Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.31, NO.2, Apr. 2021 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2021.31.2.157

부채널 공격에 안전한 전자서명 알고리즘 연구

이 훈 희,[1†] 홍 석 희[2‡]
[1,2]고려대학교 (대학원생, 교수)

A Study on Secure Digital Signature Algorithm for Side Channel Attack

HunHee Lee,[1†] SeokHie Hong[2‡]
[1,2]Korea University (Graduate student, Professor)

요 약

부채널 공격은 수학적으로 증명된 암호 알고리즘의 안전성을 손상시킬 수 있는 강력한 기술이다. 본 논문에서는 국제 표준 디지털 서명 알고리즘인 DSA(Digital Signature Algorithm)에 대한 부채널 공격 방법과 기존 대응 방법 중 가장 안전한 Kim 등이 제안한 알고리즘을 분석 하였다. 또한, Kim 등이 제안한 알고리즘의 문제점을 개선하여 안전성은 높이고 연산량은 줄이는 새로운 DSA 서명 알고리즘을 제안한다. Kim 등이 제안한 알고리즘은 모든 오류 주입 공격에 안전하지만 전력 분석 공격에 대해서는 고려되지 않았으며, 연산량이 크다는 단점이 있다. 본 논문에서 제안하는 알고리즘은 난수 2개를 사용하여 Kim 등의 대응 방법과 유사하지만, 기존 알고리즘과 달리 서명 연산 중 비밀키 값이 난수와의 곱셈으로 수행되기 때문에 오류 주입 공격뿐만 아니라 전력 분석 공격에도 안전하며, 연산 효율은 약 34% 향상 되었다.

ABSTRACT

Side channel attack is a powerful technique that can threaten the security of mathematically proven cryptographic algorithms. In this paper, side channel attack methods for DSA(Digital Signature Algorithm) and an algorithm proposed by Kim, the safest among existing countermeasures, were analyzed. In addition, a new DSA signature algorithm was proposed that increases safety and computational efficiency by improving the problems of Kim's algorithm. The Kim's algorithm is secure against all fault injection attacks, but it's not considered for power analysis attacks and requires a lot of computation. The new algorithm proposed in this paper is similar to the Kim's algorithm by using two nonce, but it's secure against not only fault injection attacks but also power analysis attacks, because secret key is multiplied by a nonce and used to generate the signature. Also, the computational efficiency was improved by about 34% compared to the Kim's algorithm.

Keywords: DSA, Side Channel Attack, Countermeasures

I. 서 론

IoT(Internet of Things), 빅 데이터, 인공지능 등 방대한 정보를 필요로 하는 산업들이 발전하면서 정보 보안의 중요성이 더욱 대두되고 있다. 대부분의 정보보호 디바이스들은 암호 알고리즘을 사용하여 중요 정보들을 보호하고 있는데, 부채널 공격은 이러한 암호 알고리즘의 안전성을 손상시킬 수 있는 강력한 기술이다. 그 결과 여러 제품들의 부채널 공격 사례가 발표되고 있으며, 공격 방법 및 대응방법에 대한 연구들이 활발하게 진행되고 있다.

부채널 공격은 대표적으로 오류 주입 공격과 전력 분석 공격으로 나누어 볼 수 있다.

오류 주입 공격은 1997년 Boneh 등에 의해 처

Received(01. 29. 2021), Modified(02. 17. 2021), Accepted(03. 04. 2021)
† 주저자, lycoslhh@gmail.com
‡ 교신저자, shhong@korea.ac.kr(Corresponding author)

음 소개된 공격으로서, 암호 알고리즘 동작 중에 오류를 주입하는 기술이다[1]. 디바이스 외부에서 글리치, 레이저, 전자기파 등의 물리적인 영향을 주어 오류를 발생 시키고, 변조된 정보들을 이용하여 비밀 정보를 찾아낼 수 있다.

전력 분석 공격은 1996년 Paul Kocher가 처음 소개한 공격으로서, 암호 알고리즘 동작 중에 발생하는 물리적 정보들을 분석하는 기술이다[2,3]. 디바이스에서 암호 알고리즘 동작으로 발생하는 전력, 전자기파 등을 측정하고, 수집된 정보들을 이용하여 비밀 정보를 찾아낼 수 있다.

