JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 23, No. 8, pp. 1539-1545, Aug. 2022

Check for updates

RISC-V 더미 명령어 삽입 기반 ARIA에 대한 부채널 분석

문 재 근[1] · 박 우 정[1] · 지 장 현[1] · 김 해 용[1] · 김 호 원[2,3*]
[1]스마트엠투엠 기업부설연구소 연구원
[2]스마트엠투엠 기업부설연구소 연구소장
[3*]부산대학교 전기컴퓨터공학부 교수

Side-channel Analysis for ARIA Based on Dummy Instruction Insertion of RISC-V

Jaegeun Moon[1] · Woojung Park[1] · Janghyun Ji[1] · Haeyoung Kim[1] · Howon Kim[2,3*]

[1]Researcher, R&D Center, SmartM2M, Busan 48058, Korea
[2]Technical Director, R&D Center, SmartM2M, Busan 48058, Korea
[3*]Professor, Department of Computer Engineering, Busan National University, Busan 46241, Korea

[요 약]

IoT 환경에서는 데이터 보호를 위해 암호화를 사용한다. 그러나 이러한 암호화는 부채널 분석에 의해 취약할 수 있음이 알려졌고 이에 안전한 통신을 위해 부채널 분석에 대한 대응기법이 요구된다. 부채널 분석에 취약하다면 마이크로 컨트롤러에서 처리하는 비밀 데이터가 부채널 정보를 통해 유출될 수 있다. 부채널 분석에 대응하기 위해 마이크로 컨트롤러는 더미 명령어 삽입 방안을 적용할 수 있다. 본 논문에서는 RISC-V에 대해 더미 명령어를 삽입했을 때 부채널 분석 취약성 변화를 분석하기 위해 RISC-V의 한 종류인 ibex를 활용하여 ARIA 암호화에 더미 명령어 삽입을 수행하였다. 더미 명령어가 부채널 분석에 미치는 영향을 확인하기 위해 더미 명령어 삽입 빈도를 변경하며 10,000개의 파형에 대해 TVLA(Test Vector Leakage Assessment)를 수행하였다.

[Abstract]

Cryptographic algorithms have been adopted in the IoT environment for data protection. However, crypto-algorithms can be vulnerable to side-channel analysis (SCA) attacks, which require countermeasures for safe communication. The vulnerability to SCA means that the confidential data processed by the microcontroller may be leaked through side channel information. To alleviate this problem, microcontroller can be designed with dummy instruction insertion. Furthermore, in this paper, we describe the analysis result of dummy instruction insertion in ARIA encryption on RISC-V against SCA attacks by using ibex, a type of RISC-V, for dummy instruction insertion. To analyze its effect, we modified the frequency of dummy instruction insertions and performed a Test Vector Leakage Assessment (TVLA) on 10,000 waveforms.

색인어 : 암호, 하드웨어 보안, 마이크로컨트롤러, RISC-V, 부채널 분석

Keyword : Cryptography, Hardware Security, Microcontroller, RISC-V, Side-channel Analysis

http://dx.doi.org/10.9728/dcs.2022.23.8.1539

Received 22 July 2022; **Revised** 16 August 2022
Accepted 17 August 2022

***Corresponding Author; Howon Kim**

Tel: (저자요청에 의해 비공개)
E-mail: howonkim@gmail.com

Ⅰ. 서 론

IoT 기기의 활용이 활발해짐에 따라 마이크로 컨트롤러를 활용한 어플리케이션 개발이 증가하였다. IoT 기기들은 서로 통신을 통해 데이터를 전송하고 그 데이터를 바탕으로 어플리케이션을 수행한다. 즉, IoT 기기의 데이터에 대해 공격을 수행하게 되면 어플리케이션의 정상적인 동작을 방해할 수 있다[1][2]. 특히 무인화 무기체계 구축을 위해 데이터 보호는 필수적이다[3]. 무기체계의 데이터가 유출된다면 안보에 큰 위협이 될 수 있다[4]. 이러한 위협으로부터 데이터를 안전하게 지키고자 데이터에 암호화를 적용한다[5].

