2차 차분전력분석 공격을 위한 새로운 전처리 기법[1)]

이철희*, 황아름*, 이동건*, 김호원*
부산대학교 컴퓨터공학과*
2fehee@gmail.gom, xlizhwang@gmail.com, guneez@gmail.com, howonkim@gmail.com

New Pre-processing method for Second-Order DPA

Chul-Hee LEE*, Ah-Reum Hwang*, Dong-Geon LEE*, Ho-Won Kim*
Department of Computer Science Engineering Pusan National University*

요 약

본 논문에서는 유비쿼터스 환경에서 그 환경을 구성하는 주요 디바이스들에 대하여 물리적인 공격을 통해 내부에 존재하는 비밀키 값과 같은 중요한 정보를 쉽게 찾아낼 수 있는 보다 효율적인 2차 차분전력분석 기법을 제안한다. 이 기법은 마스킹이 적용된 디바이스에서도 쉽게 그 키 값을 찾아내는 기법으로 기존의 전처리함수를 이용한 2차 차분전력분석 공격 기법과 제안하는 전처리함수를 적용한 기법을 실제로 각각 구현하여 성능을 분석하고 비교함으로써 제안하는 전처리함수를 이용하여 2차 DPA 공격을 했을 때 기존의 공격보다 더 강화되고 위협적인 물리적 공격임을 실험적으로 검증한다.

1. 서 론

오늘날 IT 기술의 발달로 인해 언제 어디서나 원하는 정보를 각종 네트워크에 접속하여 검색, 저장, 전송할 수 있는 유비쿼터스 환경이 이루어짐으로써 모든 사물이 지능화, 정보화, 네트워크화 되어가고 있다. 이런 환경에선 새로운 보안 문제가 부각되거나 혹은 존재하지 않던 보안 문제가 새로 발생하게 되는데 대표적인 것이 물리적 공격이다.

물리적 공격의 의미는 암호화 과정에서 발생하는 각종 물리적인 신호들을 채득하여 이로부터 암호화에 사용되는 암호키를 추출하는 것을 말하며 현재 스마트카드와 , USB 보안 token, RFID, 센서노드, Zigbee, Bluetooth와 같은 통신용 칩이나 방송용 수신 칩, 등 많은 분야에 위협적인 공격으로 응용되고 있다. 만약 물리적인 공격으로 인해 보안성이 보장되지 않게 되면 프라이버시 노출로 인한 인권침해 문제가 야기될 수 있고, 이는 관련 시장의 활성화를 가로막게 되어 매우 큰 경제적 손실 및 사회적 문제로 야기될 수 있다.

이러한 유비쿼터스 환경의 물리적인 보안에 있어 위협적인 물리적 공격은 여러 종류가 있으며 그 중에서도 부채널 공격(side-channel attack)[1-2]중의 하나인 차분 전력 분석(differential power analysis: DPA)[3] 공격은 가장 강력한 공격 방법이다.

이러한 차분 전력 분석 공격을 막기 위한 대응방법으로 마스킹[4-9]기법이 있으며 현재 가장 활발하게 연구되고 있는 대응 기법이다. 반면 이러한 마스킹 대응 기법을 공격하는 연구 또한 많이 이루어 졌는데 대표적인 것으로 2차 DPA(second-order differential power analysis)[10-12] 공격을 들 수 있다. 본 논문에서는 마스킹 기법에 대한 기존의 2차 DPA 공격에 사용된 대표적인 전처리함수(preprocessing function)의 특징에 대해서 알아보고 새로운 전처리함수를 이용하는 보다 강화된 2차 DPA공격 방법을 제시하였다. 다양한 전처리함수를 실험하기 위해서 마스킹 기법이 적용된 AES를 마이크로컨트롤러에 구현하였으며 제안하는 전처리함수를 이용한 공격 방법이 기존 공격 방법에 비해 보다 효율적이고 강력한 공격임을 실험적으로 검증하였다.

본 논문의 구성은 다음과 같다. 2장에서 마스킹 기법과 2차 DPA공격에 대해 설명한다. 3장에서는 기존의 전처리함수들을 이용한 2차 DPA공격 방법과 제안하는 새로운 전처리함수를 이용한 2차 DPA공격 방법에 대해 설명하고 각각의 성능을 비교하여 평가한다. 마지막으로 4장에서 결론을 맺는다.