DSA(Digital Signature Algorithm)에 대한 오류 주입 공격은 1997년 Bao 등에 의해서 처음 소개 되었다[4]. 이후 Giraud 등[5], Nikodem 등[6,7], Naccache 등[8], Schmidt 등[9], Jung 등[10], Kim 등[11]에 의해서 새로운 오류 주입 공격 및 이에 대응하는 안전한 알고리즘들이 제안 되었다. 그러나 이와 같은 알고리즘들은 전력 분석 공격에 대해서는 고려되지 않았다. 따라서 DSA 알고리즘을 실제 구현할 때 전력 분석 공격의 대응방법[12,13] 등을 추가적으로 적용해야 하고, 이는 많은 연산량을 필요로 하게 된다.

본 논문에서는 기존에 제안된 DSA 오류 주입 공격의 대응 방법들 중 가장 안전한 Kim의 알고리즘을 분석한 후, 새로운 DSA 서명 알고리즘을 제안한다. 제안하는 알고리즘은 난수를 2개 사용하여 Kim의 대응 방법과 유사하지만, 기존과 다르게 비밀키를 직접 사용하지 않고 서명을 생성하기 때문에 오류 주입 공격 뿐만 아니라 전력 분석 공격에도 안전하도록 설계하였다. 이로 인하여 Kim의 방법 대비 연산량도 감소시킬 수 있었다.

본 논문은 다음과 같이 구성하였다. 2장에서 기존 부채널 공격과 대응 방법에 대해 살펴보고, 3장에서는 이러한 취약점을 보완하여 부채널 공격에 안전하고 효율적인 DSA 서명 알고리즘을 제안한다. 그리고 4장에서 결론을 맺는다.

II. DSA 알고리즘의 부채널 공격 및 기존 대응 방법 분석

2.1 DSA 알고리즘

DSA는 1991년 NIST(National Institute of Standard and Technology)에서 제안 되었으며, 1994년 12월에 미국의 전자 서명 표준으로 제정 되었다[14]. DSA는 이산대수 문제에 안전성을 기반으로 하고 있고, Schnorr 서명 기법[15]의 장점을 더욱 발전시킨 서명 기법이다.

DSA 전자 서명의 키 생성, 서명, 검증 알고리즘의 동작 방법은 **Fig.1, 2.**와 같다.

올바른 서명 값(r,s)과 메시지 m은 식(1)을 만족해야 한다. 식(1)을 이용하여 식(2)와 같이 표현

Output: public key (p,q,g,y),
secret key d

1. Select a prime number q, such that $2^{159} < q < 2^{160}$.
2. Choose t so that $0 \le t \le 8$, and select a prime number p where $2^{511+64t} < p < 2^{512+64t}$, with the property that q divides $(p-1)$.
3. Select a generator g of the unique cyclic group of order q in Z_p^*.
4. Select a random integer d, such that $1 \le d \le q-1$.
5. $y = g^d \bmod p$.

Fig. 1. Key generation for DSA algorithm

Input: public key (p,q,g,y)
secret key d, message m

1) Signature generation

1. Select a random secret integer k, such that $0 < k < q$.
2. $r = (g^k \bmod p) \bmod q$, If $r=0$ then go to step 1.
3. $k^{-1} \bmod q$.
4. $s = k^{-1}(h(m) + dr) \bmod q$, If $s=0$ then go to step 1.
5. Signature for m is the pair (r,s).

2) Verification

1. $w = s^{-1} \bmod q$.
2. $v_1 = h(m)w \bmod q$, $v_2 = rw \bmod q$
3. $v = (g^{v1} y^{v2} \bmod p) \bmod q$
4. Accept the signature if $v = r$

Fig. 2. Signature generation and verification for DSA algorithm

할 수 있고, **Fig.2.** 의 verification 단계 2,3을 수행하면 식(3)이 된다. 결국 식(3)은 $v = r$과 동일하므로 **Fig.2.** 가 성립 한다

$$h(m) \equiv -dr + ks \,(\mathrm{mod} q) \tag{1}$$

$$h(m)w + drw \equiv k \,(\mathrm{mod} q) \tag{2}$$

$$(g^{v1} y^{v2} \mathrm{mod} p) \,\mathrm{mod} q = (g^k \mathrm{mod} p) \,\mathrm{mod} q \tag{3}$$

2.2 DSA 부채널 공격 방법

2.2.1 오류 주입 공격

현재까지 알려진 DSA의 오류 주입 공격 방법들은 비밀키 또는 난수에 대한 오류 주입과 제곱 연산에 대한 오류 주입으로 나누어 볼 수 있다.

