지터에 내성을 갖는 딥러닝 기반 부채널 분석 방안

김주환**, 김수진*, 우지은*, 박소연*, 한동국*, ***
*국민대학교 정보보안암호수학과
**국민대학교 수학과
***국민대학교 금융정보보안학과
zzzz2605@kookmin.ac.kr, suzin22@kookmin.ac.kr, dnwldms928@kookmin.ac.kr,
soyeonp@kookmin.ac.kr, christa@kookmin.ac.kr

Deep Learning-based Side-Channel Analysis Method with Resistance to Jitter

Ju-Hwan Kim**, Soo-Jin Kim*, Ji-Eun Woo*, So-Yeon Park*, Dong-Guk Han*, ***
*Dept. of Information Security, Cryptology, and Mathematics, Kookmin University
**Dept. of Mathematics, Kookmin University
***Dept. of Financial Information Security, Kookmin University

요 약

물리적 정보를 이용해 암호 알고리즘의 비밀정보를 분석하는 부채널분석 분야에서도 딥러닝을 접목한 분석방법들이 활발히 제안되고 있다. 본 논문에서는 소비전력이 시간축상으로 흐트러지는 현상인 지터가 있는 파형을 신경망의 특성을 기반으로 효과적으로 분석하는 방법을 제안한다. 제안한 방법을 실험적으로 검증하기 위해 지터가 있는 AES-128 파형을 Convolutional Neural Network와 Multi-Layer Perceptron을 기반으로 분석한 결과 제안한 방법을 적용한 신경망은 모든 바이트 키 분석에 성공했으나, 이외의 신경망은 일부 혹은 모든 바이트 키 분석에 실패했다.

1. 서론

부채널분석(Side-Channel Analysis)이란 암호 알고리즘이 실제 디바이스에서 동작할 때 발생하는 전력이나 전자파와 같은 부채널 정보를 이용하여 비밀정보를 분석하는 방법이다[1]. 그중 대표적인 분석방법인 전력분석공격은 디바이스가 동작할 때의 소비전력을 이용하여 비밀정보를 분석한다[2]. 따라서 공격자는 데이터와 소비전력 사이의 관계를 찾거나 소비전력에서 유의미한 분석 지점인 PoI(Point of Interest)를 찾아내야 하는 등 공격자에게 높은 분석 능력이 요구된다.

최근 딥러닝이 여러 분야에 활용되면서 부채널분석 분야에서도 딥러닝을 접목한 공격방법들이 연구되고 있다. 프로파일링 환경에서 딥러닝 기반 부채널분석은 공격자가 평문과 비밀키를 알고 있는 프로파일링 장비를 확보했다는 가정 하에서의 공격이다. 비밀키를 알고 있다면 파형에 대응되는 중간값을 계산할 수 있으므로, 신경망이 파형으로부터 중간값을 예측하도록 학습시킬 수 있다. 공격자는 학습된 신경망을 이용하여 공격 대상의 소비전력으로부터 중간값을 복구해 비밀정보를 찾을 수 있다. 딥러닝은 데이터의 특징을 알아서 학습하므로 전통적인 전력분석공격에 비해 공격자의 분석 능력에 크게 의존하지 않는다.

실제 부채널 정보에는 데이터가 시간축상으로 흐트러지는 현상인 지터가 있거나, 노이즈가 발생하는 문제가 있으므로 데이터의 특성에 맞게 적합한 신경망을 채택하는 것이 중요하다. 데이터의 이동에 민감한 MLP(Multi-Layer Perceptron)의 경우 별도의 전처리를 수행하지 않으면 부채널분석에 활용하기에 적합하지 않다. 반면, CNN(Convolutional Neural Network)은 컨볼루션층과 풀링층의 연산적 특성 덕분에 변동을 감내할 수 있으므로 CNN을 활용하면 지터가 있는 파형을 효과적으로 분석할 수 있다. 따라서 본 논문에서는 CNN을 활용해 지터에 내성을 갖는 신경망을 구성하는 방안을 제안하고 이를 실험적으로 검증한다.

