논문 2012-49-9-8

마스킹-셔플링 부채널 대응법을 해독하는 실용적인 편중전력분석

(Practical Biasing Power Analysis breaking Side Channel Attack Countermeasures based on Masking-Shuffling techniques)

조 종 원*, 한 동 국**

(Jong-Won Cho and Dong-Guk Han)

요 약

지금까지 부채널 분석은 스마트카드, 전자여권, e-ID 카드와 같은 Chip 기반의 보안 디바이스의 키를 해독하는 데 효과적임이 알려져 왔다. 이에 대한 실용적인 대응법으로 마스킹기법과 셔플링 기법을 혼용한 방법들이 제안되었다. 최근 S.Tillich는 마스킹과 셔플링 기법이 적용된 AES를 Template Attack(TA)을 이용한 biased-mask 공격기법으로 분석하였다. 하지만, S.Tillich 분석 기법을 적용하기 위해서는 사전에 masking 값에 대한 template 정보를 수집하여야 한다는 가정이 필요하다. 뿐만 아니라 분석 대상이 되는 masking 값의 시간 위치를 정확하게 알고 있어야 분석 성공 확률이 높아진다. 본 논문에서는 masking 값에 대한 시간 위치 정보와 이에 대한 template 정보를 활용하지 않고도 마스킹-셔플링 기반한 AES 대응법을 해독하는 새로운 편중전력분석 (Biasing Power Analysis, BPA)를 제안한다. 실제로 MSP430칩에서 구동되는 마스킹-셔플링 기반의 AES 대응법의 파형으로부터 BPA 공격을 통해 비밀키 128비트를 해독하는 실험을 성공하였다. 본 연구의 결과는 차세대 ID 카드 등에 활용될 스마트 칩에 대한 물리적 안전성 검증에 효율적으로 활용될 것으로 사료된다.

Abstract

Until now, Side Channel Attack has been known to be effective to crack decrypt key such as smart cards, electronic passports and e-ID card based on Chip. Combination of Masking and shuffling methods have been proposed practical countermeasure. Newly, S.Tillich suggests biased-mask using template attack(TA) to attack AES with masking and shuffling. However, an additional assumption that is acquired template information previously for masking value is necessary in order to apply this method. Moreover, this method needs to know exact time position of the target masking value for higher probability of success. In this paper, we suggest new practical method called Biasing Power Analysis(BPA) to find a secret key of AES based on masking-shuffling method. In BPA, we don't use time position and template information from masking value. Actually, we do experimental works of BPA attack to 128bit secret key of AES based on masking-shuffling method performed MSP430 Chip and we succeed in finding whole secret key. The results of this study will be utilized for next-generation ID cards to verify physical safety.

Keywords : Side channel Attack, maskning, shuffling, Biasing Power Analysis, AES

Ⅰ. 서 론

부채널 분석은 암호 알고리즘이 구현된 시스템의 물리적인 정보인 암호 연산의 시간, 소비전력 및 전자파와 같은 부채널 정보를 이용하는 공격 기법으로 스마트카드, 전자여권, e-ID카드와 같은 Chip 기반의 보안 디

* 정회원, ** 정회원-교신저자, 국민대학교 수학과
(Department of Mathematics, Kookmin University)
※ 본 연구는 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2012-0007285)
접수일자: 2012년3월19일, 수정완료일: 2012년9월19일

바이스의 키를 해독하는 데 효과적임이 알려져 있다. 알려진 부채널 분석 기법은 소비전력을 분석하는 방법으로 단순전력분석(Simple Power Analysis, SPA), 차분전력분석(Differential Power Analysis, DPA), 상관전력분석(Correlation Power Analysis, CPA) 등으로 발전되었으며, 부채널 분석에서 가장 강력한 방법들 중 하나이다[3, 6~8].

부채널 분석에 대한 취약성을 가지는 디바이스의 안전성을 높이기 위해서 다양한 대응법들이 연구 되었다. 대응법 중에서 일반적으로 사용되는 방법으로는 마스킹 기법과 셔플링 기법이 있다. 마스킹 기법은 임의의 마스킹 값을 이용하여 공격자가 분석하고자 하는 공격지점의 정보를 숨기는 방법이다. 마스킹 기법을 통해 1차 CPA를 대응할 수 있지만 2차 CPA를 통해 분석되므로 마스킹 대응법만으로는 안전성을 보장할 수 없다. 2차 CPA로 분석되는 경우 연속된 동일 연산의 순서를 바꾸는 셔플링 기법을 통해 부채널 분석에 대한 복잡도를 높이는 방법으로써 2차 CPA까지도 방어할 수 있다.

최근 S.Tillich는 마스킹 기법과 셔플링 기법이 사용된 AES에 대한 분석 방법으로 Template Attack(TA)을 이용한 biased-mask 공격기법을 제안하였다. biased-mask 공격기법은 마스킹이 적용된 대응법에 대한 공격기법으로 기존의 2차 CPA에서 분석되는 것을 1차 CPA에서 분석이 가능하게 한다. 또한 셔플링 대응기법이 적용된 알고리즘을 공격하기 위해 C.Clavier의 windowing 기법을 사용하여 마스킹과 셔플링이 모두 적용된 알고리즘을 공격하였다[1~2, 9].

하지만 S.Tillich의 biased-mask 기법은 마스킹 기법에 대한 분석 복잡도를 낮추기 위해서는 사전에 마스킹 값에 대한 template 정보를 수집하여야 한다는 것과 실제 전력이 나타나는 정확한 위치를 예측하기 어려운 마스킹 값의 시간 위치를 알고 있어야 한다는 가정이 필요하며 마스킹 값의 시간 위치는 정확하게 알고 있지 않으면 분석 가능성이 현저히 저하된다. 즉, biased-mask 기법을 사용하기 위해서는 마스킹 값의 시간 위치와 별로도 수집한 template 정보가 있어야 분석이 가능하다.

