CRT 기반의 RSA 암호 장치에 대한 레이저 오류 주입 공격 실험1)

이철희*, 추상호*, 김호원*
*부산대학교 컴퓨터공학과
e-mail:2fehee@gmail.com
e-mail:sanghochu@gmail.com
e-mail:howonkim@gmail.com

Laser Fault Injection Attack Experiment on CRT-based RSA Cryptosystem

Chul-Hee Lee*, Sang-Ho Chu*, Ho-Won Kim*
*Dept of Computer Science Engineering, Pusan National University

요 약

최근 물리적인 보안이 큰 위협이 되는 가운데 물리적 공격 중에서도 오류 주입을 통한 공격이 국내에서도 본격적으로 연구가 시작되고 있다. 특히 중국인의 나머지 정리를 이용한 RSA-CRT 알고리즘은 오류 주입 공격을 통해서 비밀 값 p, q가 쉽게 추출 되어 취약하다는 것이 실험적으로 검증이 되었다. 본 논문에서는 레이저를 통한 광학적 오류 주입 공격을 시도 했으며 외부 버퍼를 이용해서 정확하게 원하는 시점에 오류를 주입함으로 레이저 장비 특성에 따른 오류 주입 값의 변화를 확인하였다.

1. 서론

스마트 그리드, 스마트카드, USB 보안 token, RFID, 센서노드, Zigbee, Bluetooth와 같은 통신용 칩이나 방송용 수신 칩, 등 오늘날 많은 분야와 장치에서 물리적 공격에 대한 보안 문제가 부각 되거나 새롭게 발생하고 있다.

물리적 공격 중에서도 준침입성 공격에 해당하는 오류 주입 공격(fault injection attack)[1,2]은 침입성 공격과 같이 장치를 분해(Depackaging)한 후, 칩의 passivation 층을 그대로 둔 채 직접적인 전기 접촉을 하지 않고 공격을 가하며 X-Ray나 전자기장, 혹은 빛을 이용해서 오작동(fault)을 일으키는 능동적인 공격이다. 특히 이 공격의 경우 칩셋의 표면에서 정확한 위치를 찾아서 공격해야 하는 어려움이 있지만 상당히 위협적인 공격에 해당한다.

RSA-CRT 암호 알고리즘[3]은 중국인의 나머지 정리를 이용하기 때문에 일반적인 RSA 암호 알고리즘에 비해서 연산의 효율성이 높고 전력 분석 공격과 같은 부채널 공격에 강인하다는 장점을 지니고 있다. 하지만 RSA-CRT 연산 과정에서 특정 연산 부분에 공격자가 임의의 오류를 주입하는 경우 오동작을 발생 시켜 이때 출력된 연산의 결과를 이용하여 공격하려는 장치의 저장된 비밀 값 p 또는 q를 추출하는 연구가 발표되어 오류 주입 공격에 취약하다는 사실이 밝혀졌다[4,5].

본 논문에서는 RSA-CRT를 수행하는 암호 장치에 오류 주입 공격의 실질적인 공격 실험을 위해서 부채널 공격용 보드를 구성하고 레이저 장비의 특성에 따른 오류 주입 공격의 성공 여부를 테스트 하기 위해 외부 버퍼를 추가하여 레이저를 통한 광학적 오류 주입 공격을 수행하였다. 그 결과로 얻게 된 정상 서명 값과 오류 서명 값을 이용해 비밀 값인 p, q값을 추출하였고 레이저 장비에 따른 외부 버퍼의 특성을 확인 하였다.

1) 이 논문 또는 저서는 2010년 교육과학기술부로부터 지원받아 수행된 연구임 (지역거점연구단육성사업/차세대물류IT기술연구사업단)

2. 기존 RSA-CRT에 대한 오류 주입 공격

2.1. 기존의 RSA-CRT 알고리즘

RSA는 공개키 암호시스템의 하나로 파라미터 생성과정은 다음과 같다.