암호는 수학적 안전함을 기반으로 설계되었으나 구현된 환경에 따라 부채널 정보가 유출되는 경우, 부채널 정보를 통해 암호가 취약할 수 있다[6]. 부채널 정보란 전력, 전자기파, 소리, 빛 등과 같이 암호가 수행되는 환경에서 부가적으로 발생하는 정보로 단순 전력 분석(SPA; Simple Power Analysis), 차분 전력 분석(DPA; Differential Power Analysis), 오류 주입(FA; Fault Analysis) 등과 같이 다양한 방법을 활용하여 분석할 수 있다[7]. 본 논문에서는 부채널 분석을 단순 전력 분석과 차분 전력 분석과 같이 기기에 임의의 조작을 하지 않고 전력 분석만 하는 경우만을 다룬다. 전력과 관련된 부채널 정보는 반도체에서 값에 따른 전력 소비 차이가 발생함을 활용하여 분석한다[8]. IoT 기기의 전력 소비는 정적 전력 소비와 동적 전력 소비로 나뉘며 전력 소비 모델 또한 해밍 웨이트 모델과 해밍 디스턴스 모델을 적용할 수 있다[9].

부채널 분석에 대응하기 위해 더미 연산, 셔플링과 같은 하이딩 기법과 데이터를 랜덤값으로 처리하는 마스킹 기법이 존재한다[10][11]. 더미 연산이나 셔플링과 같은 하이딩 기법은 부채널 분석의 공격 복잡도를 향상하여 현실적으로 부채널 분석을 불가능하게 하며 마스킹 기법은 데이터가 랜덤값으로 처리되어 민감 정보와 그 부채널 정보의 상관관계를 제거한 대응방안이다. 마스킹 기법을 적용할 때 안전하고 오버헤드가 크지 않은 마스킹을 생성하기 위해 다양한 연구가 진행되었다[12][13][14].

본 논문에서는 RISC-V에서 제공하는 더미 명령어 삽입이 부채널 분석에 미치는 영향을 분석하였다. 기존 더미 연산, 셔플링 기법은 특수한 컴파일러를 요구하거나[15], 더미 코드 삽입을 위해 코드 사이즈가 증가하는 단점을 가졌다. 반면 RISC-V의 더미 명령어 삽입은 내부 난수 시드를 이용하여 코드 시행과정에 임의의 명령어 삽입을 수행하여, 기존 코드 수정 없이 부채널 방지 기법을 적용할 수 있는 장점을 가진다. 부채널 분석을 위한 암호는 국산 대칭키 암호인 ARIA를 선정하였다. 더미 명령어 삽입과 이에 따른 효과를 분석하기 위해 마스킹이 적용되지 않은 ARIA 128 비트 암호화 코드를 사용하였으며, 분석 결과 더미 명령어 삽입이 빈번할수록 취약 시점 수가 감소함을 확인할 수 있었다.

본 논문의 구성은 2장에서 분석 대상 암호인 ARIA와 분석 대상 플랫폼인 RISC-V, 그리고 분석 방법인 TVLA(Test Vector Leakage Assessment)에 대해 설명하고 3장에서는 RISC-V의 더미 명령어 삽입과 그 효과를 논리적으로 분석하였다. 4장에서는 RISC-V의 소비 전력을 수집하고 TVLA를 통해 분석한 결과를 다루고 5장에서 결론을 정리한다.

Ⅱ. 관련 연구

2-1 ARIA

ARIA는 Academy, Research Institute, Agency의 약자로서 학교, 연구기관, 정부 기관이 공동으로 개발한 대칭키 암호이다[16].

ARIA는 ISPN(Involutional SPN) 구조이다. 라운드 키 생성을 위해서 3라운드 함수를 feistel 구조로 라운드 키의 시드를 생성한다. 이후 라운드에 따라 시드를 비트 단위 시프트를 수행하며 라운드 키를 생성한다.

라운드 키와 라운드 입력값에 대해 $\oplus$ 연산을 수행하고 S-box를 적용한다. 이 때 4가지의 S-box(S_1, S_2, S_1^{-1}, S_2^{-1})를 사용하며 동일한 위치의 바이트에 대해 홀수 라운드와 짝수 라운드의 S-box가 달라진다. 이후 16 × 16 involution 이진 행렬을 사용하여 바이트 단위의 행렬 곱을 수행한다.

ARIA 암호의 입·출력 크기는 128 비트이며 128, 192, 256 비트 암호 키 길이를 지원하며 각각 12, 14, 16 라운드 수행하여 그 결과를 출력한다. 키 확장을 위해 마스터 키를 256 비트로 설정한 후(제로 패딩) feistel 구조의 3라운드 입·출력값을 활용하여 라운드 키를 생성한다. 본 연구에서는 한국인터넷진흥원에서 배포한 ARIA 코드를 기반으로 구현하였다.

