Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.25, NO.6, Dec. 2015 ISSN 2288-2715(Online)
http://dx.doi.org/10.13089/JKIISC.2015.25.6.1319

해밍 웨이트 누출 기반 ARIA 키 확장 SPA*

박 애 선,[1†] 한 동 국,[1] 최 준[2‡]
[1]국민대학교, [2]국방보안연구소

A Simple Power Analysis Attack on ARIA Key Expansion Based on Hamming Weight Leakage*

Aesun Park,[1†] Dong-Guk Han,[1] Jun Choi[2‡]
[1]Kookmin University, [2]Defense Security Institute

요 약

AES, ARIA와 같은 대칭키 암호 알고리즘은 각 라운드 마다 사용되는 라운드 키를 키 확장 메커니즘을 통해 생성한다. 이러한 키 확장 메커니즘이 실행 될 때 소비되는 전력은, 비밀키에 대한 정보를 보유하고 있다는 점에 기인하여, 소수의 전력 파형을 이용한 단순전력분석으로 비밀키의 후보를 현저하게 감소시킬 수 있는 취약점이 존재한다. 그러므로, 이러한 공격에 대한 연구·분석을 통해, 정보 누출을 막을 수 있는 대응방법의 연구가 시급한 실정이다. 국제 표준 암호인 AES의 키 확장 SPA에 대한 연구는 2002년 이후 현재까지 진행되고 있으나, 국내에서 많은 분야에 응용되고 있는 국내 표준 암호 알고리즘 ARIA에 대한 키 확장 SPA 연구의 진행은 미흡하다. 이에 본 논문에서는, 마스킹이 적용되어 있지 않은 ARIA-128 버전 8비트 구현시 키 확장 SPA 공격 시나리오를 제안하고, 실험을 통해 ARIA가 해밍 웨이트 누출 기반 키 확장 SPA 공격에 취약함을 보인다.

ABSTRACT

The symmetric key encryption algorithms, such as the AES or the ARIA, generate round keys by the key expansion mechanism. While the algorithm is executed, key expansion mechanism emits information about the secret key by the power consumption. The vulnerability exists that can reduce significantly the candidate of the secret key by the simple power analysis attack using a small number of the power traces. Therefore, we'll have to study about the attack and the countermeasure to prevent information leakage. While a simple power analysis attack on the AES key expansion has been studied since 2002, ARIA is insufficient. This paper presents a simple power analysis attack on 8-bit implementations of the ARIA-128 key expansion. The presented attack efficiently utilizes this information leakage to substantially reduce the key space that needs to be considered in a brute-force search for the secret key. We show that ARIA is vulnerable to a SPA attack based on hamming weight leakage.

Keywords: Side-Channel Analysis, Simple Power Analysis, ARIA key expansion

I. 서 론

현대 사회에서 암호화 기술은 인터넷 뱅킹과 전자상거래, 차세대 이동전화, 스마트 카드 등 휴대형 정보 단말기의 보안 및 컴퓨터 파일 시스템의 기밀성 및 인증 기능 확보 등 매우 다양한 분야에서 활용되

Received(07. 20. 2015), Modified(09. 10. 2015), Accepted(09. 10. 2015)
* 이 논문은 2015년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2013R1A1A2A10062137)
† 주저자, aesons@kookmin.ac.kr
‡ 교신저자, choijun1014@hotmail.com(Corresponding author)

고 있다. 최근 국내에서는 금융 IC 카드, 행정기관 IC 카드 등을 대상으로 암호화 기술을 적용한 스마트카드로 대체하고 있는 추세이다.

일반적으로, 암호화 기술을 적용한 디바이스의 경우 안전성이 검증된 암호 알고리즘을 이용한다. 하지만, 이론적으로 안전한 암호 알고리즘이라도 소프트웨어 또는 하드웨어 형태로 구현되어 디바이스에 적용 할 경우, 암호 알고리즘 실행과정에서 전자파, 소비 전력 및 알고리즘 수행 시간 등 부가적인 정보가 발생하게 된다.

