,\alpha,\alpha,0)$	1
17	$(\alpha,\alpha,0,\alpha)$	1
18	$(\alpha,0,\alpha,\alpha)$	1
19	$(0,\alpha,\alpha,\alpha)$	1
20	$(\alpha,\alpha,\alpha,\Lambda_\alpha)$	1
21	$(\alpha,\alpha,\Lambda_\alpha,?)$	1
22	$(\alpha,\Lambda_\alpha,?,?)$	1
23	$(\Lambda_\alpha,?,?,?)$	

3.2 공격 과정

본 소절에서는 Zhang의 21-라운드 차분 공격〔2〕을 22-라운드 SMS4에 대한 키 복구 공격으로 확장시킨다. 본 논문의 분석 방법은 암호문의 8-비트 여과 과정 기법에 기반을 둔다. 즉, 키 추측 과정에서 전체 32-비트 라운드 키가 아닌 각각의 바이트 단위로서 키 추측을 수행하고 원래의 공격보다 암호문쌍을 효과적으로 여과한다. 본 22-라운드 차분 공격은 앞 소절에서 소개한 18-라운드 차분 특성을 1-18 라운드까지 적용한 후, 마지막 19, 20, 21, 22 라운드 부분키를 복구한다. 그 후, 최종적으로 남아있는 비밀키에 대해 전수 조사를 실행하여 전체 128-비트 비밀키를 복구한다. 본 공격에서 Diff는 앞 소절에서 찾아낸 2^{13}개의 α값에 대한 집합 $\{(\alpha,\alpha,\alpha,0)\}$를 표기한다. 본 22-라운드 차분 공격은 다음과 같다.

1. 2^{72}개의 평문으로 구성된 2^{46}개의 구조체를 생성한 후 대응하는 암호문을 선택 평문 공격을 이용하여 획득한다. 각 구조체의 평문들은 56 비트 (0, 1, 2, 3, 7, 11, 15 번째 바이트)가 고정되어 있고 나머지 72 비트는 모든 가능한 값을 갖는다. 따라서, 각 구조체에 대해서 차분값 (*,*,*,0)을 갖는 대략 $2^{143}(=(2^{72})^2/2)$개의 평문쌍을 얻을 수 있다 (단, *는 첫번째 바이트는 0이고 나머지 바이트는 임의의 값을 갖는다). 생성된 평문쌍에 대하여 각각의 차분값이 집합 Diff에 속하는지 확인하고 만족하지 못하는 평문쌍은 버린다. 이 여과 과정 후에 $2^{130}(=2^{46}\cdot 2^{143}\cdot(2^{13}/2^{72}))$의 평문쌍이 평균적으로 남게된다.

2. 여과 후에 남아있는 평문쌍 (P^i, P^j)에 대응하는

암호문쌍 (C^i, C^j)에 대해 최상위 워드의 차분값이 집합 Λ_α에 속하는지 확인한다. ($C^i = (C_0^i, C_1^i, C_2^i, C_3^i)$)으로 표기한다). S-박스 차분 분포표에 의해 Λ_α에 속하는 원소의 개수는 총 2^{21}이므로, 이 여과 과정 후에 $2^{119}(=2^{130}\cdot(2^{21}/2^{32}))$개의 암호문쌍이 남을 것으로 기대된다.

3. 22 라운드 키의 첫번째 1 바이트 $RK_{22,0}$를 추측한다. 남아있는 암호문 쌍 (C^i, C^j)에 대해, 1 라운드 F 함수의 첫 번째 S-박스 출력 차분값을 계산하여, 다음의 식을 체크한다.

$$Sbox([C_0^i \oplus C_1^i \oplus C_2^i]_{[0]} \oplus \mathrm{RK}_{22,0}) \oplus Sbox([C_0^j \oplus C_1^j \oplus C_2^j]_{[0]} \oplus \mathrm{RK}_{22,0}) = [\mathrm{D}^{-1}(\mathrm{C}_3^i \oplus \mathrm{C}_3^j)]_{[0]}.$$

여기서 $[X]_j$는 32-비트 워드 X의 j 번째 바이트를 나타낸다. $RK_{22,0}$에 대해 올바른 키를 추측했을 때, 계산된 암호문쌍이 옳은 암호문쌍이라면 위 식이 성립하게 된다. 따라서 위 식이 성립하지 않으면, 계산된 암호문쌍은 버린다. 옳은 암호문쌍이 아니거나, 추측한 키가 옳은 키가 아니라면 위 식이 성립할 확률은 랜덤한 확률 2^{-8}을 따르게 된다. 이 과정에서 $2^{111}(=2^{119}\cdot 2^{-8})$개의 암호문쌍이 남게 된다.

