情報保護學會論文誌
第18卷 第2號, 2008. 4

오류 확산 기법을 이용한 CRT-RSA 오류 주입 공격 대응 방안*

하 재 철[1†‡], 박 제 훈[2], 문 상 재[2]
[1]호서대학교, [2]경북대학교

A Countermeasure Resistant to Fault Attacks on CRT-RSA using Fault Infective Method*

JaeCheol Ha[1†‡], JeaHoon Park[2], SangJae Moon[2]
[1]Hoseo University, [2]Kyungpook National University

요 약

최근 일반 CRT-RSA 알고리듬은 오류 주입 공격에 취약하다는 점이 실험적 결과에 의해 밝혀졌다. 본 논문에서는 CRT-RSA에 대한 오류 주입 공격 및 방어 대책을 분석하고 다양한 형태의 오류 주입 공격을 방어할 수 있는 새로운 알고리듬을 제안하고자 한다. 제안하는 알고리듬은 CRT-RSA에서 두 소수에 대한 멱승연산 시 오류가 발생하면 그 오류를 재결합 과정에서 확산되도록 설계하였다. 이 알고리듬은 판정 기법에 기반한 오류를 검사하는 과정이 없으며 공개 파라미터 e를 사용하지 않는다. 또한 계산량 측면에서도 안전성을 갖춘 타 방식에 비해 효율적이다.

ABSTRACT

Recently, the straightforward CRT-RSA was shown to be broken by fault attacks through many experimental results. In this paper, we analyze the fault attacks against CRT-RSA and their countermeasures, and then propose a new fault infective method resistant to the various fault attacks on CRT-RSA. In our CRT-RSA algorithm, if an error is injected in exponentiation with modulo p or q, then the error is spreaded by fault infective computation in CRT recombination operation. Our countermeasure doesn't have extra error detection procedure based on decision tests and doesn't use public parameter such as e. Also, the computational cost is effective compared to the previous secure countermeasures.

Keywords : *RSA, Chinese Remainder Theorem, Fault Attack, Fault Infective Method*

Ⅰ. 서 론

최근 하드웨어 기술 및 컴퓨터 성능의 향상으로 소인수분해의 어려움에 기반을 둔 RSA 암호시스템[1]의 안전성이 위협받고 있다. 현재, 안전한 암호시스템을 보장하기 위해서는 1024비트 이상의 RSA를 권고하고 있지만 비밀 키에 대한 멱승(exponentiation) 시간이 오래 걸리는 단점을 가지고 있다. 이를 해결하기 위해 RSA 시스템에서 비밀 키의 멱승은 CRT(Chinese Remainder Theorem)를 이용한 계산법(이하에서는 CRT-RSA로 표기)을 사용한다. CRT-RSA는 일반 RSA보다 계산속

접수일: 2007년 12월 13일; 채택일: 2007년 12월 27일
* 이 논문은 2007년도 호서대학교의 재원으로 학술연구비 지원을 받아 수행한 연구임(과제번호: 20070157).
† 주저자, jcha@hoseo.edu
‡ 교신저자, jcha@hoseo.edu

도가 이론적으로 약 4배가량 빨라 스마트카드뿐만 아니라 일반 시스템에도 널리 사용되고 있다[2].

하지만 최근 오류 주입 공격과 같은 부채널 공격(side-channel attack) 혹은 물리적 공격(physical attack)에 의해 시스템의 안전성이 위협당하고 있다[3]. 오류 분석 공격(fault analysis attack) 혹은 오류 주입 공격(fault insertion attack)이라 불리는 물리적 공격 방법은 1996년 Bellcore사에서 RSA 암호시스템에 대한 공격방법으로 처음 소개되었다. 이 공격 방법은 하드웨어에 대한 의도적 공격뿐만 아니라 예상치 못한 결함이나 넓게는 소프트웨어적인 버그 등에 의해서 오류가 발생할 경우 비밀 키를 찾아내는 공격이다. 또한 현재까지의 연구 자료에 따르면 이러한 오류들은 하드웨어 칩의 특정 부분에 글리치(glitch) 등을 발생하는 방법이나 전자파 방사에 의해 오류가 주입될 수 있다고 한다[4, 5].

특히, CRT-RSA는 비밀 소수 p와 q에 대한 연산을 주로 하므로 전력 분석 공격(power analysis attack)에는 비교적 강인한 속성을 가지고 있지만 단 한 번의 오류 주입만으로도 비밀 키를 알아낼 수 있어 오류 주입 공격에 특히 취약한 면이 있다[6-9]. 기존에 제시된 CRT-RSA 오류 주입 공격 대응 방법으로는 크게 오류 검사 방법과 오류 확산 방법이 있다[8, 10]. 그러나 기존 제시된 방법들 중에는 아직 보안성이 검증되지 않았거나 추후 제시된 더 발전된 공격 모델에 대해 안전성을 보증하지 못하는 경우가 많다. 또한 CRT-RSA 시스템을 구현할 경우 사용하는 메모리나 계산량 측면에서도 고려가 필요하다.

본 논문에서는 CRT-RSA에 대한 오류 주입 공격 방법과 기존 방어대책을 분석하고 이 분석을 바탕으로 다양한 형태의 오류 주입 공격을 방어할 수 있는 새로운 알고리듬을 제안하고자 한다. 제안하는 알고리듬의 기본 아이디어는 CRT-RSA에서 멱승 연산 시 오류가 발생하면 그 오류를 중간 계산 값의 재결합(recombination) 과정에서 확산되도록 하는 것이다. 또한, 제안 알고리듬은 구현의 효율성과 계산량을 고려하여 공개 파라미터 e를 사용하지 않으며 계산시간을 최소화하고자 하였다.

