Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.27, NO.1, Feb. 2017 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2017.27.1.15

프로파일링 단계에서 파형 선별을 통한 템플릿 공격의 성능 향상*

진 성 현,[1†] 김 태 원,[2] 김 희 석,[1‡] 홍 석 희[1]
[1]고려대학교, [2]SNTWORKS

Power Trace Selection Method in Template Profiling Phase for Improvements of Template Attack*

Sunghyun Jin,[1†] Taewon Kim,[2] HeeSeok Kim,[1‡] Seokhie Hong[1]
[1]Korea University, [2]SNTWORKS

요 약

템플릿 공격은 공격 대상 장비와 동일한 테스트 장비를 보유한 경우에 수행할 수 있는 강력한 부채널 분석 방법이다. 템플릿 공격은 테스트 장비를 이용하여 비밀정보에 대한 템플릿을 구성하는 프로파일링 단계와 공격 대상 장비에서 수집한 전력 파형을 템플릿과 비교하여 비밀정보를 찾는 매칭 단계로 구성된다. 템플릿 공격의 성능을 향상시키는 방법 중 하나는 가우시안 분포에 대한 템플릿의 추정을 향상시키는 것이다. 그러나 프로파일링 단계에서 각 중간값에 대한 템플릿을 계산할 때 사용되는 전력 파형의 수가 제한된다면 템플릿 계산이 부정확해진다. 본 논문에서는 프로파일링 단계에서 템플릿을 계산하기 위해 사용하는 파형의 수가 제한될 때 노이즈 파형으로 간주되는 전력 파형을 제거하는 방법론을 제시한다. 제시한 방법론에 따라 노이즈로 간주되는 전력 파형을 제외하여 템플릿을 구성할 경우에 템플릿 추정의 정확도가 향상되어 템플릿 공격의 성능이 향상된다. 또한 본 논문에서는 실험을 통해 템플릿 공격의 성능이 향상됨을 보임으로써 제시한 방법론이 타당함을 증명한다.

ABSTRACT

Template attack is a powerful side-channel analysis technique which can be performed by an attacker who has a test device that is identical to target device. Template attack is consisted of building template in profiling phase and matching the target device using template that were calculated in profiling phase. One methods to improve the success rate of template attack is to better estimate template which is consisted sample mean and sample covariance matrix of gaussian distribution in template profiling. However restriction of power trace in profiling phase led to poor template estimation. In this paper, we propose new method to select noisy power trace in profiling phase. By eliminating noisy power trace in profiling phase, we can construct more advanced mean and covariance matrix which relates to better performance in template attack. We proved that the proposed method is valid through experiments.

Keywords: Side-Channel Analysis, Profiled Attack, Template Attack, Estimation

Received(10. 18. 2016), Modified(1st: 12. 14. 2016, 2nd: 01. 03. 2017), Accepted(01. 10. 2017)
* 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 대학 ICT연구센터육성 지원사업의 연구결과로 수행되었음. (IITP-2016-R0992-16-1017)
† 주저자, mcsmonk@gmail.com
‡ 교신저자, 80khs@korea.ac.kr(Corresponding author)

I. 서 론

이론적으로는 안전하다고 알려진 암호시스템을 물리적인 장치에 구현하고 운용할 때 전력소모, 전자파, 발열, 연산소요시간 등과 같이 발생하는 부채널 정보를 이용하여 비밀정보를 복원하는 분석방법을 부채널 분석 공격(Side-Channel Analysis)이라 한다[15].

부채널 분석기법인 차분전력분석(Differential Power Analysis, DPA)[1], 상관전력분석(Correlation Power Analysis, CPA)[2] 등은 많은 수의 전력 파형을 요구하며, 통계적인 방법을 이용하여 추측된 비밀정보 중 옳은 것을 추출한다. 이와 달리 단 하나의 파형만을 이용하여 비밀정보를 복원하는 공격을 단순전력분석(Simple Power Analysis, SPA)이라 한다[1]. SPA 공격은 주로 공개키 암호 알고리즘에 많이 적용되며, 이는 비밀정보에 의존한 연산이 전력 소모량을 통해 시각적으로 구별될 수 있다는 사실을 이용한다.

템플릿 공격(Template Attack)은 공격 대상 장비와 동일한 장비가 있을 때, 수행되는 분석기법이다[3]. 템플릿 공격은 테스트 장비로부터 비밀정보를 특징화시켜 템플릿을 추정하는 프로파일링 단계와 공격 대상 장비로부터 얻은 파형의 비밀정보를 추출하는 매칭단계로 구성된다. 각 단계마다 템플릿 공격의 성능을 결정짓는 이슈가 존재하며, 가장 활발하게 연구된 것은 프로파일링 단계에서의 성능향상이다[4,5,6,7,8,9,14].

