독립된 데이터셋을 활용한 효율적인 딥러닝 기반 비프로파일링 부채널 분석 방안

김주환**, 문혜원*, 김연재*, 박아인*, 한동국*, ***
*국민대학교 정보보안암호수학과
**국민대학교 수학과
***국민대학교 금융정보보안학과
zzzz2605@kookmin.ac.kr, qwerty25879@kookmin.ac.kr, duswo0024@kookmin.ac.kr,
applepai28@kookmin.ac.kr, christa@kookmin.ac.kr

Efficient Non-Profiled Deep Learning-based Side-Channel Analysis with Independent Dataset

Ju-Hwan Kim**, Hye-Won Mun*, Yeon-Jae Kim*, A-In 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

비프로파일링 부채널 분석은 프로파일링 장비가 없는 환경에서 부채널 정보를 이용해 비밀정보를 분석하는 방법이다. 기존에 알려진 Timon의 비프로파일링 분석은 학습 데이터 집합만을 이용해 공격하므로 전력 파형의 수가 제한된다면 과적합이 발생하여 키 분석 성능이 떨어질 수 있다. 본 논문에서는 비프로파일링 환경에서의 딥러닝 기반 부채널 분석 성능을 향상시키기 위해 학습 데이터 집합과 독립적인 검증 데이터 집합을 활용해야 하는 실증적 근거를 제시한다. 이에 대한 실험으로 기존 기법과 제시한 기법의 성능을 비교해 봤을 때, 검증 데이터를 활용하면 더 적은 데이터로 비밀키 추출이 가능함을 보인다.

1. 서론

사물 인터넷 기술이 보편화됨에 따라, 수학적인 안전성과 더불어 사물에 가해지는 물리적인 공격에 대한 안전성도 중요하게 여겨지고 있다. 이때 공격자가 수행할 수 있는 대표적인 물리적인 공격으로는 암호 알고리즘이 동작할 때 발생하는 물리적인 정보(동작 시간, 소비 전력, 전자파 등)를 이용하여 비밀정보를 분석하는 방법인 부채널 분석[1]이 있다.

부채널 분석은 크게 프로파일링 부채널 분석과 비프로파일링 부채널 분석으로 나눌 수 있다. 먼저 프로파일링 부채널 분석은 분석자가 공격 대상 기기와 동일한 기기에 접근할 수 있음을 공격 가정으로 하며, 대표적으로 템플릿 공격[2]과 스토캐스틱 모델[3] 등이 있다. 반면에 비프로파일링 부채널 분석은 공격자가 모르는 비밀키에 대해 암호 알고리즘의 동작 결과만을 얻을 수 있음을 공격 가정으로 하며, 대표적으로 차분 전력 분석[1], 상관 전력 분석[4] 등이 있다. 프로파일링 분석의 경우 비교적 강한 공격 가정이 필요하므로 실제 환경에서의 적용이 현실적으로 제한될 수 있다. 따라서 공격 가정이 완화된 비프로파일링 분석이 많이 연구되고 있다.

최근에는 딥러닝을 이용한 비프로파일링 부채널 분석 방법이 제시되었다. Timon은 딥러닝 학습 결과를 구별자로 이용하여 파형을 라벨에 따라 분류했을 때, 학습 결과가 손실값, 정확도, 가중치의 측면에서 다르게 나타난다는 사실을 통해 비밀키를 분석하는 차분 딥러닝 분석을 제시했다[5]. 그러나 신경망의 용량에 비해 적은 수의 단일 데이터 집합으로 학습시킬 경우, 신경망이 데이터의 분포를 학습하지 못하고 단지 입출력을 외우는 문제가 발생할 수 있다. 따라서 본 논문에서는 학습 데이터 집합과 독립적인 데이터 집합을 활용하여 Timon의 방법보다 더 효과적인 비프로파일링 분석을 제시하고 이를 실험적으로 증명하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 제시한 방안을 이해하기 위한 소비전력모델, 딥러닝과 차분 딥러닝 분석을 소개한다. 3장에서는 독립적인 데이터셋을 사용해야 하는 근거를 제시하고, 4장에서는 실험을 통해 기존 방법[5]과 제시한 방법을 비교한다. 마지막 5장에서는 결론을 제시하면서 마친다.