Ⅱ. CRT-RSA에 대한 오류 공격

RSA 공개키 암호시스템을 정리하면 아래와 같다.

① 두 개의 큰 소수 p와 q를 선정하고 $N=p \cdot q$를 계산하여 N 을 공개한다.

② $GCD(\phi(N), e)=1$ 이 되는 e를 선정하여 공개한다. 여기서 ϕ는 오일러 totient 함수이다.

③ $e \cdot d \equiv 1 \bmod \phi(N)$이 되는 d를 계산하여 비밀키로 한다.

메시지 m에 대한 서명 시스템은 다음과 같다.

· 공개키 : N, e · 비밀키 : p, q, d

· 서명 : $S = m^d \bmod N$

· 검증 : $m = ? S^e \bmod N$

[그림 1]은 CRT-RSA 알고리듬을 이용한 $m^d \bmod N$의 계산과정을 나타낸 것이며 Gauss 방식을 이용한 CRT 재결합 방법이다. 여기서, $p_I = p^{-1} \bmod q$, $q_I = q^{-1} \bmod p$이다.

Input : p, q, d, p_p, q_p, N, m
Output : S

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

〔그림 1〕 Gauss 방법을 이용한 CRT-RSA

하지만 Gauss 방식을 이용한 CRT를 사용하면 단계 2에서와 같이 2개의 역원, q_I와 p_I 가 모두 사용되는 단점이 있어 $S = (((S_p - S_q) \cdot q_I) \bmod p) \cdot q + S_q$와 같은 Garner 방식의 재결합 방법을 쓰기도 한다.

Boneh 등은 CRT-RSA 서명 생성과정에서 S_p, S_q 중 어느 한 값에 오류가 주입되어 정확하지 않은 서명 S'을 구할 수 있을 경우, 동일한 하나의 메시지에 대해 두 개의 서명(하나의 정확한 서명 S와 하나의 틀린 서명 S')을 가진 공격자는 N을 소인수 분해하는 공격이 가능하다고 제안하였다[6]. 즉, 정확한 서명이 S이고 오류 주입에 의해 잘못 계산된 S_p'에 의한 오류 서명 값이 S'이면 $GCD(S - S', N)$를 계산하여 소수 q를 추출한다. 또한, Joye 등은 하나의 오류가 발생한 서명만 가지고도 공격자가 오류 분석 공격을 시도할 수 있다고 하였는데 위와 같은 오류 주입 시 $GCD(S'^e - m, N)$에 의해 소수 q를 추출한다[7].

Ⅲ. 오류 공격에 대한 기존 대응 방법

지금까지 알려진 CRT-RSA 암호시스템에 대한 오류 분석 공격 대응 기법들은 다음과 같이 정리할 수 있다.

3.1. 두 번 연산 확인 기법

가장 단순한 공격 방지법으로서 서명 생성 시 두 개의 값 S_p와 S_q가 정당한지 혹은 S가 정당한지를 한 번 더 계산하여 검사하는 방법이다. 그러나 이 방법은 암호 연산 자체가 많은 계산량을 요구하므로 매우 비효율적이다. 또한 영구적인 결함을 가지고 있는 시스템의 경우에 오류가 생성되었는지를 검증할 방법이 없으므로 이러한 대응기법들은 적절하지 못하다.

3.2. 메시지 복원 기법

이 대응기법은 서명 값에 대해 서명 검증을 통해 다시 원래의 메시지 $m = S^e \bmod N$으로 복원하여 오류의 유무를 판별하는 방법인데, 이 기법은 공개 키 e가 크면 계산량이 많은 것이 단점이다. 또한 자바 카드와 같은 일부 시스템에서는 서명 시 e에 대한 정보 접근을 허용하지 않기도 하기 때문에 활용에 제한성을 가지고 있다.

3.3. Shamir 기법[10]

이 방법에서는 32비트 정도의 랜덤수 r을 발생시킨 후 $S_p^* = m^d \bmod pr$, $S_q^* = m^d \bmod qr$과 같이 생성하여 $S_p^* \equiv S_q^* \bmod r$이 성립하면 서명을 출력하도록 하였다. 이 방법은 비밀 키 d가 직접 사용되는 단점이 있고, 재결합 과정에서 오류가 발생할 경우 오류 주입 공격에 취약하다.

3.4. Joye et al. 기법[11]

이 기법은 Shamir의 방법을 일반화한 것으로 랜덤수 r_1, r_2을 생성한 후 $S_p^* = m^{d_p} \bmod pr_1$, $S_q^* = m^{d_q} \bmod qr_2$와 다음을 계산한 후 $S_p^* \equiv s_1 \bmod r_1$와 $S_q^* \equiv s_2 \bmod r_2$를 검사하는 과정을 거친다.

$$s_1 = m^{d_p \bmod \phi(r_1)} \bmod r_1,$$

$$s_2 = m^{d_q \bmod \phi(r_2)} \bmod r_2$$

그러나 이 방법 역시 재결합 과정에서 오류가 발생하면 공격이 된다.