1) 큰 소수 p,q 선정, $N=p \cdot q$를 계산한다.
2) $GCD(\phi(N),e)=1$를 만족하는 e(공개키)를 선택한다
3) $e \cdot d \equiv 1 \bmod \phi(N)$이 되는 d(비밀키)를 계산한다.
4) 공개정보(N, e), 비밀정보(p, q, d)
5) 메시지 서명 혹은 복호화 $S = m^d \bmod N$
6) 메시지 검증 혹은 암호화 $S = m^e \bmod N$

(그림 1)은 CRT 기반 RSA 서명 알고리듬을 나타낸 것으로 기존의 RSA와 같이 모듈러 N 상에서 연산 하지 않고 비밀 값인 p,q 상에서 연산하여 각각의 그 결과를 재결합함으로 서명 값을 출력한다. p,q는 일반적으로 N의 1/2정도이기에 일반 RSA보다 약 4배 정도 연산 속도가

빠르고 효율적인 장점이 있다. 그리고 모듈러 연산에 사용되는 p,q 값 자체가 노출이 되지 않는 비밀 값이기 때문에 S_p와 S_q의 중간 값을 예측하기 힘들어 전력 분석 공격에도 강인한 장점이 있다.

Input : p, q, d, p_I, q_I, N, m
Output : $S = m^d \bmod N$

1. $S_p \leftarrow m^{d_p} \bmod p$, where, $d_p = d \bmod (p-1)$,
2. $S_q \leftarrow m^{d_q} \bmod q$, where, $d_q = d \bmod (q-1)$
3. $S \leftarrow (S_p \cdot (q \cdot q_I)) + (S_q \cdot (p \cdot p_I)) \bmod N$,
   where, $p_I = p^{-1} \bmod q$, $q_I = q^{-1} \bmod p$
4. Output S

(그림 1) CRT 기반 RSA 서명

2.2. RSA-CRT 알고리즘에 대한 오류 주입 공격

1996년 Bellcore사에서 처음으로 제안한 RSA CRT 알고리즘에 대한 오류 주입 공격은 이후 D.Boneh 등에 의해 멱승 과정에서 오류를 주입하여 비밀 소인수인 p와 q를 알아낼 수 있게 되었다[4]. 그 공격 방법은 다음과 같다.

1) 메시지 m을 입력으로 정상적인 서명값 S를 계산한다.
*정상 서명 값의 재 결합식:
$S = (S_p \cdot q \cdot q_I) + (S_q \cdot p \cdot p_I) \bmod N$

2) 다시 메시지 m을 입력, $S_p = m^{d_p} \bmod p$ 또는
$S_q = m^{d_q} \bmod q$가 계산되는 과정에 오류를 주입한다.
*오류가 주입 된 서명 값의 재 결합식:
$S' = (S_p' \cdot q \cdot q_I) + (S_q \cdot p \cdot p_I) \bmod N$

3) 위의 두 단계에서 생성된 서명의 차를 구한 후
$GCD(S - S', N)$를 계산하여 비밀 값 q(또는 p)를 추출한다. 여기서 GCD 는 최대공약수를 의미한다.

$$*S - S' = (S_p \cdot q \cdot q_I) + (S_q \cdot p \cdot p_I) - (S_p' \cdot q \cdot q_I) - (S_q \cdot p \cdot p_I) \bmod N$$
$$= (S_p \cdot q \cdot q_I) - (S_p' \cdot q \cdot q_I) \bmod N$$
$$*GCD(q \cdot (S_p \cdot q_I - S_p' \cdot q_I),\ p \cdot q) = q$$

위와 같은 원리를 이용하면 또 다른 비밀 값인 p 또한 구할 수 있게 된다.