● **비밀키에 대한 오류 주입**

DSA 서명 알고리즘에 대한 최초의 오류 주입 공격 방법으로서 Bao 등[4]이 비밀키 한 비트의 오류 주입 공격 방법을 소개하였고, 이를 바이트 단위로 확장한 방법을 Giraud 등[5]이 제안하였다. 그들은 공격자들이 서명 알고리즘이 수행하는 동안에 비밀키 d가 저장되어 있는 레지스터에 오류를 주입할 수 있다고 가정하였고, 다음의 분석 과정을 통해 비밀키를 찾아낼 수 있음을 증명하였다.

Step 1) 비밀키 d의 i번째 비트 오류를 가정한다. 이를 수식으로 표현하면 $d \pm 2^i$이다.

Step 2) 공격자는 오류가 주입된 비밀키 $\hat{d}$로 생성된 아래의 서명 값을 얻을 수 있다.

$r = (g^k \mathrm{mod} p) \,\mathrm{mod} q$

$\hat{s} = k^{-1}(h(M) + \hat{d}r) \,\mathrm{mod} q$

Step 3) 공격자는 오류의 서명 값 $\hat{s}$와 메시지 M을 이용하여 다음을 계산한다.

$\hat{w} = \widehat{s^{-1}} \mathrm{mod} q$

$\hat{v_1} = g^{h(M)\hat{w}} \mathrm{mod} p, \hat{v_2} = y^{r\hat{w}} \mathrm{mod} p$

$V = \hat{v_1} \bullet \hat{v_2} \mathrm{mod}\, p$

$= g^{h(M)\hat{w}} \bullet y^{r\hat{w}} \mathrm{mod} p$

$= g^{(\hat{w}h(M) + dr) \mathrm{mod} q)} \mathrm{mod} p$

그리고 검사 값 R_i을 계산한다.

$$R_i = (g^{r\hat{w}2^i}) \mathrm{mod} p, (i = 0, 1, \cdots, t-1) \tag{4}$$

Step 4) 공격자는 다음의 두 식이 성립하는지 검사하여 비밀키 중 한 비트를 찾아낼 수 있다.

$T \bullet R_i \mathrm{mod} q = r$ 이면 $x_i = 0$ 이다.

$T / R_i \mathrm{mod} q = r$ 이면 $x_i = 1$ 이다.

공격자는 서로 다른 i에 대해 반복적으로 **Step 4**를 수행하여 비밀키 d의 i번째 비트 값을 찾을 수 있다. (바이트 단위의 공격 방법도 위의 비트 공격과 유사한 분석 과정을 가지므로 생략한다.)

● **난수에 대한 바이트 리셋 오류 주입**

난수에 대한 오류 주입 공격 방법은 Naccache 등[8]에 의해 소개되었다. 그들은 공격자들이 난수가 생성되는 동안 오류를 주입하여 난수의 하위 바이트들을 0으로 리셋 시키는 것을 가정한다. 이러한 가정하에 공격자는 난수의 부분 정보를 알아낼 수 있고, 이들의 서명 쌍을 가지고 lattice 공격 방법에 적용하여 비밀키를 찾아낼 수 있음을 보여 주었다.

● **제곱 연산을 생략하는 오류 주입**

제곱 연산에 대한 오류 주입 공격은 Schmidt 등[9]이 ECDSA을 대상으로 처음 소개하였고, Jung 등[10]이 이를 DSA에 적용하였다. 그들은 난수 k의 지수승 연산이 수행되는 $r = g^k \mathrm{mod} p$ 의 $(t-i)$번째 제곱 연산을 생략하는 오류를 주입하는 것을 가정한다. 연산 오류를 통해 k의 부분 정보를 알아낼 수 있고, 이러한 서명 쌍들을 이용해 lattice 공격을 수행할 수 있음을 보여 주었다. k의 부분정보 k'를 알아내는 과정은 다음과 같다.(여기서 k'는 난수 k의 i번째 이후의 비트 값을 나타낸다.)

Step 1) 서명 생성 $r = g^k \mathrm{mod} p$의 연산 중 $(t-i)$번째 제곱 연산을 생략하는 오류를 주입하고, 오류의 서명 $\hat{r}$, $\hat{s}$를 가진다.