2. 관련 연구

2.1 소비전력모델

소비전력모델은 소비 전력과 중간값과의 관계를 수학적으로 나타내기 위해 사용되며, 대표적으로 해밍웨이트 모델과 해밍디스턴스 모델이 있다. 해밍웨이트는 데이터의 이진 표현 시 1의 개수를 의미하며 일반적으로 소프트웨어로 구현된 암호 알고리즘은 해밍웨이트 모델을 따른다고 알려져 있다. 아래 수식은 해밍웨이트 모델을 나타낸 것이다.

$$P_{total} = \epsilon \cdot HW(y) + P_{noise}$$

(P_{total} : 총 전력소비, P_{noise} : 잡음, ϵ : 상수, HW : 해밍웨이트, y : 중간값)

2.2 딥러닝

인공 신경망은 사람의 신경망으로부터 영감을 얻어 뇌의 구조를 모델로 삼은 통계학적 학습 알고리즘이다. 층을 겹겹이 쌓은 인공 신경망을 사용하여 학습하는 기계 알고리즘의 집합을 딥러닝[6]이라고 한다. 학습이란 주어진 입력을 정확한 기댓값에 사영시키기 위해 적합한 가중치를 구하는 과정이다. 일반적으로 학습은 손실함수를 통해 신경망의 출력이 기댓값에서 벗어난 정도를 계산한 후, 최적화함수를 통해 손실함수가 감소하는 방향으로 가중치를 반복 수정한다. 손실함수는 평균 제곱 오차, 오차 제곱합, 교차 엔트로피 오차 등이 있고, 최적화함수는 경사하강법, Adam, RMSProp 등이 있다.

2.2.1 다층 퍼셉트론

다층 퍼셉트론(Multi-Layer Perceptron, MLP)[7]은 퍼셉트론으로 이루어진 여러 층을 붙여 놓은 계층구조를 갖는다. 퍼셉트론은 입력에 대해 가중치 곱과 편향의 합을 계산한 후, 활성화함수를 거치는 구조로 이루어진다. 그러나 퍼셉트론의 경우, 선형 분류만이 가능하므로 이를 보완하기 위해 입력층과 출력층 사이에 여러 개의 중간층을 두어 비선형 분류가 가능하도록 MLP를 구성한다. 이때 여러 개의 중간층을 은닉층이라고 부르며, 사용하는 활성화함수로는 Sigmoid, ReLU, Softmax 등이 있다.

2.2.2 일반화 성능

모델의 일반화 성능이란 학습에 사용하지 않은 데이터에 대한 예측의 정확도를 의미한다. 그런데 일괄적인 학습 데이터 집합으로 모델을 훈련하고 평가하는 방식은 모델의 일반화 성능을 보장할 수 없다. 학습 데이터의 양에 제한이 있는 경우, 모델이 데이터의 분포를 학습하지 않고 단순히 학습 데이터를 외워버리는 문제로 인해 과적합이 일어날 수 있기 때문이다. 이때 과적합이란 모델이 실제 데이터의 분포보다 주어진 학습 데이터의 분포에 더 근접하게 학습되는 현상을 의미한다. 즉 학습 데이터만으로는 새로운 데이터에 대해 일반화가 잘 이루어졌는지를 판단할 수 없다. 따라서 학습 데이터 집합을 훈련 데이터 집합과 그와 같은 분포를 가지는 독립적인 검증 데이터 집합으로 랜덤하게 나누어 모델의 일반화 성능을 검증한다[8]. 모델의 일반화 성능을 검증하는 방법에는 홀드아웃, K-겹 교차검증 등이 있다.