이에 따라, 부가적인 정보는 암호 알고리즘의 비밀키 등 비밀 정보들을 노출 시키는 취약점으로 작용할 수 있으며, 이와 같이 부가적인 정보의 노출을 이용하여 비밀 정보를 찾아내는 분석 방법을 부채널 분석(side channel analysis)이라 한다.

부채널 분석 방법 중 1999년 Kocher에 의해 제안된 전력분석(power analysis) 공격은, 디바이스 내에서 암호 알고리즘이 실행되는 동안의 소비 전력을 분석하여 비밀 정보를 얻어 내는 효율적인 공격 방법으로서, 많은 연구자들에 의해 전력분석에 대한 대응방법이 고려되지 않은 채 암호 알고리즘을 구현한 암호 디바이스의 경우에는 취약성이 존재한다는 것이 입증되었다[1].

전력분석은 단지 몇 번의 알고리즘이 실행되는 동안 발생하는 전력 신호를 관찰하여 비밀키 값을 유도해 내는 공격인 단순전력분석(SPA, Simple Power Analysis)과, 수차례 알고리즘이 반복 실행되는 동안 수집된 다수의 전력 신호들을 두 집단으로 나누어, 두 집단의 차분을 이용하여 분석하는 차분전력분석(DPA, Differential Power Analysis)이 존재한다. 이러한 전력분석 공격은 비교적 저비용으로 쉽게 구현 할 수 있기 때문에, 암호 알고리즘이 디바이스에 구현 될 때 적절한 대응방법이 필수적으로 요구된다.

전력분석 공격으로부터 디바이스를 보호하기 위한 방법으로, 암호 알고리즘이 동작하는 동안 발생하는 중간 결과를 랜덤화 시키는 대응방법이 많이 사용된다. 이러한 랜덤화 과정을 마스킹(masking)이라 부르며, 마스킹 기법은 현재 많은 알고리즘에 사용되고 있다. 특히 AES, ARIA 등 대칭키 암호 알고리즘의 마스킹 대응 방법은 국내외에서 많이 제안되었다[2, 3].

그러나 이러한 대응방법은 대부분 키 확장 단계의 보호는 제공하지 못하고 있다. 1999년 Biham과 Shamir에 의해 키 확장 단계를 포함하고 있는 암호 알고리즘의 전력분석 공격 가능성이 제시된 이후[7], 2002년 Mangard는 키 확장 단계에서 AES에 대해 SPA 공격을 제안하였다[4]. 2005년에는 이를 향상시킨 방법을 VanLaven 등이 제안하였으며[5], 최근에는 키 확장 단계에도 마스킹을 적용한 AES를 대상으로 하는 키 확장 SPA 공격이 2014년 CHES에서 Clavier 등에 의해 발표되었을 뿐만 아니라[6], 경량 암호에 대한 키 확장 공격 방법도 제안되었다[8].

또한, 최근 암호 모듈의 비침입 공격(전력분석 등)에 대한 저항성 테스트 시, 대칭키 시스템 키 확장 단계에서의 SPA 저항성 체크를 기본적으로 요구하고 있다.

이와 같이, 국제 표준으로 사용하고 있는 AES에 대해서는 키 확장 SPA에 대한 연구가 많이 진행되고 있는 반면, 국내 표준 암호 알고리즘인 ARIA에 대한 키 확장 SPA의 연구가 미흡한 상태이다. 더욱이 키 확장 부분의 마스킹 적용 방법에 대한 연구도 미흡한 실정으로서, 국내 표준 ARIA를 대상으로 키 확장 SPA에 대한 공격 방법과 대응방법의 연구가 필요할 것이다.

이에 본 논문에서는 마스킹이 적용되지 않은 8비트 구현 ARIA를 대상으로 키 확장 부분이 SPA 공격에 취약함을 논하고, 해밍 웨이트 누출로 인한 ARIA 키 확장 SPA 공격 방법에 대해 서술한다.

본 논문은 다음과 같이 구성된다. 2장에서 키 확장 SPA와 AES-128의 키 확장 SPA에 대해 언급한 후, 3장에서 해밍 웨이트 누출 기반 ARIA-128 키 확장 SPA의 공격 시나리오를 제안한다. 또한, 4장에서 실험 결과를 제시한 후, 마지막으로 5장에서 결론을 맺는다.

