

논프로파일링 환경에서 CPA 성능향상을 위한 딥러닝 기반 Correlation Optimization 기술 연구

2019 정보보호학회 하계학술대회

Conference on Information Security and Cryptography-Summer 2019
(CISC-S 2019)

고려대학교

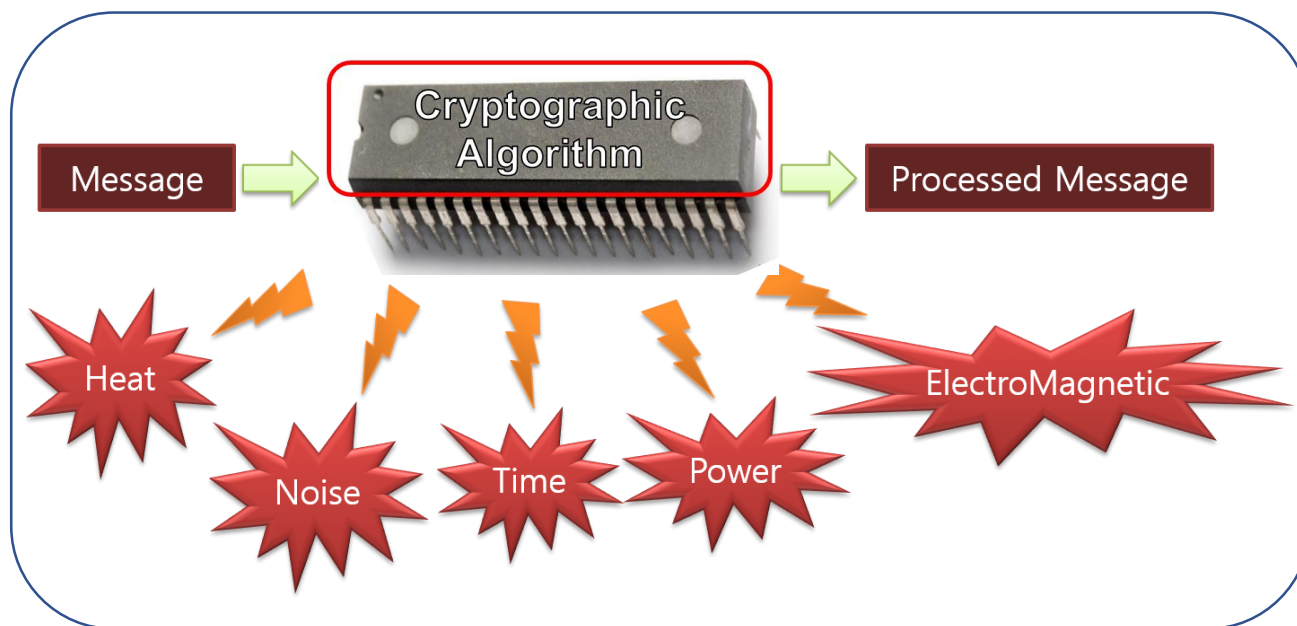
진성현, 권동근, 김희석, 홍석희

Contents

- 부채널 분석 (Side-Channel Analysis)
- 상관 전력 분석 (Correlation Power Analysis)
- 딥러닝 (Deep Learning)
- Correlation Optimization
- 논프로파일링 Correlation Optimization
- 실험
- 결론

부채널 분석 (Side-Channel Analysis)

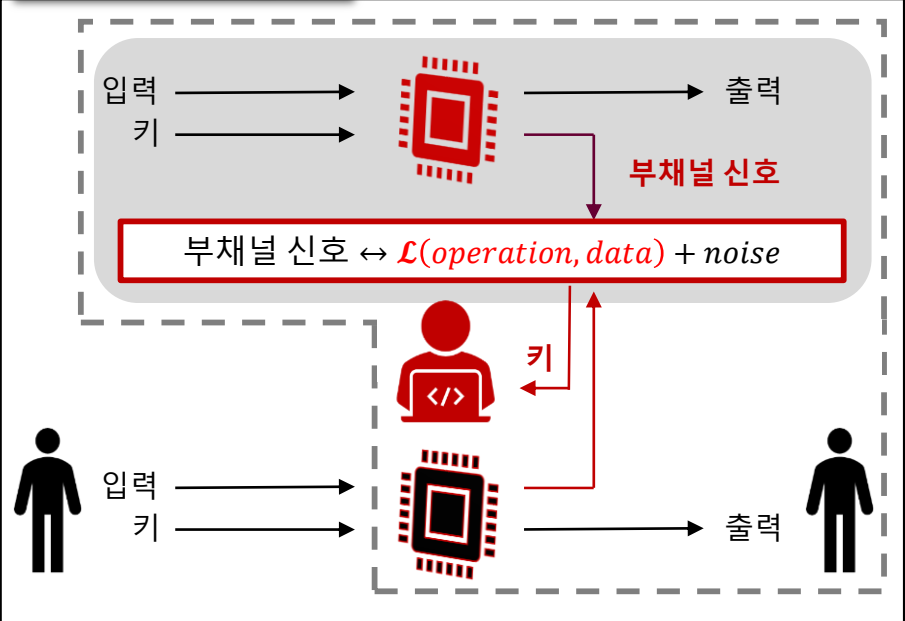
- 전자기기에서 암호호시, 입출력인 평문/암호문 외에 설계자가 고려하지 않은 부가적인 정보인 부채널 신호(Side-Channel Leakages) 발생
 - 열, 소음, 소요시간, 소모 전력, 전자파 등