본 논문의 구성은 다음과 같다. 2장에서는 부채널분석과 신경망을 설명하고 3장에서는 지터에 내성을 갖는 신경망 구성 방안을 제안한다. 4장에서는 실험을 통해 제안한 방법을 검증한다. 마지막으로 5장에서 결론을 제시하며 마친다.

2. 관련 연구

2.1 프로파일링 공격

전력분석공격이란 암호 알고리즘이 디바이스에서 동작할 때의 소비전력과 중간값의 관계를 이용하여 비밀정보를 알아내는 공격 방법이다. 프로파일링기반의 전력분석공격은 공격자가 사전에 공격 대상과 동일하며 비밀키, 평문, 소비전력을 알고 있는 장비를 조작할 수 있다고 가정한다. 프로파일링 공격은 다음과 같이 학습 단계와 공격 단계로 나눌 수 있다.

가. 공격자는 프로파일링 장비의 비밀키를 알고 있으므로 소비전력에 대응되는 중간값을 계산할 수 있다. 이를 통해 신경망이 소비전력을 입력받아 중간값을 예측하도록 학습시킨다.

나. 학습된 신경망을 이용해 공격 대상의 소비전력으로부터 중간값을 계산할 수 있다. 공격자는 평문과 계산한 중간값을 이용해 비밀키를 찾을 수 있다.

예를 들어, AES 암호 알고리즘[3]에 대한 비밀키 복구 방안은 다음과 같다. 중간값을 SubBytes 변환의 출력으로 했을 때, T, M, I, P, K, S를 각각 소비전력, 신경망, 중간값, 평문, 비밀키, SBox라 하자. 공격자는 T, M, I, P를 알고 있고 K를 찾아야 한다. 미리 학습시킨 M을 이용하면 T로부터 비밀키와 관련된 중간값 $I = M(T)$를 계산할 수 있다. 이때 $I = S[P \oplus K]$이므로 공격자는 다음 수식을 이용해 비밀키를 찾을 수 있다.

$$K = S^{-1}[M(T)] \oplus P$$

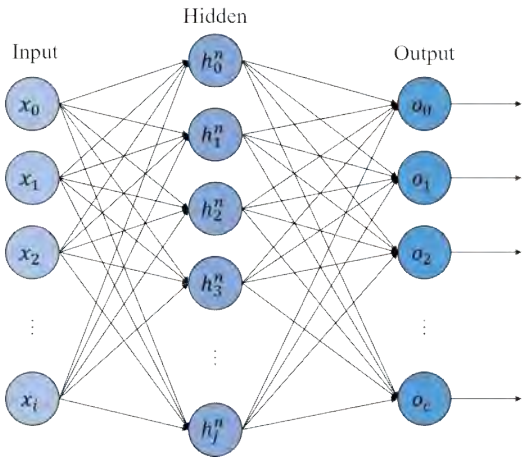
신경망이 잘못된 중간값을 예측할 수 있으므로 여러 파형에 대한 중간값을 예측한 뒤 가장 많이 예측된 키를 비밀키라 추정한다. 이때 신경망의 성능을 측정하기 위해 비율 $ratio$를 정의한다.

$$ratio = \frac{(\text{옳은 키의 수})}{(\text{틀린 키 중 가장 많이 나온 키의 수})}$$

분석결과 예측한 키가 비밀키라면 옳은 키의 수가 가장 많아야 하므로 $ratio$는 1보다 커야한다.