3. 외부 버퍼에 대한 레이저 오류 주입 공격 실험

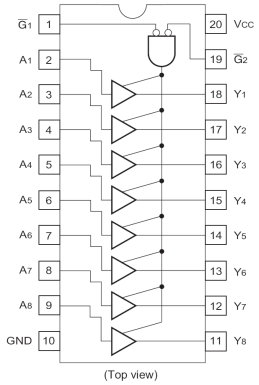
3.1. 오류 주입 공격 실험 방법

3.1.1. 실험 장비

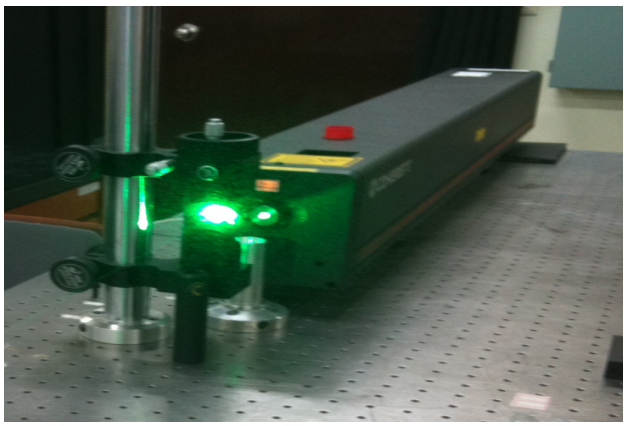
주요 실험 장비로 공격 대상 칩은 PIC18F452 8비트 마이크로 컨트롤러[6]를 사용하였고 메모리의 제한으로 인해 64비트 RSA-CRT 알고리즘을 구현하였다. <표 1>은 PIC18F452의 사양을 나타낸다. 그리고 레이저 오류를 주입할 외부 버퍼는 HD74HC541로 Octal 버퍼이며 3-state 출력이다. (그림 2)는 HD74HC541의 pin 배열을 나타낸다.

<표 1> PIC18F452 마이크로 컨트롤러 사양

On-Chip Program Memory		On-Chip RAM (bytes)	Data EEPROM (bytes)
FLASH (bytes)	# single Word instructions		
32K	16384	1536	256



(그림 2) HD74HC541 버퍼의 pin 배열



(그림 3) Laser Light Scattering System

오류 주입 공격에 사용한 레이저 장비는 (그림 3)에 Laser Light Scattering System으로 모델명은 laser high speed correlator이고 주요 구성 및 성능은 <표 2>와 같다.

<표 2> Laser Light Scattering System 구성 및 성능

Wavelength	sampling times	power
514.5/488.0nm	25ns ~ 40ms	0.001W ~ 3.89W

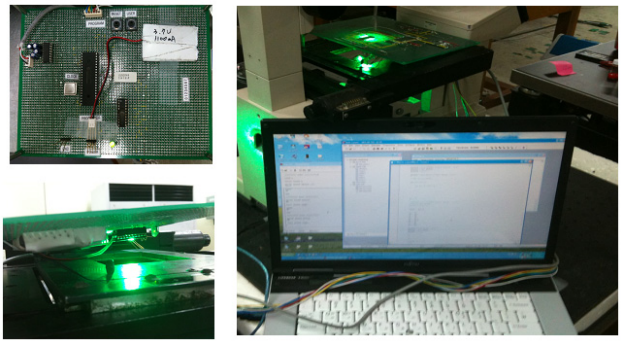
3.1.2. 외부 버퍼에 대한 디캡핑 방법

레이저를 주입 할 외부 버퍼의 핵심 코어 패드 부분을 디캡핑 하기 위해 연마기를 이용하여 해당 부위를 어느 정도 갈아 낸다. 그리고 발연질산과 황산을 적당한 비율로 섞은 뒤 혼합된 화학약품을 스포이드를 이용해서 디캡핑 하려는 부위에 살포한다. 발연질산은 플라스틱 부위를 녹임과 동시에 금속 또한 녹일 수 있기 때문에 금속을 보호하기 위해 황산을 적당히 섞어야 한다. 그리고 어느 정도 열이 유지 되어야 반응이 빨리 일어남으로 핫 플레이트 위에 버퍼를 올려 둔 채 디캡핑을 시도한다[7]. 화학 반응으로 녹은 플라스틱 잿 가루가 발생하면 3차 증류수(DIW)를 이용해서 씻어내고 다시 가열 후 원하는 부위가 드러날 때까지 앞의 절차를 반복한다. (그림 4)는 디캡핑 과정과 디캡핑된 버퍼를 보여준다.