3.5. Aumüller et al. 기법[4]

Aumüller 등은 Shamir 기법에 대해 실험적으로 공격 가능성을 지적하면서 서명 과정에서 랜덤수 r_1, r_2, 그리고 r을 생성하여 $S_p' = m^{d_p + r_1(p-1)} \bmod pr$, $S_q' = m^{d_q + r_2(q-1)} \bmod qr$와 같이 생성하며 다양한 검사 절차를 거치도록 하였다. 그러나 d_p나 d_q에 대한 영구적 오류 주입 공격에 취약하다[9].

3.6. BOS 기법[12]

BOS 기법에서는 먼저 랜덤 수 r_1과 r_2를 택한 후 $d_1 = d \bmod \phi(pr_1)$, $d_2 = d \bmod \phi(qr_2)$, $e_1 = d_1^{-1} \bmod \phi(r_1)$, $e_2 = d_2^{-1} \bmod \phi(r_2)$를 사전 계산한다. 이 기법은 사전 계산된 d_1, d_2, e_1, e_2를 이용하여 각 단계들에서 오류가 발생하면 그 오류가 서명에 확산되도록 설계하였다.

S1. $S_p^* = m^{d_1} \bmod pr_1$, $S_q^* = m^{d_2} \bmod pr_2$

S2. $S^* = CRT(S_p^*, S_q^*) \bmod r_1 \cdot r_2 \cdot N$

S3. $c_1 = (m - S^{*e_1} + 1) \bmod r_1$

$c_2 = (m - S^{*e_2} + 1) \bmod r_2$

$S = S^{*c_1 c_2} \bmod N$

그러나 이 방법은 Wagner에 의해 안전하지 않음이 밝혀져 저자들에 의해 이 알고리듬을 수정하여 개선한 알고리듬이 제안되었으나 아직 안전성에 대한 논쟁이 일고 있다[13, 14]. 이 외에도 비밀 키 d가 필요하며 타 기법에 비해 계산량이 비교적 많으며 재결합 연산을 위한 모듈러스의 크기가 $r_1 \cdot r_2 \cdot N$로 큰 것도 단점이다.

3.7. Ciet-Joye 기법[15]

이 방법은 Shamir의 기법을 일반화한 Joye et al. 기법[11]을 오류 확산 기법으로 변경한 것이다. 그러나 이 방법 역시 d_p나 d_q에 대한 오류 공격에 취약하다. 특히, 연산 순서를 조정하여 일시적인 오류에는 방어하도록 하였다고 하지만 d_p나 d_q에 대한 영구적인 오류 공격에 의해 비밀 키가 노출될 수 있다.

3.8. Giraud 기법[16]

멱승 알고리듬으로 Montgomery Ladder 알고리듬을 이용하는 Giraud 기법은 멱승 결과로 다음을 출력한다.

$(S_p^*, S_p) = (m^{d_p}, m^{d_p+1})$, $(S_q^*, S_q) = (m^{d_q}, m^{d_q+1})$

그리고 CRT 재결합에서 $S^* = CRT(S_p^*, S_q^*)$, $S = CRT(S_p, S_q)$를 계산한 후 S와 $m \cdot S^* \bmod N$을 비교하는 과정을 거치게 된다. 그러나 Montgomery Ladder 알고리듬 자체가 Relative 단순 전력 분석 공격에 취약한 것으로 알려져 있다[17].

3.9. BNP 기법[18]

이 기법은 멱승 연산 시 Left-to-Right 방법을 이용하면서 중간에 계산된 $m^{2^l} \bmod p$나 $m^{2^l} \bmod q$값을 이용하여 검사하는 방법이다. 여기서 $l = |p| = |q|$이다. 그러나 이 방법을 CRT-RSA에 사용할 경우 d_p에 대한 오류가 주입되면 비밀 정보가 노출된다[19].

3.10. Kim et al. 기법[20]

이 기법은 중간에 계산된 S_p나 S_q를 공개 키 e를 이용하여 검증하는 방법으로서 CRT 재결합 과정에서 오류를 확산하도록 한 방법이다. 그러나 이 방식에서는 공개키 e를 사용해야 하며 e의 크기가 클 경우에는 계산량이 일반 CRT-RSA 보다 두 배 이상 필요하다는 결정적인 약점을 가지고 있다.

3.11. Kim-Quisquater 기법[21]

이 기법은 문헌 [21]에서 사용한 전력 공격에 대응하는 멱승 알고리듬과 BOS 기법[12]을 결합한 방법으로서 현재까지는 안전한 것으로 알려져 있다. 그러나 연산량이 다소 많은 것이 단점이다.

Ⅳ. 제안하는 오류 확산 기법

본 논문에서는 상기한 분석을 바탕으로 안전하고 효율적인 CRT-RSA 기법을 제안하고자 한다. 제안 알고리듬을 설계하는데 고려한 요소는 아래와 같다.