2.2 Muti-Layer Perceptron

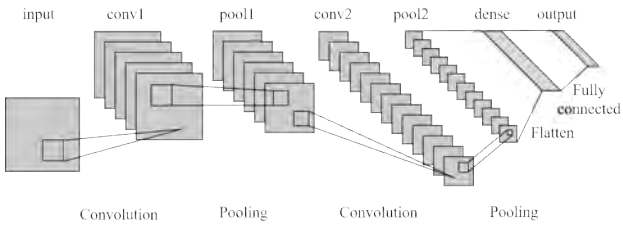
퍼셉트론(Perceptron)이란 인공 신경망의 한 구조로 고차원의 데이터를 입력받아 가중치 벡터를 곱한 후 비선형함수인 활성 함수를 통과시켜 일차원 데이터를 출력한다[4]. 이러한 퍼셉트론을 여러 개 결합하여 비선형적인 데이터에 대해서도 분류가 가능하도록 하는 Multi-Layer Perceptron(MLP)이 제안되었다[5]. MLP는 (그림 1)과 같이 입력층, 은닉층, 출력층 총 세 가지 계층으로 이루어져 있다. 은닉층에서는 데이터의 특징을 학습한다. 활성 함수로는 Sigmoid, Tanh, ReLU 등을 사용한다. MLP는 각 벡터의 원소마다 다른 가중치를 곱하므로 특징 벡터의 이동에 민감한 특성이 있다.



(그림 1) MLP의 구조

2.3 Convolutional Neural Network

Convolutional Neural Network(CNN)은 영상처리나 신호처리 분야에서 일반적으로 사용하는 신경망이다[6]. CNN은 (그림 2)와 같이 컨볼루션 연산과 풀링 연산을 이용해 특징 맵을 생성하고 마지막에 완전연결층을 통해 데이터를 분류한다. 컨볼루션 연산은 이동에 동변이고 풀링 연산은 특징 벡터를 압축하므로 MLP와 달리 CNN은 이동에 내성을 갖는다.



(그림 2) CNN의 구조

3. 신경망의 특성에 기반한 지터에 내성을 갖는 신경망 구성 방안

본 절에서는 신경망의 특성을 기반으로 지터가 심한 파형에 내성을 갖는 신경망 설계 방법을 제안한다.

MLP는 특징 벡터의 차원별로 가중치를 가지므로 데이터의 이동이 심하고 데이터가 적으면 데이터의 분포를 제대로 학습할 수 없다. 반면, CNN의 컨볼루션 연산은 이동에 동변이고, 풀링 연산은 특징 벡터를 압축하므로 CNN은 데이터의 이동을 감내할 수 있다. 따라서 CNN을 이용하면 지터가 심한 파형을 다른 신경망에 비해 효과적으로 데이터의 분포를 학습할 수 있다.

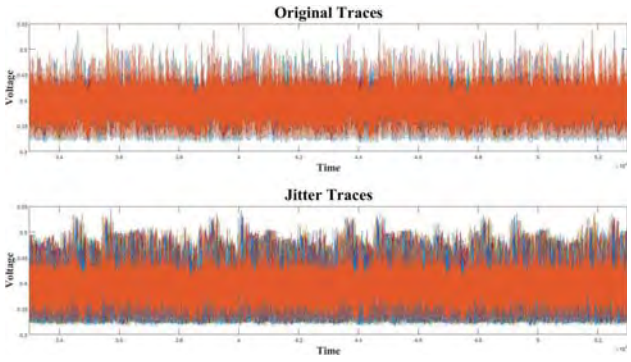
컨볼루션 연산과 풀링 연산은 특징 벡터의 이동에 영향을 받지 않지만, 완전연결 층은 특징 벡터의 이동에 영향을 받는다. 따라서 지터에 내성을 갖는 신경망을 구성하기 위해서는 충분히 많은 풀링층을 거치도록 구성해야 한다. 즉, 특징 벡터가 이동하더라도 마지막 풀링층에서는 유사한 특징 벡터가 나오도록 구성해야 한다. 풀링층의 커널의 크기를 p라 했을 때, 보폭이 커널의 크기와 같다면 풀링층을 한 번 거칠 때마다 p차원 정보가 1차원으로 요약된다. 유사하게 신경망의 층의 수를 l이라 했을 때, 마지막 풀링 결과 신경망의 입력의 p^l 차원의 정보가 1차원으로 집약된다. 수집된 파형의 위상의 차의 최댓값을 z라 하면, 즉 동일한 데이터와 관련된 파형 정보가 최대 z 포인트 이동한다면, 해당 정보와 관련된 특징 벡터가 마지막 풀링 결과 1차원으로 집약되기 위해서는 수식 (1)과 같은 관계를 만족하도록 신경망을 구성해야 한다.