II. 선행 연구

2.1 키 확장 SPA

SPA는 암호 알고리즘이 동작할 때 소비되는 소수의 소비 전력을 측정함으로써, 암호 알고리즘에 사용되는 비밀 정보를 직접 공격하는 방법이다. 디바이스에서 암호 알고리즘이 동작할 때, 프로세서의 명령에 따라 각기 다른 전력을 갖는다는 사실을 외부에서 관측할 수 있으므로, 공격자는 이로부터 키 또는 순간 동작 중인 명령에 대한 정보를 추론 할 수 있다.

SPA를 이용한 공격은 공격자가 암호 알고리즘에 대해서 자세히 알고 있어야 한다. 즉, 알고리즘의 연산과정에서 소비되는 전력을 얻었을 때 어느 부분에서 어떠한 연산을 하는지, 또는 어떤 명령어들을 사용하는지를 판단할 수 있는 경우에, 이를 이용하여 암호 알고리즘에 대한 SPA 공격을 수행할 수 있다.

AES, ARIA와 같은 대칭키 암호 알고리즘의 1차 DPA 공격은 아래의 4가지 요구사항을 기반으로 이루어진다[4].

1. 공격자는 동일키와 랜덤 데이터를 이용하여 암호화(복호화)가 이루어지는 동안 디바이스의 소비 전력을 측정 할 수 있다.
2. 공격자는 소비 전력에 대응하는 평문(암호문)을 알 수 있다.
3. 암호화(복호화)가 이루어지는 동안 발생하는 중간 결과는 키 비트와 평문(암호문)의 관계로 이루어지는 함수이다.
4. 중간 결과 과정의 소비 전력은 두 개의 다른 값에 대하여 다르게 나타나야 한다.

실제 1, 2번 요구사항은 대부분의 암호 알고리즘이 동작하는 디바이스에서 수행 가능하다. 또한 마스킹 AES 등 요구사항 3, 4번에 대해 내성을 제공할 수 있는 여러 가지 대응 방법이 알려져 있다.

대칭키 암호 알고리즘의 키 확장 SPA 공격에 대한 요구사항은 다음과 같다.

1. 공격자는 키 확장 전체 과정에 대해, 적어도 한번 수행한 소비 전력을 측정할 수 있다.
2. 공격자는 수집 된 소비 전력 파형 일부에 키 확장 중에 발생하는 각각의 중간 결과를 매칭할 수 있다.
3. 수집 된 소비 전력 파형으로부터 디바이스의 전력 소비 정보를 획득 할 수 있다.
4. 적어도 한 쌍의 (평문, 암호문) 쌍을 알 수 있다.

키 확장 SPA의 첫 번째 요구사항은 1차 DPA 공격의 첫 번째 요구사항과 비슷하다. 사실 이는, 디바이스의 입력 데이터가 중요하지 않기 때문에 1차 DPA 보다 더 쉽게 접근 가능하다.

키 확장 SPA 공격은 키 후보를 줄이는 역할을 수행하기 때문에 무차별 대입 공격(brute force attack)을 고려해야 한다. 그러므로 남아 있는 키 후보들에 대한 효과적인 무차별 대입 공격을 수행하기 위해 하나의 암호문에 대응하는 평문이 필요하다.

2.2 AES 키 확장 SPA

2002년 Mangard[4]는 AES 키 확장이 이뤄지는 동안 누출되는 8비트 해밍 웨이트 정보를 이용하여 라운드 키를 찾아내는 방법을 제안하였다. 본 절에서는 Mangard가 제안한 AES 암호 알고리즘 키 확장 SPA 공격 방법 및 결과에 대해 간단히 언급한다.