Step 2) 오류의 서명 값 $\hat{r}$, $\hat{s}$을 이용하여 아래의 식을 만족하는 $\sqrt{g^k} \,\mathrm{mod} p$ 을 계산한다.

$$\sqrt{g^k} \,\mathrm{mod} p = \sqrt{g^{(\hat{s})^{-1} h(m)} \mathrm{mod} p} \bullet \sqrt{y^{(\hat{s})^{-1} \hat{r}} \mathrm{mod} p}$$

Step 3) 위 과정에서 얻은 $\hat{r}$과 $\sqrt{g^k} \,\mathrm{mod} p$를 이용하여 식(5)를 만족하는 k'를 찾을 수 있다.

$$\hat{r} = (\sqrt{g^k} \bmod p \cdot \sqrt{g^{k'}} \bmod p) \bmod q \qquad (5)$$

위 공격에서는 $(\hat{s})^{-1}h(m) \bmod q$, $(\hat{s})^{-1}\hat{r} \bmod q$, k 값이 모두 짝수라는 제한 조건이 있다.

2.2.2 전력 분석 공격

DSA의 전력 분석 공격은 비밀키 d와 난수 k에 대한 공격으로 나누어 볼 수 있다.

- **비밀키에 대한 전력 분석**

DSA의 안전성은 비밀키 d에 의존하므로 d가 연산되는 $k^{-1}(h(M)+dr) \bmod q$ 등을 통해 전력 분석 공격을 수행할 수 있다. 비밀키 d는 항상 고정값이고, 이와 연산되는 r은 항상 랜덤하게 바뀌기 때문에 일반적인 전력 분석 방법 DPA (Differential Power Analysis), CPA (Correlation Power Analysis) 등의 공격들을 수행할 수 있다.

- **난수에 대한 전력 분석**

Nguyen 등[16,17]은 난수 k의 일부 정보가 노출된 서명 쌍들을 수집하여 lattice 공격으로 비밀키를 복구할 수 있음을 보여 주었다. 이는, 난수 또한 전력 분석 공격의 대상이 될 수 있다는 것이다. 즉, 난수 생성 알고리즘[18] 또는 난수 k가 연산되는 $r = (g^k \bmod p) \bmod q$ 등을 통해 전력 분석 공격을 수행할 수 있다.

2.3 DSA 부채널 공격 기존 대응 방법

부채널 공격 중 전력 분석 공격은 masking과 같은 일반적인 대응 방법들이 사용되고 있다[12,13]. 그리고 오류 주입 공격의 대응 방법들은 Bao가 공격 방법을 처음 제안한 이후 다양한 방법들이 연구되어 왔다. 본 논문에서는 이러한 오류주입 공격의 대응 방법들 중 가장 안전한 Kim 등이 제안한 방법에 대해서 설명한다.

2.3.1 Kim 등의 방법

Kim 등[11]이 제안하는 방식은 Nikodem[6]의 오류 주입 공격에 안전한 DSA 서명 알고리즘을 변형 시킨 것으로 **Fig.3.** 에서 기술한다.

Input: secret key d, public key y
message M, generator g
modulus p, q
Output: signature(r, s)

1. Select random number k_1, k_2
   with $\left\lceil \frac{q}{2} \right\rceil < k_1,\ k_2 < q$
2. $k = k_1 k_2 \bmod q$
3. $r = (g^k \bmod p) \bmod q$
4. $v = k_1 + dr \bmod q$
5. $V = ((g^v y^{-r} \bmod p)^{k_2} \bmod q) - r \bmod q$
6. $k' = (k \oplus V)^{-1} \bmod q$
7. $s = k'(h(m) + v) - k_2^{-1} \bmod q$

Fig. 3. Kim's DSA signature algorithm

서명 생성 중에 오류가 주입되면 서명문 전체에 오류가 확산 되고, 이렇게 확산된 오류는 검증 단계를 통해 확인할 수 있도록 설계하였다. 또한, 난수의 오류 주입 공격을 막기 위해서 두 개의 난수 k_1, k_2를 선택하여 새로운 난수 k를 생성한다. 단계 4, 5가 오류 주입 여부를 검증하는 단계이다. 만약 단계 5 이전에 비밀키 d 혹은 난수 k에 오류가 주입될 경우, $v \neq 0$이 되어 서명 s는 추측하기 어려운 난수가 된다. 또한 **Fig.3.** 의 서명 알고리즘을 사용하더라도 일반적인 DSA 서명 알고리즘과 동일하게 서명의 유효함을 검증할 수 있다.