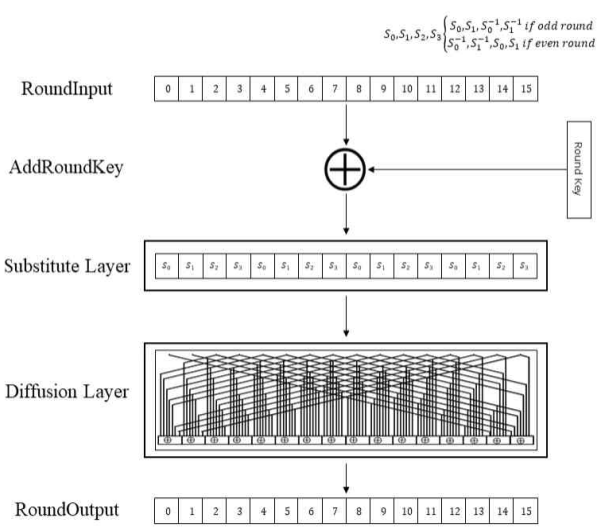

그림 1. ARIA 라운드 함수
Fig. 1. The round function of ARIA

2-2 RISC-V

RISC-V는 오픈 소스 기반의 ISA(Instruction Set Architecture)이며 이를 기반으로 많은 종류의 마이크로 컨트롤러가 파생되었다[17][18]. RISC-V 기반 오픈 소스를 통해 개발자는 FPGA 등과 같은 장비를 통해 마이크로 컨트롤러를 구현하고 해당 마이크로 컨트롤러를 위한 프로그램을 업로드하여 시험 및 검증할 수 있다. RISC-V의 명령어 셋은 32 비트 기본 명령어 셋인 RV32I, 64 비트 기본 명령어 셋인 RV64I, 곱셈과 나눗셈을 위한 명령어를 지원하는 RV32M(RV64M), 축약된 명령어를 지원하는 RV32C (RV64C) 등이 존재한다. 오픈 소스 코드 기반으로 개발자는 마이크로 컨트롤러 내부를 파악할 수 있을 뿐만 아니라 하드웨어 모듈 및 모듈 동작 명령어를 추가하는 등의 방안을 적용함으로써 성능을 향상할 수 있다.

다양한 RISC-V 기반 마이크로 컨트롤러 중 ibex는 32 비트 프로세서이며 저전력/저면적을 특징으로 제시한다[19]. 또한 ibex는 보안을 위해 더미 명령어 삽입, 캐시 에러 검출 코드, Shadow CSR, 버스 무결성 검증 등의 기능을 제공한다[20]. 이를 통해 보안이 강화된 임베디드 환경을 구축할 때 유용할 수 있다.

2-3 TVLA(Test Vector Leakage Assessment)

TVLA는 부채널 분석의 취약점을 검증하기 위해 제안된 방법이다[21]. TVLA는 고정 평문을 입력한 파형 셋과 랜덤 평문을 입력한 파형 셋에 대해 Welch's t-test 기반으로 취약점을 판단한다. 부채널 정보가 누출된다면 고정 평문을 사용하는 경우와 랜덤 평문을 사용하는 경우는 통계적으로 유의미한 차이를 보인다는 점을 이용한 것이다. 파형의 주어진 포인트에 대해 다음 수식에 따라 t 값을 계산하고 t 값이 ±4.5가 넘는 시점을 취약한 시점으로 간주하였다.

$$t = \frac{\mu_0 - \mu_1}{\sqrt{\frac{s_0^2}{n_0} + \frac{s_1^2}{n_1}}} \quad (1)$$

μ는 데이터 셋의 평균, s는 데이터 셋의 표준 편차, n은 데이터 수를 의미한다. Welch's t-test를 수행하기 위한 두 데이터 셋은 공격자가 원하는 공격 대상 중간값으로 선택할 수 있으며 단순히 고정 평문과 랜덤 평문으로 나누어 비교(테스트 0)하는 방법 또한 적용 가능하다. 고정 평문과 랜덤 평문을 비교하는 방법은 공격 대상 기기의 소스 코드 혹은 내부 구조 등을 알 수 없어 정확한 공격 대상 중간값을 알 수 없는 경우에도 부채널 정보 노출 가능성을 제시할 수 있다.

TVLA는 데이터 셋에 대해 동일한 시점의 통계적 특성을 구해 비교하는 것으로 동일한 시점에 동일한 연산이 수행될 때 유의미한 비교를 수행할 수 있다.