4. 남아있는 22, 21, 20 라운드의 각 바이트 단위의 부분키에 대해 단계 3의 과정을 순차적으로 적용한다. 마찬가지로 S-박스의 8-비트 출력 차분값을 비교하여 부분적으로 암호화된 암호문쌍을 여과할 수 있다. 이 단계까지 대략 $2^{23}(=2^{111}\cdot(2^{-8})^{11})$개의 암호문쌍이 남게 된다.

5. 19 라운드 키의 최하위 1 바이트 $RK_{19,0}$를 추측한다. 마찬가지로 S-박스의 출력 차분값을 비교하여 암호문쌍을 여과한다. 이 단계에서는 이미 공격의 시작 과정에서 19 라운드 F 함수의 출력 차분값들에 대해 Λ_α를 통한 여과 과정을 수행했기 때문에 대략 $2^{-7}(\approx 1/127)$의 확률로 암호문쌍이 여과된다. 남아있는 $RK_{19,1}, RK_{19,2}$에 대하여도 동일한 과정을 반복한다. 따라서 살아남는 암호문쌍의 개수는 $2^2(=2^{23}\cdot(2^{-7})^3)$로 기대할 수 있다. 반면, 19 라운드까지 올바른 키가 추측됐을 때 18-라운드 차분 특성을 따르는 올바른 암호문쌍의 개수는 $16(=2^{130}\cdot 2^{-126})$개로 기대할 수 있다. 즉, 마지막 단계까지 남아 있는 암호문쌍의 개수가 15개 이상인 키를 올바른 $RK_{19,0}, RK_{19,1}, RK_{19,2}, RK_{20}, RK_{21}, RK_{22}$에 대한 후보키로 출력한다.

6. 출력된 $RK_{19,0}, RK_{19,1}, RK_{19,2}, RK_{20}, RK_{21}, RK_{22}$값에 대해서 비밀키의 남아있는 8 비트를 전수 조사하여 복구한다.

본 차분 공격의 데이터 복잡도는 2^{118} 선택 평문이며, 테이블에 2^{118}개의 선택 평문에 대한 암호문을 저장해야 하므로 메모리 복잡도는 $2^{123}(=2^{118}\cdot 16\cdot 2)$ 바이트를 요구한다.

또한, 본 차분 공격의 시간 복잡도는 다음과 같이 계산할 수 있다. 단계 3에서의 시간 복잡도는 추측된 8 비트의 키 $RK_{22,0}$에 대해, 2^{119}개의 암호문쌍을 한 개의 S-박스를 통하여 부분적으로 암호화하므로 $2^{121.54}(\approx 2^{119}\cdot 2^8\cdot 2\cdot(1/22)\cdot(1/4))$ 22-라운드 SMS4의 암호화 연산이다. 각각의 8-비트 키 추측과정에서 여과되어 버리는 암호문쌍의 개수와 추가적으로 추측해야 되는 키 비트들의 수는 동일하므로 단계 4에서의 각각 시간 복잡도는 $2^{121.54}$ 암호화 연산으로 동일하다. 그러므로 단계 4까지의 시간 복잡도는 대략 $12\cdot 2^{121.54}$ 암호화 연산이다. 단계 5에서의 키 추측 과정에서는 2^{-7}의 확률로 암호문쌍이 여과되므로, 각각의 시간 복잡도는 $2^{122.54}$ 암호화 연산이다. 전수 조사과정의 시간 복잡도를 계산하기 위해서는 단계 5까지 남아있는 부분키의 개수를 계산해야 된다. 포아송 분포 $X \sim Poi(\lambda = 2^2), \Pr_X[X > 14] \approx 2^{-15.61}$에 의해서, 단계 5까지 살아남는 키의 개수는 대략 $2^{104.39}(=2^{120}\cdot 2^{-15.61})$로 계산할 수 있다. 이는 단계 6의 시간 복잡도가 대략 2112.39의 암호화 연산이다. 따라서, 본 차분 공격의 전체 시간 복잡도는 $2^{125.71}(\approx 12\cdot 2^{121.54} + 3\cdot 2^{122.54})$22-라운드 SMS4 암호화 연산이다. 본 차분 공격의 성공률은 포아송 분포 $X \sim Poi(\lambda = 2^4)$에 의해서 $\Pr(X > 14) = 0.6$이다.