2.3 차분 딥러닝 분석

차분 딥러닝 분석(Differential Deep Learning Analysis, DDLA)[5]은 차분 전력 분석을 딥러닝에 적용한 것이다. 공격 대상 기기에서 수집한 파형들을 학습의 입력 데이터로 사용하여 딥러닝 학습을 수행하는데, 이때 모델의 라벨은 중간값의 최상위 비트, 최하위 비트 혹은 해밍웨이트를 사용한다. 학습의 성능 지표로는 손실값, 정확도 혹은 가중치를 이용하는데, 만약 성능 지표로 손실값을 사용했다면 딥러닝 모델이 학습을 잘 수행했다고 가정했을 때 옳은 라벨에 대한 손실값이 틀린 라벨에 대한 손실값보다 작을 것이라고 예상한다. 따라서 가장 작은 손실값을 가지는 라벨을 계산할 때 사용된 키를 비밀키라고 추측할 수 있다.

3. 신경망의 일반화성능 비교를 통한 효율적인 딥러닝 기반 비프로파일링 부채널 분석 방안

본 논문에서는 신경망의 일반화 성능을 비교함으로써 기존 방법보다 더 적은 정보로도 비밀키를 찾는 방안을 제시하고, 논리적 타당성을 실증적으로 보인다.

기존 방법은 하나의 데이터 집합에 대한 학습 결과를 기반으로 추정한 키가 비밀키인지를 검증한다. 이 방법에서는 파형과 추정한 키로 계산한 라벨이 관련 없더라도 신경망이 단순히 입출력을 외워서 손실값이 감소하는 문제가 있다. 즉, 옳은 키와 틀린 키 모두 손실값이 감소한다. 우리는 이를 해소하기 위해 일반화 성능을 기반으로 신경망의 학습 여부를 판정한다. 신경망이 데이터의 분포를 학습하지 못했다면 새로운 데이터에 대해서는 정확한 라벨을 예측

할 수 없으므로 검증 데이터 집합에 대한 손실값은 감소하지 않는다. 반면, 분포를 학습했다면 새로운 데이터에 대한 라벨을 높은 정확도로 예측할 수 있으므로 손실값이 감소한다.

기존 방법은 비밀키를 찾기 위해 신경망이 입출력을 외울 수 없을 정도로 많은 데이터가 필요하다. 한편, 제시한 방법은 분포를 학습할 수 있을 정도의 데이터만 있으면 비밀키를 찾을 수 있으므로 기존 방법보다 더 적은 정보로도 분석이 가능하다.

4. 실험 결과

본 장에서는 기존 방법과 제시된 방법을 비교한다. 실험은 Atmel XMEGA128D4 (8-bit 프로세서) 칩을 사용하는 ChipWhisperer-Lite 전력 수집 보드에서 부채널 대응기법이 적용되지 않은 8-bit 단위 AES-128 암호 알고리즘[9]이 10,000번 동작할 때 발생하는 소비전력을 Sampling Rate 29.538MS/s로 수집하였다. 신경망은 TensorFlow 2.0.0 버전을 백엔드로 하는 Keras 2.3.1 버전을 이용해 구현하였다.

신경망의 입력은 1라운드 SubBytes를 수행할 때의 소비전력 696포인트이고, 출력은 바이트별 1라운드 SubBytes 출력의 해밍웨이트이다. 과적합을 방지하기 위해 은닉층이 1개인 MLP를 구성하였으며, 은닉층은 8개의 노드를 가진다. 학습은 200에포크만큼 수행했다. 제시한 방법에서 학습 데이터 집합과 검증 데이터 집합의 비율은 7 : 3으로 지정했다. 성능 지표로는 Timon이 제시한 손실값, 정확도, 가중치 중 손실값을 활용한다.