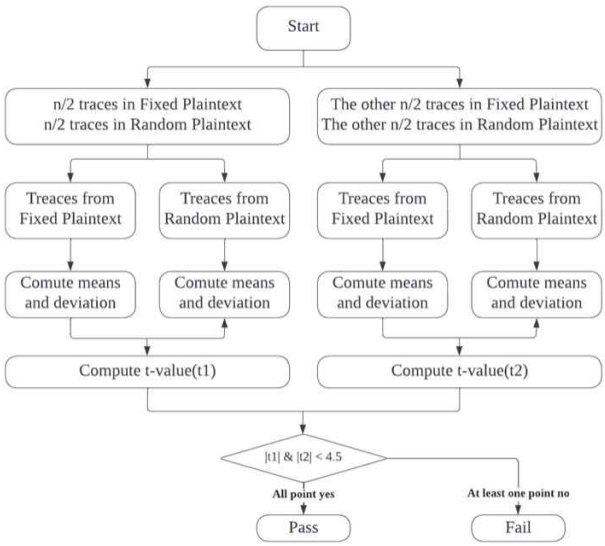

그림 2. TVLA 테스트 0 흐름도
Fig. 2. TVLA test 0 flowchart

2-4 더미 명령어 삽입

부채널 분석에 대응하기 위해 더미 명령어 삽입 방안이 제안되었다[22]. 해당 연구에서는 더미 명령어는 제로 레지스터만 활용한 명령어, 고정된 레지스터 번호와 제로 레지스터를 활용한 명령어, 랜덤 레지스터 번호와 제로 레지스터를 활용한 명령어로 구분하였다. 해당 연구에서는 랜덤 레지스터 번호와 제로 레지스터를 활용할 때 소비 전력에 변동성이 커져 부채널 분석 대응 방안으로 적절하다고 판단하였다. 이 연구에서는 소프트웨어 코드에 더미 명령어를 삽입할 구간을 정하여 해당 구간에만 발생할 수 있도록 제안하였다. 더미 명령어 삽입 구간 정보를 하드웨어에 전달하기 위해 플래그 신호를 활용한다. 플래그 신호를 하드웨어에서 인지하면 난수 생성 모듈이 랜덤한 시점에 프로그램 카운터를 중지하고 더미 명령어를 생성한다. 해당 연구에서는 제로 레지스터만을 활용한 명령어, 고정된 레지스터와 제로 레지스터를 활용한 명령어, 랜덤 레지스터와 제로 레지스터를 활용한 명령어에 대해 부채널 정보를 수집하여 랜덤 레지스터와 제로 레지스터를 활용하는 것이 부채널 분석 대응에 유리함을 보였다.

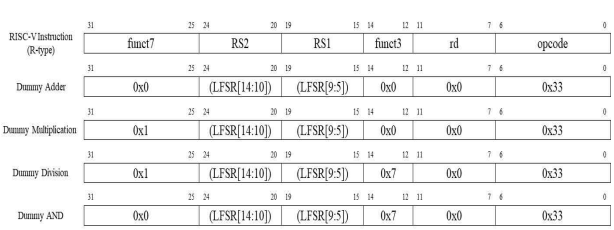

그림 3. ibex 더미 명령어
Fig. 3. ibex dummy instruction

해당 방식은 ibex에서 사용하는 더미 명령어 삽입 방식과 유사하다. 다만, 해당 연구는 부채널 분석에 대응 가능함을 객관적 지표로 제시한 것이 아닌 소수의 파형을 통해 직관적으로 제시하였다. 이에 대해 본 논문에서는 ibex에서 ARIA 암호를 소프트웨어로 구현하고 TVLA를 통해 부채널 분석에 대한 안전성을 객관적 지표로 증명하였다.

Ⅲ. ibex의 더미 명령어 기반 ARIA에 대한 부채널 분석

3-1 ibex의 더미 명령어 삽입 기능

ibex는 보안을 위해 여러 기능을 제공하며 그 중 더미 명령어 삽입 기능이 있다[20]. ibex의 더미 명령어 삽입 방안은 [22]에서 제시한 방안과 유사하게 구현하였다. ibex에서의 더미 명령어 생성과 관련된 설정은 다양하며 ibex 내부의 LFSR(Linear Feedback Shift Register)를 활용하여 더미 명령어 후보군 중 하나로 설정하여 명령어를 패치할 수 있다. 더미 명령어가 삽입되는 간격은 LFSR의 결과에 따라 랜덤으로 삽입한다. LFSR의 시드값은 소프트웨어를 통해 재설정할 수 있으며 이를 통해 LFSR의 난수성을 조절할 수 있다.