① 안전성 : 현재까지의 영구적 오류 혹은 일시적 오류 주입 공격을 방어할 수 있는 알고리듬

② 계산의 효율성 : 서명을 생성하는데 소요되는 시간이 적은 효율성

③ 구현의 용이성 : 가급적 사용되는 레지스터나 메모리의 사용량이 적고 구현 환경이 제한된 조건에서도 사용할 수 있는 용이성

④ 확장 대응성 : 제안하는 CRT-RSA에 사용되는 멱승 알고리듬이 전력 분석 공격 등 다른 부채널 공격에 강인한 특성을 지님

제안 알고리듬은 문헌 [20]의 오류 주입 대응 방식을 개선하여 큰 길이를 갖는 공개 키 e를 사용함으로써 발생하는 응용의 제한성과 그로 인한 계산량 증가 문제를 해결하고자 한 것이다. 또한, 문헌 [5]에서 시도하는 이중 오류 공격에 대응하기 위함이다. 제안 방식에서는 몇 가지 사전 정의 및 가정이 필요한데 먼저 랜덤 수 r_1과 r_2 를 선정하는데 이 두 수는 서로 소(coprime)의 관계를 가지며 제곱형 소인수를 가지지 않아야(squarefree) 한다. 또한, $r_i \equiv 3 \bmod 4, i \in 1,2$와 $r_2 \nmid X = pr_1 \cdot ((pr_1)^{-1} \bmod qr_2)$를 만족해야 한다. 또한 $e_p = d_p^{-1} \bmod r_1$와 $e_q = d_q^{-1} \bmod r_2$를 사전 계산하는데. 여기서 e_p와 e_q를 구하기 위해서 $GCD(d_p, r_1) = 1$이고 $GCD(d_q, r_2) = 1$인 r_1과 r_2를 선택한다. 또한 중간 값들을 랜덤화하기 위해 랜덤 수 $R \in Z_N^*$을 사용하며 $R_1 = R \bmod 2^{(l-n)}$이다. 여기서 l은 p나 q의 길이를 의미하며 n은 r_1이나 r_2의 길이를 나타낸다. 즉, R_1은 3단계에서 최종 T의 비트를 l로 맞추기 위해 R의 하위 비트를 절삭(truncation)한 값이다. 제안 알고리듬을 요약한 것이 그림 2이다.

[그림 2]의 알고리즘에서 T_p, T_q, T는 초기에 랜덤한 값으로 둔다. 그리고 공격자는 연산 중간에 이 값들을 강제적으로 0으로 초기화할 수 없다고 가정한다. 여기서 r_1과 r_2는 안전도를 고려하여 확장할 수 있으나 안전도를 고려하여 약 60~80비트 정도로 한다[12]. 제안 알고리듬에서 사용한 핵심 아이디어를 단계별로 설명하면 다음과 같다.

단계 1 : 서명 연산 중 가장 많은 연산량을 차지하는 단계이므로 멱승의 지수는 d_p와 d_q를 그대로 사용하여 필요한 연산을 수행한다.

단계 2 : 단계 1의 계산이 정확한지 확인하는 단계로서 정상적일 경우는 T_p, T_q가 모두 0이므로 $T=0$이 된다.

단계 3 : 랜덤 수 R_1을 이용하여 오류가 발생한 경우

Input : $p, q, d_p, d_q, e_p, e_q, p_I, q_I, r_1, r_2, N, m$
Output : S

1. $S_{pr} = m^{d_p} \bmod pr_1,\ S_{qr} = m^{d_q} \bmod qr_2$
2. $T_p = (m - S_{pr}^{e_p}) \bmod r_1,\ T_q = (m - S_{qr}^{e_q}) \bmod r_2$
3. $T = (T_p \oplus T_q),\ T = T(R_1 \oplus T)$,
   R is a random number
4. $S = (S_p \cdot (q \oplus T) \cdot q_I) + (S_q \cdot (p \oplus T) \cdot p_I) + R \bmod N$
   where $S_p = S_{pr} \bmod p,\ S_q = S_{qr} \bmod q$
5. $c = ((S - (S_{pr} + R)) \bmod p \oplus (S - (S_{qr} + R)) \bmod q)R + 1$
6. $S = (S - R)^c \bmod N$

〔그림 2〕 오류 공격에 대응하는 CRT-RSA 서명

오류의 크기를 p나 q의 크기 정도로 확장하기 위해 사용한다. 또한, 공격자에 의해 악의적으로 랜덤 수 발생기 자체를 손상시켜 $R=0$으로 초기화 한 후 S_{pr}이나 S_{qr}에 오류를 넣는 공격에 대비하여 $T=(T_p \oplus T_q)$, $T=T(R_1 \oplus T)$와 같이 설계하였다. 물론, 오류가 없는 정상 상태일 경우에는 T_p, T_q가 모두 0이며 $T=0$이다.

단계 4 : 단계 3에서 오류가 발생하지 않으면 정상적으로 재결합 연산을 수행한다. 그러나 $T \neq 0$이면 재결합 수식의 양변에 오류가 확산된다. 이 경우 오류가 주입된 경우에는 비밀키를 유추하는 계산식이 성립하지 않도록 오류 서명이 생성된다. 여기서 랜덤 수 R을 더하게 되고 단계 6에서 이를 다시 빼는 과정을 거치게 되는데 그 이유는 다음과 같다, 만약 R을 더하는 연산이 없는데 재결합 과정에서 $S_p = S_{pr} \bmod p$에 오류가 주입되었고 단계 5와 단계 6을 거치지 않게 하는 공격자에 의해 단계 4의 S가 노출되었다고 하자. 그러면 공격자는 $S \bmod N$을 계산한 후 비밀 정보를 얻을 수 있다. 실제로 이러한 이중 오류 공격이 성공한 경우도 있어 대응책으로 제안한 것이다[5]. 따라서 이러한 이중 오류 공격에 대응하기 위해 최종 결과 값이 출력되기 전에는 최종 서명 값을 출력 레지스터에 저장하지 않는 것이 더욱 안전하다.