프로파일링 단계에서 공격자의 목표는 각 비밀정보마다 고유한 특징을 갖도록 프로파일링을 하는 것이다. 이를 위하여 유의미한 부채널 정보만을 사용하기 위해 적절한 전력 파형의 시점(Points Of Interest, POIs)을 선택하는 방법이 연구되었다[3,4,5,6,7]. 또한 많은 응용분야에서 이미 효율성이 증명된 차원축소기법인 주성분분석(Principal Component Analysis, PCA)과 선형판별분석(Linear Discriminant Analysis, LDA)을 템플릿 공격에 적용한 연구도 진행되었다[8,9]. 템플릿 구성 단계에서 두 기법들은 공통적으로 노이즈로 간주되는 데이터가 제외되도록 정보를 집약시키므로써 질 좋은 템플릿을 얻게 하며, 축소된 차원만큼 템플릿 구성시 소요되는 연산량을 감소시킨다는 이점이 있다. 하지만 유의미한 시점보다 적은 파형만 이용하여 템플릿을 구성해야 되는 경우, 선형판별분석을 적용할 수 없다는 단점이 있다. 이를 해결하기 위한 방법으로 [14]에서 공통공분산행렬(Pooled Covariance Matrix)이 소개되었다.

매칭 단계에서 사용할 수 있는 파형이 여러 개일 경우에 각 파형에 대한 템플릿 공격의 결과를 조합하여 매칭 성공률을 높이는 연구가 진행되었다[5,14]. 여러 파형의 평균에 템플릿 공격을 적용하는 방법은 노이즈를 감소시켜 매칭 성공률을 높인다. 또 다른 방법인 각 파형에 대한 템플릿 공격의 결과인 최대우도추정의 확률을 더하거나 곱함으로써 공격의 실패율을 감소시켜 매칭성공률을 향상시킨다.

프로파일링 단계와 매칭 단계에서 파형을 수집하는 시간과 환경의 차이에 의한 템플릿 공격의 성능 하락을 해결하기 위한 연구도 진행되었다[10,11,12]. 각 단계에서 사용되는 파형들을 파형들의 평균과 분산으로 정규화하여 템플릿 공격의 성능 하락을 방지한다.

본 논문에서는 템플릿 공격 성능을 향상시키는 방법을 소개한다. 제안한 기법은 템플릿 특징화를 저해시키는 파형들을 선별할 수 있도록 방법과 기준을 제시하며, 이를 통해 질 좋은 템플릿을 구성할 수 있다. 제안한 방법론을 입증하기 위해 8비트 마이크로컨트롤러에서 동작하는 AES(Advanced Encryption Standard) 암호 알고리즘을 구현하여 실험을 진행하였다. 실험한 결과로 평균성공률이 약 3.8% 포인트 증가하였으며, 파형의 수가 적을수록 성능이 향상되는 것을 확인하였다. 이는 제한된 파형을 이용하여 템플릿 공격을 수행해야 되는 공격 환경에서 적용할 수 있는 기법임을 의미한다.

논문은 다음과 같이 구성된다. 2장에서 템플릿 공격과 공통공분산행렬에 대해 소개한다. 3장에서는 프로파일링 단계에서 사용되는 파형의 수가 제한될 때 파형 선별을 통해 전력 파형에 대한 분포 추정을 향상시켜 템플릿 공격의 성능을 향상시키는 방법론을 소개한다. 4장에서는 본 논문에서 제안한 방법론에 대한 실험 결과를 보이고, 5장에서 결론을 짓는다.

II. 관련 연구

이번 장에서는 템플릿 공격에 대해 간단하게 서술하며, 제안하는 방법의 이해를 위한 배경지식을 소개한다.

2.1 템플릿 공격

공격 대상 장비에서 수집 가능한 전력 파형의 수가 제한될 때 차분전력분석, 상관전력분석과 같은 부채널 분석기법은 적용이 힘들어진다. 2002년에 S. Chari 등이 제안한 템플릿 공격[3]은 공격 대상 장비에서 수집 가능한 전력 파형의 수가 제한됨을 가정한 공격이며, 프로파일링 단계와 매칭 단계로 이루어진 대표적인 다변량 프로파일링 부채널 분석기법이다.

