여러 가지 연산에 대한 DPA 공격

김한필*, 엄대현**, 이필중**

포항공과대학교 정보통신대학원*, 포항공과대학교 전자전기공학과**

{feelhk, dhyum, pjl}@postech.ac.kr

DPA attacks on the various operations

Han Pil Kim*, Dae Hyun Yum**, Pil Joong Lee**

GSIT*, Department of EEE**, POSTECH

{feelhk, dhyum, pjl}@postech.ac.kr

요 약

본 논문에서는 여러 가지 연산에 대해서 Hamming weight를 이용한 DPA 공격이 어떻게 가능하고, 그 결과는 어떠한가를 살펴본다. 각 연산에 대해서 먼저 1, 2차 DPA 공격이 어떻게 가능한지를 보인다. 각 연산 별로 얻어지는 결과들을 비교해 보고, 연산들이 DPA 공격에 대해 내부 정보를 얼마나 유출하며, 공격에 대해 안전한지를 알아본다.

I. 서론

최근 들어 비밀키 값에 의존하는 알고리즘의 실행시간이나 전력 소모량의 차이, 전자파 방출량 등을 관찰/분석하여 키 값을 유도해 낼 수 있는 부가채널 공격기법(side channel attacks)들이 개발되어 실질적인 위협으로 부각되고 있다[1, 6]. 그 중에서도 전력 소모량 분석(power analysis)에 의한 공격은 특히 스마트카드나 하드웨어 구현 등과 같이 물리적으로 안전하게 실계된 장치에 대해서도 이에 대한 대책을 구현하지 않는 경우 내장된 비밀키를 간단히 유도해 낼 수 있는 매우 강력하고 실질적인 공격방법이다[3, 5, 7].

1998년, Kocher는 차분 전력 소모량 분석(DPA: Differential Power Analysis) 기법을 제안했다[2]. 이 DPA는 고정된 비밀키가 개입된 암호 연산의 서로 다른 데이터에 대한 다수의 수행과정에서 비밀키의 특정 비트 값에 의존하는 중간 계산 값과 해당 전력 소모량 사이의 상관관계를 통계적으로 분석하는 방법으로, 강력한 공격 기법으로 알려져 있다.

T. S. Messerges는 간단한 전력 누설 모델을 가정하고, 입력 데이터와 비밀키의 XOR 연산이 Hamming weight를 기반으로 한 DPA 공격에 취약함을 보이고, 이에 대한 대응책을 제안하였다[4]. 그리고 이 대응책에 대해서 2차 DPA 공격이 가능함도 보였다.

본 논문에서는 이러한 공격기법에 대해 간단히 살펴보고, 이 공격기법이 다른 연산(+ mod 2^N, AND, OR)에 대해서 어떻게 적용되는지 간단한 전력 누설 모델을 통하여 알아본다. 각 연산별 결과를 비교해 보고, 연산들이 DPA 공격에 대해 내부 정보를 얼마나 유출하며, 공격에 대해 안전한지를 알아본다.

1. 정의

Kocher가 정의한 고차 DPA 공격은 단일 전력 정보 내에서 하나 이상의 표본을 조합하는 방법이다. 1차 DPA 공격에서 공격자는 전력 소모 신호를 관찰하고, 각 표본 시간에 따라 신호의 개별적인 통계적 특성을 분석하는 반면, 고차 DPA 공격에서는 다수의 표본 시간에서 전력 소모 신호의 공통된 통계적 특성을 계산한다. n차 DPA의 정의는 아래와 같다.

정의 1. n차 DPA 공격은 전력 소모 신호에서, n개의 다른 중간 값에 해당하는 각기 다른 n개의 표본을 사용한다.

2. 전력 누설 모델

본 논문에서 소개되는 공격에 대해서, 연산 장치는 계산되는 데이터의 Hamming weight 정보를 누출하는 것으로 가정한다. 또한, 높은 Hamming weight를 가지는 데이터의 계산이 전력 소모가 크고, 이들은 거의 선형적인 관계를 가지는 것으로 가정한다. 특정 시간 j에서의 전력 소모를 $P[j]$로 표현하자. $P[j]$는 세 부분으로 나눌 수 있는데, 첫 번째는 데이터의 Hamming weight에 따라 변하는 전력 소모를 나타낸다. 두 번째는

일정한 부가 부분이며, 세 번째는 노이즈를 나타낸다. 따라서 $P[j]$는 다음과 같은 선형관계를 가진다.