2.2.1 AES-128 키 확장

AES-128를 계산하기 위해 128비트 비밀키는 11개의 128비트 라운드 키로 확장된다. Fig.1.은 AES-128 키 확장을 나타내는 의사코드이며, 11개의 라운드 키는 32 비트 워드 단위로 키 워드 배열 $W[0\ldots43]$으로 저장된다.

```
RC[1..10] = ('01','02','04','08','10','20','40','80','1B','36')
Rcon[i]   = (RC[i],'00','00','00')

for(i = 0; i < 4; i++)
   W[i] = (key[0], key[1], key[2], key[3])

for(i = 4; i < 44; i++)
{
   temp = W[i - 1]
   if (i mod 4 == 0)
       temp = SubWord(RotWord(temp)) xor Rcon[i / 4]
   W[i] = W[i - 4] xor temp
}
```

Fig. 1. Pseudo code for the AES-128 key expansion

$W_{i,j}$: i번째 키 워드의 $(j \bmod 4)$번째 바이트
$RotWord$: 1 바이트 왼쪽 로테이트

2.2.2 AES 키 확장 SPA 공격 방법

Mangard에 의해 제안된 공격은 Fig.2.와 같이, 찾기를 원하는 AES 라운드 키를 4개의 중복된 부분으로 나누는 것을 기반으로 시행된다. 라운드 키의 한 개 파트(5 바이트)는, 다른 라운드 키와 다양한 중간 값의 많은 바이트들을 계산할 수 있다(자세한 내용은 참고논문[4]의 Table 1.을 참고). Fig.2.는 3라운드 키 128비트를 나타낸 것으로, 기호의 의미는 아래와 같다.

$Key_3 = W_{12} \| W_{13} \| W_{14} \| W_{15}$
$Part_i = \{W_{12,i}, W_{13,i}, W_{14,i}, W_{15,i}, W_{15,(i+1)\bmod 4}\}$
$0 \le i \le 3$
W_i: 32 비트워드 단위

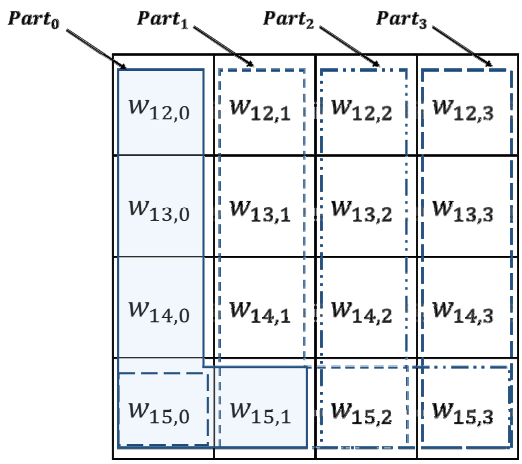


Fig. 2. Key_3 is split into four overlapping parts

t : 공격하려는 라운드 키의 처음 워드

Mangard의 공격 방법은 라운드 키를 찾기 위해 해당 라운드와 연관 되어 찾을 수 있는 총 81 개의 바이트를 주목한다. 예를 들어, 3라운드 키를 찾기 위해 $Key_3 = W_{12} \| W_{13} \| W_{14} \| W_{15}$ 중 $Part_0$와 관련된 아래의 바이트를 이용하여 $Part_0$에 해당하는 라운드 키를 찾은 후, 비슷한 방법을 이용해 나머지 $Part_i$에 대한 키를 찾는다.

$$W_{3,0}, W_{6,0}, W_{7,0}, W_{9,0}, W_{10,0}, W_{11,0},$$
$$W_{16,0}, W_{17,0}, W_{18,0}, W_{19,0}, SBox(W_{3,0}),$$
$$SBox(W_{7,0}), SBox(W_{11,0}), SBox(W_{15,0}),$$
$$SBox(W_{15,1}), SBox(W_{19,0}), SBox(W_{15,1}) \oplus Rcon$$

2.2.3 공격 결과

위의 공격 방법을 이용한 공격 결과를 다음의 3가지로 나타내었다. 첫 번째 81개의 바이트를 모두 알 수 있는 경우, 라운드 상수와 XOR 한 결과를 알 수 없는 경우(76 바이트), 마지막으로 $SBox$의 입·출력만을 알 수 있는 경우(40 바이트)를 실험한 결과이다.

Table 1.은 3가지 경우에 대한 결과를 나타낸 것으로서, 81 바이트와 76 바이트를 사용한 경우에 대해서는 1000번을 실험한 결과이고, 40바이트를 사용한 실험은 100번을 실험한 결과이다.