- 부채널 분석이란 부채널 신호를 이용하여 비밀정보를 분석하는 기법

부채널 분석 (Side-Channel Analysis)

프로파일링 환경

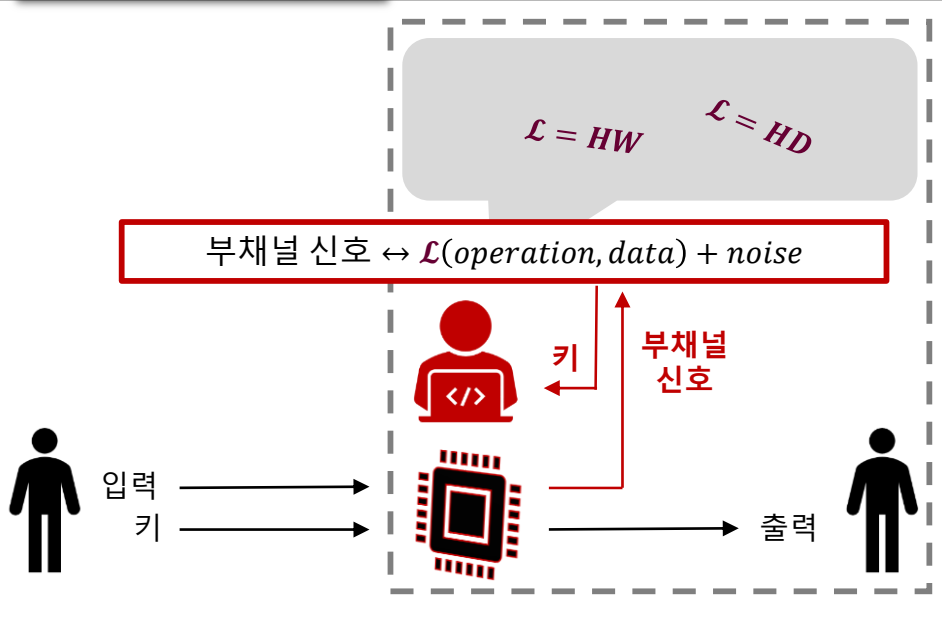


입력 키 → [Chip] → 출력 부채널 신호

부채널 신호 ↔ $\mathcal{L}(\text{operation}, \text{data}) + \text{noise}$

입력 키 → [Person/Device] → 출력

논프로파일링 환경



입력 키 → [Chip] → 출력 부채널 신호

부채널 신호 ↔ $\mathcal{L}(\text{operation}, \text{data}) + \text{noise}$

$\mathcal{L} = HW$ $\mathcal{L} = HD$

입력 키 → [Person/Device] → 출력

- 공격 대상 장비와 동일한 장비를 이용하여 데이터에 따른 부채널 신호 특성 분석 가능
- 강한 공격자 환경

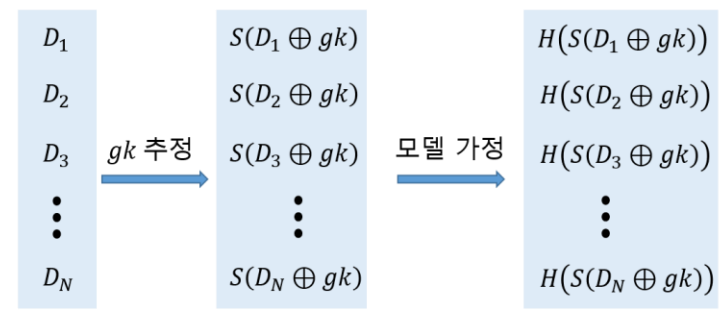
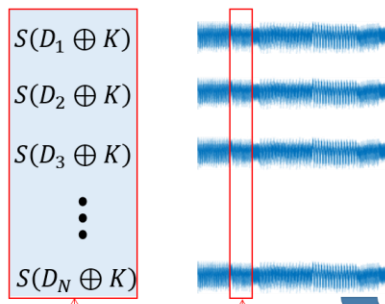
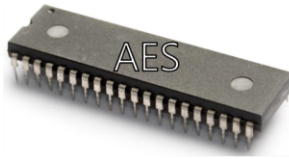
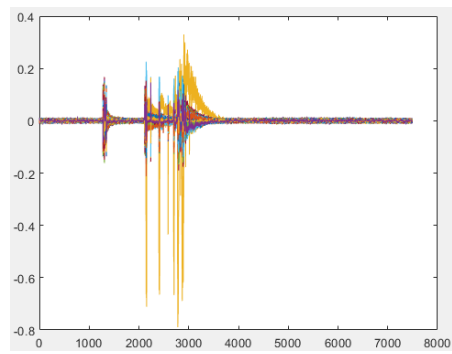
- 공격 대상 장비만을 가지고 분석
- 데이터에 따른 부채널 신호 모델 가정
- 약한 공격자 환경