본 논문에서는 biased-mask 기법에서 반드시 필요로 하는 두 가지 가정인 마스킹 값의 시간 위치를 알고 있어야 한다는 가정과 template 정보를 사전에 수집해야 한다는 가정 없이 마스킹-셔플링 대응법이 적용된 AES를 해독할 수 있는 방법인 편중전력분석(Biasing Power Analysis, BPA)을 제안하였다. 본 방법은 마스킹 값에 대한 template 정보를 이용하지 않고 이미 수집된 파형만을 이용해서 분석할 수 있는 방법이며, 마스킹 값의 시간 위치를 모르는 상태에서도 분석이 가능하다.

본 논문의 구성은 다음과 같다. Ⅱ장에서는 부채널 분석에 대한 대응법과 그에 대한 기존의 분석 기법인 biased-mask 기법을 설명하고, Ⅲ장에서는 제안된 BPA 기법에 관한 설명과 BPA 기법의 사용에 필요한 편중 분류방법에 대해 설명 하였으며, Ⅳ장에서는 실제 MSP430 소프트웨어 보드에서 전력분석 실험을 통해 BPA분석의 성능과 분석 가능함을 입증한다. 마지막 Ⅴ장에서는 실험 결과를 토대로 제시된 방법의 장, 단점을 논하고 논문을 결론지었다.

Ⅱ. 기존 분석 및 대응법

부채널 분석에 대한 대응법으로는 다양한 방법이 존재하지만 본 논문에서는 일반적으로 사용되는 부채널 대응법인 마스킹 기법과 셔플링 기법을 설명하고, 사용된 대응법에 대한 분석으로 windowing 기법과 S.Tillich가 제안한 biased-mask 기법을 설명할 것이다.

2.1 마스킹 및 셔플링

마스킹 기법은 부채널 분석에 사용되는 중간값을 숨기는 방법으로 8bit단위로 연산되는 AES의 각 중간값 D에 임의의 마스킹 값 m을 xor연산하여 $D \oplus m$으로 변환하는 방법으로 m값을 알 수 없기 때문에 변환된 중간값 $D \oplus m$도 알 수 없으므로 공격자로 하여금 단순 CPA분석으로는 키를 찾아내지 못하게 하는 방법이다.

셔플링 기법은 Sbox연산과 같이 동일한 연산이 독립적으로 발생하는 곳에 적용되는 기법이다. 즉, 첫번째 Sbox에 대한 연산이 수행되고 두번째 Sbox 가 수행되며 차례대로 마지막 Sbox까지 이루어지는 연산의 순서를 임의적으로 섞어주는 방법으로써, 공격 지점이 되는 연산이 16개의 Sbox연산 중 어느 것인지 알 수 없게 하는 방법이다. 셔플링 연산이 사용되지 않은 알고리즘의 연산은 분석 Sbox가 고정된 반면 셔플링이 사용된 알고리즘은 공격 지점이 되는 Sbox가 파형마다 동일한

시간위치에 출현할 확률이 1/16이 되어 분석 복잡도가 증가하게 된다[2].

2.2 windowing 분석법

S.Tillich는 셔플링 기법에 대한 분석 복잡도를 낮추는 방안으로 C.Clavier의 windowing 기법을 사용하였다[1]. 사용한 windowing 기법은 셔플링 된 반복연산 구간들을 더하는 방법으로 구간의 연산들을 하나의 시간에 분포시킴으로써 셔플링의 효과를 제거하는 방법이다.

그림 1은 C.Clavier의 windowing 방법을 나타낸 것이다. 16개의 Sbox 연산이 수행되는 구간은 연산들의 시간이 동일하고 유사한 패턴을 지니게 된다. 따라서 셔플링 기법이 사용될 경우 16번의 연산중에서 어떤 연산이 먼저 수행 됐는지 알 수 없다. windowing 기법은 16개의 독립된 연산 시간에서 발생하는 전력을 하나의 시간으로 합쳐서 공격자가 원하는 연산의 셔플링 대응 효과를 감소시킨다.

하지만 windowing 기법으로는 셔플링의 분석 복잡도를 줄일 수 있지만 하나의 시간영역에 여러 개의 연산이 중첩되어 존재하므로 공격 복잡도를 온전히 낮추지 못하며 마스킹과 셔플링이 적용된 대응 알고리즘의 완벽한 분석 방법은 아니다.

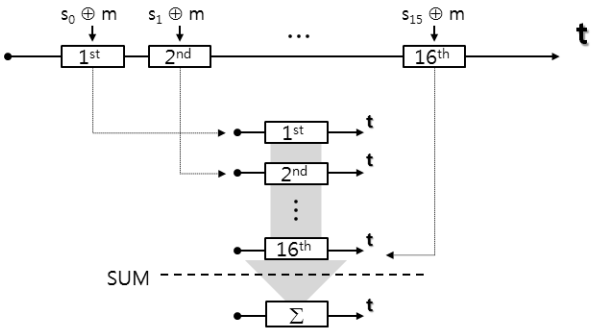


그림 1. windowing 기법
Fig 1. windowing method.

2.3 biased-mask 분석법

S.Tillich는 마스킹-셔플링 기법이 사용된 AES를 분석하는 전력분석법으로 1차 CPA를 이용하는 방법인 biased-mask 기법을 제안하였다[9].

그림 2는 S.Tillich의 분석방법을 나타낸 것이다. biased-mask 기법은 임의의 값인 m을 이용해 파형을 선택해서 분석하는 방법으로 사전에 수집된 마스킹 값의 template정보를 알고 있는 마스킹 값의 시간 위치를 이용하여 분석하여 마스킹 값의 헤밍웨이트(HW)가 8이 되는 파형만을 선택하여 분석 하는 방법이다.