본 절에서는 템플릿 공격의 원리와 공격 방법에 대해 소개한다. 사용할 표기법은 다음과 같다. 공격 시점에 가능한 모든 중간값이 S개로 공격 대상인 중간값은 $v \in \{V_1, V_2, ..., V_S\}$로 나타내며 유의미한 시점을 p개라 할 때, 수집한 n개의 전력 파형 벡터는 $t_{v,1}, t_{v,2}, ..., t_{v,n} \in \mathbb{R}^{p \times 1}$로 표기한다.

2.1.1 원리

암호시스템이 동작하는 동안 발생하는 부채널 정보인 전력 파형은 암호시스템 내부의 연산과 중간값에 의해 결정되는 신호 부분과 노이즈 부분으로 구성된다[15]. 암호시스템이 동작하는 동안에 발생하는 노이즈는 랜덤하기 때문에 같은 연산하에서 일정 개수 이상 수집된 각 중간값에 대한 전력 파형들의 노이즈들은 확률 분포를 형성한다. 템플릿 공격은 노이즈에 의해 형성된 확률 분포를 가우시안 분포로 가정하며 형성된 확률 분포에 기반한 최대우도추정을 이용하여 수행된다[3].

각 중간값마다 형성된 다변량 가우시안 분포는 고유의 평균 파형 벡터와 공분산행렬을 가지게 된다. 공격 대상 장비로부터 수집한 전력 파형도 해당하는 중간값과 노이즈에 의해 결정되기 때문에 해당 중간값에 대한 가우시안 분포의 파형으로 취급될 수 있다. 이러한 성질을 이용하여 프로파일링 단계에서는 테스트 장비를 이용하여 공격하고자 하는 연산에 대해 모든 가능한 중간값에 대한 가우시안 확률 분포의 표본평균과 표본공분산행렬인 템플릿을 계산한다. 그 후 매칭 단계에서는 공격 대상 장비로부터 수집한 전력 파형을 프로파일링 단계에서 계산한 템플릿과 비교한다. 전력 파형과 템플릿을 비교하기 위해 가우시안 확률밀도함수를 이용한다. 각 템플릿을 이용하여 전력 파형에 대한 가우시안 확률분포값을 계산하였을 때, 최대우도추정에 의하여 확률분포값을 가장 크게 하는 템플릿에 대한 중간값을 공격 대상 전력 파형의 중간값으로 추정한다. 만약 전력 파형의 실제 중간값을 찾을 경우에 중간값이 평문 혹은 암호문과 비밀정보인 키에 의해 연산된다는 사실을 이용하여 비밀정보인 키를 복구해낼 수 있다.

2.1.2 프로파일링 단계

템플릿 공격의 첫 번째 단계인 프로파일링 단계에서는 테스트 장비를 이용하여 각 중간값 $v \in \{V_1, V_2, ..., V_S\}$에 대한 템플릿을 계산하기 위해 각 중간값마다 전력 파형을 n_v개씩 수집한다. 수집된 전력 파형에 대해 공격 대상인 연산 부분에 해당하는 유의미한 시점을 선택하기 위해 상관전력분석(Correlation Power Analysis, CPA), SOST(Sum of Squared pairwise T-difference)등과 같은 방법을 사용하여 p개 시점을 선택한다[3,4,5,6,7]. 선택한 시점 p개로만 이루어진 중간값 v에 대한 n_v개의 전력 파형 벡터 $t_{v,1}, t_{v,2}, ..., t_{v,n_v} \in \mathbb{R}^{p \times 1}$을 얻은 후, 식 (1),(2)를 통해 중간값 v에 대한 템플릿 $T_v = (m_v, cov_v)$을 계산한다. 식 (2)에서 사용되는 $(t_{v,i} - m_v)'$은 전력 파형 벡터$(t_{v,i} - m_v)$의 전치(transpose)를 의미한다.

$$m_v = \frac{1}{n_v} \sum_{i=1}^{n_v} t_{v,i} \tag{1}$$

$$cov_v = \frac{1}{n_v - 1} \sum_{i=1}^{n_v} (t_{v,i} - m_v)(t_{v,i} - m_v)' \tag{2}$$

2.1.3 매칭 단계

프로파일링 단계에서 계산한 템플릿 $T_{V_1}, T_{V_2}, ..., T_{V_S}$을 이용하여 매칭 단계를 진행한다. 프로파일링 단계에서 선택한 시점 p개로 구성된 파형 $t \in R^{p \times 1}$을 공격 대상 장비로부터 수집한다. 수집한 전력 파형 t에 대한 가우시안 확률밀도함수(3)의 값을 계산한다.