상관 전력 분석 (Correlation Power Analysis)

■ 대표적인 논프로파일링 부채널 분석 기법

- 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.

$$Corr(P, H) = \frac{COV(P, H)}{\sigma_P \cdot \sigma_H}$$

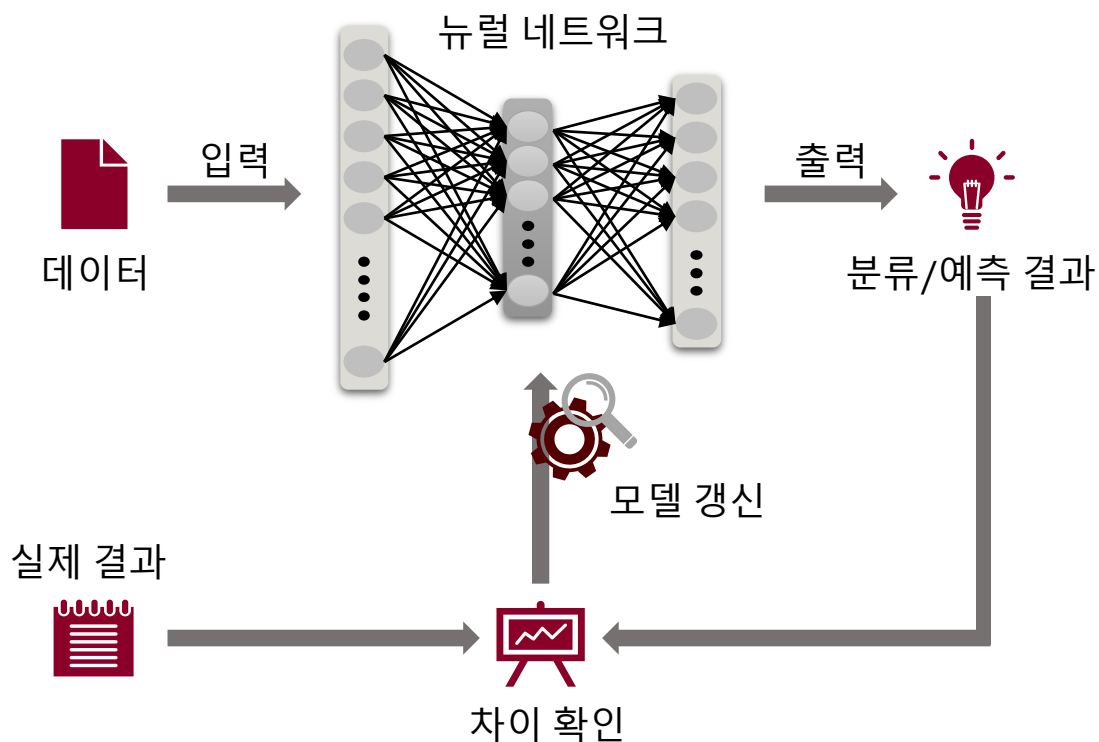


■ 한계점

- 전력 모델 가정 필요
- 키 추측 시 한 시점의 절대적 수치만 고려됨
⇒ 모든 이용 가능한 정보가 활용되지 않음

딥 러닝 (Deep Learning)