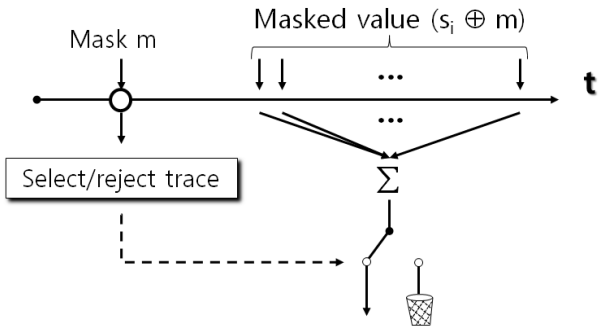


그림 2. S.Tillich의 biased-mask 기법
Fig 2. S.Tillich's biased-mak method.

기존에 제안된 biased-mask 기법은 m을 편중시킴으로써 2차 CPA로 분석될 파형을 1차 CPA로 분석할 수 있다는 장점을 가지고 있지만, 사전에 마스킹 값에 대한 template 정보의 수집해야 하고 마스킹 값의 시간 위치 또한 알아야 한다는 단점을 가지고 있다.

Ⅲ. 편중전력분석

본 논문에서 제안된 편중전력분석(Biasing Power Analysis, BPA)은 수집된 전력 파형들이 동일 시간에서 전력의 편중을 가지는 것을 이용해 헤밍웨이트 값과 같은 $\{0, 1, \cdots, 8\}$ 의 값으로 전력값을 분류한 후 그것을 이용해 상관전력분석을 하는 방법을 말한다. 이때 사용되는 분류방법에 따라 수집된 전력 파형들의 각 시간의 전력값은 0부터 8까지의 9개의 헤밍웨이트로 분류할 수 있다. 따라서 우리의 방법에 의하면 정확한 마스킹 시간에서 나타나는 template 정보를 가지지 않더라도 마스킹 값을 분류할 수 있음을 말한다.

제안된 분석 방법은 분류방법을 마스킹의 전력 부분에 적용하여 분석하는 방법과 공격지점이 되는 Sbox 부분에 적용하여 분석하는 방법으로 나누어진다. 마스킹의 전력 위치를 알고 있다면, 분류방법을 통해 마스킹의 전력 부분에서 헤밍웨이트를 구분지어 분석할 수 있게 되어 별도의 template 정보를 필요로 하지 않게 된다. 만일 마스킹의 전력 위치를 모르고 있는 경우 공격지점이 되는 Sbox에서 제안된 수식을 통해 키와 연

관된 마스킹 값의 헤밍웨이트를 구하는 방법으로 분석할 수 있다. 따라서, 본 장에서는 제안된 분석 방법인 마스킹 위치에서의 편중전력분석과 Sbox 위치에서의 편중전력분석에 대해 이론적으로 설명할 것이고, 두 분석 방법의 설명에 앞서 분석에 필요로 하는 분류방법을 설명할 것이다.

3.1 전력편중을 이용한 분류 방법

본 절에서는 헤밍웨이트에 대한 분류 함수를 적용하는 방법을 자세히 설명한다. 분류방법은 그림 3에서와 같이 분석 지점에서의 전력값의 분포는 그림 4에서와 같이 일반적으로 Gaussian 분포를 따르는 헤밍웨이트 분포와 실제로 다르게 나타나기 때문에 전력값과 헤밍웨이트 사이의 적절한 분류 방법이 설계되어야 한다.

제안하는 분류 방법은 전력값을 일정 간격의 비율로 분류하는 방법으로 전체 N개의 파형에서 분석지점이 되는 전력의 전력값 중 최대값과 최소값을 구하고, 그

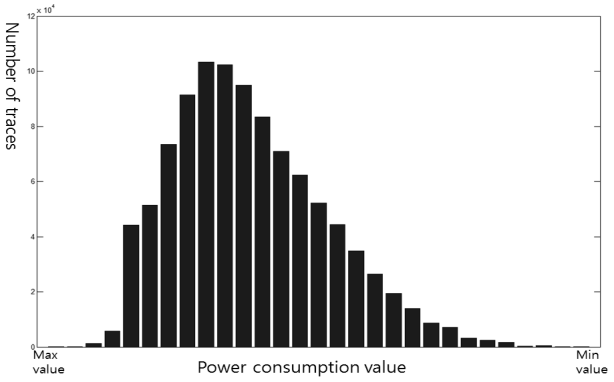


그림 3. 분류 지점의 전력 분포 예
Fig 3. Classified point of power distribution.

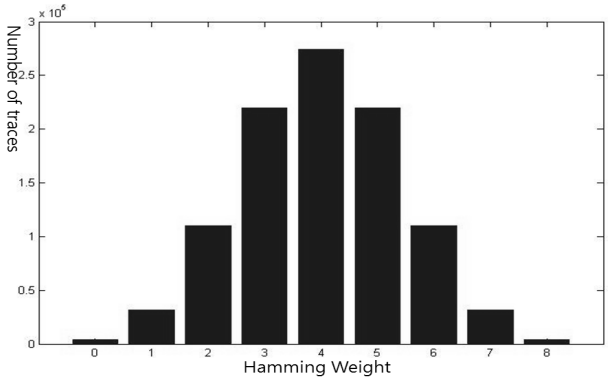


그림 4. 8bit 헤밍웨이트의 확률분포
Fig 4. the probability distribution of 8bit Hamming-Weight.

최대값과 최소값을 9개의 동일한 크기로 나누어 각 부분에 해당되는 전력값을 큰 값부터 순차적으로 8~0의 값으로 분류하는 방법이다. 즉, 분류지점이 되는 시간에서 헤밍웨이트 0에서 8의 값으로 분류하기 위해 그림 5와 같이 최대값에서 최소값까지의 변화량을 9등분하고 그것을 하나의 범위로 하여 전력이 큰 값의 범위부터 헤밍웨이트 값을 8에서 0으로 분류하는 것이다.