(그림 4) 외부 버퍼를 디캡핑 하기 위한 환경

3.1.3. 외부 버퍼에 대한 레이저 오류 주입 공격

레이저 오류 주입을 위해서 (그림 5)와 같이 먼저 부채널 공격용 보드의 PIC 칩에 입출력 선을 외부로 내어 버퍼에 연결 하였다. 즉 RSA-CRT 연산 도중에 중간 결과 값의 일부를 버퍼에 저장 할 수 있도록 한 것이다. 이때 $S_p = m^{d_p} \bmod p$ 연산 과정 중 m 값의 일부를 버퍼에 저장 시키고 버퍼에 레이저를 주입하여 버퍼 내부에 저장



(그림 5) 외부 버퍼에 레이저 오류 주입 공격을 위한 환경

된 정상적인 평문 값을 다른 값으로 변경 되도록 함으로서 레이저를 통해 오류가 주입되도록 하였다. 이렇게 변경된 평문 값은 오류 서명 값을 만들게 되고 결국 정상 서명 값과 오류 서명 값 사이의 차이 값에 모듈러 N과의 최대 공약수를 계산함으로서 비밀 값 q를 구하게 된다. <표 3>은 오류 서명 값을 얻기 전에 미리 설정 된 64비트의 RSA-CRT 알고리즘의 파라미터 값들을 보여준다.

<표 3> 64비트의 RSA-CRT 알고리즘의 파라미터 값

공개키	e	13
모듈러	N	4F 74 12 87 BD CF 54 CB
비밀값	p	5F 5C C5 05
비밀값	q	D5 4A DB 8F
비밀키	d	3A 8B 6B F7 57 96 84 CB
평문	m	26 27 EA 52 E1 46 7B D5

현재 평문 m의 값은 26 27 EA 52 E1 46 7B D5 이며 이때 마지막 값인 5를 버퍼에 저장하고 레이저를 주입하여 오류를 발생 시키면 버퍼에 저장 된 값이 어떤 값으로 변경 되느냐에 따라 (그림 6)과 같은 오류 서명 값들이 발생한다.

```
m_err.hw[35]:
3e

sign:
04 0e 0c 0f 0b 07 01 0b 09 0d 09 0d 0d 0a 06 05

sign_err:
04 09 09 05 0a 09 0c 0e 09 05 0c 0b 0b 02 01 01

GCD(sign - sign_err, N) = key q:
00 00 00 00 00 00 00 00 0d 05 04 0a 0d 0b 08 0f
```

```
m_err.hw[35]:
fe

sign:
04 0e 0c 0f 0b 07 01 0b 09 0d 09 0d 0d 0a 06 05

sign_err:
03 05 0d 09 09 05 05 0d 0b 07 0c 08 0c 03 05 00

GCD(sign - sign_err, N) = key q:
00 00 00 00 00 00 00 00 0d 05 04 0a 0d 0b 08 0f
```

```
m_err.hw[35]:
ff

sign:
04 0e 0c 0f 0b 07 01 0b 09 0d 09 0d 0d 0a 06 05

sign_err:
01 04 01 0d 01 08 0c 0c 01 07 03 0e 0e 0a 07 01

GCD(sign - sign_err, N) = key q:
00 00 00 00 00 00 00 00 0d 05 04 0a 0d 0b 08 0f
```

(그림 6) 외부 버퍼의 오류 값에 따라 출력되는 오류 서명 값과 비밀 값 q

m_err.hw[35]는 평문의 마지막 배열 값인 5가 저장되어 있는 외부 버퍼로 레이저를 주입시켜 오류가 발생 되도록 했을 때 버퍼 내부에 변경된 값을 나타낸다. 그리고 sign

은 정상적인 서명 값을 나타내며 sign_err는 오류 서명 값을 타나낸다. 마지막으로 key q는 비밀 소인수 값인 q값이 얼마인지를 나타낸다. 평문의 값이 어떤 값으로 바뀌느냐에 따라 계산되어지는 오류 서명 값도 달라지지만 결국 추출되는 비밀 값 q는 동일함을 확인할 수 있다.