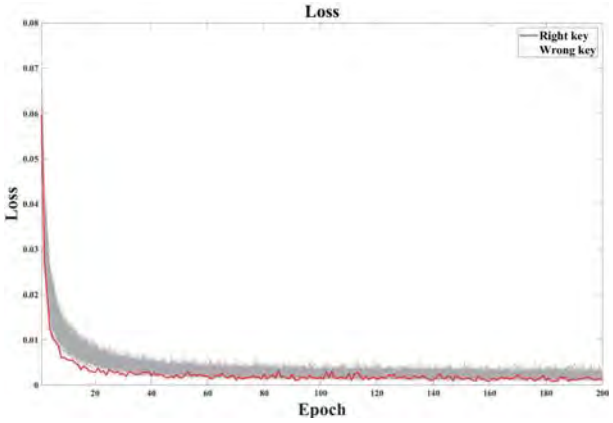
신경망의 목표가 중간값의 해밍웨이트를 찾는 회귀 문제를 해결하는 것이므로 손실함수는 평균 제곱오차를 채택하였으며, 손실값이 가장 낮을 때의 키를 비밀키라고 추정하였다.

학습 결과를 비교하기 위해 $ratio$를 다음과 같이 정의한다.

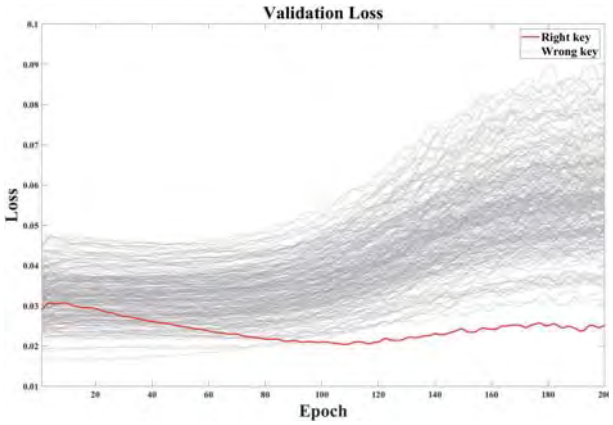
$$ratio = \frac{(\text{틀린 키의 손실값 중 최솟값})}{(\text{옳은 키의 손실값})}$$

올바르게 추측했다면 옳은 키의 손실값이 가장 작으므로 $ratio$는 1보다 커야 한다.

(그림 1), (그림 2)는 200개의 파형을 이용해 첫 번째 바이트 키를 각각 기존 방법, 제시한 방법으로 분석했을 때의 손실값을 나타낸 것이다.



(그림 1) 기존 방법을 이용했을 때의 에포크별 손실값

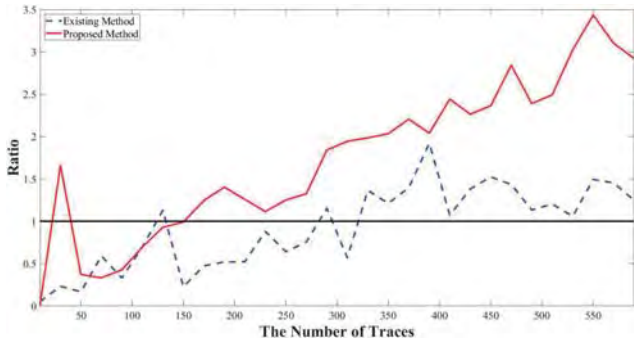


(그림 2) 제시한 방법을 이용했을 때의 에포크별 손실값

기존 방법의 경우 틀린 키를 이용해 라벨을 계산했더라도 신경망이 입출력을 외우기 때문에 모든 키의 손실값이 낮아지며 $ratio \approx 0.77$으로 비밀키 분석에 실패했다.

반면, 제시한 방법은 틀린 키를 이용해 라벨을 계산할 경우 신경망이 데이터의 분포를 학습하지 못하므로 틀린 키의 손실값은 증가하지만, 옳은 키의 손실값은 감소한다. 또한, $ratio \approx 1.22$로 비밀키 분석에 성공했다.