Table 1. Experimental results for AES

Number of used HW	Number of keys that remain for brute-force search	
81	mean	11.18
	max	280.0
	min	1.0
76	mean	16.41
	max	1152.0
	min	1.0
40	mean	1.68×10^{12}
	max	81.71×10^{12}
	min	11.22×10^{6}

III. ARIA 키 확장 SPA

3.1 ARIA-128 키 확장 알고리즘

ARIA 알고리즘은 암호화와 복호화를 수행하는 라운드 함수 및 키 확장 부분으로 구성되어 있다. ARIA 라운드 함수의 기본 구조는 Involution SPN 구조이며, 입·출력의 크기는 128비트이고 키의 크기는 128/192/256비트 중에서 선택할 수 있다.

ARIA 알고리즘은 라운드 키 덧셈 부분과 치환계층 그리고 확산 계층으로 구성된다. 이후, 본 논문에서는 치환계층을 SubstLayer 기호로 사용한다.

특히 치환계층은 두 가지의 S박스 S_1, S_2를 사용한 두 가지 유형의 치환계층을 사용한다. 홀수 라운드는 치환 계층으로 (LS, LS, LS, LS)를 사용하고, 짝수 라운드는는 치환 계층으로 $(LS^{-1}, LS^{-1}, LS^{-1}, LS^{-1})$를 사용한다. 여기에서 LS는 $(S_1, S_2, S_1^{-1}, S_1^{-1})$을 나타낸다.

특히, ARIA의 키 확장은 초기화 과정과 라운드 키 생성 과정으로 나뉘며, Fig.3.은 초기화 과정을 나타내고 있다. Fig.3.에서 KL은 키의 상위 128비트 값이고, KR은 키의 나머지 부분에 0으로 패딩한 128비트 값이다. CK_1, CK_2, CK_3의 세 개의 고정된 상수 값은 각 라운드함수(F_o, F_e, F_o)의 키 역할을 수행하고 이로 인해 얻은 W_0, W_1, W_2, W_3을 통해 각 라운드에 사용 될 라운드 키를 생성한다. F_o, F_e는 각 홀수 라운드와 짝수 라운드에서 사용되는 함수로 치환계층 부분을 제외한 나머지 부분은 동일하다.

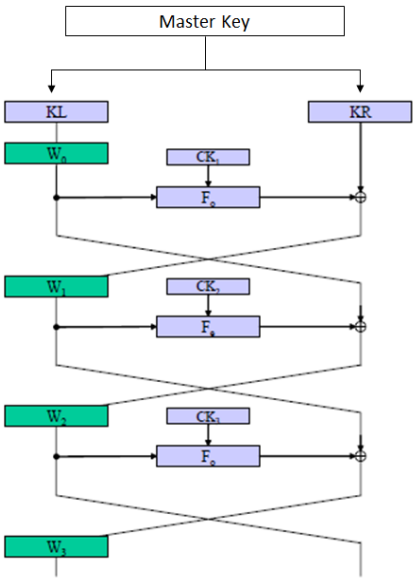


Fig. 3. ARIA-128 key expansion

3.2 해밍 웨이트 기반 SPA 공격 시나리오

본 논문에서는 ARIA-128의 키 확장 알고리즘을 공격하는 방법으로, 키 초기화 과정의 8비트 소프트웨어 구현에 대한 SPA 공격을 제안한다. 본 장에서는 해밍 웨이트 누출을 이용한 ARIA 키 확장 SPA의 실제 공격 시나리오를 서술한다.

제안하는 ARIA 키 확장 SPA 공격자 가정은 다음과 같다. 2장의 SPA 공격 요구사항에서 언급했듯이, 공격자는 대칭키 암호 시스템에서 발생하는 중간 값의 해밍 웨이트를 확인 할 수 있어야 하며, ARIA 키 확장 부분이 8비트 구현으로 이루어졌다는 가정에서 출발한다. 키 확장 내의 중간 값들의 종속성을 사용하여 비밀키의 후보들을 줄이고, 마지막으로 알려진 한 쌍의 평문과 암호문을 이용하여 비밀키를 결정한다.