단계 5 : CRT 재결합 과정에서 발생할 수 있는 오류 주입 공격에 대응하여 단계 4에서 생성한 S에 대한 검사를 수행하는 단계이다. 재결합 과정이 맞으면 $c=1$이 된다.

단계 6 : 최종적인 서명을 출력하는데 $c=1$이면 정상적인 서명을 출력하고 오류 주입에 의해 $c \neq 1$이면 오류 서명을 출력한다. 그러나 이 오류로부터 비밀 키는 유추할 수 없다.

V. 제안 기법 비교 분석

5.1. 안전성

제안 CRT-RSA 알고리듬은 멱승 시 발생하는 오류를 확산하는데, 오류가 발생하면 재결합에 사용되는 두 비밀 키 p, q와 직접 XOR하여 오류를 확산시키는 것이 특징이다. 또한, 재결합 과정에서 발생할 수 있는 또 다른 오류를 확산하기 위해 단계 5를 두었다. 따라서 이러한 오류 확산 기법은 검사 과정을 강제적으로 우회하도록 할 가능성이 있는 판정 기법에 기반한 오류 검사 방법에 비해 더욱 안전하다. 또한 제안 알고리듬은 기존에 메시지나 사용하는 비밀 키 혹은 재결합 과정의 중간 값에 오류를 넣는 공격뿐만 아니라 최근 제안된 이중 오류 주입 공격까지도 방어할 수 있도록 설계하였다.

본 논문에서 제안한 방식의 안전성을 증명하기 위해 각 파라미터에 오류가 주입되었을 경우를 가정하여 단계적으로 분석해 보기로 하자. 물론, 공격의 방법은 어느 특정한 레지스터나 메모리에 얼마만큼의 시간적 정확성을 가지고 오류를 주입하는가에 따라 여러 가지로 분류할 수 있다. 지금까지의 실험적 공격은 세밀하지 않지만 일정한 시점에 글리치 등을 주입하여 오류를 삽입하는 정도이며 어떤 특정 값이 주입되는 것까지는 알지 못하는 수준이다[5].

① 단계 1의 S_{pr} 혹은 S_{qr}의 연산과정에서 오류가 주입된 경우
이 경우 S_{pr}에 오류가 주입되었다고 가정하면, 그 오류는 단계 2에서 $T_p \neq 0$의 결과를 유도하고 T는 랜덤한 값이 된다. 따라서 단계 4, 5, 6을 거쳐 출력되는 서명 값은 랜덤한 값이 되어 공격에 유용한 정보가 되지 못한다.

② 단계 2의 T_p 혹은 T_q의 연산과정에서 오류가 주입된 경우
이 경우 T_p에 오류가 주입되었다고 가정하면, 위에서과 같이 T는 랜덤한 값이 된다. 이 공격 역시

단계 4, 5, 6을 거쳐 출력되는 서명 값은 랜덤한 값이 되어 공격에 유용한 정보가 되지 못한다.

③ 단계 3의 T 연산과정에서 혹은 R_1에 오류가 주입된 경우

정상적인 서명인 경우 $T=0$가 된다. 그러나 이 과정에서 랜덤한 값이 주입되면 단계 4에서 오류가 확산되어 역시 랜덤한 서명 값을 출력한다. 즉, 단계 1, 2, 3에서 발생한 오류는 대부분 T값을 0으로 만들지 못하며 이는 단계 4에서 오류가 확산된 서명 값을 만든다.

④ 단계 4의 재결합 과정에서 오류가 주입된 경우

단계 4에서 $S_p = S_{pr} \bmod p$ 연산 과정에서 오류가 주입되었다고 가정하자. 그러면 단계 5에서 $S-(S_{pr}+R) \bmod p$ 값이 0이 되지 않아 c가 랜덤한 값이 되고 랜덤한 오류 서명 값을 출력한다.

⑤ 단계 5의 지수를 계산하는 과정에서 오류가 주입된 경우

정상적인 경우에는 $S-(S_{pr}+R) \bmod p$과 $S-(S_{qr}+R) \bmod p$이 0이 되고 c값은 1이 된다. 그러나 이 과정에서의 오류가 발생하면 c를 1이 아닌 랜덤한 값으로 만들므로 단계 6을 거치면 서명 값은 공격에 유용한 정보가 되지 못한다.

⑥ 단계 6의 과정에서 오류가 주입된 경우

이 경우에 오류가 발생하면 정확한 서명을 출력하지 못할 뿐만 아니라 단계 5까지의 연산은 랜덤한 R이 S값에 포함되어 있어 공격에 필요한 서명 값은 되지 못한다.

⑦ 단계 1과 6에서 이중 오류가 주입된 경우

문헌 [5]에서의 공격과 같이 단계 1에서 S_{pr}에 오류를 주입하고 단계 6의 과정을 수행하지 않더라도 단계 4에서 오류가 확산될 뿐만 아니라 랜덤수 R이 더해져서 오류 공격에 대응할 수 있다.

⑧ 파라미터(p, q, p_I, q_I, r_1, r_2, N)에 오류가 주입된 경우

단계 1에서 파라미터를 읽어올 경우 오류가 발생하면 T가 0이 될 수 없고 따라서 단계 4에서 오류가 확산되므로 공격이 성공할 수 없다. 또한, 단계 4, 5에서 사용되는 파라미터 오류는 c를 1로 만들지 못하며 단계 6에서의 N에 대한 오류 주입 공격 역시 공격에 유용한 서명 값을 얻을 수 없다.