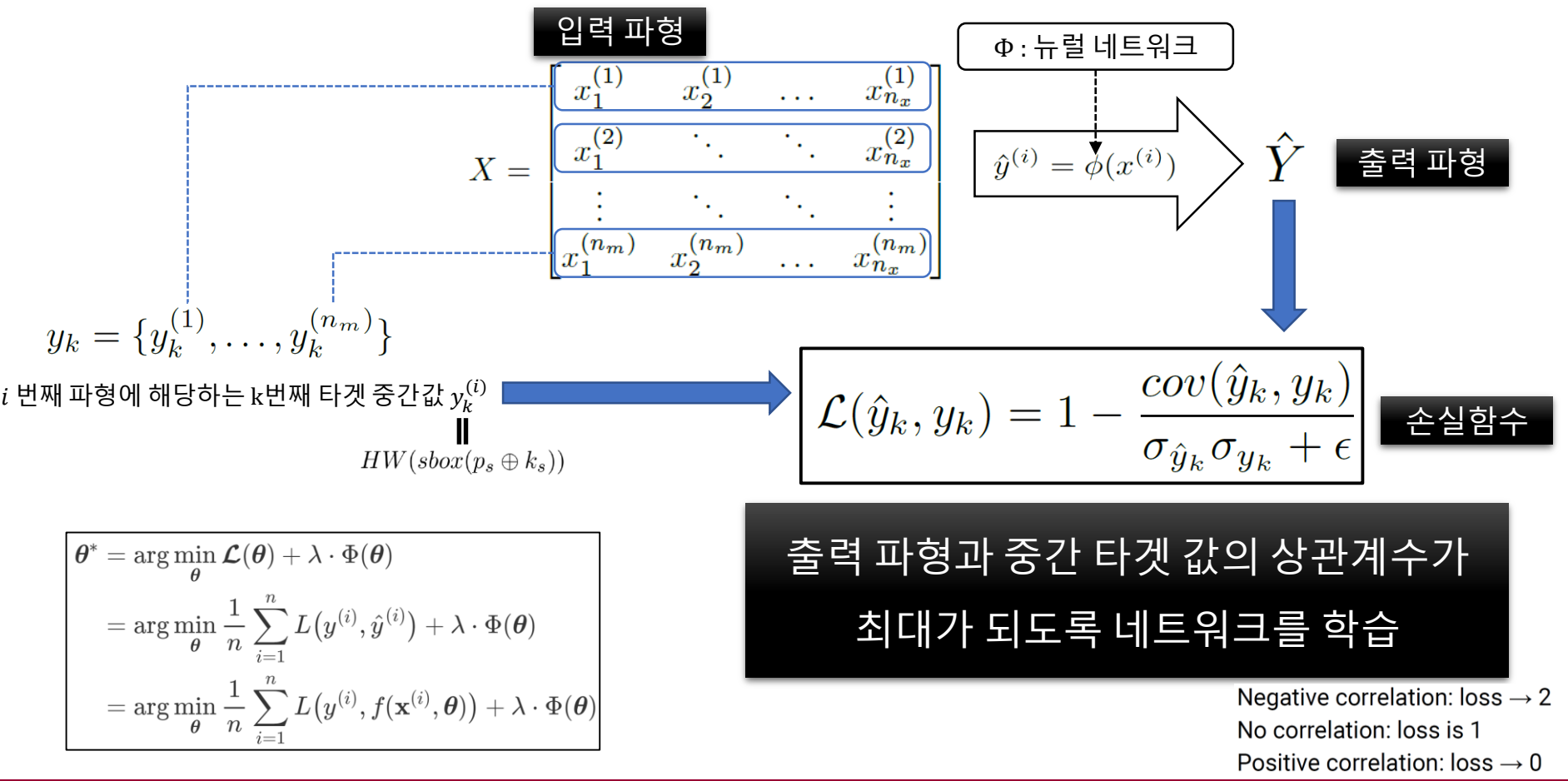
- 다층의 연산층을 통해 데이터를 순차적으로 추상화
- 특징을 자체적으로 학습이 가능하기 때문에 선별할 필요 없음
- 다량의 학습 데이터를 필요로 함



Correlation Optimization [1/3]

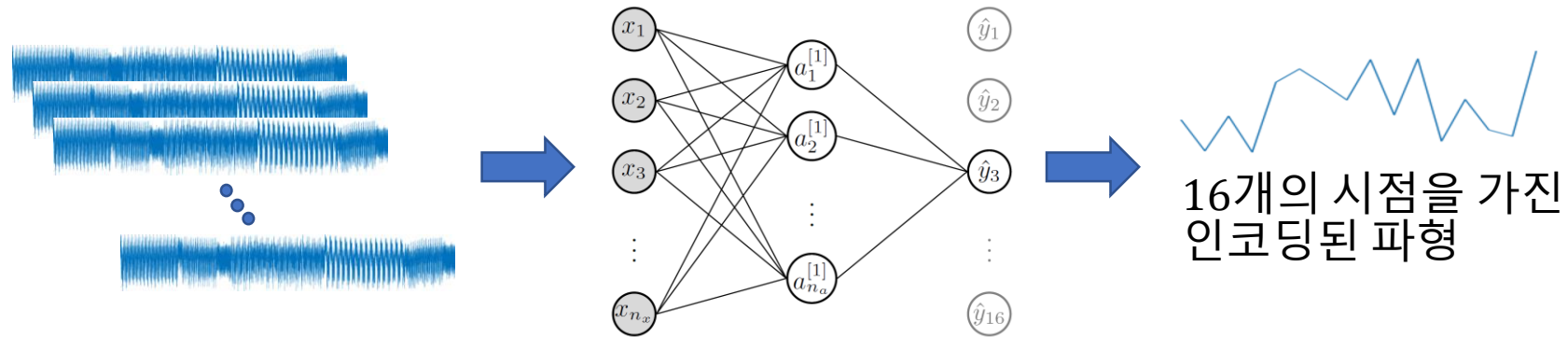
■ CPA 수치가 커지도록 뉴럴 네트워크를 파형 인코더로 학습시킴

- Robyns, P, Quax, P, & Lamotte, W. (2018). Improving CEMA using Correlation Optimization. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2019(1), 1-24.



Correlation Optimization [2/3]