더미 명령어는 명령어 패치 단계에서 생성되며 LFSR 내부의 값과 타임아웃 값 설정을 동시에 고려하여 더미 명령어 삽입 신호(insert_dummy_instr)가 발생한다. 더미 명령어 삽입 간격은 설정값을 통해 정할 수 있다. 또한, 엔트로피 및 시드의 크기도 설정값을 통해 변화할 수 있으며 LFSR 내부에 permutation 추가 등의 설정도 가능하다.

LFSR 출력 중 더미 명령어 생성을 위해 사용하는 값은 17 비트이며 상위 2 비트에 의해 더미 명령어 셋이 결정된다. 더미 명령어 셋은 덧셈, 곱셈, 나눗셈, and 연산이 존재한다. LFSR의 최하위 5 비트는 더미 명령어 삽입을 여부 결정을 위한 기준값에 사용된다. 중간의 10 비트는 5 비트씩 나누어 연산을 수행하기 위한 두 값을 불러오기 위해 사용한다. 이와 같은 전체 과정을 통해 더미 명령어가 생성되고 더미 명령어 삽입 시점을 결정한다. 더미 명령어 결과는 더미 명령어 구조에서 확인할 수 있듯이 연산 결과를 레지스터 0에 저장한다. 레지스터 0은 더미 명령어에 제한하여 값이 갱신할 수 있다.

표 1. 설정에 따른 더미 명령어 삽입 간격
Table 1. The setting for interval of dummy instruction insertion

Dummy Instruction Insertion Setting (Dummy_instr_mask)	Interval
000	Between real instruction 0~4
001	Between real instruction 0~8
011	Between real instruction 0~16
111	Between real instruction 0~32

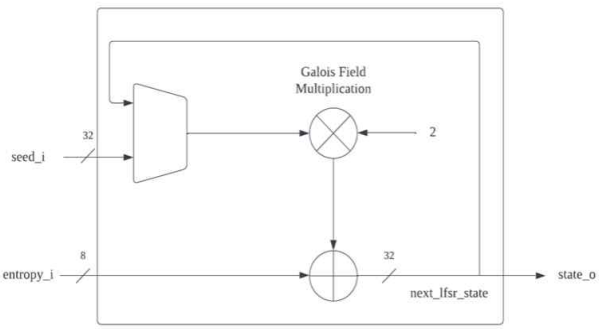

그림 4. ibex 더미 명령어 생성 과정
Fig. 4. The creation process for dummy instruction of ibex

그러나 실제 명령어가 레지스터 0에 접근할 때는 항상 0을 반환하도록 제어하기 때문에 더미 명령어의 연산 결과가 실제 명령어 연산에 영향을 미치지 않는다. 즉, 연산 과정에서 필요한 파이프라인 레지스터 등의 값은 더미 명령어 수행 결과로 갱신되지만 연산 결과는 실제 명령어에서 접근이 금지된다. 이와 같은 방법으로 ibex에서는 실제 명령어 연산에 영향을 주지 않고 더미 명령어를 수행한다.

3-2 부채널 분석 관점에서의 RISC-V의 더미 명령어 삽입 기능 효과

더미 명령어는 임의의 타이밍에 삽입되며 연산에 사용하는 값 또한 임의의 값들을 사용한다. 그 과정에서 연산을 위해 거쳐야 하는 레지스터 값이 갱신되고 연산 수행 시점에 변동성을 주어 부채널 분석 결과에 영향을 미친다.

해밍 디스턴스 모델은 데이터 값이 변하는 순간의 부채널 정보에 적용하는 모델로 명령어 삽입에 의해 변경 전후 값이 변한다면 누출되는 부채널 정보에 영향을 미칠 것으로 해석할 수 있다. 더미 명령어의 경우 수행 과정에서 변동성을 가지는 부분은 삽입 타이밍과 RS1, RS2의 값이 될 수 있다.

$$r = HD(Val(RS_x), s) = HW(Val(RS_x) \oplus s) \quad (2)$$

해밍 디스턴스 모델을 적용한다면 부채널 정보 r은 민감 정보 s와 RS1 혹은 RS2의 값의 $\oplus$연산 결과의 해밍 웨이트 모델로 나타난다. 이 때 RS1 혹은 RS2의 값이 민감 정보와 독립적이며 변수이기 때문에 해밍 디스턴스 모델의 부채널 정보는 마스킹 효과를 가진다.