2. 마스킹 기법과 2차 DPA공격

2.1 마스킹 기반 대응기법

마스킹 대응 기법은 랜덤한 마스크 값을 삽입하여 암호화 연산중에 발생하는 중간 값을 알 수 없게 함으로 추정하는 모델 값과 실제 소비 전력 값 사이에 상관관계를 제거하는 방법이다. 이 방법은 구현이 쉽고 비용도 적게 들기 때문에, 차분전력분석 공격을 막기 위해 가장 많이 사용되고 있다. 본 논문에서는 3장에 마스킹 된 암호 알고리즘에 2차 DPA 공격을 하기 위해 AES에 마스킹을 적용하였으며 구현 하였다. 구현한 마스킹에 대한 설명은 다음과 같다. 우선 AES의 암호화 연산 과정의

1) 이 논문 또는 저서는 2010년 교육과학기술부로부터 지원받아 수행된 연구임 (지역거점연구단육성사업/차세대물류IT기술연구사업단)

중간 결과 값을 랜덤하게 만들기 위해 S-box에 두 개의 랜덤한 마스크 mask1과 mask2를 이용하여 다음과 같은 식 (1)의 성질을 만족시키는 마스킹 된 S-box를 생성한다.

$$S_m(p \oplus k \oplus mask1) = S(p \oplus k) \oplus mask2 \quad (1)$$

마스킹 된 S-box의 입력 값 p⊕k⊕mask1은 키 값 k에 마스크 mask1을 XOR하고 AddRoundKey 단계에서 평문p와의 XOR 연산을 거쳐 생성된 값이다. 이 입력 값이 마스킹 된 S-box을 통과 했을 때의 출력 값 S_m(p⊕k⊕mask1)은 정상적인 입력이 원래의 S-box를 통과 한 후의 출력 값 S(p⊕k)에 또 다른 마스크 mask2를 XOR한 결과 값인 S(p⊕k)⊕mask2의 연산 결과와 같아지는 특징이 있다. 결국 mask1을 통해 키 값을 랜덤하게 만들며 마스킹 된 S-box을 통해서 SubByte단계를 지난 중간 결과 값을 공격자가 추정할 수 없게 만드는 효과를 주는 것이다. [그림 1]은 두 마스크 mask1과 mask2를 이용하여 마스킹 된 S-box을 생성하는 알고리즘이다.

입력 : mask1, mask2
출력 : m_sbox(p⊕k⊕mask1) = sbox(p⊕k)⊕mask2

```
1 : for i = 0 to 255 do
2 :       m_sbox(i⊕mask1) = sbox(i)⊕mask2;
3 : end for
4 : return m_sbox
```

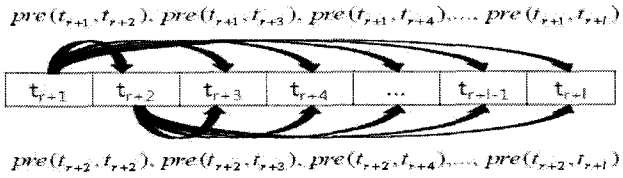
[그림 1] 마스킹 된 S-box 생성을 위한 알고리즘

2.2 2차 DPA 공격

1차 DPA 공격은 측정한 부채널 신호의 한 시점 정보만 이용하기 때문에 2.1에서 설명한 마스킹 대응 기법을 통해서 1차 DPA공격을 막을 수 있게 된다. 하지만 마스킹 대응 기법에도 취약성이 존재하는데 그것은 다음과 같다. 두 마스크 mask1과 mask2가 같게 되는 경우 마스킹 된 S-box의 입력 값 p⊕k⊕mask1과 마스킹 된 S-box의 출력 값 S_m(p⊕k⊕mask1)을 XOR하면 마스크가 제거 된 (p⊕k)⊕S(p⊕k)값을 구할 수 있게 된다. 왜냐하면 식 (1)에 의해 S_m(p⊕k⊕mask1) = S(p⊕k)⊕mask2 이기 때문에 (p⊕k⊕mask1)⊕S_m(p⊕k⊕mask1) = (p⊕k⊕mask1)⊕S(p⊕k)⊕mask2 = (p⊕k)⊕S(p⊕k) (단 mask1=mask2)가 성립하는 것이다. 이 결과는 키와 데이터에 의존한 값이 되기 때문에 결국 측정한 소비전력 트레이스와의 상관계수를 구하여 키 값을 도출 해낼 수 있게 된다. 따라서 S-box의 입력 값과 출력 값이라는 두 시점의 값을 어떻게 찾고 이 두 시점의 XOR 연산 관계를 어떻게 이끌어 낼 것인가가 2차 DPA공격의 핵심 포인트가 되는 것이다. 2차 DPA 공격을 위해 먼저 마스크가 제거된 값을 얻을 수 있는 식 (2)와 같은 해밍 웨이트 모델식을 사용한다.