⑨ 메시지 m과 지수(d_p, d_q, e_p, e_q)에 대한 오류 공격이 있을 경우

단계 1과 2를 수행하는 과정에서 메시지 m과 지수(d_p, d_q, e_p, e_q)에 대한 일시적 오류가 있을 경우는 T가 0이 될 수 없으므로 단계 4에서 오류가 확산된다.

따라서 제안 알고리듬에서는 공격자가 연산 오류, 레지스터 오류, 파라미터 오류 등 여러 가지 공격을 가하더라도 공격에 유용한 서명 값을 얻을 수 없다. 제안 알고리듬에 사용된 멱승 알고리듬이 전력분석 공격과 같은 다른 부채널 공격에 안전한지도 살펴보아야 한다. 예를 들어 Giraud 방법[16]은 Montgomery Ladder 알고리듬을 고정적으로 사용하게 되는데, 이 알고리듬은 오류 주입 공격을 방어할 수는 있지만 단순 전력분석 공격에 취약한 특성이 있으므로 주의해야 하며 다른 전력

〔표 1〕 오류 공격 방어용 CRT-RSA의 안전도 비교

구 분	안전성 여부	공격 형태 및 안전성
Shamir [10]	X	재결합 과정 취약, d 사용
Joye et al. [11]	X	재결합 과정 취약, d_p에 오류 주입 공격 가능
Aumüller et al. [4]	X	d_p에 오류 주입 공격 가능, 영구적 오류에 취약
BOS [12]	X	Wagner 공격에 취약, d 사용
Ciet-Joye [15]	X	d_p에 오류 주입 공격 가능
Giraud [16]	△	멱승 알고리듬이 전력분석 공격에 취약
BNP [18]	X	d_p에 오류 주입 공격 가능, L-to-R이진 멱승 방법만 사용
Kim et al. [20]	O	공개 키 e사용, 큰 e인 경우 계산량이 많음
Kim-Quisquater [21]	O	논문 [20]의 멱승알고리듬과 BOS 기법의 결합
Proposed	O	e_p, e_q 사전 계산, 오류 확산용 추가 레지스터 사용

분석 공격 대응 알고리듬과 병용하여야 한다. 특히, CRT-RSA 알고리듬은 멱승 시 사용하는 모듈러스(modulus)가 비밀히 사용되기 때문에 차분 전력 분석(differential power analysis)에는 강인한 특성은 있으나 문헌 [17]과 같은 Relative doubling 전력 공격에 취약한 경우는 주의해야 한다. 예로서 문헌 [22]에서 제안한 스칼라 곱셈 알고리듬을 응용하여 CRT-RSA 멱승 연산을 수행하면 다양한 전력 분석 공격도 방어할 수 있다.

제안 기법에서는 사전 계산 과정을 통해 작은 크기의 오류 검사용 e_1과 e_2를 생성하여 사용하므로 계산량이 최소화되도록 하였다. 또한, 모든 연산에 사용된 모듈러스가 최대 N을 넘지 않도록 하여 기존 시스템과의 호환성을 유지하고자 하였다. 이외에도 별도의 d나 e가 필요하지 않은 장점이 있어 구현 측면에서도 유리하다. 그러나 타 방식에 비해 사전 역수 과정과 저장해야 할 파라미터가 약간 증가하는 단점은 있다. 다음의 [표 1]은 앞서 언급한 안전도 분석들을 요약하여 나타내고 있다.

5.2. 연산량

[표 2]에서는 기존의 오류 주입 공격 대응책들과 제안하는 기법의 계산량을 비교하기 위해 CRT-RSA 연산에서 계산량에 가장 큰 영향을 미치는 모듈러스 비트 길이와 지수의 비트 길이를 나타내고 있다. 계산량 비교에는 편의상 사전 연산 과정과 멱승 이외의 연산은 제외하였다. 계산량의 비교를 위해 필요한 표기는 아래와 같다.

- $l=|p|=|q|$: 두 소수의 길이(bit)
- $n=|r_1|=|r_2|=|r|$: 모듈러 연산체 확장을 위한 랜덤수의 길이(bit)
- $k=|e|$: 멱승에서 지수(exponent)로 사용되는 정수의 길이(bit)
- a^b : a비트 길이를 갖는 모듈러 연산체내에서 이루어지는 b비트의 지수 멱승

  (예, $S=m^{d_p} \bmod p$에서 $l=512$, $k=512$이면 $a^b=l^k=512^{512}$로 표시)

[표 2]에서는 CRT-RSA 연산량을 100%로 두고 각 대응 알고리듬의 연산량을 이론적으로 비교 분석하였다. 일반적으로 연산 비트가 l인 $GF(p)$ 체상에서 모듈라 덧셈(addition)은 길이의 $O(l)$의 시간 복잡도를 가지며 모듈라 곱셈(multiplication)은 $O(l^2)$, 그리고 멱승은 $O(l^3)$의 시간 복잡도를 가진다. 따라서 표 2에서 보는 바와 같이 CRT-RSA에 비해 일반 RSA는 약 4배의 연산량을 가지게 된다.