- 각 시점은 타겟 중간 값과 상관계수가 높아지도록 파형을 인코딩



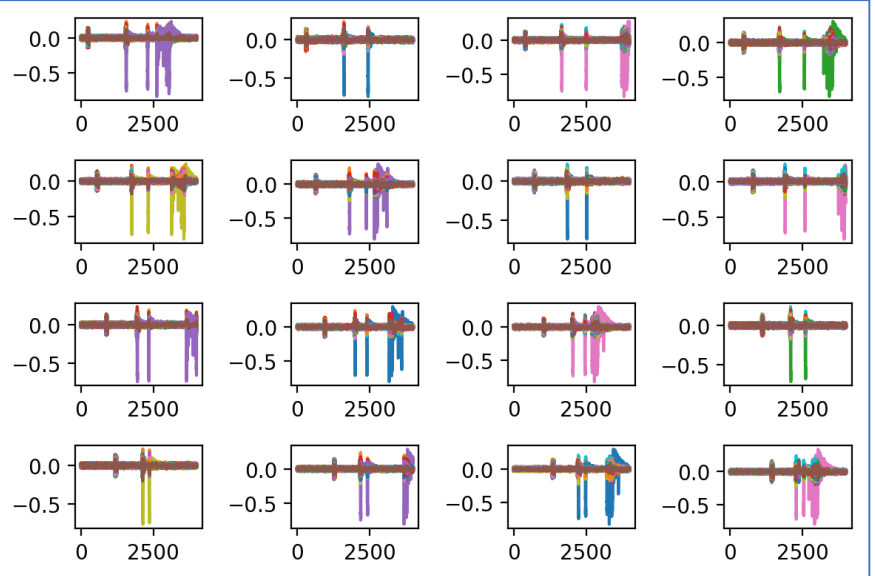
- 장점

1. 뉴럴 네트워크가 모든 정보를 이용하도록 파형을 인코딩
2. 중간 값에 대한 전력모델을 가정하지 않더라도 가능
3. 주파수 도메인 정보 이용 시 (shallow) MLP로도 미 정렬 파형 인코딩 가능

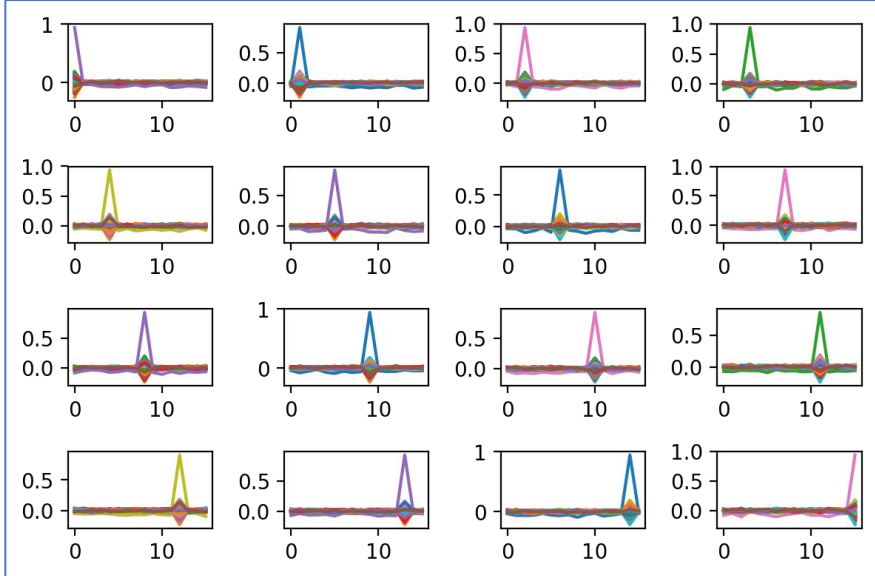


Correlation Optimization [3/3]

Example → AES 1 라운드 HW ($s(P_i \oplus K_j)$)로 학습한 경우



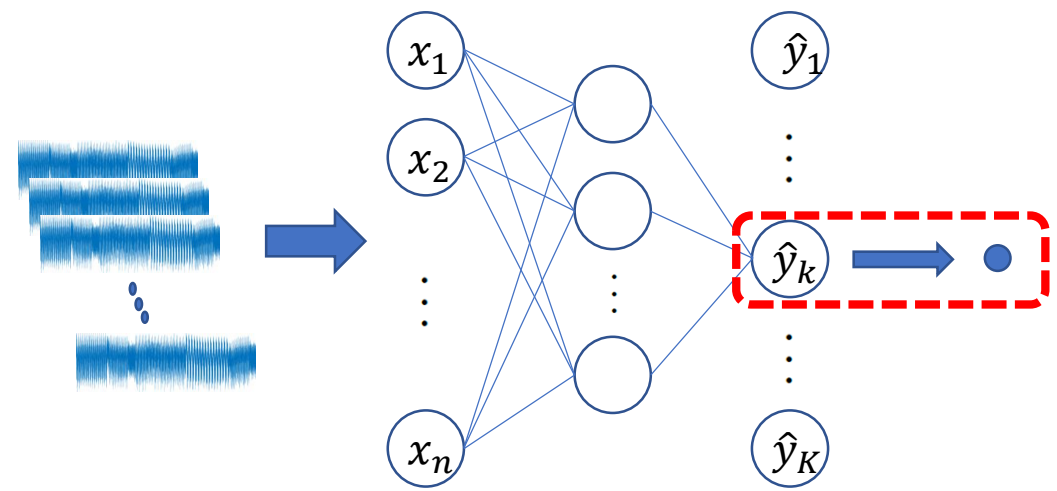
- 01 Byte guess
: FE (254) at 2622 : 1st peak 0.831525 : confidence 3.559137
- 02 Byte guess
: DC (220) at 2446 : 1st peak 0.739401 : confidence 3.092180
- 03 Byte guess
: BA (186) at 3814 : 1st peak 0.832767 : confidence 3.529079
- 04 Byte guess
: 98 (152) at 3430 : 1st peak 0.782777 : confidence 3.441878