IV. SMS4의 22-라운드 선형 공격

4.1 SMS4의 새로운 18 라운드 선형 근사식

차분 공격과 마찬가지로, SMS4의 전체 선형 근사식을 찾기 위해서는 SMS4의 비선형 함수인 S-박스

에 대한 선형 근사식을 연구할 필요가 있다. 컴퓨터 프로그램을 통하여 얻어진 S-박스의 선형 분포표에서 가장 높은 bias값은 2^{-4}이며 그 다음은 $2^{-4.19}$의 값을 갖는다. 차분 공격과 동일하게, SMS4의 불균형 Feistel network 구조로 인해서 비선형 함수 F에 동일한 입출력 비트 마스크를 적용하였을 때 SMS4의 선형 근사식을 효율적으로 구성할 수 있다. 따라서 본 논문에서는 F 함수의 모든 가능한 동일한 입출력 비트 마스크에 대한 선형 근사식을 조사했다. 본 선형 공격에서는 bias $2^{-10.38}$을 갖는 입출력 비트 마스크 $\Gamma_\alpha=[0,64,6f,fe]$를 이용한다. F 함수에 대한 동일한 입출력 마스크 Γ_α를 갖기 위해서는, 하위 세 개의 S-박스 입력값의 입력 비트 마스크 Γ_α값은 S-박스 출력값에 대한 비트 마스크 $\Gamma_\beta=[0,6d,13,3]$값을 가져야 한다. 이 때의 bias는 세 개의 능동 S-박스에 대해 각각 $2^{-4.19}, 2^{-4.19}, 2^{-4}$를 갖는다. S-박스를 통과한 후의 Γ_β의 값은 다시 D 함수를 통과하여 원래의 입력 비트 마스크 Γ_α값과 같게 된다. F 함수에 대한 본 선형 근사식은 5, 6, 10, 11, 15, 16 라운드의 F 함수에 적용된다. [그림 2]는 SMS4의 전체 18-라운드 선형 근사식을 나타내고 있으며 다음의 식들은 각 라운드에서의 선형 근사식을 표현한다.

$$\Gamma_\alpha\cdot(P_5\oplus P_6\oplus P_7\oplus RK_5)=\Gamma_\alpha\cdot F(P_5\oplus P_6\oplus P_7\oplus RK_5),$$
$$\Gamma_\alpha\cdot(P_6\oplus P_7\oplus P_8\oplus RK_6)=\Gamma_\alpha\cdot F(P_6\oplus P_7\oplus P_8\oplus RK_6),$$
$$\Gamma_\alpha\cdot(P_{10}\oplus P_{11}\oplus P_{12}\oplus RK_{10})=\Gamma_\alpha\cdot F(P_{10}\oplus P_{11}\oplus P_{12}\oplus RK_{10}),$$
$$\Gamma_\alpha\cdot(P_{11}\oplus P_{12}\oplus P_{13}\oplus RK_{11})=\Gamma_\alpha\cdot F(P_{11}\oplus P_{12}\oplus P_{13}\oplus RK_{11}),$$

$$\Gamma_\alpha\cdot(P_{15}\oplus P_{16}\oplus P_{17}\oplus RK_{15})=\Gamma_\alpha\cdot F(P_{15}\oplus P_{16}\oplus P_{17}\oplus RK_{15}),$$
$$\Gamma_\alpha\cdot(P_{16}\oplus P_{17}\oplus P_{18}\oplus RK_{16})=\Gamma_\alpha\cdot F(P_{16}\oplus P_{17}\oplus P_{18}\oplus RK_{16}).$$

여기서 각 라운드의 선형 근사식은 bias $2^{-10.38}$을 갖는다. $F(P_i\oplus P_{i+1}\oplus P_{i+2}\oplus RK_i)$는 $P_{i-1}\oplus P_{i+3}$과 동일하므로, 위의 선형 근사식은 다음과 같이 표현된다.