$$p^l > z \Leftrightarrow l > \log_p z \quad \cdots \quad (1)$$

4. 실험 결과

본 장에서는 MLP와 CNN을 기반으로 지터가 있는 파형에 대한 프로파일링 분석을 수행한다. 실험은 TrusThings에서 제공하는 공개 데이터셋을 활용했다. 해당 데이터는 AVR ATmega 128 (8-bit 프로세서)에서 부채널 대응기법이 적용되지 않는 AES-128 암호 알고리즘이 동작할 때의 소비전력을 SCARF 전력수집 보드에서 수집한 것이다. 학습을 위한 임의의 키로 암호화를 5,000번 수행할 때의 데이터, 공격을 위한 고정된 키로 암호화를 2,000번 수행할 때의 데이터로 나누어져 있다. 신경망은 Tensor Flow 2.0.0 버전을 백엔드로 하는 Keras 2.3.1 버전을 이용해 구현하였다.

TrusThings의 파형은 지터가 거의 없는 파형이므로 실험을 위해 임의로 파형의 위치를 변환한다. 우리는 집합 $[-250, 250]$에서 균등분포로 임의의 정수를 선택한 뒤, 선택된 정수만큼 파형을 이동시켰다. (그림 3)은 각각 원본 파형과 변환된 파형을 나타낸다.



(그림 3) TrusThings의 원본 파형과 변형된 파형

본 실험에서는 7개의 CNN와 2개의 MLP로 분석한 결과를 비교한다. 실험한 CNN은 신경망의 입력, 컨볼루션 출력, 완전연결층 사이에 배치정규화를 수행한다. 신경망의 컨볼루션층(C), 풀링층(P)의 커널의 수와 완전연결층(FC)의 노드의 수는 다음과 같다.

$$[\mathrm{CNN}-2^i] := C(2^1) - P(2^1) - C(2^2) - P(2^2) - \cdots - C(2^i) - P(2^i) - FC(2^9) - FC(2^8) \ (5 \le i \le 11)$$

신경망을 위와 같이 구성한 이유는 모델의 용량을 유사하게 하기 위함이다. CNN의 전체 파라미터 중 대부분은 완전연결층과 관련된 파라미터이므로 모델의 용량을 유사하게 만들기 위해서는 평탄화된 특징 벡터의 차원이 비슷해야 한다. 위의 신경망은 풀링으로 인해 압축되는 비율만큼 커널의 개수가 증가하므로 평탄화된 특징 벡터의 차원이 유사하다.

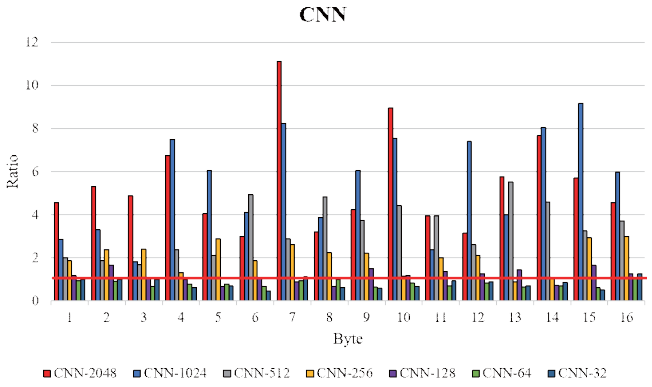
CNN의 컨볼루션 커널의 크기는 3, 풀링 커널의 크기는 2로 지정했다. 각 [CNN-2^i]는 풀링 연산을 i번 수행하므로 수식 (1)의 $2^i > 500$를 만족하려면 i는 9보다 커야한다.

MLP는 신경망의 입력, 완전연결층 출력에 배치정규화를 수행한다. 각 층의 노드의 수는 다음과 같다.