부채널 분석 대응 측면에서 가장 중요한 특징은 더미 명령어 삽입 시점이 변동성이 가진다는 점이다. 더미 명령어 삽입으로 인해 민감 정보의 부채널 정보 누출 시점이 변하게 되는데 이는 부채널 분석 대응 기법의 하이딩 기법 중 랜덤 지터 효과와 동일한 효과를 가진다. 이로 인해 수집된 민감 정보 관련 부채널 정보는 동일한 시점에 존재하지 않을 가능성이 발생한다. 이로 인해 전력 소비 모델과 무관하게 부채널 분석 대응이 가능하다. 부채널 정보 분석을 위해서는 공격 대상 연산 수행 중 민감 정보와 관련된 변동성만 존재해야 한다.

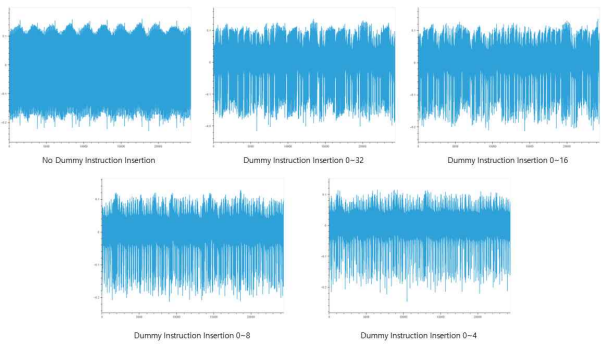

그림 5. 더미 명령어 삽입 설정에 따른 소비 전력 파형
Fig. 5. The power waves with the settings of dummy instruction insertion

더미 명령어 삽입은 민감 정보와 무관한 변동성을 가져온다. 따라서 더미 명령어 삽입 기능을 사용한 경우 TVLA를 적용하더라도 취약한 시점을 발견하기 어려울 것으로 분석할 수 있다.

Ⅳ. 실험 결과

본 논문에서는 ibex 더미 명령어 삽입 기능이 부채널 분석에 미치는 영향을 확인하기 위해 ibex를 자일링스 사의 arty-7 시리즈인 xc7a100tfftg256에 구현하여 실험을 수행하였다.

파형은 ISO/IEC 17825:2016의 보안 수준 3의 조건인 10,000개를 수집하였으며 수집의 용이성을 위해 24400 포인트(6100 싸이클)까지 수집하여 분석하였다. 또한 더미 명령어 삽입 시 성능에 미치는 영향을 측정하기 위해 ARIA 암호화 1 회당 걸리는 싸이클 수를 10,000 회 측정하고 평균을 계산하여 비교하였다. 그 결과 더미 명령어를 추가했을 때 싸이클 수는 기존 대비 최소 약 6배에서 최대 20배 정도의 싸이클 수가 소요됨을 확인하였다.

더미 명령어 삽입이 없는 경우 ARIA 암호의 라운드가 육안으로 구분되나 더미 명령어 삽입이 빈번할수록 육안으로 라운드 구분이 어려워짐을 알 수 있다.

표 2. 더미 명령어 설정에 따른 성능 및 취약 시점 수
Table 2. The performance and vulnerable points along with the settings of dummy instruction insertion

Dummy Instruction Insertion Setting (Dummy_instr_mask)	Cycles	Vulnerability Points
Not Applied	7,524	37,118
000 (Between real instrucction 0~4)	141,678	4
001 (Between real instrucction 0~8)	78,738	6
011 (Between real instrucction 0~16)	53,489	7
111 (Between real instrucction 0~32)	42,563	14

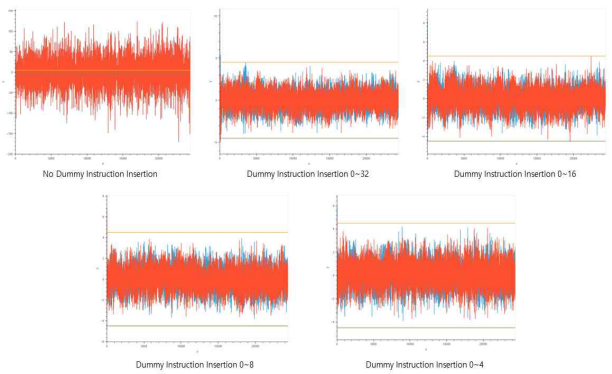

그림 6. 더미 명령어 삽입 설정에 따른 TVLA 결과
Fig. 6. The result of TVLA along with dummy instruction insertion