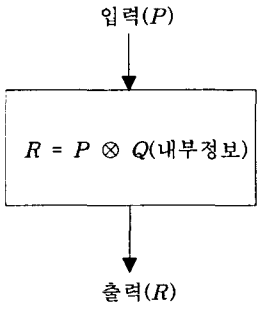
$$P[j] = \varepsilon \cdot d[j] + L + n \quad (1)$$

여기서 $d[j]$는 j에서 중간 결과 값의 Hamming weight를 나타내고, ε은 Hamming weight '1'에 따라 증가하는 전력 소모량이다. L은 일정한 부가 부분이며, n은 노이즈이다. 이 노이즈는 평균치를 '0'으로 가정하여, 충분한 통계 결과를 사용할 때 무시할 수 있다.

II. 본문

1. 1차와 2차 DPA의 비교[4]

1차와 2차 DPA 공격의 이해를 돕기 위해서 [그림 1]과 같은 간단한 모델을 이용하는 것이 유용하다. 이 모델에서는 입력(P)과 내부 값(Q)에 대해 연산을 수행하여 출력(R)을 내고, 출력의 Hamming weight 정보를 누출하게 된다.



[그림 1] DPA 공격을 위한 일반적인 모델

1. 1차 DPA 공격

[그림 1]의 시스템에서 XOR 연산자가 사용될 때 가능한 DPA 공격은 아래 알고리즘과 같이 나타낼 수 있다.

[알고리즘 1]. 전력 소모와 Hamming weight 사이에 선형관계가 있는 N비트 처리장치에서 이 암호시스템이 실행될 때, 다음 DPA 공격이 가능하다.

1. for i=0 to N-1 {
2. for b=0 to 1 {
3. 평균 신호 전력 $A_b[j]$ 계산 {
4. 입력 데이터의 i번째 비트를 b로 설정
5. 나머지 비트는 랜덤 값으로 설정
6. 알고리즘의 전력 신호 수집 } }
7. DPA 바이어스 신호 $T[j]=A_0[j]-A_1[j]$ 계산
8. $T[j]$는 내부 정보의 비트가 1일 때 양의 값을 가지는 파형으로 치우칠 것이고, 0일 때는 음의 값을 가질 것이다. }

분석. 출력 값이 계산될 때에 해당하는 시간을 표본 시간 j^*로 두자. 식(1)을 이용하여 이때의 전력소비를 표현하며, 이때 d는 출력의 Hamming weight값이다. 입력 데이터와 내부 값의 i번째 값을 각각 p_i와 q_i로 두면, Hamming weight의 기대값은 다음과 같다.

$$E[d \mid p_i \oplus q_i = 0] = \frac{N-1}{2}$$

$$E[d \mid p_i \oplus q_i = 1] = \frac{N+1}{2}$$

q_i=0일 때, $A_0[j^*]$와 $A_1[j^*]$에 대한 식은 P의 기대값으로 나타내어진다.

$$A_0[j^*] \approx E[P \mid p_i=0,\ q_i=0] = \frac{N-1}{2}\varepsilon + L \quad (2)$$

$$A_1[j^*] \approx E[P \mid p_i=1,\ q_i=0] = \frac{N+1}{2}\varepsilon + L \quad (3)$$

(2)와 (3)의 차를 구하면,

$$T[j^*] = A_0[j^*] - A_1[j^*] \approx -\varepsilon \quad (4)$$

q_i=1일 때도 유사한 방법을 이용하면,

$$T[j^*] = A_0[j^*] - A_1[j^*] \approx \varepsilon \quad (5)$$

(4)와 (5)로부터, q_i=1일 때 양의 값을 가지는 파형으로 치우치고, q_i=0일 때는 음의 값을 가진다.

2. 2차 DPA 공격

위와 같은 연산(앞에서는 XOR)을 수행하기 전에 임의의 값을 먼저 연산해 주면 [알고리즘 1]의 공격을 받지 않는다. 이것은 바로 연산을 수행하지 않고, RandomMask 값을 먼저 생성해서 계산하므로, 내부의 값이 사용되더라도 결과의 Hamming weight가 임의의 값이 되기 때문이다. 따라서 시스템의 전력 소비는 내부 값이나 입력 데이터의 값과 상관관계가 줄어든다. 이 방법은 1차 공격에는 안전하지만, 2차 DPA 공격에는 취약할 수 있다.

[알고리즘 2].