3.2. 실험 결과

외부 버퍼에 레이저를 주입 할 때 버퍼의 디캡된 상태에 따라 오류가 발생되는 레이저 세기가 달랐다. 어떤 버퍼는 레이저 포인터로도 오류가 주입되었으며 또 다른 버퍼들은 각각 0.029W, 0.45W에 오류가 주입이 되었다. 최종적인 실험 데이터를 얻기 위해 사용한 버퍼의 경우에는 50A, 512nm파장에서 3.92W로 순간적으로 레이저를 주입했을 때 오류가 발생하였다. 사용한 외부 버퍼의 경우 일정시간 레이저를 주입 하였을 때 오류 값이 순간에만 남아 있는 것이 아니라 일정시간 유지 되는 것을 확인 하였으며 레이저가 주입 되는 각도나 파장 또한 각 버퍼의 특성에 맞게 맞춰줘야 오동작이 발생됨을 확인하였다. 또한 레이저의 세기는 강할수록 오류가 잘 들어가지만 너무 강할 때는 버퍼 자체가 강한 열을 발생하거나 순간적인 강한 에너지에 의해 일정시간 동작을 하지 않음을 확인할 수 있었다.

4. 결론

RSA-CRT 알고리즘에 광학적 오류 주입 공격의 경우 멱승 연산이 이루어지는 시점에 오류가 주입 되어야 한다. 그리고 레이저의 파장과 세기, 레이저가 주입되는 부위 등에 따라서 오류 주입 여부가 결정 된다. 우리는 이러한 특징에 따른 오류 공격 성공 가능성을 파악하기 위해 직접 마이크로 컨트롤러를 디캡핑 하여 정확한 공격 위치를 찾아서 공격하는 것이 아니라 외부 버퍼를 통해서 공격하려는 시점과 공격하려는 부위에 따른 제한적인 요소들을 제거하여 칩의 상태라든지 레이저 장비의 스펙에 따라 효율적으로 공격할 수 있는 환경을 구성하였으며 그러한 실험 환경에서 비밀 값 q를 알아 낼 수 있음을 실험적으로 검증하였다.

참고문헌

[1] D. Boneh, R. DeMillo, and R. Lipton, “New Threat Model Breaks Crypto Codes,” Bellcore Press Release, Sep. 1996.

[2] A. Lenstra, “Memo on RSA Signature Generation in the Presence of Faults,” private communication(available from the author), Sep. 1996.

[3] C. Couvreur and J. Quisquater, “Fast decipherment algorithm for RSA public-key cryptosystem,”Institution of Engineering and Technology IET, Electronics Letters, vol. 18, no. 21, pp. 905 - 907, Oct. 1982.

[4] D. Boneh, R.A. DeMillo, and R.J. Lipton, ″On the importance of checking cryptographic protocols for faults,″ EUROCRYPT’97, LNCS Vol. 1233, pp.37-51, 1997.

[5] M. Joye, A.K. Lenstra, and J.-J. Quisquater, ″Chinese remaindering based cryptosystems in the presence of faults,″ Journal of Cryptology 12(4), pp. 241-245, 1999.

[6] http://www.microchip.com/wwwproducts/devices.aspx?ddocname=en010296

[7] S. P. Skorobogatov, “Semi-invasive attacks: a new approach to hardware security analysis,” University of Cambridge, Technical Report UCAM-CL-TR-630, April 2005.