Kim 등이 제안한 DSA 알고리즘은 비밀키 오류, 난수에 대한 오류 등 알려진 오류 주입 공격들에는 안전하나 연산량이 크다는 단점이 있다. 또한 전력 분석 공격에 안전하기 위해서는 추가적인 대응 방법이 필요하므로 더욱 많은 연산량이 요구 된다.

III. 제안하는 DSA 서명 알고리즘

본 장에서는 위에서 설명한 Kim 등이 제안한 알고리즘의 단점들을 해결하는 새로운 DSA 서명 알고리즘을 제안하고, 이에 대한 안전성을 분석한다.

3.1 제안하는 DSA 서명 기법

제안하는 새로운 DSA 서명 알고리즘은 **Fig.4.** 에서 기술한다. 이는 오류주입 공격 및 전력분석 공격을 막기 위해서 아래의 해결 방법들을 수행한다.

두개의 난수 k_1, k_2를 선택하여 서명 생성에 사용한다. 이 때 k_1, k_2의 비트 길이는 k의 반으로 설정한다. 두 개의 난수를 사용하여 서명 값 r을 생성하기 때문에 두 번의 지수승 연산을 수행하게 된다. 하지만, 두 난수 곱이 k와 같기 때문에 연산량은 기존과 동일하다.

비밀키 d를 공격자가 알 수 없는 난수 k_1과의 곱셈과 그 결과의 역원으로 t_1을 계산하고, 이를 서명 생성에 사용한다.

제안하는 DSA 서명 방법은 정리 1.과 같이 기존 DSA와 동일한 서명 값이 생성되므로, 서명 검증 또한 동일하게 수행 할 수 있다.

정리 1. **Fig.4.** 는 표준 DSA와 동일한 서명 결과를 생성한다.

증명. 식(6)과 식(7)을 이용하여 증명할 수 있다.

$$\begin{aligned} r &= (v^{k_2} \bmod p) \bmod q = ((y^{t1})^{k2} \bmod p) \bmod q \\ &= (y^{d^{-1}k_1^{-1}k_2} \bmod p) \bmod q \\ &= ((g^d)^{d^{-1}k_1^{-1}k_2} \bmod p) \bmod q \\ &= (g^{k_1^{-1}k_2} \bmod p) \bmod q \\ &= (g^k \bmod p) \bmod q \end{aligned} \quad (6)$$

$$\begin{aligned} s &= t_2 + r(t_1 k_2)^{-1} \bmod q \\ &= h(M)(k_1 k_2^{-1}) + r(d^{-1}k_1^{-1}k_2)^{-1} \bmod q \\ &= (k_1 k_2^{-1})(h(M) + dr) \bmod q \\ &= k^{-1}(h(M) + dr) \bmod q \end{aligned} \quad (7)$$

Input: secret key d, public key y, message M, generator g, modulus p, q
Output: signature(r, s)

1. Select random number k_1, k_2 with $0 < k_1, k_2 < \frac{q}{2}$
2. $t_1 = (dk_1)^{-1} \bmod q$
3. $v = y^{t_1} \bmod p$
4. $r = (v^{k_2} \bmod p) \bmod q$
5. If $r = 0$ then go to step1
6. $t_2 = h(M)k_1 k_2^{-1} \bmod q$
7. $s = t_2 + r(t_1 k_2)^{-1} \bmod q$

Fig. 4. Proposed DSA signature algorithm

추가로 상기 수식들에 대한 시뮬레이션 검증을 진행하였다. **Fig.5.**과 **Fig.6.**는 제안하는 DSA 서명 방법과 표준 DSA 알고리즘을 Maple 6. 계산 프로그램을 사용하여 검증한 결과이다. 입력 파라미터들은 CAVS (Cryptographic Algorithm Validation System)[19]의 test vector를 사용하였고, 동일한 k값으로 연산할 경우 서명 값 r, s가 일치하는 것을 확인할 수 있다. □

```
k1:= 128467230848540:
k2:= (k*k1)modq:
t1:=((d*k1)^(-1))modq:
v:=mod_exp(y,p,t1):
r:=(mod_exp(v,p,k2))modq:
t2:=(h_M*k1*(k2^(-1)))modq:
s:=(t2+(r*((t1*k2)^(-1))))modq:
r:=convert(r, hex, decimal);
s:=convert(s, hex, decimal);

r:=0xED4715B8D218D31B7ADF0BEA5165777A7414315E
s:=0x29C70A036AA83EB0742F1FA3F56CCEAD0FC0F61D
```