$$HW((p \oplus k \oplus mask1) \oplus (S(p \oplus k) \oplus mask1)) = HW((p \oplus k \oplus) \oplus S(p \oplus k)) \quad (2)$$

다음으로 측정한 소비전력 트레이스로부터 식 (2)와 동등한 소비전력 패턴을 얻기 위해서 전처리 단계를 거치는데 이때 식 (2)에서 사용된 S-box의 입력 값과 출력 값이 실제로 측정된 소비전력 트레이스의 각각 어느 시점에 존재 하는지 알아야 하기 때문에 S-box에 해당하는 연산이 다 포함되어 있는 적절한 구간을 선택하여 뽑아낸 뒤 [그림 2]와 같이 그 구간에 대해서 서로 다른 두 시점마다 모든 경우의 수를 두어 전처리 함수를 수행함으로 구간을 확장한다. 즉 원래의 구간 길이가 l일 때 확장된 구간 길이는 $(l-1)+(l-2)+...2+1 = l \cdot (l-1)/2$가 되는 것이다. [그림 2]에서 t는 측정한 트레이스이며 pre()는 전처리함수를 뜻하고 r은 몇 번째로 측정한 트레이스인지를 의미하며 +숫자는 하나의 소비전력 트레이스에 각각의 시간 포인트를 뜻한다. 전처리함수를 사용하는 이유는 측정한 소비전력에서 두 시점에 대한 값의 관계가 식 (2)의 XOR 연산 후의 해밍 웨이트를 취한 값과 동등한 관계를 가지도록 하기 위해서이다. 즉 전처리 함수를 어떠한 것을 사용하느냐에 따라 얼마나 동등한 관계가 이루어질지가 결정되어 지고 이것은 공격자가 추정하는 소비전력 모델과 측정한 소비전력 트레이스 사이에 높은 상관관계를 이끌기 위한 아주 중요한 요소가 되게 된다.



[그림 2] 두 시점을 찾기 위한 구간 확장 과정과 전처리 함수

3. 기존의 전처리함수와 제안하는 전처리함수

3.1 기존의 전처리함수

전처리함수는 2.2장에서 설명했듯이 측정한 소비전력 트레이스에서 추정하는 소비전력 모델과 동등한 패턴의 결과를 이끌어 내기 위해 DPA공격 이전에 적용시켜 최대한 높은 상관계수를 이끌어 내는 함수를 말한다. 이러한 전처리함수에는 현재 여러 형태의 함수가 있으며 이전에 연구 되어 진 대표적인 전처리함수들로 두 포인터에 대한 곱인 $pre(t_a,t_b) = t_a \cdot t_b$,[13], 두 포인터에 대한 차이값의 절대치인 $pre(t_a,t_b) = |t_a-t_b|$[14], 두 포인터에 대한 합의 제곱인 $pre(t_a,t_b) = (t_a+t_b)^2$[15] 등이 있다. 표 1은 이 함수들 중에서 $pre(t_a,t_b) = |t_a-t_b|$가 추정하는 소비 전력 모델인 $HW(t_a \oplus t_b)$와 가장 높은 상관관계를 가짐을 보여주고 있다. 하지만 표 1은 1비트 상에서 각각의 전처리함수들을 통해 얻은 결과 값이기 때문에 1비트에서 8비트로 비트수가 늘어나게 되면 $HW(t_a \oplus t_b)$의 연산 특성으로 출력되는 더 많은 경우의 결과 값을 고려해

야 하기 때문에 상관계수가 떨어지게 된다. 즉 각각의 전처리 함수들이 $HW(t_a \oplus t_b)$ 연산 결과 값을 정확하게 표현하지 못하게 되는 것이다. 그래서 우리는 기존의 전처리함수들 보다 $HW(t_a \oplus t_b)$ 연산 결과 값을 더 정확하게 표현 할 수 있는 새로운 전처리함수를 제안한다.