Table 2.는 한 바이트의 해밍 웨이트에 따른 가능한 키 후보의 개수를 정리해 놓은 표이다. Table 2.에서 볼 수 있듯이 1바이트의 해밍 웨이트는 총 9가지(0~8)가 존재하며, 각 해밍 웨이트의 경우에 따라 한 바이트는 최대 70(약 $2^{6.13}$)개 키 후보를 갖는다.

또한 한 바이트가 갖는 해밍 웨이트의 평균은

$$E(HW)=\sum_{r=0}^{8} r\ {}_8C_r\left(\frac{1}{2}\right)^8=4$$

이므로, 만약 16바이트의 W_0 해밍 웨이트 값만을 정확하게 알 수 있다면, 비밀키 후보는 평균 $(2^{6.13})^{16}\approx 2^{98}$개가 남는다. 이는 현실적으로 비밀키를 찾기 힘든 시간이다.

Table 2. Key candidates according to the number of HW(Hamming Weight)

HW	0	1	2
Key candidates	$1(2^0)$	$8(2^3)$	$28(2^{4.81})$
HW	3	4	5
Key candidates	$56(2^{5.81})$	$70(2^{6.13})$	$56(2^{5.81})$
HW	6	7	8
Key candidates	$28(2^{4.81})$	$8(2^3)$	$1(2^0)$

그러므로 본 논문에서는 키 후보를 줄이기 위해 W_0와 관련 있는 다른 지점의 해밍 웨이트를 이용한다.

ARIA-128 키 확장 SPA를 수행하기 위해 공격자는 키 초기화 과정에서 W_0, W_1, $W_0\oplus CK_1$, $SubstLayer(W_0\oplus CK_1)$의 해밍 웨이트를 확인 할 수 있다고 가정한다. 즉, 키 초기화 과정 중 첫 번째 F_0의 입력 값 및 중간 계산 값과 출력 값 64 바이트의 해밍 웨이트를 이용하여 비밀키를 복구한다.

Fig.4.는 ARIA 키 확장 SPA를 위해 초기화 과정 중에서 타겟이 되는 공격 지점을 나타낸다.

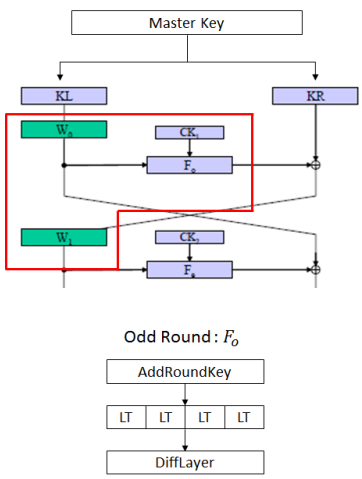


Fig. 4. Attack point

ARIA 키 확장 SPA는 다음과 같은 방법으로 비밀키(W_0)의 후보를 줄인다.

Step1. W_0의 해밍 웨이트를 이용하여 W_0의 가능한 후보를 추측한다.

Step2. 추측된 W_0의 후보들을 이용하여 바이트 단위로 $W_0 \oplus CK1$의 해밍 웨이트를 계산 후 전력 소비 관찰을 통해 얻은 $W_0 \oplus CK1$의 해밍 웨이트와 비교한다. 이 과정을 통해 가능한 W_0의 후보들을 줄일 수 있다.

Step3. 줄여진 W_0의 후보들을 이용하여 $SubstLayer(W_0 \oplus CK_1)$ 계산 후 알고 있는 $SubstLayer(W_0 \oplus CK_1)$의 해밍 웨이트와 비교한다.

Step4. Step1부터 Step3까지의 해밍 웨이트 비교를 통해 최종 구해진 W_0의 후보를 이용하여 확산 계층 출력 값을 계산 후 하고 계산된 결과 값과 W_1의 해밍 웨이트를 비교한다.

Step2와 Step3에서 계산되어지는 $W_0 \oplus CK1$와 $SubstLayer(W_0 \oplus CK_1)$의 해밍 웨이트 비교는 한 바이트씩 비교가 가능 하다. 그러나 W_1은 확산 계층을 통과하기 때문에 바이트 비교가 불가능하다. 앞쪽의 해밍 웨이트 비교를 통해 최종 구해진 W_0의 후보를 이용하여 16바이트 확산 계층 출력 값을 계산하고, 계산된 결과 값과 W_1의 해밍 웨이트를 비교해 비밀키 후보를 줄인다.