$$\Gamma_\alpha\cdot P_5\oplus\Gamma_\alpha\cdot P_6\oplus\Gamma_\alpha\cdot P_7\oplus\Gamma_\alpha\cdot RK_5=\Gamma_\alpha\cdot P_4\oplus\Gamma_\alpha\cdot P_8, \quad (4.1)$$
$$\Gamma_\alpha\cdot P_6\oplus\Gamma_\alpha\cdot P_7\oplus\Gamma_\alpha\cdot P_8\oplus\Gamma_\alpha\cdot RK_6=\Gamma_\alpha\cdot P_5\oplus\Gamma_\alpha\cdot P_9, \quad (4.2)$$
$$\Gamma_\alpha\cdot P_{10}\oplus\Gamma_\alpha\cdot P_{11}\oplus\Gamma_\alpha\cdot P_{12}\oplus\Gamma_\alpha\cdot RK_{10}=\Gamma_\alpha\cdot P_9\oplus\Gamma_\alpha\cdot P_{13}, \quad (4.3)$$
$$\Gamma_\alpha\cdot P_{11}\oplus\Gamma_\alpha\cdot P_{12}\oplus\Gamma_\alpha\cdot P_{13}\oplus\Gamma_\alpha\cdot RK_{11}=\Gamma_\alpha\cdot P_{10}\oplus\Gamma_\alpha\cdot P_{14}, \quad (4.4)$$
$$\Gamma_\alpha\cdot P_{15}\oplus\Gamma_\alpha\cdot P_{16}\oplus\Gamma_\alpha\cdot P_{17}\oplus\Gamma_\alpha\cdot RK_{15}=\Gamma_\alpha\cdot P_{14}\oplus\Gamma_\alpha\cdot P_{18}, \quad (4.5)$$
$$\Gamma_\alpha\cdot P_{16}\oplus\Gamma_\alpha\cdot P_{17}\oplus\Gamma_\alpha\cdot P_{18}\oplus\Gamma_\alpha\cdot RK_{16}=\Gamma_\alpha\cdot P_{15}\oplus\Gamma_\alpha\cdot P_{19}. \quad (4.6)$$

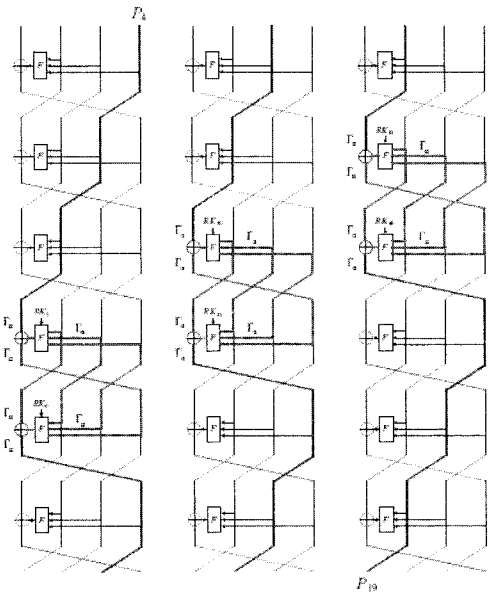
그러므로 식 (4.1),...,(4.6)을 XOR하여 2 라운드부터 19 라운드까지의 18-라운드 선형 근사식을 아래와 같이 얻을 수 있다 (주의: 본 선형 근사식은 첫번째 라운드가 아닌 두번째 라운드부터 적용된다).

$$\Gamma_\alpha\cdot P_4\oplus\Gamma_\alpha\cdot P_{19}=\Gamma_\alpha\cdot RK_5\oplus\Gamma_\alpha\cdot RK_6\oplus\Gamma_\alpha\cdot RK_{10}\oplus\Gamma_\alpha\cdot RK_{11}\oplus\Gamma_\alpha\cdot RK_{15}\oplus\Gamma_\alpha\cdot RK_{16}. \quad (4.7)$$

18-라운드 선형 근사식에서 P_4, P_{19}는 각각 1 라운드, 19 라운드 후의 중간값이다. 본 선형 근사식은 Pilling-up lemma에 의해서 bias $2^{-57.28}$을 갖는다. 선형 공격의 성공률은 필요로 하는 기지 평문의 개수 $c\cdot\epsilon^{-2}$에 의존한다. 본 선형 공격은 기지 평문 공격 시나리오에 따라 $2^{117}(\approx 8\cdot(2^{-57.28})^{-2})$개의 평문에 대해 대응하는 22-라운드 SMS4 암호문을 준비한다. [7]에서 소개된 Matsui의 성공 확률에 따라 본 선형 공격은 약 96%의 성공 확률을 갖는다.