표 1. 전처리함수에 따른 상관 계수

	값				상관 계수
t_{am}	0	0	1	1	
t_{bm}	0	1	0	1	
$HW(t_a \oplus t_b)$	0	1	1	0	
$HW(t_{am}) \cdot HW(t_{bm})$	0	0	0	1	-0.57
$\lvert HW(t_{am}) - HW(t_{bm})\rvert$	0	1	1	0	1
$(HW(t_{am}) + HW(t_{bm}))^2$	0	1	1	4	-0.33

3.2 제안하는 전처리함수

먼저 $HW(t_a \oplus t_b)$와 가장 높은 상관계수를 나타내는 $|HW(t_{am}) - HW(t_{bm})|$ 함수가 왜 8bit 상에서 상관계수가 낮아지는지 다음의 예를 통해 설명하려고 한다. 식 (2)에 의해 $HW(t_a \oplus t_b) = HW(t_{am} \oplus t_{bm})$이며 이때 t_{am}가 $00010001_{(2)}$ 이고 t_{bm}가 $00110001_{(2)}$ 라고 한다면 $t_{am} \oplus t_{bm} = 00100000_{(2)}$ 이 될 것이고 $HW(t_{ma} \oplus t_{bm})=1$이 되어 $|HW(t_{am}) - HW(t_{bm})| = |2-1| = 1$과 같아지게 된다. 하지만 t_{am}가 $00100100_{(2)}$ 이고 t_{bm}가 $00000010_{(2)}$ 라고 한다면 $t_{am} \oplus t_{bm} = 00100110_{(2)}$ 이 될 것이고 $HW(t_{am} \oplus t_{bm})=3$이 되어 $|HW(t_{am}) - HW(t_{bm})| = |2-1| = 1$과 다르게 된다. 이 경우 t_{am}값과 t_{bm}의 비율을 고려한다면 이 문제를 어느 정도 해결할 수 있게 된다. 그래서 제안한 사전처리 함수가 식 (3)이다.

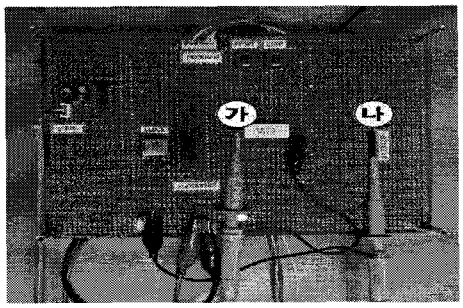
$$\left| \frac{HW(t_{am}) - HW(t_{bm})}{HW(t_{am})} + \frac{HW(t_{bm}) - HW(t_{am})}{HW(t_{bm})} \right| \quad (3)$$