IV. 실험 결과

본 장에서는 논문에서 제안한 ARIA 키 확장 SPA 공격의 실험 결과를 보여준다. 실험은 Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz 프로세서 및 메모리 16GB가 장착된 일반 PC에서 시행 되었다.

첫 번째 실험은 3.2절에서 설명하고 있는 방법을 이용한 실험이다. 즉, 사용하는 바이트는 W_0 , W_1 , $W_0 \oplus CK_1$, $SubstLayer(W_0 \oplus CK_1)$의 64바이트를 모두 사용한다. 두 번째 실험은 W_1의 해밍 웨이트 값을 얻지 못했을 때의 경우로 48바이트를 사용한 실험이다. 모든 실험은 총 500번 시행하였다.

Table 3. Experimental results 1 for ARIA

Number of used HW	Number of keys that remain for brute-force search	
64	mean	1.38×10^4 ($2^{13.75}$)
	max	2.33×10^6 (2^{21})
	min	1
48	mean	4.42×10^5 ($2^{36.23}$)
	max	8.04×10^{10} ($2^{43.97}$)
	min	1.72×10^{13} ($2^{18.75}$)

Table 3.은 두 가지 실험에 대한 실험 결과로 남아있는 비밀키 후보의 수를 비교 한 것이다. 우선 첫 번째 실험은 임의의 키에 대한 64바이트의 해밍 웨이트 정보를 정확히 알고 있는 경우에 대하여 500번 실험한 결과이다. 남아 있는 후보 키의 개수는 최대 2.33×10^6(약 2^{21})개이고, 정확히 비밀키를 찾은 경우(남은 키 후보 1개)는 500번 중 53번 발생하였으며, 평균 1.38×10^4(약 $2^{13.75}$)개가 비밀키의 후보로 남았다.

W_0, $W_0 \oplus CK_1$, $SubstLayer(W_0 \oplus CK_1)$의 48바이트 해밍 웨이트를 이용하여 남아있는 키 후보수를 측정한 결과는 평균 4.42×10^5(약 $2^{36.23}$)로 64바이트의 해밍 웨이트를 이용한 결과보다 평균 $2^{22.48}$배 많은 후보 키가 남아 있음을 알 수 있다.

Table 4.는 64바이트를 사용한 500번의 실험의 결과를 나타낸 표이다. 왼쪽은 키 후보의 개수를 1

Table 4. Experimental results 2 for ARIA(64Bytes)

Range of key candidates	Number of keys that included in the range
0~10,000	449
10,000~20,000	14
20,000~30,000	7
30,000~40,000	8
40,000~50,000	3
50,000~60,000	2
60,000~70,000	1
70,000~80,000	3
80,000~90,000	0
90,000~100,000	1
100,000~	12

만개 단위로 나눈 것으로 $a \sim b$의 표시는 무차별 대입 공격을 위해 남아있는 키의 수가 a이상 b미만이라는 표시이다. 또한 오른쪽은 좌측의 범위에 들어가는 횟수를 나타낸 것이다. 즉, 후보 키가 10,000개 미만으로 나온 횟수가 총 500번의 실험 중 449번이 나왔다는 것을 나타낸다. 이는 높은 확률로 키 후보를 줄 일 수 있음을 의미한다.

V. 결 론

본 논문에서는 국내 표준으로 사용되고 있는 대칭키 암호 알고리즘 ARIA-128이 8비트 기반으로 구현되었을 때, 키 초기화 과정 64바이트 해밍 웨이트에 관심을 갖고, 키 확장 SPA 공격을 통해 비밀키의 후보를 현저하게 감소시킴으로써, 무차별 대입 공격을 용의하게 수행 할 수 있음을 보였다.

ARIA-128 키 확장의 키 초기화 과정은 확산 계층을 포함한다. 라운드 키 생성 부분은 키 초기화 과정을 거쳐 생성된 값을 이용하기 때문에 라운드 키 생성 부분에서 얻어오는 해밍 웨이트 정보는 제안하는 키 확장 SPA 공격보다 공격성능이 월등히 좋아지지 않음을 알 수 있다.