4.2 공격 과정

본 선형 공격은 [7]에서 소개된 Matsui의 Algorithm 2를 적용하고 [8]에서 소개된 기법을 이용한다. 일반적인 Matsui의 Algorithm 2는 전체 r 라운드의 마지막 r′ 라운드가 선형 근사식에 포함된 키 비트들을 추측하여 복호화된 후, (r-r′)-라운드 선형 근사식이 적용된다. 전체 평문-암호문쌍에 대하여 선형 근사식이 만족하는 개수를 카운트하여 카운트 값이 가장 많이 절반에서 치우친 부분키를 올바른 키 후보



[그림 2] SMS4의 18-라운드 선형 근사식

로 출력한다. 본 22-라운드 선형 공격은 첫 1-라운드, 마지막 3-라운드 부분키를 추측한 후, 2 라운드 부터 19 라운드까지의 18-라운드 선형 근사식이 적용된다. 최종적으로 1, 20, 21, 22 라운드 부분키를 복구한 후, 남아있는 비밀키를 전수 조사를 통하여 복구한다.

Matsui는 [9]에서 선형 공격의 시간 복잡도를 상당히 줄일 수 있는 개선된 Algorithm 2를 제안했다. 제안된 공격의 시간 복잡도는 추측되는 부분키가 관여하는 능동 S-박스의 개수에 의존한다. 개선된 Algorithm 2가 본 선형 공격에 적용된다. 본 22-라운드 선형 공격에서 1, 20 라운드의 능동 S-박스의 개수는 각각 3개씩이며, 21, 22 라운드의 능동 S-박스의 개수는 각각 4개씩이다. 따라서 전체 2^{112}개의 부분키에 대해서 선형 근사식의 bias를 계산할 필요가 있다. 본 선형 공격에서는 모든 암호문이 아닌, 능동 S-박스에 대응하는 암호문 비트들에 대해서 bias를 계산한다.

먼저, 선형 근사식 (4.7)을 평문과 22-라운드 SMS4 암호문 그리고 1, 20, 21, 22 라운드 부분키에 대한 선형 근사식으로 확장시킨다. 확장된 선형 근사식은 다음과 같다.

$$\begin{aligned}\Gamma_\alpha\cdot \mathrm{P}_0 \oplus \Gamma_\alpha\cdot \mathrm{C}_1 \oplus \Gamma_\alpha\cdot \mathrm{F}(\mathrm{P}_1\oplus \mathrm{P}_2\oplus \mathrm{P}_3\oplus \mathrm{RK}_1)\\ \oplus \Gamma_\alpha\cdot \mathrm{F}(\mathrm{C}_0\oplus \mathrm{C}_2\oplus \mathrm{C}_3\oplus \mathrm{RK}_{20}\\ \oplus F(C_0\oplus \mathrm{C}_1\oplus \mathrm{C}_2\oplus \mathrm{RK}_{22})\\ \oplus \mathrm{F}(\mathrm{C}_0\oplus \mathrm{C}_1\oplus \mathrm{C}_3\oplus \mathrm{RK}_{21}\\ \oplus F(C_0\oplus \mathrm{C}_1\oplus \mathrm{C}_2\oplus \mathrm{RK}_{22})))\\ = K(RK).\end{aligned} \tag{4.8}$$

여기서 K(RK)는 각 라운드의 부분키로 구성된 식 (4.7)의 우변이다. 위의 식을 아래의 방정식으로 다시 표현한다.

$$g(\mathrm{P},\mathrm{C})\oplus \mathrm{f}(\mathrm{RK},\mathrm{C}) = \mathrm{K}(\mathrm{RK}). \tag{4.9}$$