$$P(t | m_v, cov_v) = \frac{\exp(-\frac{1}{2}(t - m_v)' cov_v^{-1} (t - m_v))}{\sqrt{(2\pi)^p \det(cov_v)}} \tag{3}$$

$$where \ \ v \in \{V_1, V_2, ..., V_S\}$$

$$\hat{v} = \arg max_{v \in \{V_1, V_2, ..., V_S\}} \; p(t|m_v, cov_v) \qquad (4)$$

전력 파형 t에 대한 확률밀도함수값과 최대우도추정(4)을 이용하여 전력 파형에 적합한 중간값 $\hat{v}$을 구한다. 최대우도추정을 통해 결정된 중간값 $\hat{v}$이 실제 전력 파형의 중간값이라면 평문 혹은 암호문을 이용하여 비밀정보인 키를 복구해낼 수 있다.

2.2 공통공분산행렬을 사용한 템플릿 공격

프로파일링 단계에서 구한 모든 템플릿의 공분산행렬들의 평균을 공통의 공분산행렬로 사용하는 템플릿 공격은 Oswald 등이 처음으로 제안하였다[13]. 이러한 공통의 공분산행렬로 사용되는 공분산행렬을 공통공분산행렬(pooled covariance matrix)이라 한다. 공통공분산행렬은 식 (5)를 통해 계산된다.

$$\begin{aligned} cov_{pooled} &= \frac{1}{S}\sum_{v \in V} cov_v \\ &= \frac{1}{S}\sum_{v \in V} \frac{1}{(n_v - 1)} \sum_{i=1}^{n_v} (t_{v,i} - m_v)(t_{v,i} - m_v)' \\ &where \; V = \{V_1, V_2, ..., V_S\} \end{aligned} \qquad (5)$$

템플릿 공격의 매칭 단계에서 다변량 가우시안 확률밀도함수값을 계산하기 위해 공분산행렬의 행렬식과 역행렬 등을 이용한다. 그런데 프로파일링 단계에서 유의미한 시점의 개수보다 적은 전력 파형을 이용하여 공분산행렬을 계산하였을 경우 특이행렬이 되기 때문에 행렬식과 역행렬을 계산할 수 없게 된다. 이러한 계산 문제를 회피하기 위해 공통공분산행렬을 사용한다[14]. 식 (5)에서 공통공분산행렬을 계산할 때 전력 파형을 $\sum_{v \in \{V_1, V_2, ..., V_S\}} n_v$개 사용하기 때문에 각 중간값마다 필요한 전력 파형의 개수가 기존 템플릿 계산보다 적어지게 된다.

본 논문에서 제안한 기법인 프로파일링 단계에서 노이즈로 간주되는 파형을 제외하여 템플릿을 계산함으로 인해 각 중간값마다 사용되는 전력 파형의 개수가 줄어든다. 이를 공통공분산행렬을 이용하여 특이행렬로 인해 발생할 수 있는 계산문제를 방지한다.

III. 제안기법

본 장에서는 프로파일링 단계에서 노이즈로 간주되는 파형을 제거하는 방법과 기준을 제시하고, 선별된 파형만을 이용하여 템플릿 공격을 수행하는 방법을 설명한다.

3.1 제안기법의 원리

템플릿을 계산할 때 사용되는 전력 파형의 수가 많다면 각 전력 파형이 가우시안 분포의 평균과 공분산행렬의 계산에 미치는 영향이 줄어든다. 즉, 노이즈로 간주되는 전력 파형이 템플릿 계산에 미치는 영향이 줄어든다고 할 수 있다. 그러나 한 템플릿을 계산할 때 사용되는 파형의 수가 제한된다면 템플릿에 대한 각 전력 파형의 영향이 커지게 된다. 이는 노이즈로 간주되는 전력 파형의 영향력이 커지게 됨을 의미하며, 템플릿 계산을 부정확하게 만든다. 결국 프로파일링 단계에서 템플릿 계산에 사용되는 전력 파형의 수가 제한되면 노이즈로 간주되는 전력 파형에 의해 템플릿 공격의 성능이 낮아진다. 이러한 상황을 극복하기 위해 프로파일링 단계에서 템플릿을 계산할 때 노이즈로 간주되는 파형을 제외하고 템플릿을 계산하여 더 정확한 템플릿을 얻는다.