- 01 Byte guess
: FE (254) at 0 : 1st peak 0.943175 : confidence 3.781685
- 02 Byte guess
: DC (220) at 1 : 1st peak 0.906889 : confidence 4.004834
- 03 Byte guess
: BA (186) at 2 : 1st peak 0.935978 : confidence 4.004632
- 04 Byte guess
: 98 (152) at 3 : 1st peak 0.940171 : confidence 4.211693

논프로파일링 Correlation Optimization

- Correlation Optimization : 프로파일링 환경이어야만 실제 중간값 계산 가능
- Proposed method
 - 논프로파일링 환경에서는 학습에 필요한 실제 중간 값을 알 수 없음
 - 각 키 가정에 따른 중간 값과 연관되도록 출력 노드를 구성



\hat{y}_i 출력 노드를 위한 손실함수
 $1 - Corr(\phi(Tr)[k], I(P_l, k))$

키가 k 라고 가정할 때의 중간 값과
연관성이 높아지도록 파형을
인코딩시키는 출력 노드

논프로파일링 Correlation Optimization

■ 손실 함수 : $\sum_{gk=0}^{255} loss_{gk} = \sum_{gk=0}^{255} 1 - Corr(\phi(Tr)[gk], I(P_I, gk))$

- 각 키 추측에 대한 CPA 성능이 높아지도록 뉴럴 네트워크를 학습
- 학습되는 이유
 - 옳은 키 추측 시 연관성을 찾을 수 있어 손실 함수 감소하도록 학습 가능
 - 틀린 키 추측 시 연관성을 찾을 수 없어 손실 함수 감소하도록 학습 불가
- 이상적인 상황에서 전체 손실함수는 다음과 같음
 - 학습 초기(=학습이 안되었을 때) 손실 함수 수치는 256 근방
 - 올바른 키 관련 노드만 학습, 나머지는 연관성 없으므로 상관계수가 0에 가까움
- 실제로는 틀린 키에 대해서도 CPA 결과가 0이 아니므로 손실함수 감소함

실험 [1/5]

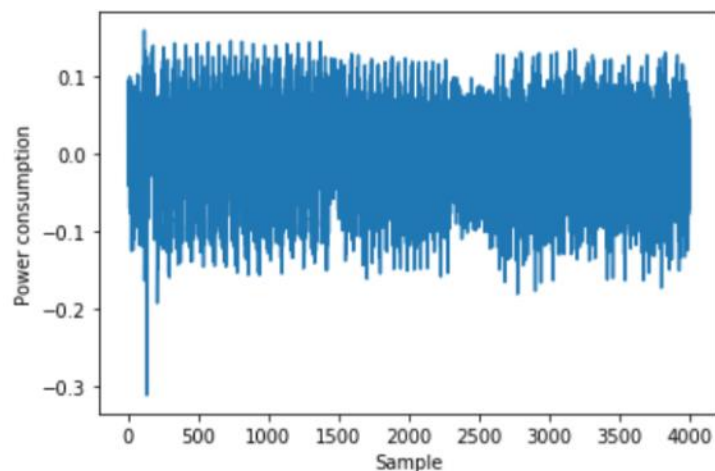
- 다음과 같은 사항을 확인하기 위한 실험을 진행
 - 1. 논프로파일링 환경에서 학습 가능 여부
 - 2. 프로파일링 환경과 논프로파일링 환경 각각에서의 성능 차이
 - 3. Correlation Optimization 적용 유무에 따른 성능 차이

⇒ 위 사항을 우선적으로 확인하기 위해 정렬된 파형에 대한 실험 결과를 제시

실험 [2/5]

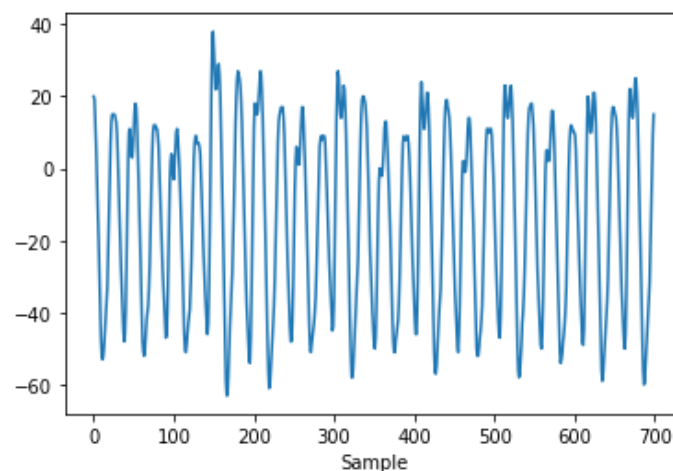
실험 대상 1

- 대응기법이 없는 AES
- CW303 Atmel XMEGA128 (8-bit)
- 소모 전력 파형 수집
- preAddRoundKey + 1 라운드 부분