3.2 마스킹 위치에서의 편중전력분석

마스킹 위치에서의 BPA분석은 2.3에서 설명한 biased-mask 분석법과 유사한 방법으로, 사용된 마스킹 값을 제한하는 방법으로 분석한다. 마스킹 값 m의 헤밍웨이트 HW(m)이 8이나 0인 값만 사용된다면, 계산되어지는 분류함수에 '0xFF'를 xor 하거나 그대로 사용하여 분석할 수 있다. HW(m)이 0이라면 분석지점이 되는 Sbox출력 값 s에 마스킹 값 m이 xor연산된 값 $s \oplus m$이 $s \oplus 0x00 = s$ 가 되기 때문이고, HW(m)가 8이라면 m이 '0xFF'로 $s \oplus 0xFF$와 같기 때문에 마스킹이 사용된 중간값을 알 수 있게 된다. 하지만 이것은 마스킹 값의 헤밍웨이트를 알아야 쓸 수 있는 방법이다. 실제 수집된 정보에서 마스킹 값에 관해 알 수 있는 것은 마스킹 값 m의 전력값인 $C(m)$뿐이다. 따라서 3.1절에서 설명한 그림 5의 분류 방법을 통해 $C(m)$을 헤밍웨이트와 같은 위상의 값으로 변환 해야만 마스킹 값을 원하는 헤밍웨이트로 제한할 수 있다. 결과적으로 마스킹 값 m의 시간 위치만 알고 있다면 분류방법을 통해 특정 마스킹 값만을 선택함으로써 분석할 수 있게

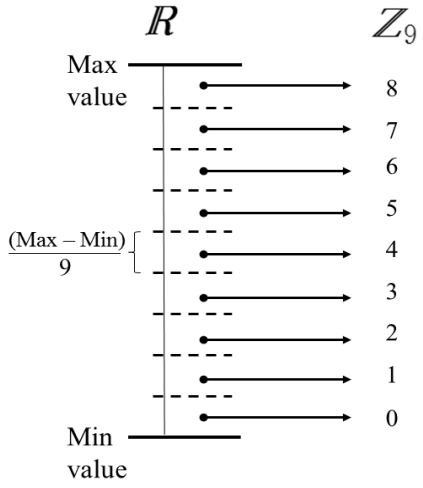


그림 5. 균일등분 분류방법
Fig 5. Classification method for uniform division.

되는 것이다.

전력값을 헤밍웨이트로 변환시키는 방법을 마스킹 값에 적용시킴으로 분석에 필요한 파형을 선택할 수 있다. 분석에는 HW(m)이 0, 8인 파형을 선택하는 것이 분석 효율에 좋지만, 문제는 분류된 파형의 수가 수집된 파형에 비해 수치적으로 약 1/128 정도로 줄어들기 때문에 분석에 실패할 수 있다는 것이다. 따라서 파형 수에 대한 문제를 해결하고자 HW(m)이 0, 8인 파형 외에도 1, 7인 파형을 사용할 수 있다. HW(m)이 0, 1인 값은 HW(m)이 0인 것과, HW(m)이 7, 8인 값은 HW(m)이 8인 것과 연관성이 있기에 가능한 분석이다.

분류방법을 통해 선택된 두 종류의 파형을 독립적으로 사용하는 것은 효율적인 면에서 낭비라고 할 수 있다. 그것은 두 종류의 파형을 동시에 사용하여 분석 할 수 있기 때문이다. 당연하게도 HW(m)이 8이라는 것은 마스킹 값 m이 '0xFF'라는 것이고, 이것은 분류함수에 '0xFF'를 xor연산 해준다면 m을 제거하는 것과 같은 효과를 지닌다. 따라서 HW(m)이 8인 것과 연관성을 지닌 파형을 분석할 때는 분류함수의 값에 '0xFF'를 xor연산을 해주고 그렇지 않은 경우엔 아무런 추가 연산 없이 사용하여 분석한다면 두 종류의 파형을 모두 이용하여 분석할 수 있게 된다.

3.2 Sbox 위치에서의 편중전력분석

Sbox 위치에서의 편중전력분석은 마스킹 위치에서의 분석과 다르게 Sbox의 출력 부분을 분석지점으로 하여 분석 된다. 공격지점이 되는 Sbox출력 연산 s_{ij} ($= Sbox[p_i \oplus k_j]$, p_i: i번째 입력평문, k_j: j번째 키 후보)는 마스킹 값 m_i와 xor 되면서 $s_{ij} \oplus m_i$로 연산되기 때문에 m_i값을 알지 못하는 한 1차 전력 분석이 불가능하다. 하지만 m_i값을 알지 못할 뿐 계산에 의해 HW(s_{ij})값을 알고 수집된 파형으로 $s_{ij} \oplus m_i$의 전력값인 $C(s_{ij} \oplus m_i)$를 알고 있기 때문에 BPA는 이 두 가지 값 HW(s_{ij})와 $C(s_{ij} \oplus m_i)$를 이용하여 분석이 가능하며 BPA 분석 원리는 다음과 같다.

일반적인 2차 CPA의 경우 다음과 같은 수식이 성립함이 전제된다[5].

$$\begin{aligned} \mathrm{HW}(s_{ij}) &\approx |\mathrm{HW}(s_{ij} \oplus m_i) - \mathrm{HW}(m_i)| \\ &\approx |C(s_{ij} \oplus m_i) - C(m_i) \end{aligned} \quad (1)$$

수식 (1)과 같이 전력값 $C(s_{ij} \oplus m_i)$와 $C(m_i)$의 차는 공격자가 계산 가능한 중간값 HW(s_{ij})와 상관도가 존재하며 완벽한 헤밍웨이트 모델에서 상관계수가 대략 0.24임이 알려져 있다. 하지만 우리는 m의 전력값 $C(m_i)$을 모르는 것을 가정했기 때문에 수식 (1)을 이용한 분석이 불가능하다. 따라서 본 논문에서는 근사 수식 (1)을 수식 (2)와 같이 변환하였다.

$$\mathrm{HW}(m_i) \approx |\mathrm{HW}(s_{ij} \oplus m_i) - \mathrm{HW}(s_{ij})| \quad (2)$$