비슷한 방법으로 표 2의 연산량을 산출해 보면, 먼저 $(l+n)^{(l+n)}$의 연산량은 l^l의 연산량에 비해 $((l+n)/l)^3$의 연산량을 가진다. 예를 들어 $l=512$ 이고 $n=80$이면 약 1.5458배의 연산량을 갖는다. 또한, 모듈라 곱셈의 연산체만 확장된 경우 $(l+n)^l$의 연산량은 l^l의 연산량에

〔표 2〕 오류 공격 방어용 CRT-RSA의 계산량 비교($l=512$, $n=80$일 때)

구 분	멱승 연산 길이	연산량 비율(%)	비고
RSA [1]	$(2l)^{2l}$	400.00	-
CRT-RSA [2]	$2(l^l)$	100.00	-
Shamir [10]	$2((l+n)^{(l+n)})$	154.58	연산체 확장, 지수 확장
Joye et al. [11]	$2((l+n)^l+n^n)$	134.07	연산체 확장, n-bit 모듈러스에서 멱승 두 번 수행
Aumüller et al. [4]	$2((l+n)^{(l+n)}+n^n)$	154.96	연산체 확장, 지수 확장 n-bit 모듈러스에서 멱승 두 번 수행
BOS [12]	$2((l+n)^{(l+n)}+n^n)$	154.96	연산체 확장, 지수 확장 n-bit 모듈러스에서 멱승 두 번 수행
Ciet-Joye [15]	$2((l+n)^l+n^n)$	134.07	연산체 확장, n-bit 모듈러스에서 멱승 두 번 수행
Giraud [16]	$2(l^l)$	100.00	멱승 연산의 중간값 이용.
BNP [18]	$2(l^l)$	100.00	멱승 연산의 중간값 이용.
Kim et al. [20]	$4(l^l)$	200.00	공개 키 e 사용
Kim-Quisquater [21]	$2((l+n)^l+2n^n)$	155.34	연산체 확장, 지수 확장 n-bit 모듈러스에서 멱승 네 번 수행
Proposed	$2((l+n)^l+n^n)$	134.07	연산체 확장, n-bit 모듈러스에서 멱승 두 번 수행

비해 $((l+n)/l)^2$의 연산량을 가지므로 1.3369배의 연산량을 갖는다. 마지막으로 n^n의 연산량은 l^l의 연산량에 비해 $(n/l)^3$의 연산량을 가지므로 0.0038배의 연산량을 갖는다.

이와 같은 방법으로 이론적 계산량을 산출해 보면 제안하는 알고리듬은 문헌 [11]와 [15] 방식과 비슷한 연산량을 가지며 일반 CRT-RSA보다 약 1.34배 정도의 오버헤드를 갖게 된다. 또한, 현재까지 공격되지 않아 안전한 것으로 보이는 Kim 등의 방식[20, 21]에 비해 매우 효율적임을 알 수 있다.

V. 결 론

본 논문에서는 고속 RSA 서명이나 복호에 사용되는 CRT-RSA에 대한 오류 주입 공격을 분석하였다. 기존의 방법들은 새로운 공격에 지속적으로 위협받고 있으며 이에 대한 대응책이 필요하다. CRT-RSA에 대한 오류 주입 공격 대응책은 다양한 상황의 오류 주입 가능성에 대하여 안전해야 하며 계산 속도, 시스템 구현의 효율성, 호환성을 제고하여야 한다. 본 논문에서는 안전한 CRT-RSA 알고리듬을 구현하기 위해 오류 확산 기법을 사용하였다. 제안하는 기법은 멱승 연산 시 발생한 오류가 재결합 과정의 소수 p나 q에 직접적으로 영향을 주게 함으로써 공격이 불가능하도록 하였으며 이중 오류 주입 공격에도 대응할 수 있다. 또한, 멱승 값의 정확성을 점검하는 과정에서 짧은 길이의 지수를 사용하도록 설계하여 일반 CRT-RSA에 비해 계산량이 크게 늘어나지 않는 장점이 있어 계산 능력과 구현 여건이 부족한 스마트카드나 암호 칩 디바이스를 설계 시 매우 효과적으로 사용할 수 있다.

참고문헌

[1] R. Rivest, A. Shamir, L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Comm. of the ACM* 21, pp. 120 - 126, 1978.

[2] C. Couvreur, J. J. Quisquater, "Fast decipherment algorithm for RSA public-key cryptosystem," *Electronics Letters* Vol. 18 pp. 905 - 907, 1982.

[3] J. S. Coron, "Resistance Against Differential Power Analysis for Elliptic Curve Cryptosystems" *Cryptographic Hardware and Embedded Systems - CHES'99.* LNCS Vol. 1717, pp. 292 - 302, 1999.

[4] C. Aumüller, P. Bier, W. Fischer, P. Hofreiter, and J. P. Seifert, "Fault attacks on RSA with CRT: Concrete results and practical countermeasures," *Cryptographic Hardware and Embedded Systems - CHES '02,* LNCS Vol. 2523, pp. 260-275, 2002.

[5] C. H. Kim and J. J. Quisquater, "Fault Attacks for CRT Based RSA: New Attacks, New Results, and New Countermeasures," *Workshop in Information Security Theory and practices-WISTP'07,* LNCS Vol. 4462, pp. 215-228, 2007.

[6] 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.

[7] 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.

[8] S. M. Yen, S. J. Kim, S. G. Lim, and S. J. Moon, "RSA speedup with residue number system immune against hardware fault cryptanalysis," *International Conference on Information Security and Cryptology - ICISC'01* LNCS V.2288, pp.397-413, 2001.