템플릿을 계산할 때 사용되는 전력 파형이 노이즈를 얼마나 포함하는지 판별하기 위해 가우시안 확률밀도함수를 이용한다. 한 템플릿을 계산할 때 사용되는 각 전력 파형에 대해 각 파형을 제외한 나머지 파형으로 템플릿을 계산하고 구한 템플릿에 대한 전력 파형의 가우시안 확률밀도함수값을 계산한다. 같은 중간값을 가지는 모든 전력 파형에 대해 가우시안 확률밀도함수값을 계산하였을 때 확률밀도함수값이 낮다면 해당 파형이 다른 파형들이 형성하는 분포의 평균 벡터와 거리가 먼 것을 의미하고 이는 다른 파형들보다 해당 전력 파형이 노이즈가 심하다는 것을 의미한다. 그러므로 가우시안 확률밀도함수값이 낮다면 노이즈 파형으로 판단하고, 그러한 파형들을 일부분 제외하고 프로파일링 단계를 진행한다.

3.2 제안기법

프로파일링 단계에서 각 중간값 v에 대한 템플릿을 계산하기 위해 n개의 전력 파형 $t_{v,1}, t_{v,2}, ..., t_{v,n}$이 있을 때, 다음과 같은 과정을 통해 파형을 선별한다.

Algorithm 1	Trace selection for a template profiling

Input : traces $t_1, t_2, ..., t_n$, rate α

Output : selected traces $t_{1*}, t_{2*}, ..., t_{r*}$

1. If $\alpha = 0$, return $t_1, t_2, ..., t_n$
2. For each t_i, calculate template T_i based on $\{t_1, t_2, ..., t_n\} \setminus \{t_i\}$
3. Calculate ordered pair set $\{(t_i, p_i) \mid p_i = p(t_i|T_i) \text{ for } i = 1,2,...,n\}$
4. Sort $\{(t_i, p_i)\}$ based on second coordinate in decreasing order and get $\{(t_{i*}, p_{i*})\}$
5. Select $t_{1*}, t_{2*}, ..., t_{r*}$ where $r^* = n \times (100 - \alpha)/100$

1) 각 파형 $t_{v,i} \in \{t_{v,1}, t_{v,2}, ..., t_{v,n}\}$을 제외한 전력 파형들 $\{t_{v,1}, t_{v,2}, ..., t_{v,n}\} \setminus \{t_{v,i}\}$로 템플릿 $T_{v,i} = (m_{v,i}, cov_{v,i})$을 계산한다.
2) 계산한 템플릿과 제외된 파형에 대한 가우시안 확률밀도함수값 $p_{v,i} = P(x_{v,i}|m_{v,i}, cov_{v,i})$를 계산한다.
3) 파형과 가우시안 확률밀도함수값의 순서쌍의 집합 $\{(t_{v,i}, p_{v,i}) \mid i = 1,2,...,n\}$을 확률밀도함수값을 기준으로 정렬한다.
4) 확률밀도함수값이 가장 낮은 파형들을 노이즈 파형으로 간주하여 일정비율을 제외하고 남은 파형들로만 템플릿을 계산한다.

주어진 Algorithm 1은 프로파일링 단계에서 한 템플릿을 계산할 때 파형을 선별하는 방법을 나타낸다.

제시한 방법을 통해 노이즈로 간주되는 전력 파형을 제외하여 템플릿 추정을 향상시킬 수 있게 된다. 그러나 이러한 방법론을 사용할 경우에 프로파일링 단계에서 사용되는 파형의 수를 제외하는 비율이 커진다면 공분산행렬이 특이행렬이 될 가능성이 존재한다. 공통공분산행렬을 사용함으로써 공분산행렬의 특이행렬과 같은 계산문제를 방지한다.

IV. 실 험

본 장에서는 3장에서 제안한 방법론이 효과가 있음을 실험을 통하여 입증한다.

Table 1. Number of Points of Interest for each target byte in template attack

Byte	#POIs	Byte	#POIs
1	30	9	67
2	53	10	104
3	71	11	131
4	80	12	91
5	102	13	80
6	82	14	87
7	85	15	74
8	108	16	141

4.1 실험환경

부채널 분석 보드 KLA-SCARF MSP430[16]을 이용하여 실험을 진행하였다. 자세한 환경은 아래와 같다.

◎ 프로세서: MSP430F2618 (16비트)
◎ 동작 주파수: 8 MHz
◎ 오실로스코프: Lecroy WaveRunner 204Xi-A
◎ 샘플링 속도: 200 MSample/s
◎ 알고리즘: AES-128 Encryption 1 round
◎ 수집한 파형의 개수: 256,000개
◎ 클락당 샘플수: 25 point/clock
◎ 전처리기법: 적용안함

실험에 사용한 AES(Advanced Encryption Standard) 암호 알고리즘에 대한 자세한 사항은 [17]을 참고하면 된다. Fig. 1은 하나의 랜덤평문을 암호화할 때 수집한 전력 파형을 나타낸다.