1. for i=0 to N-1 {
2. for b=0 to 1 {
3. 평균 통계치 $S_b = |P_B - P_C|$ 계산{
4. 입력 데이터의 i번째 비트를 b로 설정
5. 나머지 비트는 랜덤 값으로 설정
6. 랜덤 값 생성 시(B)와 후 연산 시(C)의 전력 소비 정보를 수집, 각각 P_B, P_C로 정함} }
7. DPA 바이어스 통계치 $T = S_0 - S_1$ 계산

8. T>0이면 내부 정보의 i번째 비트는 1이고, 아니면 0이다.)

분석. B와 C에서 전력소모는 식(1)을 이용하면, 다음과 같다.

$$P_B = d_B \cdot \varepsilon_B + L_B \text{and } P_C = d_C \cdot \varepsilon_C + L_C$$

계산을 간단히 하기 위해서, B와 C의 L과 ε값을 같다고 가정하자. 따라서 [알고리즘 2]의 바이어스 통계치는 다음과 같다($\varepsilon=\varepsilon_B=\varepsilon_C$).

$$S_b = | P_B - P_C | = \varepsilon | d_B - d_C |$$

[알고리즘 2]에서 p_i와 q_i는 전과 같고, m_i는 RandomMask 비트를 나타낸다. 따라서, d_B의 기대값은 m_i에, d_C는 p_i, q_i, m_i에 의존한다.

$$E[d_B | m_i = 1] = E[d_C | m_i \oplus p_i \oplus q_i = 1] = \frac{N+1}{2} \quad (6)$$

$$E[d_B | m_i = 0] = E[d_C | m_i \oplus p_i \oplus q_i = 0] = \frac{N-1}{2} \quad (7)$$

따라서 q_i=0일 때, S_0은 다음과 같다.

$$S_0 = \frac{1}{2}[\varepsilon | d_B - d_C | \; | \; m_i = p_i = q_i = 0] +$$

$$\frac{1}{2}[\varepsilon | d_B - d_C | \; | \; m_i = 1, \; p_i = q_i = 0] = 0$$

또한 유사한 방법으로 S_1은 ε이 된다. 그리고 두 값의 차이 T는 다음과 같다.

$$T = S_0 - S_1 = -\varepsilon$$

q_i=1일 때도 비슷한 방법으로 계산이 가능하며, 그 결과는 반대로 되어, S_0은 ε, S_1은 0이 된다. 따라서 이때 T는 ε이 된다. 즉, T의 부호가 내부 값의 비트를 지시한다.

2. 다른 연산에 대한 적용

1. (+ mod 2^N)에 대한 DPA 공격

앞장에서 제안한 XOR에 대한 공격을 (+ mod 2^N)에 대해서도 적용이 가능하다. (+ mod 2^N) 연산에 1차 공격을 적용해보면, 같은 결과를 얻을 수 있다. 즉, q_i=1일 때 양의 값을 가지고, 0일 때는 음의 값을 가진다. 이에 대한 대응책으로 사용되는 RandomMask 방법에 다시 2차 공격을 적용해보면, 다시 같은 결과를 얻을 수 있다. q_i=1일 때는 T가 ε이 되고, q_i=0일 때는 T가 $-\varepsilon$값을 가짐을 알 수 있다. 따라서 (+ mod 2^N) 연산에 대해서도 Messerges의 방법과 같은 방법으로 DPA 공격이 가능함을 알 수 있다. 계산 과정은 XOR과 유사하므로 생략한다.

2. AND 연산에 대한 DPA 공격

내부 연산을 AND 연산으로 생각해보자. 1차 DPA 공격을 하기 위해, 마찬가지로 [알고리즘 1]을 이용해서 기대값을 구하면 다음과 같다.

$$E[d | p_i \& q_i = 0] = \frac{N-1}{2} \quad (8)$$

$$E[d | p_i \& q_i = 1] = \frac{N+1}{2} \quad (9)$$

q_i=0일 때, b값에 따른 전력 신호는 식(2)과 같은 값이 나온다.

$$A_0[j^*] = A_1[j^*] = \frac{N-1}{2}\varepsilon + L$$

따라서 두 값의 차이는 T = 0이다.

q_i=1일 때, b값에 따른 P의 기대값은

$$A_0[j^*] \approx E[P | q_i = 1, \; p_i = 0] = \frac{N-1}{2}\varepsilon + L$$

$$A_1[j^*] \approx E[P | q_i = 1, \; p_i = 1] = \frac{N+1}{2}\varepsilon + L$$

이고, 두 값의 차이 T는 $-\varepsilon$이 된다.