수식 (2)의 HW(m_i)는 HW(s_{ij})가 계산가능하기 때문에 수식 (1)과는 달리 HW(m_i)를 계산 가능한 변수들의 조합으로 일부 변환이 가능하다. 또한 HW(m_i)는 s_{ij}($= Sbox[p_i \oplus k_j]$)에 의해 계산되기 때문에 k_j값에 의존한다. 따라서 수식 (2)의 계산에서 HW($s_{ij} \oplus m_i$)를 알고 있다는 가정 하에, s_{ij}에 사용된 키후보 k_j가 찾고자 하는 옳은 키라면 수식 (2)의 우변이 정확히 계산되어 알 수 없었던 m_i와 연관성을 가지는 HW(m_i)가 추측된다. 추측된 HW(m_i)은 키에 의존한 값이기 때문에 키 값을 분석할 수 있게 되는 것이다. 하지만 위에서의 가정과 달리 실제로 수식 (2)의 HW($s_{ij} \oplus m_i$)는 m_i값을 모르기 때문에 불가능하기 때문에 수식 (2) 역시 실제 공격자가 이용 가능한 정보는 아니다.

$$\begin{aligned} \mathrm{HW}(m_i) &\approx |\mathrm{HW}(s_{ij} \oplus m_i) - \mathrm{HW}(s_{ij})| \\ &\approx |C(s_{ij} \oplus m_i) - \mathrm{HW}(s_{ij})| \end{aligned} \quad (3)$$

한편 $C(s_{ij} \oplus m_i)$는 실제 전력에 존재하는 전력 정보이므로 $\mathrm{HW}(s_{ij} \oplus m_i) \approx C(s_{ij} \oplus m_i)$임을 이용하면 수식 (3)이 유도되어진다. 하지만 완벽한 헤밍웨이트 모델에서도 HW(s_{ij})와 $C(s_{ij} \oplus m_i)$가 정확히 동일한 위상을 갖지 않으므로 수식 (2)를 수식(3)으로 대치할 수 없으며, 실제 분석을 적용하기 위해서는 두 수식이 호환될 수 있도록 만드는 헤밍웨이트와 전력간의 분류기준이 수식 (4)과 같이 정의되어야 한다. 분류방법에 대한 자세한 설명은 3.1절에서 설명하였다.

$$F: \quad \mathbb{R} \rightarrow \mathbb{Z}_9 \quad (4)$$

수식 (4)은 같이 전력값을 헤밍웨이트와 같은 $\mathbb{Z}_9$의 값들로 바꾸는 것이다. 이렇게 $\mathbb{Z}_9$의 값으로 변환된 $C(s_{ij} \oplus m_i)$의 변환값 $F(C(s_{ij} \oplus m_i))$를 이용해

HW(m_i)를 다음과 같이 계산할 수 있게 된다.

$$\mathrm{HW}(m_i) \approx |F(C(s_{ij} \oplus m_i)) - \mathrm{HW}(s_{ij})| \quad (5)$$

따라서, 공격자는 이론적으로 수식 (5)를 통해 HW(m_i), 즉, 변환된 $F(C(s_{ij} \oplus m_i))$을 이용해 HW(m_i)를 계산할 수 있다.

수식 (5)는 일반적인 2차 전력 분석과 달리 전력을 헤밍웨이트로 변환하는 수식(4)에 의해 정보가 왜곡되므로, 수식 (1)에서의 대략적인 유사성인 0.24보다 낮은 상관도를 갖으며, 수식 (5)를 이용하여 실제 분석을 하기가 쉽지 않다. 그래서 우리는 수식(5)를 이용하여 전력분석을 시행하기 위해 분석 상관도를 높이는 방법으로 편차가 높은 헤밍웨이트만을 적용하여 HW(m_i)와 실제 m_i값과의 연관성을 높이는 방법을 이용할 것이다[10].

$F(C(s_{ij} \oplus m_i))$의 값이 0과 8에 가까울수록 전력값과 헤밍웨이트의 오류 가능성이 줄어들기 때문에 m_i와 연관성이 높아지며, 반대로 $F(C(s_{ij} \oplus m_i))$의 값이 4에 가까울수록 m_i와의 연관성이 떨어진다. 따라서 k_j와 연관된 HW(m_i)를 추측하기 위해서는 알고리즘 1.의 단계 2처럼 $F(C(s_{ij} \oplus m_i))$ 값이 7이상 1이하인 값을 사용할 것이다. 알고리즘 1은 $F(C(s_{ij} \oplus m_i))$이 7이상 1이하인 값으로 결정하는 방법을 나타낸 것이다. 공격자는 분석에 사용될 256개의 키후보 k_j에 대해 연산으로 구해진 HW(m_i)의 256 종류의 중간값들의 집합인 BM_j를 구하고 이와 짝이 되는, m_i와의 연관성을 높이기 위해 분류하는, 파형들의 집합인 V_j를 이용하여 분석한다.

알고리즘 1. HW(m_i)값 추측 방법
Input : p_i, $F(C(s_{ij} \oplus m_i))$ Output : 중간값 집합 BM_j, 파형의 집합 V_j
1. for j= 0 to 255 2. for i= 0 to $N-1$ 3. if $F(C(s_{ij} \oplus m_i)) \geq 7$ or $F(C(s_{ij} \oplus m_i)) \leq 1$ 3.1. $s_{ij} = p_i \oplus j$ 3.2. $\mathrm{HW}(m_i) = \lvert F(C(s_{ij} \oplus m)) - \mathrm{HW}(s_{ij}) \rvert$ 3.3. HW(m_i)를 집합 BM_j에 포함시킨다. i번째 파형을 V_j에 포함시킨다. 4. return(BM_j, V_j)

추측되는 HW(m_i)는 k_j값에 의존하여 결정되며, k_j값을 모르기 때문에 알고리즘 1.과 같이 k_j에 해당하는 256개의 값을 모두 추측해야 한다. 또한, HW(m_i)는 256개의 키 후보에 의존하기 때문에 알고리즘 1.처럼 256종류의 집합 BM_j가 구해지고 그것과 짝으로 이루는 V_j를 구하게 된다. 따라서 BM_j로 이것과 짝을 이루는 V_j를 분석한다면 비밀 키를 구할 수 있게 된다. 이것은 HW(m_i)가 알고리즘 1.의 3.2와 같이 $\mathrm{HW}(m_i) = \lvert F(C(s_{ij} \oplus m_i)) - \mathrm{HW}(s_{ij}) \rvert$로 계산되어지기 때문에 나오는 결과이다. 즉, 알고리즘 1.에서 키후보 256개의 값을 추측하면 HW(m_i)역시 256종류의 후보 값이 생성되어 추측에 사용되는 것이다.