g(P,C)는 평문과 암호문으로 구성된 선형 근사식에 대한 방정식(식 (4.8)의 첫번째, 두번째 항)이며 f(RK,C)는 5개의 F 함수로 구성된 방정식이다(식 (4.8) 좌변의 나머지 항). 1, 19 라운드 F 함수의 선형 근사식에서, Γ_α 는 D^{-1}을 통하여 $\Gamma_\beta = [0,6d,13,3]$이 되므로 F 함수 출력값의 모든 32-비트 값을 계산할 필요는 없다. 따라서, $(P_1\oplus \mathrm{P}_2\oplus \mathrm{P}_3), (\mathrm{C}_0\oplus \mathrm{C}_2\oplus \mathrm{C}_3)$의 각각 하위 24 비트가 Γ_α의 비트 마스크 계산에 포함되고 $(C_0\oplus \mathrm{C}_1\oplus \mathrm{C}_2), (\mathrm{C}_0\oplus \mathrm{C}_1\oplus \mathrm{C}_3)$의 각각 32 비트가 Γ_α의 비트 마스크 계산에 포함되므로 f의 값은 평문, 암호문과 라운드 키 각각 112 비트에 의해 결정된다. 각 평문, 암호문과 라운드 키에 대해서, γ를 $(RK_1, RK_{20}, RK_{21}, RK_{22})$의 112 비트와 δ를 $(P_1\oplus \mathrm{P}_2\oplus \mathrm{P}_3, C_0\oplus \mathrm{C}_2\oplus \mathrm{C}_3, C_0\oplus \mathrm{C}_1\oplus \mathrm{C}_2, \mathrm{C}_0\oplus \mathrm{C}_1\oplus \mathrm{C}_3)$의 112 비트로 표기한다. 또한 $Z = Z_3|Z_2|Z_1|Z_0$를 $\gamma\oplus\delta$의 112 비트로 표기한다. 여기서 $Z_0 = RK_{22}\oplus \mathrm{C}_0\oplus \mathrm{C}_1\oplus \mathrm{C}_2, \mathrm{Z}_1 = \mathrm{RK}_{21}\oplus \mathrm{C}_0\oplus \mathrm{C}_1\oplus \mathrm{C}_3, \mathrm{Z}_2 = RK_{20}\oplus \mathrm{C}_0\oplus \mathrm{C}_2\oplus \mathrm{C}_3, \mathrm{Z}_3 = \mathrm{RK}_1\oplus \mathrm{P}_1\oplus \mathrm{P}_2\oplus \mathrm{P}_3$와 같다. 결국, f(RK,C)는 다음의 방정식으로 표현된다.

$$\begin{aligned}f(RK, C) = \Gamma_\alpha\cdot \mathrm{F}(\mathrm{Z}_3)\oplus\\ \Gamma_\alpha\cdot \mathrm{F}(\mathrm{Z}_2\oplus \mathrm{F}(\mathrm{Z}_0)\oplus \mathrm{F}(\mathrm{Z}_1\oplus \mathrm{F}(\mathrm{Z}_0))).\end{aligned} \tag{4.10}$$

2^{112}개의 모든 키 후보들에 대해서 2^{117}개의 기지 평문-암호문쌍과 식 (4.10)을 이용하여 22-라운드 선형 근사식 (4.9)에 대한 bias를 계산할 수 있다. SMS4에 대한 4-라운드 공격(4R-Attack)은 다음과 같다.

1. f에 사용된 δ의 112 비트에 대응하는 2^{112}개의 원소를 가진 벡터 X를 초기화한다.

2. 각각의 γ, δ에 대하여, f(RK,C)의 parity를 계산한다(f의 값이 0이면 1, 1이라면 -1로 계산한다). 이 값들을 $2^{112}\times 2^{112}$ 행렬 M 에 저장한다 (δ는 행, γ는 열을 나타낸다).
   즉, $M[\gamma][\delta] = f(\gamma,\delta)$와 같다.

3. 2^{117}개의 평문-암호문쌍들에 대해 g(P,C)의 parity를 계산한다. 만약 parity가 1이라면 X의 대응하는 카운터를 증가시키고 0이라면 감소시킨다.

4. $M\cdot X$의 행렬-벡터 곱셈을 통하여, 2^{112}개의 키 후보(γ의 값)에 대한 bias를 계산한다.

5. 계산된 모든 bias ϵ에 대해서, $\epsilon_{\max}$와 $\epsilon_{\min}$를 각각 최대값, 최소값이라고 표기한다.
   - $|\epsilon_{\max}| > |\epsilon_{\min}|$라면 $\epsilon_{\max}$에 대응하는 키를 올바른 키로 추측하고 K(RK)=0로 추측한다.
   - $|\epsilon_{\max}| < |\epsilon_{\min}|$라면 $\epsilon_{\min}$에 대응하는 키를 올바른 키로 추측하고 K(RK)=1로 추측한다.

6. $RK_{20}, RK_{21}, RK_{22}, RK_1, K(RK)$의 113-비트가 추측되면 비밀키의 나머지 15 비트를 전수조사를

통하여 복구한다.