Fig. 5. Simulation of proposed DSA.

```
r:=(mod_exp(g,p,k))modq:
t:=(k^(-1))modq:
s:=(t*(h_M+(d*r)))modq:
r:=convert(r, hex, decimal);
s:=convert(s, hex, decimal);

r:=0xED4715B8D218D31B7ADF0BEA5165777A7414315E
s:=0x29C70A036AA83EB0742F1FA3F56CCEAD0FC0F61D
```

Fig. 6. Simulation of original DSA

3.2 안전성 분석

3.2.1 오류 주입 공격에 대한 안전성 분석

현재까지 알려진 DSA 서명 알고리즘의 오류주입 공격 방법은 세 가지이다. 1) 비밀키 d에 대한 오류 주입, 2) 제곱 연산을 생략하는 오류 주입, 3) 난수 k에 대한 byte reset 오류 주입 방법들이 있다. 각 공격 방법들의 안전성을 분석하면 정리 2.와 같다.

정리 2. **Fig.4.**의 제안하는 알고리즘은 상기 3가지 공격에 안전하다.

증명.

Case 1) 비밀키 d에 대한 오류 주입: 제안하는 방법은 비밀키 d를 단계 2에서 한 번 사용하는데, 오류가 주입된 경우를 $\hat{d} = d \pm \alpha$로 표현하여 서명 값을 생성하면 식(8), 식(9)와 같다

$$\begin{aligned}\hat{r} &= ((\hat{v})^{k_2} \bmod p) \bmod q = ((y^{\hat{t_1}})^{k2} \bmod p) \bmod q \\ &= (g^{d})^{(\hat{d}k_1)^{-1}k_2} = ((g^{d})^{(d\pm\alpha)^{-1}k} \bmod p) \bmod q \\ &= ((g^{k})^{d(d\pm\alpha)^{-1}} \bmod p) \bmod q \qquad (8)\end{aligned}$$

$$\begin{aligned}\hat{s} &= t_2 + \hat{r}(t_1 k_2)^{-1} \bmod q \\ &= h(m)k_1 k_2^{-1} + \hat{r}(\hat{d}k_1)^{-1}k_2)^{-1} \bmod q \\ &= h(m)k^{-1} + \hat{r}((d\pm\alpha)^{-1}k)^{-1} \bmod q \\ &= k^{-1}(h(m) + (d\pm\alpha)\hat{r}) \bmod q \qquad (9)\end{aligned}$$

서명 값 r, s는 각각 $g^{d(d\pm\alpha)^{-1}}$와 $(d\pm\alpha)\hat{r}$의 확산된 오류의 값을 가지는 랜덤 값으로 생성되기 때문에 공격자는 비밀 정보 없이 식(4)의 검사 값 R_i를 구할 수 없다. 따라서 제안하는 방법은 비밀키에 대한 오류 주입 공격으로부터 안전하다.

Case 2) 제곱 연산을 생략하는 오류 주입: 제안하는 방법은 단계 3, 단계 4에서 제곱 연산이 수행되므로 각 단계로 나누어 안전성을 분석하였다.

● 단계 3의 i번째 연산 중 제곱 연산이 생략 되어 졌다면 식(10), 식(11)을 만족하게 된다.

$$\hat{v} = \left(\sqrt{y^{\sum_{j=i+1}^{t} 2^j (dk_1)^{-1}_j}}\right)^{\sum_{j=0}^{i} 2^j (dk_1)^{-1}_j} \bmod p \qquad (10)$$

$$\hat{r} = ((\hat{v})^{k_2} \bmod p) \bmod q \qquad (11)$$

공격자는 난수 k의 i번째 이후 값인 k'를 찾아내기 위해 식(5)를 수행하게 된다. 제안하는 알고리즘의 경우 오류의 서명 값 $\hat{r}$이 k의 부분정보가 아닌 k_2와 k_1의 부분 정보의 곱으로 계산되기 때문에 공격자는 식(5)를 만족하는 k'를 찾아낼 수 없다.