제안한 전처리함수는 다중 비트일 때를 고려해서 이 함수는 단순히 t_{am}과 t_{bm}의 차이 값만을 사용하는 것이 아니라 각각 t_{am}과 t_{bm}상에서 $HW(t_{am})$와 $HW(t_{bm})$값에 따른 비율을 고려하도록 했다. 그래서 앞서 적용한 예를 제안하는 사전처리함수에 적용해 보면 t_{am}가 $00010001_{(2)}$ 이고 t_{bm}가 $00110001_{(2)}$ 라고 한다면 $t_{am} \oplus t_{bm} = 00100000_{(2)}$ 이 될 것이고 $HW(t_{am} \oplus t_{bm})=1$이 되며 $|((HW(t_{am})-HW(t_{bm}))/HW(t_{am}))+((HW(t_{bm})-HW(t_{am}))/HW(t_{bm}))| = |((2-3)/2)+((3-2)/3)| = |(-1/2)+(1/3)| = |(-3+2)/6| = 1/6$이 된다. 그리고 t_{am}가 $00100100_{(2)}$ 이고 t_{bm}가 $00000010_{(2)}$ 라고 한다면 $t_{am} \oplus t_{bm} = 00100110_{(2)}$ 이 될 것이고 $HW(t_{am} \oplus t_{bm})=3$이 되며 $|((HW(t_{am})-HW(t_{bm}))/HW(t_{am}))+((HW(t_{bm})-HW(t_{am}))/HW(t_{bm}))| = |(2-1)/2+(1-2)/1| = |1/2+(-1)| = |-1/2| = 1/2 = 3/6$이 된다. 결국 첫 번째 경우와 두 번째 경우 각각 $HW(t_{am} \oplus t_{bm})=1$, $HW(t_{am} \oplus t_{bm})=3$으로 1:3의 비율을 가지는데 제안한 사전처리함수의 경우 연산 결과가 1/6과 3/6으로 똑같이 1:3의 비율을 가지는 것을 확인 할 수 있다. 항상 예로 든 경우처럼 비율이 정확하게 맞는 것은 아니지만 $HW(t_{am} \oplus t_{bm})$의 연산 결과의 크기가 작고 큼의 비율은 어느 정도 잡아 낼 수 있기 때문에 기존의 $|HW(t_{am}) - HW(t_{bm})|$함수보다 성능이 더 좋다고 할 수 있으며 이에 대한 검증은 실험을 통한 결과로 보일 것이다.

3.3 2차 DPA 공격을 통한 실험 및 결과 분석

3.3.1 실험 환경

주요 실험 장비로 공격 대상 칩은 PIC18F452 마이크로컨트롤러를 사용하였고 AES-128비트를 공격할 암호 알고리즘으로 칩에 구현하였다. 칩의 동작 과정을 측정하기 위해 Ageilent사의 MS06102A 오실로스코프를 사용하였다. 또한 전원 공급을 위해 DAP-3005t모델의 power supply를 사용하여 전압은 5V, 전류는 0.5A를 주었고 전력 소비를 측정하기 위해 1Ω 시멘트 저항을 삽입하였다. [그림 3]은 앞서 설명한 부채널 공격을 위한 실험 환경으로 오실로스코프의 ㉮의 프로브로 시멘트 저항 양단을 연결하여 소비 전력을 측정하게 하고 또한 여러 전력 트레이스간의 시작지점을 일치시키기 위하여 General Purpose I/O 핀을 이용하여 암호화가 시작될 때 별도의 시작 신호를 트리거 하도록 하였다. 그리고 이 신호를 ㉯의 프로브를 이용하여 측정하며, 이를 트리거 신호로 사용하여 시작점을 일치시킨 상태에서 원하는 구간의 전력 파형을 측정 할 수 있게 하였다.



[그림 3] 부채널 공격을 위한 실험 환경

그리고 계측기를 통한 소비 전력 측정을 쉽게 작성 가능한 Vee-Pro 언어를 이용하여 자동화함으로 오실로스코프에서 측정한 전력 파형을 필요로 하는 개수만큼 지정한 파일이름으로 쉽게 저장할 수 있게 하였다. 마지막으로 공격자가 추정하는 소비전력 모델을 Matlab상에서 구현한 뒤 측정한 파형들과의 상관계수를 구하여 최종적인 결과 값을 이끌어 내었다.

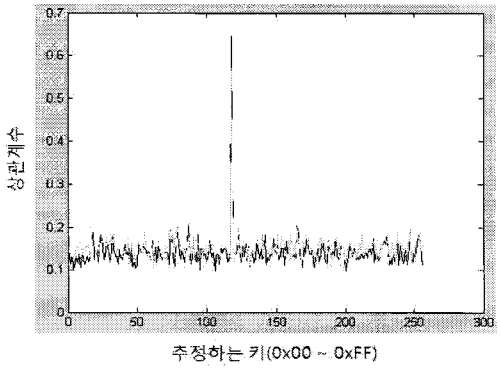
3.3.2 전처리함수 결과 분석

2차 DPA공격에 기반한 제안된 새로운 사전처리함수의 성능을 확인하기 위해, AES 암호화 알고리즘의 첫 번째 라운드에 사용된 16byte의 키를 모두 적용하여 파형을 측정하였다. 먼저 성능을 평가할 사전처리 함수는 기존의 가장 우수한 성능을 보이는 $pre(t_a,t_b) = |t_a-t_b|$와 제안한 $pre(t_a,t_b) = |((t_a-t_b)/t_a)+((t_b-t_a)/t_b)|$ 함수를 선택했다. 그리고 각각의 키에 대해서 총 500개의 소비전력 트레이