이렇게 추측된 HW(m_i)값을 이용하여 CPA분석을 하게 되면 정확한 m_i의 시간 위치는 알 수 없지만, 알고리즘에서 처음에 위치한 마스킹이 생성되는 구간 전체를 분석한다면 존재하는 m_i의 시간 위치에서 상관계수 피크가 생성되어 분석된다.

3.2절의 마스킹 위치에서의 분석은 단순하게 마스킹 값을 분류하여 CPA 분석을 하는 것이지만 3.3절은 그 과정이 3.2절의 분석방법에 비해서 복잡하다. 그 과정을 보다 쉽게 보기 위해 알고리즘으로 나타낸 것이 알고리즘 2이다.

알고리즘 2. BPA 수행 과정
단계 1. 분류방법을 사용 $C(s_{ij} \oplus m_i)$의 분류값 계산 - $F(C(s_{ij} \oplus m_i))$ 계산 단계 2. 알고리즘 1.HW(m_i)값 추측 방법 수행 - BM_j, V_j결정 단계 3. CPA분석 - 최대 상관계수를 갖는 key결정 (단, $\rho(A,B)$는 A와 B의 피어슨 상관계수) 3.1. $a \leftarrow 0$, $b \leftarrow 0$, $Key \leftarrow 0$ from $j = 0$ to 255 do 3.2. $b \leftarrow \mathrm{MAX}(\rho(V_j, BM_j))$ 3.3. if $b > a$ 3.4 $Key \leftarrow j$ $a \leftarrow b$ 단계 4. return(Key)

Ⅳ. 실험 환경 및 분석 결과

4.1 실험 환경

본 논문에서 제시한 BPA에 대한 실험을 위해서 MSP430 Chip board를 이용하였다. 사용된 AES는 C.Herbst가 논문에서 제안한 방법으로 구현한 것으로 마스킹-셔플링 기법이 사용된 것이다[2]. 구현된 AES 알고리즘의 파형 수집은 오실로스코프에서 250MS/s의 Sampling rate로 1,000,000개의 파형을 수집하였다.

그림 6은 수집된 파형을 나타낸 그림으로써 16개의 1round Sbox 연산 부분이며 앞에 좁은 피크는 실험에 의해 불러온 마스킹 값 6개의 전력이다. 그림 6의 파형을 그대로 사용하여도 분석이 가능하지만 분석의 효율을 높이기 위해서 압축을 이용한 파형 전처리를 통해 파형을 분석 이 용이한 형태로 변형하였다.

그림 7은 연속된 포인트를 더하는 방식의 압축을 사용하여 250MS/s의 파형을 5MS/s의 파형으로 변환시킨 파형으로 그림 6보다 Sbox연산이 확연히 구분된다.

본 논문에서 분석하는 파형은 마스킹과 셔플링 대응

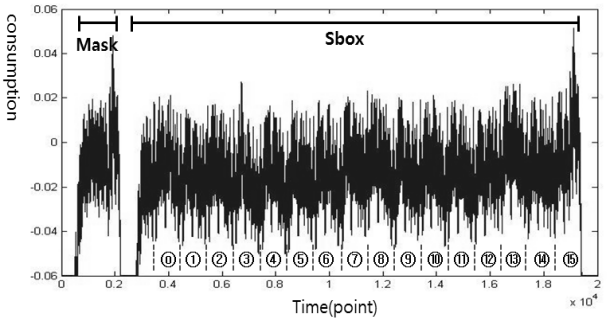


그림 6. AES, mask value & 1round Sbox
Fig 6. AES, mask value & 1 round Sbox.

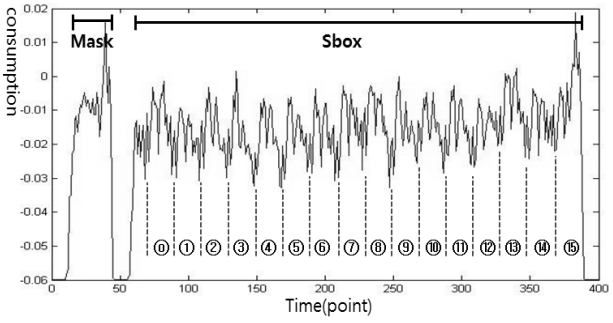


그림 7. AES, 5MS/s로 변환, 1/50 압축
Fig 7. AES, convert to 5MS/s, 1/50 compression.

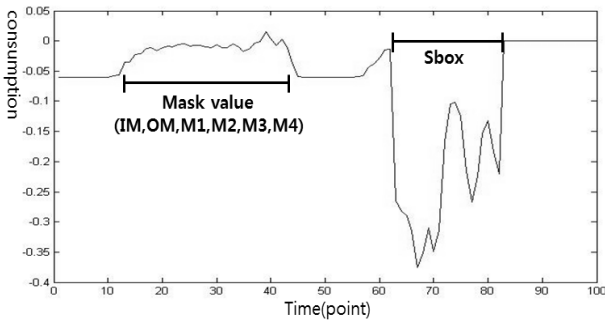


그림 8. AES, windowing 기법 적용 파형
Fig 8. AES, traces to applied windowing method.

법이 모두 적용된 파형이므로 C.Clavier가 제안한 windowing 기법을 적용하여 분석할 것이다. 그림 7은 그림 6보다 확연히 구분되는 Sbox연산으로 분석을 위한 기법 중 windowing 기법을 보다 정확하게 사용할 수 있다.