즉, 더미 명령어 삽입이 이루어지면 실제 명령어 관련 부채널 정보를 정렬하는 것이 용이하지 않음을 알 수 있다. 수집한 ibex의 소비 전력 파형에 대해 취약점 검증을 위해 TVLA를 수행한 결과 더미 명령어 삽입이 빈번할수록 취약 시점 수가 줄어듦을 확인할 수 있었다. 아래 표의 취약 시점 수는 t1과 t2의 취약 시점 수를 합한 결과이다. 이 결과에 따라 암호화 초반부는 더미 명령어 삽입이 수행되지 않아 TVLA를 통과하는 것은 불가능하였으나 더미 명령어 삽입이 수행되는 시점부터는 취약점이 발견되지 않았음을 알 수 있다.

Ⅴ. 결 론

ibex의 더미 명령어 삽입은 마이크로 컨트롤러가 연산에 사용하는 레지스터에 대해 임의의 값으로 갱신하는 효과를 가져온다. 이는 해밍 디스턴스 모델 기반 부채널 정보에 마스킹을 수행하는 효과를 가질 수 있다. 또한 민감 정보 누출 시점에 변동성을 가져 부채널 분석 대응 방안으로 적용 가능하다. 부채널 정보를 분석하기 위해서는 공격 대상 연산이 수행되는 시점에 대한 파악이 필요하며 이를 정렬 과정을 통해 해결한다. ibex의 더미 명령어 삽입은 정렬을 어렵게 하여 부채널 분석에 대해 대응 방안이 된다. 더미 명령어 삽입을 ARIA 암호에 적용하고 TVLA를 통해 객관적 지표를 제시하며 부채널 분석 대응 방안으로 적용 가능함을 보였다.

더미 명령어 삽입을 빈번하게 수행할 경우 정렬의 어려움은 증가하나 ARIA 암호화의 성능이 감소하므로 오버헤드를 고려하여 더미 명령어 삽입을 수행하여야 한다.

또한 부채널 분석에 대한 안전성을 높이기 위해 마스킹 기법과 결합하여 사용할 수 있다. 향후 연구로는 더미 명령어 삽입과 마스킹을 함께 적용할 때 최적화한 모델을 제시하는 것을 목표로 한다.

감사의 글

본 연구는 국방과학연구소(계약번호: UD210023XD)의 지원에 의한 연구 결과임

참고문헌

[1] E. Hellman, "New Directions in Cryptography," *IEEE transactions on Information Theory*, Vol. 22, No. 6, pp. 644-654, November 1976. https://doi.org/10.1109/TIT.1976.1055638

[2] Rahman, Mohammad Ghulam, and Hideki Imai, "Security in wireless communication," *Wireless personal communications* Vol. 22, No. 2, pp. 213-228, August 2002. https://doi.org/10.1023/A:1019968506856

[3] M. W. Lee, "Applying Cybersecurity and Anti-Tamper Methods for Secure Operating of Unmanned Weapon Systems," *Journal of the Korean Society of Systems Engineering*, Vol. 16, No. 1, pp. 36-42, June 2020. https://doi.org/10.14248/JKOSSE.2020.16.1.036

[4] C. L. Cain, Anti-Tamper Technology: Preventing and/or Delaying Exploitation of Critical Technologies, M.S. dissertation, Utica College, Auguest 2013.

[5] Miller, Sandra Kay, "Facing the challenge of wireless security," *Computer*, Vol. 34, No. 7, pp. 16-18, July 2001. https://doi.org/10.1109/2.933495

[6] C. Kocher Paul, "Timing attacks on implementation of Diffe-Hellman, RSA, DSS and other systems," *Crypto 96,* pp. 104-113, August 1996. https://doi.org/10.1007/3-540-68697-5_9

[7] Zhou, YongBin, and DengGuo Feng, "Side-channel attacks: Ten years after its publication and the impacts on cryptographic module security testing," *Cryptology ePrint Archive*, October 2005. Available: http://eprint.iacr.org/2005/388

[8] Kocher, Paul, Joshua Jaffe, and Benjamin Jun, "Differential power analysis," *Crypto 99*, pp. 338-397, August 1999. https://doi.org/10.1007/ 3-540-48405-1_25

[9] Mangard, Stefan, Elisabeth Oswald, and Thomas Popp, "Power analysis attacks: Revealing the secrets of smart cards," Vol. 31. *Springer Science & Business Media*, 2008. https://doi.org/10.1007/978-0-387-38162-6