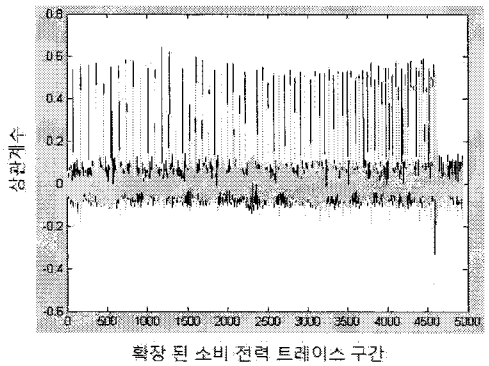
스를 측정하였고 500개의 트레이스를 각각의 사전처리함수를 이용하여 2차 DPA 공격을 한 결과를 표 2에 나타내었다. 표 2에서 확인 할 수 있듯이 두 가지 사전처리함수는 AES 첫 번째 라운드의 16byte의 키 모두를 정확하게 추출 하는데 성공하였다. 그리고 두 사전처리함수의 가장 높은 피크가 뜨는 상관계수를 비교해 보면 16byte의 키 모두에서 제안한 $|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)|$ 함수가 $|t_a-t_b|$에 비해 더 높은 상관계수 값을 나타내는 것을 볼 수 있다. 그리고 [그림 4]은 추정하는 다른 키들에 비해 정확한 키가 가지는 상관계수의 차이가 어느 정도인지를 보여주기 위해 15번째 키 '0x75'(10진수로 '117') 에 대한 상관계수 그래프를 그린 것이다. 회색 그래프는 $|t_a-t_b|$ 함수에 대한 것이고 검은색 그래프는 $|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)|$ 함수에 대한 것이다. 표 2의 결과와 같이 정확한 키에 해당하는 '0x75' 부분에서 검은색 그래프가 회색 그래프보다 0.08 정도가 높은 것을 확인할 수 있다. [그림 5]는 15번째 키 '0x75'에 대한 확장된 전체 트레이스 상에서 상관계수를 다 표현한 것이다. 여기서도 검은색 그래프가 $|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)|$ 함수의 결과이고 회색 그래프는 $|t_a-t_b|$ 함수의 결과이다. 전체적으로 검은색 그래프가 회색 그래프 보다 전체 시간 영역에서 상관계수 값들이 높게 나타남을 확인할 수 있다.

표 2 사전처리함수에 따른 2차 DPA 공격 결과
(500개 소비 전력 트레이스)

키 값	사전처리함수	상관계수	공격성공
1번째 키 '0x49'	$\|t_a-t_b\|$	0.3371	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.3401	O
2번째 키 '0x38'	$\|t_a-t_b\|$	0.5494	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.5778	O
3번째 키 '0x47'	$\|t_a-t_b\|$	0.5923	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6316	O
4번째 키 '0x56'	$\|t_a-t_b\|$	0.5606	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6272	O
5번째 키 '0x65'	$\|t_a-t_b\|$	0.5835	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.5909	O
6번째 키 '0x74'	$\|t_a-t_b\|$	0.6109	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6670	O
7번째 키 '0x83'	$\|t_a-t_b\|$	0.5686	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.5783	O
8번째 키 '0x92'	$\|t_a-t_b\|$	0.6114	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6256	O
9번째 키 '0x11'	$\|t_a-t_b\|$	0.5711	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6223	O
10번째 키 '0x20'	$\|t_a-t_b\|$	0.5917	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6275	O
11번째 키 '0x39'	$\|t_a-t_b\|$	0.5708	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6259	O
12번째 키 '0x48'	$\|t_a-t_b\|$	0.6240	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6501	O
13번째 키 '0x57'	$\|t_a-t_b\|$	0.6119	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6140	O
14번째 키 '0x66'	$\|t_a-t_b\|$	0.5635	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6079	O
15번째 키 '0x75'	$\|t_a-t_b\|$	0.5687	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6464	O
16번째 키 '0x84'	$\|t_a-t_b\|$	0.5721	O
	$\|((t_a-t_b)/t_a)+((t_b-t_a)/t_b)\|$	0.6346	O