그림 8은 그림 7에서 windowing 기법을 적용하여 16개의 Sbox연산을 하나로 합하여 나타낸 파형이며, 16개의 Sbox 연산이 하나로 합쳐졌기 때문에 뒤에 아래로 형성된 피크가 Sbox연산이 된다.

실험에 사용된 Sbox 시간 위치는 가장 상관성이 높은 시간위치 1 point를 사용하였다. 이것은 압축에 의해 50 points가 1point로 압축된 것이기 때문에 결과적으로 50 points의 시간을 BPA분석에 이용한 것이다.

우리의 실험에서는 첫 번째 Sbox, 두번째 Sbox, 세번째 Sbox, 네번째 Sbox를 분석하며, Sbox 출력값과 연산되는 마스킹 값은 m_0, m_1, m_2, m_3 4개의 값이 사용된다.

BPA 실험 결과를 보인 후에 새롭게 제시한 BPA 분석 방법을 다른 분석과 비교해 보기 위해서 일반적인 분석법으로 사용되는 2차 CPA분석을 사용하여 결과를 비교할 것이다. 2차 CPA분석의 방법은 그림 8과 같이 압축과 windowing이 사용된 상태에서 마스킹 값의 시간 위치와 연산의 시간위치를 차분하는 방법을 사용할 것이다. 2차 CPA분석은 BPA와 마찬가지로 상관관계가 가장 높은 마스킹 값의 시간 위치 1 point를 사용하여 분석 할 것이다.

4.2 실험 결과

분석은 1,000,000개의 파형을 사용하였으며, 분류되어 실제로 분석에 사용된 파형수는 마스킹 위치에서의

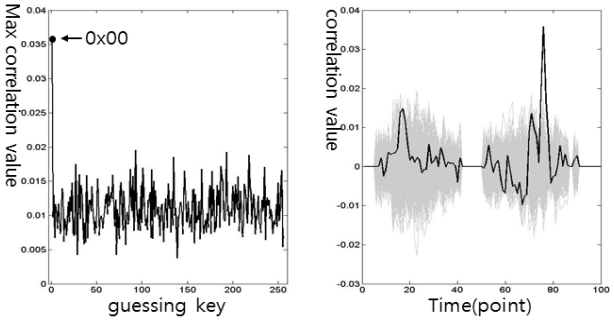


그림 8. BPA with mask,1st Sbox, 키 후보들에 대한 최대상관계수(좌), 키 후보들의 CPA 분석 파형(우)

Fig 9. BPA with mask, 1st Sbox, max correlation value for guessing keys(left), correlation trace of guessing keys(right)

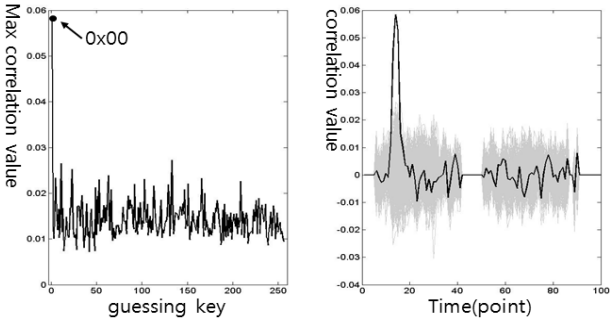


그림 10. BPA with Sbox,1st Sbox, 키 후보들에 대한 최대상관계수(좌), 키 후보들의 CPA 분석 파형(우)

Fig 10. BPA with Sbox, 1st Sbox, max correlation value for guessing keys(left), correlation trace of guessing keys(right).

BPA는 전체의 1/10 정도인 약 11만개의 파형만을, Sbox 위치에서의 BPA 분석은 전체의 1/20 정도인 약 5만개의 파형만을 사용되었다.

그림 9는 마스킹 위치에서 BPA 분석한 결과를 나타낸 것으로 좌측에 256개의 키 후보의 최대상관계수를 표시한 것이고, 우측은 좌측의 최대값인 첫 번째 키 '0x00'에 대한 CPA 분석 파형을 검은색으로 다른 키 후보들의 CPA 분석 파형을 회색으로 나타낸 것이다.

마스킹 값의 HW가 0과 8이라고 추측되는 값으로 제한하였고, 분류함수를 Sbox의 출력값으로 했기 때문에 분석 파형인 그림 8의 Sbox 위치에서 peak가 형성되는 것을 볼 수 있다.

그림 10은 Sbox 위치에서 BPA 분석한 결과를 나타낸 것으로 그림 9와 마찬가지로 좌측에 256개의 키 후보의 최대상관계수를 표시한 것이고, 우측은 좌측의 최대값인 첫 번째 키 '0x00'에 대한 CPA 분석 파형을 검은색으로 다른 키 후보들의 CPA 분석 파형을 회색으로 나타낸 것이다.

분석에 사용된 분류함수는 키와 연관된 HW(m_{ij})로 분석한 것이기 때문에 그림9와는 다르게 그림 8의 마스킹이 생성되는 구간에서 최대 peak가 발생되는 것을 확인할 수 있다.

4.3 2차 CPA분석과 BPA분석의 비교

2차 CPA분석과 BPA분석과의 비교는 1,000,000개의 파형을 분석한 결과만을 비교하였다. 2차 CPA분석은 [그림 8]의 파형에서 마스킹 값과 전력값을 $|C(s_{ij} \oplus m) - C(m)|$와 같이 차분하여 분석한 것이다.

표 1의 분석에서 회색으로 음영진 것은 키 분석에 성공한 것으로 2차 CPA 분석은 3개의 키만을 찾는데 그친 반면, 두 방법의 BPA분석은 16개의 키 전부를 찾아내었다. 하나의 분석 결과만으로 분석 방법의 우열을 가릴 수는 없지만 우리의 실험 결과에서는 기존의 2차 전력 분석방법과 비교했을 때, 현격한 성능의 차이를 확인하였다.

표 1. 2차 CPA와 BPA분석의 비교

Table 1. Compare S.O.CPA with BPA.