[10] Coron, Jean-Sébastien, and Louis Goubin, "On boolean and arithmetic masking against differential power analysis," *International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 231-237, August 2000. https://doi.org/ 10.1007/3-540-44499-8_18

[12] E. Trichina, D.D. Seta, L. Germani, "Simplified Adaptive Multiplicative Masking for AES," *CHES*, pp.187-197, September 2003. https://doi.org/10.1007/3-540-36400-5_ 15

[13] L. Goubin, "A sound method for switching between boolean and arithmetic masking," *CHES*, pp. 3-15, May 2001. https://doi.org/10.1007/3-540-44709-1_2

[14] M. Nassar, Y. Souissi, S. Guilley, J. L. Danger, "RSM: A small and fast countermeasure for AES, secure against 1st and 2nd-order zero-offset SCAs" *2012 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 1173-1178, March 2012. https://doi.org/10.1109/DATE.2012. 6176671

[15] A. G. Bayrak, N. Velickovic, P. Ienne, W. Burleson, "An architecture-independent instruction shuffler to protect against side-channel attacks." *ACM Transactions on Architecture and Code Optimization (TACO)*, Vol. 8, No. 4, pp. 1-19, January 2012. https://doi.org/10.1145/2086696.2086699

[16] D. Kwon, J. Kim, S. Park, S. H. Sung, Y. Sohn, J. H. Song, J. Hong, "New block cipher: ARIA," *In International conference on information security and cryptology*, Vol. 2971, pp. 432-445, November 2003. https://doi.org/10.1007/978-3-540-24691-6_32

[17] A. Waterman, Y. Lee, D. A. Patterson, A. K. Asanovic, "The risc-v instruction set manual, volume i: Base user-level isa", EECS Department, Technical Report UCB/EECS-2011-62, May 2011.

[18] RISC-V. RISC-V: The Open era of computing [Internet]. Available: https://riscv.org/.

[19] L. Elsadek, E. Y. Tawfik, "RISC-V resource-constrained cores: A survey and energy comparison", *In 2021 19th IEEE International New Circuits and Systems Conference (NEWCAS)*, pp. 1-5, June 2021. https://doi.org/10.1109/NEWCAS50681.2021.9462781

[20] ETH Zurich and University of Bologna, ibex Documentation [Internet]. Available: https://ibex-core .readthedocs.io/en/ latest/03_reference/security.html.

[21] B. J. Gilbert Goodwill, J. Jaffe, P. Rohatgi, "A testing methodology for side-channel resistance validation", *In NIST non-invasive attack testing workshop*, pp. 115-136, September 2011. Available: https://csrc.nist.gov/csrc/media/events/non-invasive-attack-testing-workshop/documents/08_goodwill.pdf.

[22] J. A. Ambrose, R. G. Ragel, and S. Parameswaran, "RIJID: Random code injection to mask power analysis based side channel attacks", *Proceedings of the 44th annual Design Automation Conference*, pp. 489-492, June 2007. https://doi.org/10.1109/DAC.2007.375214

문재근(Jaegeun Moon)

2015년 : 경북대학교 전자공학과 (공학학사)
2017년 : 경북대학교 대학원 (공학석사)

2017년~2018년: 한국정보통신기술협회
2021년~현 재: 스마트엠투엠
※관심분야 : 정보보호, 하드웨어 보안, 부채널 분석

박우정(Woojung Park)

2021년 : 부산대학교 컴퓨터공학과 (공학학사)

2021년~현 재: 스마트엠투엠
※관심분야 : 하드웨어 보안

지장현(Janghyun Ji)

2016년 : 부산대학교 컴퓨터공학과 (공학학사)
2021년 : 부산대학교 대학원 석박통합과정 수료

2020년~현 재: 스마트엠투엠
※관심분야 : 정보보호, 하드웨어 보안

김해용(Haeyoung Kim)

2015년 : 부산대학교 전자공학과 (공학학사)
2019년 : 부산대학교 대학원 석박통합과정 수료

2018년~현 재: 부산대학교 대학원 박사과정
2022년~현 재: 스마트엠투엠
※관심분야 : IoT, 하드웨어 보안

김호원(Howon Kim)

1995년 : 포항공과대학교 대학원 공학 석사
1999년 : 포항공과대학교 대학원 공학 박사
2004년 : Ruhr University Bochum, Post Doctorial

1998년~2008년: 한국전자통신연구원 팀장
2008년~현 재: 부산대학교 전기컴퓨터공학부 교수
2020년~현 재: 스마트엠투엠 연구소장
※관심분야 : 정보보호, IoT, 인공지능, 블록체인