다시 RandomMask에 2차 공격을 해보자. 이 때의 기대값은 식(6), (7)과 같은 값을 가진다(($m_i \oplus p_i$)&q_i만 다름). q_i=0일 때, S_0은 다음과 같다.

$$S_0 = \frac{1}{2}[\varepsilon | d_B - d_C | \; | \; m_i = p_i = q_i = 0] +$$

$$\frac{1}{2}[\varepsilon | d_B - d_C | \; | \; m_i = 1, \; p_i = q_i = 0] = \frac{\varepsilon}{2}$$

비슷한 방법으로 S_1은 $\varepsilon/2$가 됨을 알 수 있다.

$$T = S_0 - S_1 = 0 (when \; q_i = 0)$$

q_i=1에 대해서 S_0은 0, S_1은 ε값을 가진다.

$$T = S_0 - S_1 = -\varepsilon (when \; q_i = 1)$$

내부 정보의 비트에 따른 결과에서 보듯이 앞의 두 연산과는 다른 결과를 가짐을 알 수 있다.

3. OR 연산에 대한 DPA 공격

OR연산에 대해서도 1, 2차 DPA 공격이 가능한데, 앞의 계산 과정과 비슷한 방법으로 할 수 있다. 1차 DPA 공격에서 두 기대값은 식(8), (9)와 같다. q_i=0일 때의 두 전력신호의 기대값과 그 차는 아래와 같다.

$$A_0[j^*] = \frac{N-1}{2}\varepsilon + L, \; A_1[j^*] = \frac{N+1}{2}\varepsilon + L$$

$$T[j^*] = A_0[j^*] - A_1[j^*] \approx -\varepsilon (when \; q_i = 0)$$

q_i=1일 때는 T=0이다.

$$A_0[j^*] = A_1[j^*] = \frac{N+1}{2}\varepsilon + L$$

$$T[j^*] = A_0[j^*] - A_1[j^*] = 0 (when \; q_i = 1)$$

역시 2차 DPA 공격에 대해서도 같은 결과를 나타낸다.

$$T = S_0 - S_1 = -\varepsilon (when\ q_i = 0)$$

$$T = S_0 - S_1 = 0 (when\ q_i = 1)$$

이 결과는 앞의 AND 연산의 결과와 반대의 값을 가짐을 알 수 있다.

III. 결론

본 논문에서는 여러 가지 연산에 대한 DPA 공격과 그 결과를 살펴보았다. XOR과 (+ mod 2^N)에 대한 1, 2차 DPA 공격에 대하여 두 전력 신호의 차이는 비트에 따라서 양의 값과 음의 값으로 각각 ε만큼 바이어스 됨을 볼 수 있었다. 이 값을 관찰함으로서 시스템의 내부에 있는 정보의 비트 값을 알 수 있다. 그리고 두 연산 AND와 OR에 대해서는 한 값에 대해서만 음의 값으로 바이어스 된 결과를 볼 수 있었다. 즉, 한쪽만이 바이어스 됨으로서 ε만큼의 격차를 나타내고, 이것은 XOR이나 (+mod 2^N)의 경우 격차가 2ε인데 비해서 누출되는 정보의 양이 반으로 줄어든 것이다. 이러한 결과로 볼 때, AND와 OR 연산이 DPA 공격만을 고려할 때는, XOR과 (+mod 2^N) 연산보다는 강한 것을 알 수 있다. 그러나 실제 시스템 설계 시 연산의 선택은 다른 다양한 요소들을 고려하여 결정하여야 한다.

참고문헌

[1] J. Kelsey, B. Schneier, D. Wagner, and C. Hall, "Side Channel Cryptanalysis of Product Ciphers," ESORICS'98, pp. 97-110, 1998.

[2] P. Kocher, J. Jaffe, and B. Jun, "Differential Power Analysis," CRYPO'99, pp. 388-397, 1999.

[3] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Power Analysis Attacks of Modular Exponentiation in Smartcards," CHES'99, LNCS 1717, pp. 144-157, 1999.

[4] T. S. Messerges, "Using Second-Order Power Analysis to Attack DPA Resistant Software," CHES 2000, pp. 238-251, 2000.

[5] T. S. Messerges, E. A. Dabbish, R. H. Sloan, "Examining Smart-Card Security under the Threat of Power Analysis Attacks," IEEE Transactions on Computers, Vol. 51, No. 5, pp. 541-552, 2002.

[6] J. Muir, "Techniques of Side-Channel Cryptanalysis," Master Thesis, dept. Math, Univ. of Waterloo, 2001.

[7] 임채훈, "부가채널 공격에 안전한 효율적인 타원곡선 상수배 알고리즘", 정보보호학회논문지, 제12권, 제4호, pp. 99-114, 2002.