본 공격에서는 2^{112} 개의 원소를 갖고 있는 벡터 X를 저장하기 위해 2^{112}-비트 (2^{108}-바이트) 메모리가 요구되며, 각각의 단계에서 요구되는 공격 복잡도는 다음과 같다. 단계 2에서 행렬 M의 circulant 구조 [3]로 인해서 M은 $2^{112}\times 2^{112}$의 연산보다 효율적으로 계산이 가능하다. 2^{112}개의 Z에 대해서, $f(\gamma,\delta)$의 값을 계산하고, 이 값을 M의 대응하는 위치에 저장한다. f는 5개의 F 함수로 구성되어있기 때문에, 이 과정에서 $2^{109.86}(\approx 2^{112}\cdot(1/22)\cdot 5)$ 22-라운드 SMS4 암호화 연산이 요구된다. 또한, Z의 값을 저장하여 행렬 M의 각각의 원소값을 계산할 수 있으므로, Z의 값을 저장하기 위한 2^{112}-비트 메모리가 요구된다. 단계 3에서는 비트 추출과 카운트 증가와 같은 간단한 연산만이 사용되며, 이 과정에서 2^{117} 대수적 연산이 요구된다. 단계 4에서 $2^{112}\times 2^{112}$ 크기의 행렬-벡터 곱셈을 수행하므로 2^{224} 대수적 연산이 요구된다. 그러나 행렬 M의 circulant 구조로 인해서 행렬-벡터 곱셈의 시간 복잡도를 상당히 줄일 수 있다. B. Collard는 [8]에서 Fast Fourier Transform을 이용하여 이러한 방법을 처음으로 제시했다. 제안된 방법은 특수한 성질을 갖는 정방 행렬을 Discrete Fourier Transform matrix을 포함하는 세 개의 행렬로 대각화 시킨다. Discrete Fourier Transform matrix의 $m\times m$ 크기의 행렬 곱셈은 시간 복잡도를 $m\cdot\log(m)$으로 줄여준다. 따라서 단계 4에서의 $2^{112}\times 2^{112}$ 크기의 행렬-벡터 곱셈에 대한 시간 복잡도는 대략 $2^{120.39}(\approx 3\cdot 112\cdot 2^{112})$ 대수적 연산이다. SMS의 한 라운드는 대략 16 대수적 연산의 수행 시간이 걸리므로, 단계 4는 $2^{111.93}(\approx 2^{120.39}\cdot(1/16)\cdot(1/22))$ 22-라운드 암호화 연산이 요구된다. 또한, 단계 5의 비밀키의 나머지 키를 찾기 위한 시간 복잡도는 $2^{15}(=2^{128-113})$ 22-라운드 암호화 연산이다.

결론적으로 본 22-라운드 선형 공격에 대한 시간 복잡도는 $2^{112.24}(\approx 2^{109.8}+2^{111.93})$ 22-라운드 암호화 연산, 데이터 복잡도는 2^{117} 기지 평문과 2^{109}-바이트 메모리 복잡도를 요구한다.

V. 결 론

본 논문에서는 중국 표준 암호 알고리즘으로 사용되고 있는 SMS4에 대한 향상된 차분 공격 및 선형 공격을 소개했다. 기존의 20, 21-라운드 SMS4 차분 공격을 키 복구 부분에서 좀 더 세밀한 divide and conqure 기법을 적용하여 22-라운드 SMS4 차분 공격으로 향상시켰다. 또한, 새로운 18-라운드 선형 근사식을 이용하여, 기존의 22-라운드 SMS4 선형 공격에 대한 데이터 복잡도를 4배, 시간 복잡도를 약 32배 향상시켰다. 본 논문의 분석 결과는 SMS4에 대해 알려진 공격 중 최상의 공격이다.

참 고 문 헌

[1] 김태현, 김종성, 성재철, 홍석희, "축소된 20-라운드 SMS4에 대한 차분 공격," 정보보호학회논문지, 18(4), pp. 37-44, 2008년 8월.

[2] L. Zhang, W. Zhang, and W. Wu, "Cryptanalysis of Reduced-Round SMS4 Block cipher," ACISP'08, LNCS 5107, pp. 216-229, 2008.

[3] J. Etrog and M.J.B. Robshaw, "The Cryptanalysis of Reduced-Round SMS4," SAC'08, To appear.

[4] J. Lu, "Attacking Reduced-Round Versions of the SMS4 Block Cipher in the Chines WAPI Standard," ICICS'07, LNCS 4861, pp. 306-318, 2007.