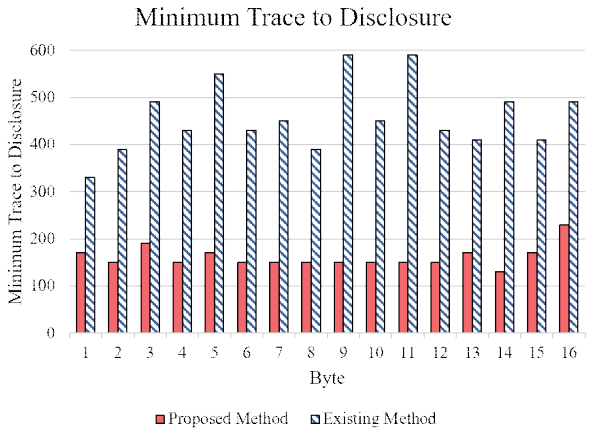
(그림 3)은 데이터 수에 따른 $ratio$의 변화를 나타낸 것이다.



(그림 3) 데이터 수에 따른 *ratio*

기존 방법은 비밀키 분석을 위해 330개 이상의 데이터가 필요하지만, 제시한 방법은 170개의 데이터만으로도 비밀키 분석에 성공했다.

(그림 4)는 바이트별 비밀키 분석을 위해 필요한 파형의 수를 도식화한 것이다. 기존 방법은 전체 비밀키 분석을 위해 590개의 파형이 필요하지만, 제시한 방법은 230개의 파형으로도 비밀키 분석에 성공했다. 따라서 제시한 방법을 사용하면 기존 방법의 약 1/3의 데이터만으로도 비밀키 분석이 가능하다.



(그림 4) 기존 방법과 제시한 방법의 최소 분석 파형 수

5. 결론

본 논문에서는 비프로파일링 환경에서 학습 데이터 집합과 독립적인 검증 데이터 집합을 활용하여 딥러닝 기반의 부채널 분석 성능을 향상할 수 있는 방법을 제시하였다. 제시한 방법은 신경망의 일반화 성능을 검증함으로써 입력 데이터와 라벨 사이에 규칙이 존재함을 보인다는 점에서 기존 방법보다 논리적 타당성을 갖는다. 실험 결과, 기존 방법으로는 키 분석에 590개의 파형이 필요했지만 제시한 방법을 이용하면 230개의 파형으로 비밀키 분석이 가능했다.

사사

본 논문은 산업통상자원부 국제공동기술개발사업으로 지원된 연구임. (P0011922, 딥러닝을 이용한 RISC-V 기반 하드웨어 보안성 검증 도구 개발)

참고문헌

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

[2] Suresh Chari, Josyula R. Rao, and Pankaj Rohatgi "Template attacks" International Workshop on Cryptographic Hardware and Embedded Systems. Springer, Berlin, Heidelberg, 2002.

[3] Werner Schindler, Kerstin Lemke, and Christof Paar "A stochastic model for differential side channel cryptanalysis" International Workshop on Cryptographic Hardware and Embedded Systems. Springer, Berlin, Heidelberg, 2005.

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

[5] Benjamin Timon, "Non-profiled deep learning-based side-channel attacks with sensitivity analysis" IACR Transactions on Cryptographic Hardware and Embedded Systems (2019): 107-131.

[6] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton "Deep learning" nature 521.7553 (2015): 436-444.

[7] Marius-Constantin Popescu et al. "Multilayer perceptron and neural networks" WSEAS Transactions on Circuits and Systems 8.7 (2009): 579-588.

[8] Ron Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection" Ijcai Vol. 14. No. 2. 1995.

[9] Federal Information Processing Standards Publication (FIPS 197), "Advanced Encryption Standard(AES)", 2001