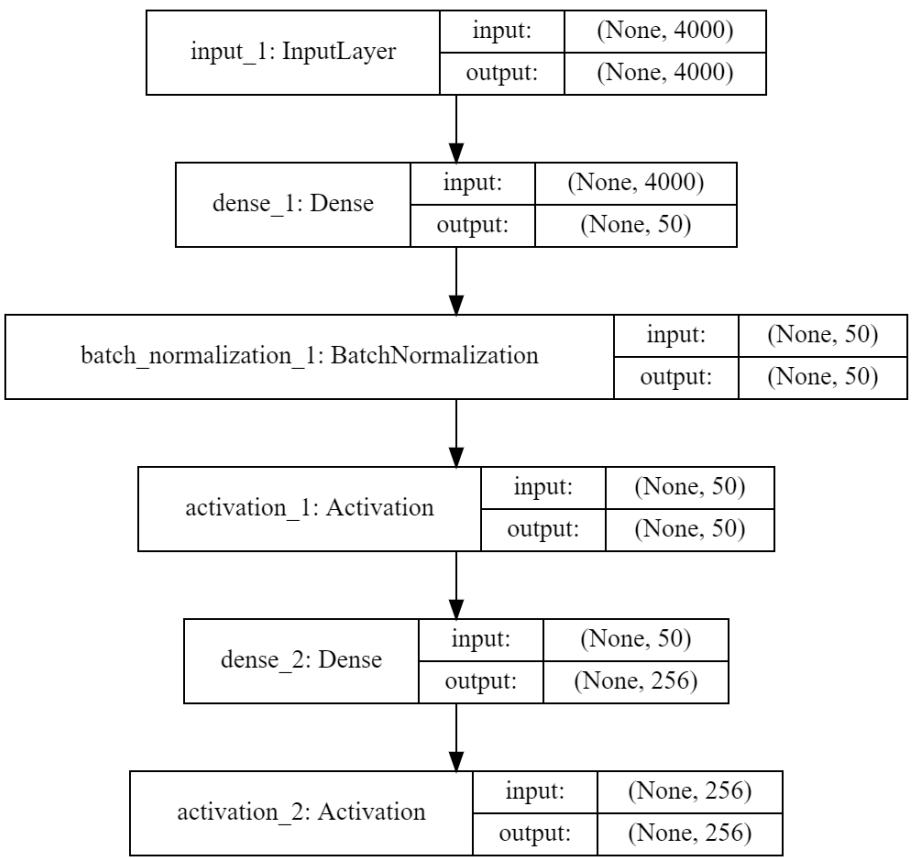
실험 대상 2

- 1st order masked AES SW : ASCAD
- ATmega8515 (8-bit AVR architecture)
- EM 파형 수집
- 1라운드 3번째 SubBytes 부분



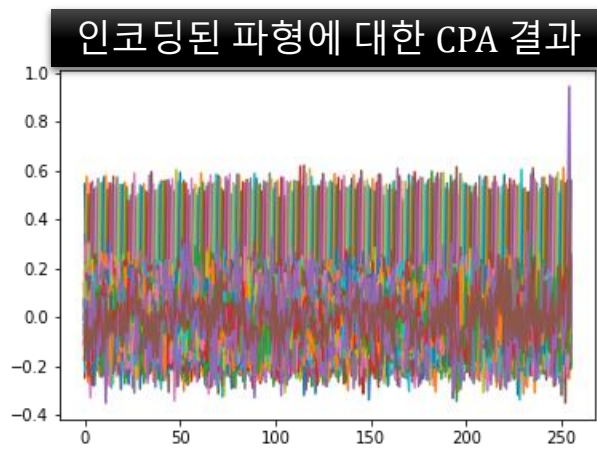
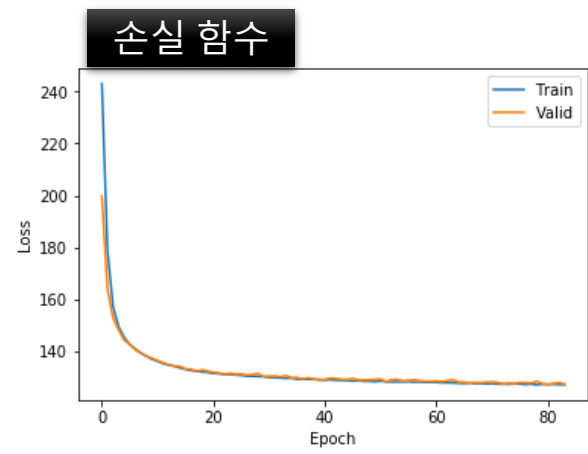
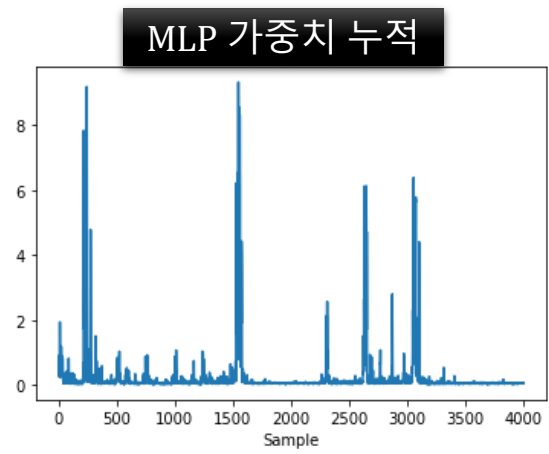
실험 [3/5]