[그림 4] 전처리함수에 따른 추정하는 키들의 상관계수 값



[그림 5] 전처리함수에 따른 15번째 키 '0x75'에 대한 상관계수 값

4. 결론

본 논문에서는 마스킹 기법에 대한 2차 DPA 공격에 대해 알아보고 이러한 공격을 더 강화시키는 새로운 전처리함수를 제안하였다. 그리고 실험 환경을 구축하여 실제 2차 DPA 공격을 구현 후 전처리함수에 따른 공격 성능을 분석하였으며 실험결과, 같은 수의 트레이스 상에서, 제안한 사전처리함수가 기존의 가장 우수한 성능을 보인 전처리함수에 비해 더 높은 상관계수를 나타냄을 확인하였다. 이로서 유비쿼터스 환경을 구성하는 주요 디바이스들 내부에 저장되어 있는 비밀키 값을 이전보다 더 정확하게 추출할 수 있게 되었고 2차 DPA 공격이 상당히 위협적인 물리적 공격임을 검증하였다.

참 고 문 헌

[1] John Kelsey, Bruce Schneier, David Wagner, and Chris Hall, "Side Channel Cryptanlysis of Product Cipher," Proceedings of ESORICS'98, pp. 97-112, Springer-Verlag, Sep. 1998.

[2] John Kelsey, Bruce Schneier, David Wagner, and Chris Hall, "Side Channel Cryptanlysis of Product Cipher (final version)," in the site, 2000.

[3] P. Kocher, J. Jaffe and B.Jun, "Differential Power Analysis," CRYPTO'99, LNCS 1666, pp.388-397, Springer-Verlag, 1999.

[4] M. L. Akkar and C. Giraud. "An Implementation of DES and AES, Secure against Some Attacks," In CHES2001, LNCS, vol. 2162, pp. 309-318, Springer-Verlag, 2001.

[5] J. D. Golic and C. Tymen. "Multiplicative masking and power analysis of AES," In CHES2002, LNCS, vol.2523, pp. 198-212, Springererlag, 2002.

[6] T. S. Messerges, "Securing the AES finalists against power analysis attacks, In FSE'00, LNCS 1978, pp. 150-164, Springer-Verlag, 2000.

[7] E. Trichina, D. D. Seta, and L. Germani. "Simplified adaptive multiplicative masking for AES," In CHES'02, LNCS 2535, pp. 187-197, Springer-Verlag, 2003.

[8] J. Blomer, J. Guajardo, and V. Krummel, "Provably secure masking of AES", in Proc. SAC'04, LNCS 3357, pp. 69-83, Springer-Verlag, 2004.

[9] E. Oswald, S. Mangard, and N. Pramstaller, and V. Rijmen, "A side-channel analysis resistant description of the AES S-box," In FSE'05, LNCS 3557, pp. 413-423, Springererlag, 2005.

[10] T. Messerges, "Using Second-Order Power Analysis to Attack DPA Resistant Software," In CHES'00, LNCS 1965, pp. 238-251, Springer-Verlag, 2004.

[11] J. Waddle and D. Wagner, "Towards Efficient Second-Order Power Analysis," In CHES'04, LNCS 3156, pp. 1-15, Springer-Verlag, 2004.

[12] M. Joye, P. Paillier, and B. Schoenmakers, "On Second-Order Differential Power Analysis," In CHES'05, LNCS 3659, pp. 293-308, Springer-Verlag, 2005.

[13] S. Chari, C. Jutla, J. Rao, and P. Rohatgi, "Towards Sound Approaches to Counteract Power-Analysis Attacks", "In CRYPTO'99, LNCS 1666, pp. 398-412. Springer-Verlag, 1999.

[14] T. Messerges, "Using Second-Order Power Analysis to Attack DPA Resistant Software," In CHES'00, LNCS 1965, pp. 238-251, Springer-Verlag, 2004.

[15] J. Waddle and D. Wagner, "Towards Efficient Second-Order Power Analysis," In CHES'04, LNCS 3156, pp. 1-15, Springer-Verlag, 2004.