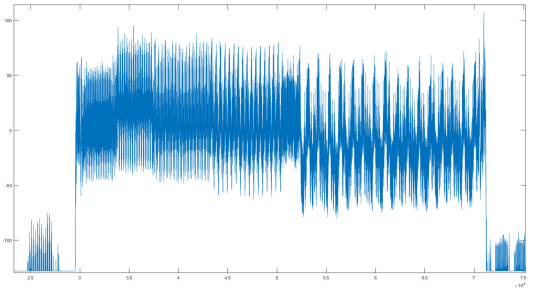

Fig. 1. A power consumption trace of AES 1 round encryption.

4.2 실험방법

부채널 분석 보드를 통해 수집한 전력 파형에 대해 AES 1라운드의 모든 SubBytes 1 바이트 출력을 대상으로 템플릿 공격을 진행하였다. 먼저 템플릿 공격을 진행하기 위해 각 바이트마다 CPA와 SOST를 이용하여 유의미한 시점을 선택하였다[3,4,5,6,7]. Table 1은 각 바이트마다 선택한 유의미한 시점의 개수를 나타낸다. 공격 대상인 SubBytes 출력에 대해 가능한 중간값(0x00~0xFF)마다 300개를 사용하여 총 76,800개의 전력 파형을 프로파일링 단계에서 사용하였다. 프로파일링에 사용되지 않은 나머지 179,200개의 전력 파형을 매칭 단계에 이용하였다.

제안기법을 적용할 때 파형을 제외하는 비율을 다르게 하여 실험을 진행하였다. 이 때 본 논문에서 제안한 방법의 효과를 입증하기 위해 각각의 16바이트에 대해 제안기법과 공통공분산행렬을 동시에 사용한 템플릿 공격의 각 중간값에 대한 공격 성공률과 공통공분산행렬만 적용한 템플릿 공격의 각 중간값에 대한 공격 성공률을 비교하였다.

4.3 실험결과

Fig. 2는 SubBytes 출력 중에 두 번째 바이트에 대해 공통공분산행렬을 사용하는 템플릿 공격에 제안기법을 적용하여 30%의 전력 파형을 제외하고

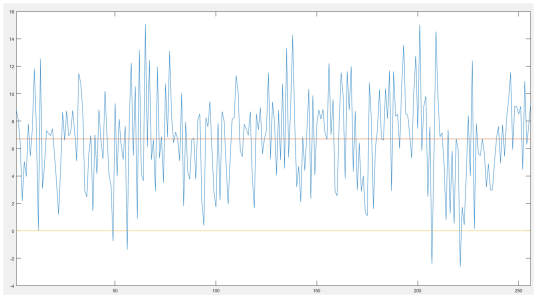

Fig. 2. Difference of success rates between template attack uinsg proposed method and pooled covariance matrix and template attack using pooled covariance matrix.

템플릿을 계산하였을 때 각 중간값별 공격 성공률의 변화를 나타낸다. 가로축은 두 번째 출력 바이트에 대한 각 중간값을 의미하고 세로축은 기존 템플릿 공격 대비 제안기법을 적용한 템플릿 공격의 공격성공률 변화량을 나타낸다. 전력 파형을 선별하여 템플릿을 구성할 경우 대부분의 중간값별 템플릿 공격의 매칭 성공률이 향상됨을 알 수 있다. 그러나 어떤 중간값에 대해서는 공격 성공률이 하락함을 알 수 있다. 이는 각 중간값별로 템플릿을 계산할 때 제외할 전력 파형의 비율이 과함을 의미한다. 이러한 결과를 통해 각 중간값에 대한 템플릿을 계산할 때 노이즈 파형을 제외할 비율이 다르게 적용되어야 함을 알 수 있다.

Table 2는 각 SubBytes 출력마다 본 논문에서

Table 2. For each byte, average difference of success rate of our method and original template attack for each byte of intermediate value