- 사용한 뉴럴 네트워크
 - Standardization 전처리
 - Adam 최적화기
 - 배치정규화 사용
 - 200 epoch
 - 조기학습종료
 - mini-batch : 512



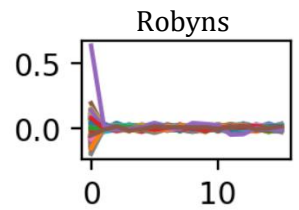
실험 [4/5]

■ 실험 대상 1에 대한 1번째 Sbox 출력 학습 결과



■ 해석

- 1. MLP 가중치 누적을 확인함으로써 학습됨을 확인 가능
- 2. 논프로파일링 환경에선 틀린 키에 대한 noise 증가
- 3. 실제 키 peak 는 옳은 키 노드 출력에서 발생



	key peak	confidence
Raw	0.831525	3.559137
Robyns	0.961740	3.890858
Proposed	0.955060	1.512278

실험 [5/5]

■ 실험 대상 1에 대해 전체 Sbox 출력에 대한 학습 결과

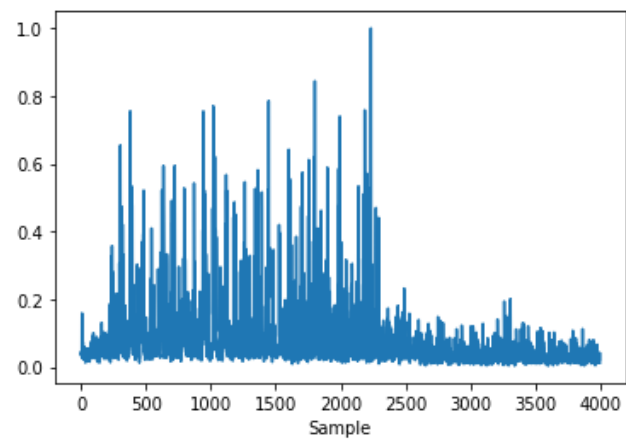
$$\sum_{i=0}^{16} \sum_{gk=0}^{255} loss_{i,gk} = \sum_{i=0}^{16} \sum_{gk=0}^{255} 1 - Corr(\phi(Tr)[i], HW(Sbox(P_i \oplus gk)))$$

- 각 바이트당 256 개 포인트 출력 ➔ $16 \times 256 = 4096$ 개의 출력 노드
- 출력 노드가 증가되어 학습 소요시간 증가됨

Epoch 001/200 - loss: 3992.7410

...

Epoch 132/200 - loss: 3301.8871



학습된 네트워크로 인코딩한 파형에 대한 CPA 결과

01 Byte guess	: FE (254)	at 254	: 1st peak 0.906036	: confidence 3.173388
02 Byte guess	: DC (220)	at 476	: 1st peak 0.837130	: confidence 3.157836
03 Byte guess	: BA (186)	at 698	: 1st peak 0.914904	: confidence 3.151379
04 Byte guess	: 98 (152)	at 920	: 1st peak 0.878419	: confidence 3.116646
05 Byte guess	: 76 (118)	at 1142	: 1st peak 0.900328	: confidence 3.088771
06 Byte guess	: 54 (84)	at 1364	: 1st peak 0.873158	: confidence 3.132627
07 Byte guess	: 32 (50)	at 1586	: 1st peak 0.909242	: confidence 3.386676
08 Byte guess	: 10 (16)	at 1808	: 1st peak 0.886794	: confidence 2.961424
09 Byte guess	: FE (254)	at 2302	: 1st peak 0.917453	: confidence 3.253718
10 Byte guess	: DC (220)	at 2524	: 1st peak 0.892550	: confidence 3.195467
11 Byte guess	: BA (186)	at 2746	: 1st peak 0.895463	: confidence 3.085051
12 Byte guess	: 98 (152)	at 2968	: 1st peak 0.715834	: confidence 2.672757
13 Byte guess	: 76 (118)	at 3190	: 1st peak 0.891674	: confidence 3.171602
14 Byte guess	: 54 (84)	at 3412	: 1st peak 0.877490	: confidence 3.196246
15 Byte guess	: 32 (50)	at 3634	: 1st peak 0.893792	: confidence 3.161286
16 Byte guess	: 10 (16)	at 3856	: 1st peak 0.875942	: confidence 3.115353

- 1바이트 단독에 대한 논프로파일링 환경과 달리 confidence 하락 없음

결론

- (Robyns, CHES2019) Correlation Optimization 소개
- Correlation Optimization 이 부채널 분석에 유의미함을 실제 실험을 통해 확인
- **논프로파일링 환경에서도 Correlation Optimization 기술 적용 가능함을 확인**
- 추후 연구 방향
 - 다른 데이터셋에서도 논프로파일링 환경의 학습이 가능한지 추가 검증 필요
 - 틀린 키 관련 노드에서 틀린 키 CPA peak가 증가되는 원인 규명



Q&A

Thanks