[5] D. Toz and O. Dunkelman, "Analysis of two Attacks on Reduced-Round Versions of the SMS4," ICICS'08, LNCS 5308, pp. 141-156, 2007.

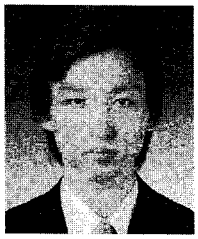
[6] F. Liu, W. Ji, L. Hu, J. Ding, S. Lv, A. Pyshkin, and R.P. Weinmann, "Analysis of the SMS4 block cipher," ACISP'07, LNCS 4586, pp. 85-100, 2007.

[7] M. Matsui, "Linear Cryptanalysis Method for DES Cipher," EUROCRYPT 1993, LNCS 765, pp. 386-397, 1994.

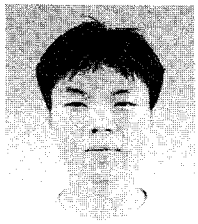
[8] B. Collard, F.X. Standaert, and J.J. Quisquater, "Improving the Time Complexity of Matsui's Linear Cryptanalysis," ICISC'07, LNCS 4817, pp. 77-88, 2007.

[9] M. Matsui, "The First Experimental Cryptanalysis of the Data Encryption Standard," CRYPTO'94, LNCS 839, pp. 1-11, 1994.

〈著者紹介〉

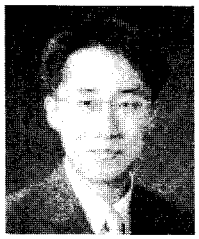


김 태 현 (Taehyun Kim) 학생회원
2005년 2월: 고려대학교 수학과 학사
2008년 8월: 고려대학교 정보경영공학전문대학원 석사
2008년 8월~현재: LG 전자
〈관심분야〉 블록암호, 스트림 암호 및 해쉬 함수의 분석 및 설계

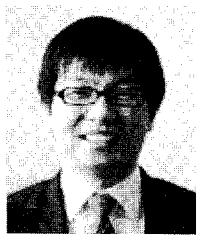


김 종 성 (Jongsung Kim) 정회원
2000년 8월: 고려대학교 수학과 학사
2002년 8월: 고려대학교 수학과 석사
2006년 11월: K.U.Leuven, ESAT/SCD-COSIC 박사
2007년 2월: 고려대학교 정보보호대학원 박사
2007년 3월~현재: 고려대학교 정보경영공학전문대학원 연구교수
〈관심분야〉 대칭키 암호의 분석/설계, 멀티미디어/유비쿼터스 정보보호, 포렌식, 부채널 공격

홍 석 희 (Seokhie Hong) 종신회원
1995년 2월: 고려대학교 수학과 학사
1997년 2월: 고려대학교 수학과 석사
2001년 2월: 고려대학교 수학과 박사
1999년 8월~2004년 2월: (주) 시큐리티 테크놀로지스 선임연구원
2003년 2월~2004년 2월: 고려대학교 정보보호기술연구센터 선임연구원
2004년 4월~2005년 2월: K.U.Leuven, ESAT/SCD-COSIC 박사후연구원
2005년 3월~2008년 8월: 고려대학교 정보보호대학원 조교수
2008년 9월~현재: 고려대학교 정보경영공학전문대학원 부교수
〈관심분야〉 대칭키 암호의 분석 및 설계, 컴퓨터 포렌식



성 재 철 (Jaechul Sung) 종신회원
1997년 8월: 고려대학교 수학과 학사
1999년 8월: 고려대학교 수학과 석사
2002년 8월: 고려대학교 수학과 박사
2002년 8월~2004년 1월: 한국정보보호진흥원 선임연구원
2004년 2월~현재: 서울시립대학교 수학과 조교수
〈관심분야〉 암호 알고리즘 설계 및 분석



이 창 훈 (Changhoon Lee) 정회원
2001년 2월: 한양대학교 수학과 학사
2003년 2월: 고려대학교 정보보호대학원 석사
2008년 2월: 고려대학교 정보경영공학전문대학원 박사
2008년 4월~2009년 2월: 고려대학교 정보경영공학전문대학원 연구교수
2009년 3월~현재: 한신대학교 컴퓨터공학부 전임강사
〈관심분야〉 정보보호, 암호 분석 및 설계, 멀티미디어 보안, 게임