Byte	1%	3%	5%	10%	15%	20%	25%	30%	40%	50%
1	0.147	0.203	0.246	0.305	0.371	0.350	0.285	0.234	-0.028	-0.311
2	0.746	1.586	2.308	3.566	4.659	5.502	6.107	6.697	7.214	7.118
3	0.465	1.216	1.867	3.128	4.000	4.685	5.145	5.348	4.902	4.267
4	0.366	0.764	1.158	1.891	2.351	2.556	2.668	2.625	2.526	2.274
5	0.348	0.668	0.823	0.762	0.361	-0.314	-1.055	-1.869	-3.309	-4.475
6	0.453	1.187	1.948	3.544	4.390	4.922	5.108	5.228	5.076	4.565
7	0.761	1.817	2.518	4.042	5.102	5.849	6.399	6.939	7.589	7.375
8	0.914	2.096	3.185	4.637	5.076	5.218	4.992	4.595	3.617	2.394
9	0.839	2.060	2.827	3.548	3.640	3.415	3.043	2.634	1.751	0.761
10	0.385	1.136	1.804	3.125	4.002	4.629	4.968	5.153	5.065	4.233
11	0.355	1.184	1.745	2.528	2.857	2.782	2.374	1.796	0.412	-1.222
12	0.543	1.492	2.165	3.378	4.017	4.312	4.403	4.373	3.748	2.922
13	0.580	1.591	2.305	3.137	3.470	3.465	3.517	3.389	2.776	1.909
14	0.457	1.169	1.942	3.287	4.281	4.955	5.407	5.765	6.266	6.425
15	0.570	1.539	2.157	3.488	4.534	5.267	6.068	6.743	8.049	9.124
16	0.382	0.829	1.209	1.965	2.291	2.081	1.726	1.076	-0.592	-2.405
avg.	0.520	1.284	1.888	2.896	3.463	3.730	3.822	3.795	3.441	2.810

제안한 방법에 대해 노이즈 전력 파형을 제거하는 비율을 다르게 할 때 각 중간값에 대한 템플릿 공격의 성공률 변화의 평균을 나타낸 표이다. 각 템플릿을 계산할 때 본 논문에서 제안한 방법에 의해 프로파일링 단계에서 20~30%의 전력 파형을 제외하여 템플릿을 계산하면 중간값별 템플릿 공격의 성공률이 평균적으로 3.730~3.822%포인트 향상된다. 5번째 바이트에서 공격성공률이 약간 하락하였지만 그 외의 모든 바이트 블록에서는 공격성공률이 향상됨을 확인하였다. 이는 제안하는 기법인 템플릿 프로파일링 단계에서 파형을 선별하는 방법이 템플릿 공격의 성능 향상을 가져올 수 있음을 의미한다.

V. 결 론

본 논문에서는 템플릿 공격의 성능을 향상시키기 위해 프로파일링 단계에서 사용되는 전력 파형들을 선별하여 템플릿 추정의 정확도를 향상시키는 방법론을 제안하였다. 이러한 방법이 타당함을 확인하기 위해 실험을 통해 입증하였다. 제안한 방법에 의해 20~30%의 전력 파형을 제외하고 템플릿을 구성할 경우 중간값별 템플릿 공격의 평균성공률이 약 3.73~3.822%포인트 증가하였다. 이는 본 논문에서 제시한 방법이 템플릿 공격의 성능 향상을 가져올 수 있음을 의미한다. 그러나 제안한 기법의 이론적인 증명과 환경에 따른 파형을 선별하는 비율에 대한 연구는 추후 연구되어야 할 것이다.

제안한 기법이 타당함을 입증하기 위한 실험에서는 공통 공분산 행렬을 사용한 템플릿 공격에 제안한 기법만을 적용하였다. 그러나 전처리 기법인 PCA, LDA, SSA 등을 함께 사용하여 템플릿 공격을 진행하면 더욱 효율적인 템플릿 공격이 가능할 것으로 기대된다.

References

[1] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," Advances in Cryptology, CRYPTO '99, LNCS 1666, pp. 388-397, Aug. 1999.

[2] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," Cryptographic Hardware and Embedded Systems-CHES 2004, LNCS vol. 3156, pp. 16-29, Aug. 2004.

[3] S. Chari, J.R. Rao, and P. Rohatgi. "Template Attacks," Cryptographic Hardware and Embedded Systems-CHES 2002, LNCS 2523, pp. 13 - 28, Aug. 2002.

[4] C. Rechberger, and E. Oswald. "Practical template attacks." Information Security Application-WISA 2004, LNCS 3325, pp. 440-456, Aug. 2004.

[5] B. Gierlichs. "Signal Theoretial Methods in Differential Side Channel Cryptanalysis." master's thesis, Ruhr-Universit¨at Bochum, 2006.

[6] M. Bär, D. Hermann, and P. Jürgen. "Improved template attacks." Constructive Side-Channel Analysis and Secure Design-COSADE 2010, 2010.

[7] G. Fan, Y. Zhou, H. Zhang, and D. Feng. "How to Choose Interesting Points for Template Attacks More Effectively?." International Conference on Trusted Systems-INTRUST 2014, LNCS 9473, pp. 168-183, Dec. 2014.