[9] S. M. Yen, S. J. Moon, and J. C Ha, "Hardware fault attack on RSA with CRT revited," *International Conference on Information Security and Cryptology-ICISC'02*, LNCS 2587, pp. 374-388, 2003.

[10] A. Shamir, "Method and apparatus for protecting public key schemes from timing and fault attacks," United States Patent p5,991,415, November 23, 1999. Also presented at the rump session of EUROCRYPT'97.

[11] M. Joye, P. Pailler, S. M. Yen, "Secure evaluation of modular functions," International *Workshop on Cryotpology and Network*

Security 2001, pp.227-229, 2001.

[12] J. Blömer, M. Otto, and J. P. Seifert, "A new CRT-RSA algorithm secure against Bellcore attacks," *10th ACM Conference on Computer and Communications Security,* pp. 311-320, 2003.

[13] D. Wagner, "Cryptanalysis of a provably secure CRT-RSA algorithm," *11th ACM Conference on Computers and Comm- unications Security,* pp. 92-97, 2004.

[14] J. Blömer and M. Otto, "Wagner's attack on a secure CRT-RSA algorithm recondiered," *Fault Diagnosis and Tolerance in Cryptography -FDTC'06* LNCS Vol. 4236, pp. 13-23, 2006.

[15] M. Ciet and M. Joye, "Practical fault countermeasures for Chinese Remain- dering based RSA," *Fault Diagnosis and Tolerance in Cryptography -FDTC'05,* pp. 124-131, 2005.

[16] C. Giraud, "Fault resistant RSA implementation," *Fault Diagnosis and Tolerance in Cryptography-FDTC'05,* pp. 142-151, 2005.

[17] S. M. Yen, L. C. Ko, S. J. Moon and J. C. Ha, "Relative Doubling attack against Montgomery Ladder," *International Conference on Information Security and Cyptography-ICISC'05,* LNCS 3935, pp. 117-128, 2006.

[18] A. Boscher, R. Naciri, and E. Prouff, "CRT-RSA Algorithm Protected Against Fault Attacks," *Workshop in Information Security Theory and practices-WISTP'07,* LNCS Vol. 4462, pp. 237-252, 2007.

[19] 권은정, 신종훈, 이필중, "SPA-FA에 안전한 exponentiation 알고리듬에 대한 Fault Attack," *한국정보보호학회 하계학술대회(CISC-S'07) 논문집*, pp. 345-249, 2007.

[20] C. K. Kim, J. C. Ha, S. H. Kim, S. K. Kim, S. M. Yen, and S. J Moon, "A secure and practical CRT-Based RSA to resist side channel attacks," *International Conference on Computational Science and Its Applications-ICCSA'04*, LNCS 3043, pp. 150-166, May, 2004

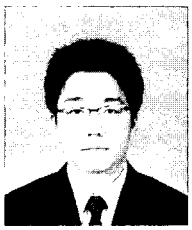
[21] C. H. Kim and J. J. Quisquater, "How can we overcome both side channel analysis and fault attacks on RSA-CRT?," *Fault Diagnosis and Tolerance in Crptography-FDTC'07*, pp. 21-29, 2007

[22] J. C. Ha, J. H. Park, S. J. Moon, and S. M. Yen, "Provably Secure Countermeasure Resistant to Several Types of Power Attack for ECC," *Workshop on Information Security Applications-WISA'07,* LNCS 4867, pp. 333-344, 2007.

〈著者紹介〉

하 재 철 (JaeCheol Ha) 종신회원

1989년 2월 : 경북대학교 전자공학과 졸업
1993년 8월 : 경북대학교 전자공학과 석사
1998년 2월 : 경북대학교 전자공학과 박사
1998년 3월~2006년 1월 : 나사렛대학교 전자계산소장, 학술정보관장, 입시학생처장
1998년 3월~2007년 2월 : 나사렛대학교 정보통신학과 부교수
2006년 7월~2006년 12월 : QUT in Australia 연구 교수
2007년 3월~현재 : 호서대학교 정보호호학과 부교수
2002년 3월~현재 : 한국정보보호학회 이사
<관심분야> 정보보호, 네트워크 보안, 스마트카드 보안



박 제 훈 (JeaHoon Park) 학생회원

2004년 2월 : 경북대학교 전자·전기공학부 졸업
2006년 2월 : 경북대학교 전자공학과 석사
2006년 3월~현재 : 경북대학교 전자공학과 박사과정
<관심분야> 정보보호, 네트워크 보안, 스마트카드 보안

문 상 재 (SangJae Moon) 종신회원

1972년 2월 : 서울대학교 공업교육(전자전공)과 학사
1974년 2월 : 서울대학교 전자공학과 석사
1984년 6월 : 미국 UCLA 전기공학과 박사
1984년 7월~1985년 6월 : UCLA Postdoctor 근무
1984년 7월~1985년 6월 : 미국 OMNET 컨설턴트
1997년 9월~1998년 8월 : 경북대학교 전자전기공학부 학부장
1974년 12월~현재 : 경북대학교 전자전기컴퓨터공학부 교수
2000년 8월~현재 : 경북대학교 이동네트워크 정보보호기술 연구센터장
2002년 2월~현재 : 한국정보보호학회 명예회장
<관심분야> 정보보호, 디지털 통신, 이동 네트워크