본 논문의 결과는 전력 분석 공격의 대응방법을 고려할 때, 키 확장 SPA에 안전한 대응방법도 같이 연구되어야 함을 시사하고 있으며, 국내에서도 키 확장 SPA에 대한 위험성을 인지하고, 이에 대한 대응방법 연구가 이루어져야 할 것이다.

또한, 향후 유추한 해밍 웨이트의 오류가 있는 경우에 대한 효과적인 분석 방법에 대한 연구가 필요할 것이다.

References

[1] P.Kocher, J.Jaffe and B.Jun, "Differential power analysis," Advances in Cryptology-CRYPTO'99, LNCS 1666, pp. 388-397, Jan. 1999

[2] K. Itoh, M. Takenaka, and N. Torii, "DPA Countermeasure Based on the Masking Method," Information Security and Cryptology - ICISC 2001, LNCS 2288, pp. 440 - 456, Dec. 2001

[3] Junki Kang, Dooho Choi, Yong-Je Choi, and Dong-Guk Han, "Secure hardware implementation of ARIA based on adaptive random masking technique," ETRI Journal, 34(1), pp. 76-86, Feb. 2012

[4] S. Mangard, "A Simple Power-Analysis (SPA) Attack on Implementations of the AES Key Expansion," Information Security and Cryptology-ICISC 2002, LNCS 2587, pp. 343 - 358, Nov. 2001

[5] J. VanLaven, M. Brehob and K. J. Compton, "A Computationally Feasible SPA Attack on AES via Optimized search," IFIP TC11 20th International Information Security Conference, IFIP ICT 181, pp. 577 - 588, Jun. 2005

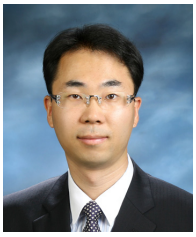
[6] C. Clavier, D. Marion and A. Wurcker, "Simple Power Analysis on AES Key Expansion Revisited," Cryptographic Hardware and Embedded Systems - CHES 2014, LNCS 8731, pp. 279-297, Sep. 2014

[7] E. Biham and A. Shamir, "Power Analysis of the Key Scheduling of the AES Candidates," Proceedings of second AES Candidate Conference, pp. 115-121, Mar. 1999

[8] V. Banciu, E. Oswald and C. Whitnall, "Exploring the Resilience of Some Lightweight Ciphers Against Profiled Single Trace Attacks," Constructive Side-Channel Analysis and Secure Design - COSADA 2015, Revised Selected Papers. Springer, pp. 51-63, Apr. 2015

〈저자소개〉

박 애 선 (Aesun Park) 학생회원
2011년 2월: 국민대학교 수학과 졸업 (학사)
2013년 2월: 국민대학교 수학과 석사 (이학석사)
2014년 3월~현재: 국민대학교 금융정보보안학과 박사과정
〈관심분야〉 부채널 분석 및 대응법, 신호처리, 스마트 카드 평가 등



한 동 국 (Dong-Guk Han) 종신회원
1999년 2월: 고려대학교 수학과 졸업(학사)
2002년 2월: 고려대학교 수학과 석사 (이학석사)
2005년 2월: 고려대학교 정보보호대학원 박사 (공학박사)
2004년 4월 ~2005년: 4월 일본 Kyushu Univ., 방문연구원
2005년 4월~2006년: 4월 일본 Future Univ.-Hakodate, Post.Doc.
2006년 6월~2009년: 2월 한국전자통신연구원 정보보호연구단 선임연구원
2009년 3월~현재: 국민대학교 수학과 부교수
관심분야: 공개키 암호시스템 안전성 분석 및 고속 구현, 부채널 분석 및 대응법 설계, IoT 정보보호 기술

최 준 (Jun Choi) 정회원
2001년 2월: 경희대학교 이학부(수학) 이학사
2003년 2월: 고려대학교 정보보호대학원 공학석사
2008년 2월: 고려대학교 정보보호대학원 공학박사
2014년 11월~현재: 국방보안연구소 선임연구원
〈관심분야〉 암호알고리즘, 암호키관리, 정보보호프로토콜