[8] C. Archambeau, E. Peeters, F.-X. Standaert, and J.-J. Quisquater. "Template attacks in principal subspaces." Cryptographic Hardware and Embedded Systems-CHES 2006, LNCS 4249, pp. 1-14, Oct. 2006.

[9] F.-X. Standaert, and C. Archambeau. "Using subspace-based template attacks to compare and combine power and electromagnetic information leakages." Cryptographic Hardware and Embedded Systems-CHES 2008, LNCS 5154, pp. 411-425, Aug. 2008.

[10] M. A. Elaabid, and S. Guilley. "Portability of templates." Journal of Cryptographic Engineering, Vol. 2, Issue 1, pp. 63-74, 2012.

[11] D. P. Montminy, R. O. Baldwin, M. A. Temple, E. D. Laspe. "Improving cross-device attacks using zero-mean unit-variance normalization." Journal of

Cryptographic Engineering, Vol. 3, Issue 2, pp. 99-110, 2013.

〔12〕 O. Choudary, and M.G. Kuhn. "Template attacks on different devices." International Workshop on Constructive Side-Channel Analysis and Secure Design-COSADE 2014, LNCS 8622, pp. 179-198, Apr. 2014.

〔13〕 D. Oswald, and C. Paar. "Breaking mifare DESFire MF3ICD40: power analysis and templates in the real world." Cryptographic Hardware and Embedded Systems-CHES 2011, LNCS 6917, pp. 207-222, Oct. 2011.

〔14〕 O. Choudary, and M.G. Kuhn. "Efficient template attacks." International Conference on Smart Card Research and Advanced Applications-CARDIS 2013, LNCS 8419, pp. 253-270, Nov. 2013.

〔15〕 S. Mangard, E. Oswald, and T. Popp. Power analysis attacks: Revealing the secrets of smart cards. Vol. 31. Springer Science & Business Media, 2008.

〔16〕 YongJe Choi, DooHo Cho, and JaeCheol Ryou, "Implementing Side Channel Analysis Evaluation Boards of KLA-SCARF system," Journal of The Korea Institude of Information Security & Cryptology, 24(1), pp. 229-240, Feb.2014.

〔17〕 "Information Technology - Security Techniques - Encryption Algorithms - Part 3: Block Ciphers," ISO/IEC 18033-3:2005, 2005.

〈저자소개〉

진 성 현 (Sunghyun Jin) 학생회원
2015년 2월: 서울시립대학교 수학과 학사
2015년 3월~현재: 고려대학교 정보보호대학원 석사과정
〈관심분야〉 부채널 공격

김 태 원 (Taewon Kim) 학생회원
2010년 2월: 광운대학교 수학과 학사
2012년 8월: 고려대학교 정보보호대학원 석사
2012년 8월~2016년 2월: 고려대학교 정보보호대학원 박사수료
2016년 3월~현재 :(주)SNTWORKS 책임연구원
〈관심분야〉 부채널 공격, 스마트 카드 보안, 암호시스템 안전성 분석 및 고속구현

김 희 석 (HeeSeok Kim) 정회원
2006년: 연세대학교 수학과 학사
2008년: 고려대학교 정보보호대학원 석사
2011년: 고려대학교 정보보호대학원 박사
2011년 9월~2012년 12월: Bristol University 박사후 연구원
2013년~2016년 8월: 한국과학기술정보연구원(KISTI) 선임연구원
2015년~2016년 8월: 과학기술연합대학원대학교(UST) 조교수
2016년 9월~현재: 고려대학교 과학기술대학 수학과 조교수
〈관심분야〉 부채널 공격, 암호시스템 안전성 분석 및 고속구현, 암호칩 설계 기술, 보안관제, 네트워크 보안

홍 석 희 (Seokhie Hong) 종신회원
1995년: 고려대학교 수학과 학사
1997년: 고려대학교 수학과 석사
2001년: 고려대학교 수학과 박사
1999년 8월~2004년 2월: ㈜시큐리티 테크놀로지 선임연구원
2003년 3월~2004년 2월: 고려대학교 정보보호기술연구센터 선임연구원
2004년 4월~2005년 2월: K.U. Leuven ESAT/SCD-COSIC 박사후 연구원
2005년 3월~2013년 8월: 고려대학교 정보보호대학원 부교수
2013년 9월~현재: 고려대학교 정보보호대학원 정교수
〈관심분야〉 대칭키 및 공개키 암호 알고리즘, 부채널 공격 및 대응기법, 디지털 포렌식