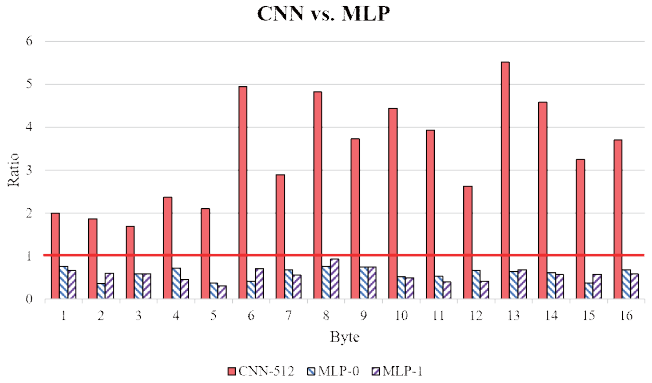
$[\mathrm{MLP}-0] := 2^{10}-2^{10}-2^{9}-2^{9}-2^{8}-2^{8}-2^{8}-2^{8}$
$[\mathrm{MLP}-1] := 2^{9}-2^{9}-2^{9}-2^{9}-2^{8}-2^{8}-2^{8}-2^{8}$

CNN과 MLP는 모두 최적화함수는 Adam, 손실함수는 categorical cross entropy, 활성함수는 ReLU를 사용하며, 레이블은 바이트별 SubBytes 출력값이다. 데이터는 총 4,500개의 학습파형과 500개의 검증파형를 사용하여 50 에포크만큼 학습했다.

(그림 4)는 CNN의 구조별 *ratio*를 도식화한 것이다. 수식 (1)을 만족하는 신경망 [CNN-2048], [CNN-1024], [CNN-512]는 모든 바이트 키 분석에 성공했지만, [CNN-256]은 16바이트 중 15바이트 비밀키만 복구했고, [CNN-128]은 11바이트, [CNN-64는 1바이트, CNN-32]는 2바이트 비밀키를 복구했다.



(그림 5)는 수식 (1)을 만족하는 가장 얕은 신경망인 [CNN-512]와 두 구조의 MLP의 *ratio*를 비교한 것이다. 특징 벡터의 이동에 민감한 MLP는 모든 바이트에서 키 분석에 실패했지만, CNN은 전체 비밀키를 복구했다. 이는 데이터의 특성을 기반으로 적합한 신경망을 채택하는 것이 분석의 성공 여부를 결정할 정도로 중요함을 시사한다.



(그림 5) [CNN-512]와 MLP 구조별 실험 결과

5. 결론

본 논문에서는 프로파일링 환경에서 파형이 흔들려서 나타나는 노이즈, 즉 지터가 있는 파형에 대한 효과적인 분석방법을 제안했다. 이를 실험적으로 보이기 위해 다양한 구조의 CNN, MLP로 지터가 있는 파형을 분석한 결과, 제안된 조건을 만족하는 CNN은 모든 바이트 키 분석에 성공했지만, 그렇지 않은 신경망과 MLP는 분석에 실패했다. 따라서 제안된 조건을 이용하면 지터가 있는 파형을 효과적으로 분석할 수 있다.

사사

본 연구는 고려대 암호기술 특화연구센터(UD170109ED)를 통한 방위사업청과 국방과학연구소의 연구비 지원으로 수행되었습니다.

참고문헌

[1] Kocher, Paul, Joshua Jaffe, and Benjamin Jun. "Differential power analysis." Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 1999.

[2] Mangard, Stefan, Elisabeth Oswald, and Thomas Popp. Power analysis attacks: Revealing the secrets of smart cards. Vol. 31. Springer Science & Business Media, 2008.

[3] Brier, Eric, Christophe Clavier, and Francis Olivier. "Correlation power analysis with a leakage model." International workshop on cryptographic hardware and embedded systems. Springer, Berlin, Heidelberg, 2004.

[4]Rosenblatt, Frank. "The perceptron: a probabilistic model for information storage andorganization in the brain." Psychological review 65.6 (1958): 386.

[5] R. Collobert and S. Benjio, "Links between perceptrons, MLPs and SVMs," Proceedings of the twenty-first international conference on Machine learning, ICML'04, 2004

[6] O'Shea, Keiron, and Ryan Nash. "An introduction to convolutional neural networks." arXiv preprint arXiv:1511.08458 (2015).