	Analysis method		
Key	Second order CPA	BPA with mask	BPA with Sbox
1st 0x00	0x84	0x00	0x00
2nd 0x01	0x23	0x01	0x01
3rd 0x02	0xDC	0x02	0x02
4th 0x03	0xF3	0x03	0x03
5th 0x04	0x04	0x04	0x04
6th 0x05	0xC6	0x05	0x05
7th 0x06	0x32	0x06	0x06
8th 0x07	0xF7	0x07	0x07
9th 0x08	0x08	0x08	0x08
10th 0x09	0xEA	0x09	0x09
11th 0x0A	0x6C	0x0A	0x0A
12th 0x0B	0x0E	0x0B	0x0B
13th 0x0C	0x0C	0x0C	0x0C
14th 0x0D	0xD0	0x0D	0x0D
15th 0x0E	0x34	0x0E	0x0E
16th 0x0F	0x64	0x0F	0x0F

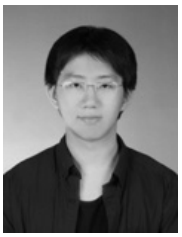
V. 결 론

본 논문에서 마스킹-셔플링 기반의 부채널 대응법을 2차 CPA를 사용하지 않고 공격하는 실제적인 BPA 분석 방법을 제안하였다. 기존의 S.Tillich의 방법은 template 정보와 마스킹 값에 대한 시간 정보를 알고 있다는 가정이 필요하지만, 본 논문에서 제안하는 BPA 분석은 이러한 가정 중 하나를 제외하거나 두 가정 모두 없이 적용 가능한 분석 방법이다. 제안된 분석 방법을 검증하기 위해 MSP430 Chip에서 얻은 부채널 정보를 이용해 AES-128 알고리즘에 대해 비밀키를 찾는 실험을 수행하였고, 백만개의 전력 파형으로 실험을 성공하였다. 하지만 MSP430 칩과 같은 MCU 칩의 경우 탐지된 부채널 정보가 노이즈가 적고, 다른 하드웨어 대응법이 없는 환경이기에 백만개 정도의 파형으로 BPA 가 가능했지만, 실제 사용되는 IC칩 카드와 같은 경우는 자체 H/W 대응법이 디폴트로 구동되어 동작함으로 제안된 BPA를 활용하여 분석할 경우 공격 성공 확률이 낮아 질 것으로 예상된다. 따라서, 향후 연구의 진행 방향은 다양한 H/W 노이즈가 존재하는 실제 IC 칩 환경에서 BPA분석이 가능하도록 향상된 신호전처리, 분류 기준과 차분 기술 개발 하는 것이다.

참 고 문 헌

[1] C.Clavier, J.-S.Coron, and N.Dabbous. “Differential Power Analysis in the Presence of Hardware Countermeasures“, CHES 2000 , LNCS 1965, pages 252 - 263, 2000.
[2] C.Herbst, E.Oswald, and S. Mangard, “An AES Smart Card Implementation Resistant to Power Analysis Attacks”, ACNS 2006, LNCS 3989, pp 239-252, 2006.
[3] E.Brier, C.Clavier, and F.Olivier, “Correlation power analysis whit a leakage model”, CHES 2004, LNCS 3156, Spreinger-Verlag, pp16-29, 2004.
[4] E.Oswald and S.Mangard, “Template Attacks on Masking—Resistance is Futile”, CT-RSA 2007, LNCS 4377 , pages 243 - 256, 2007.
[5] E.Oswald, S.Mangard, C.Herbst, and S.Tillich. “Practical Second-Order DPA Attacks for Masked Smart Card Implementations of Block Ciphers”, CT-RSA 2006, LNCS 3860 Springer, pp. 192 - 207. 2006.
[6] P.Kocher, “Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems”, CRYPTO 96, LNCS1109 pp. 104-113, 1996.
[7] P.Kocher, J.Jaffe, and B.Jun, “Differential power analysis”, CRYPTO 1999, LNCS 1666, Springer-Verlag, pp388-397, 1999.
[8] P. Kocher, J.Jaffe, and B.Jun, “Introduction to differential power analysis and related attack”, Cryptography Research, White Paper, 1998.
[9] S.Tillich, C.Herbst, and S.Mangard, “Protecting AES Software Implementations on 32-Bit Processors Against Power Analysis,” Computer Science 2007, LNCS 4521, pp. 141-157, 2007.
[10] Y.Kim, T.Sugawara, N.Homma, T.Aoki, and A.Satoh, “Biasing power traces to improve correlation in power analysis attacks”, COSADE 2010, Darmstadt, Germany, February 4-5, 2010.
[11] 박종연, 최지선, 한동국, 이옥연, “RSA에 대한 향상된 등간격 선택 평문 전력 분석 방법”, 대한전자공학회 학술대회, 1877-1880쪽, 2010년.
[12] 박종연, 한동국, 이옥연, 최두호, “RSA-CRT의 향상된 등간격 선택 평문 전력 분석”, 전자공학회논문지-CI, pp. 117-126, 2011.

저 자 소 개



조 종 원(정회원)

2010년 국민대학교 수학과 학사 졸업.
2012년 국민대학교 수학과 석사 졸업
2012년 ~ 현재 한국시스템보증 연구원

<주관심분야 : 부채널 분석 및 대응법, 정보보안, 스마트카드 보안>

한 동 국(정회원)-교신저자

1999년 고려대학교 수학과 학사 졸업
2002년 고려대학교 수학과 석사 졸업
2005년 고려대학교 정보보호 대학원 박사
2004년 4월~2005년 4월 일본 Kyushu Univ. 방문연구원
2005년 4월~2006년 4월 일본 Future Univ. -Hakodate, Post Doc.
2006년 6월~2009년 2월 한국전자통신연구원 정보보호연구본부 선임연구원
2009년 3월~현재 국민대학교 수학과 조교수

<주관심분야 : 공개키 암호시스템 안전성 분석 및 고속 구현, 부채널 분석, RFID/USN 정보보호 기술>