● 단계 4의 i번째 연산 중에 제곱 연산이 생략 되어 졌다면 식(12), 식(13)를 만족하게 된다.

$$v = y^{(dk_1)^{-1}} \bmod p = g^{k_1^{-1}} \bmod p \qquad (12)$$

$$\hat{r} = ((\sqrt{v^{\sum_{j=i+1}^{t} 2^j k_{2j}}})^{\sum_{j=0}^{i} 2^j k_{2j}} \bmod p) \bmod q \qquad (13)$$

공격자는 난수 k의 i번째 이후 값인 k'를 찾아내기 위해 식(5)를 수행하게 된다. 제안하는 알고리즘의 경우 오류의 서명 값 $\hat{r}$이 k의 부분정보가 아닌 k_1과 k_2의 부분정보의 곱으로 계산되기 때문에 공격자는 식(5)를 만족하는 k'를 찾아낼 수 없다. 따라서 제안하는 알고리즘은 제곱 연산을 생략하는 오류 주입 공격에 안전하다.

Case 3) 난수 k에 대한 바이트 리셋 오류 주입: 난수 k_1의 최하위 한 바이트가 0이 되었다고 가정하면, 단계 2의 t_1을 계산할 때 모듈러스 곱셈 연산을 수행하기 때문에 최하위 바이트는 0이 아닌 랜덤 값으로 생성 된다. 이를 이용하여 서명 값을 생성하면 랜덤 오류가 확산 되어 k의 부분 정보를 알 수 없는 서명 값을 가지게 된다. 따라서 공격자는 k의 부분 정보를 알 수 없기 때문에 lattice 공격에 적용할 수 없다. 이는 k_2 혹은 k_1, k_2 동시에 바이트의 리셋 오류가 주입되더라도 동일하게 랜덤 서명 값이 생성되므로 제안하는 방법은 난수의 바이트 리셋 오류 주입 공격에 안전하다. □

3.2.2 전력 분석 공격에 대한 안전성 분석

DSA 서명 알고리즘의 전력 분석 공격 방법은 두 가지이다. 1) 비밀키 d에 대한 전력 분석, 2) 난수 k에 대한 전력분석 방법이 있다. 각 공격 방법들의 안전성을 분석하면 정리 3.과 같다.

정리 3. **Fig.4.** 의 제안하는 알고리즘은 상기 공격에 안전하다.

증명.

Case 1) 비밀키 d에 대한 전력 분석: 제안하는 방법은 단계 2에서 비밀 키 d의 연산이 수행된다. 단계 2는 비밀키 d와 공격자가 알 수 없는 난수 k_1의 곱셈과 역원 계산으로 이루어져 있고, 그 연산 결과인 t_1 역시 공격자가 알 수 없는 값이다. 따라서 전력 분석 공격을 위한 조건이 성립하지 않으므로 안전하다.

Table 1. Computational Complexity of DSA Algorithms.

Method	Computational Complexity (p=512, q=160)	
Original DSA	($\|p\|$ modulus exp.×1)+($\|q\|$ inversion×1)+($\|q\|$ modulus mul.×2) ⇒ $(3p^3+6p^2+3p)+(5q^2+4q)$ = 404,356,224	1
Kim's DSA	($\|p\|$ modulus exp.×3)+($\|q\|$ modulus exp.×1)+($\|p\|$ inversion×1) +($\|q\|$ inversion×2)+($\|p\|$ modulus mul.×1)+($\|q\|$ modulus mul.×3) ⇒ $(9p^3+21p^2+11p)+(3q^3+14q^2+9q)$ = 1,226,118,048	3.03
Proposed DSA	**($\|p\|$ modulus exp.×2)+($\|q\|$ inversion×3) +($\|q\|$ modulus mul.×5)** **⇒ $(6p^3+12p^2+6p)+(13q^2+10q)$ = 808,789,568**	**2.00**

Case 2) 난수 k에 대한 전력 분석: 제안하는 방법은 두 개의 난수 k_1, k_2를 사용하여 서명 값을 생성한다. 전력 분석 공격으로 k_1 혹은 k_2를 알아낼 수 있다고 가정하여도 이들은 난수 k의 부분정보가 아니기 때문에 lattice 공격에 사용될 수 없다. 따라서 제안하는 알고리즘은 난수 k에 대한 전력 분석 공격에 안전하다.

Case 3) 추가적으로 단계 3과 단계 4에서는 공격자가 t_1과 k_2를 알아낼 수 없도록 지수승 연산의 SPA(Simple Power Analysis) 공격을 막기 위한 atomicity 혹은 montgomery ladder 방법을 사용하여 구현해야 한다. 본 논문에서는 atomicity modulus exponentiation 방법을 사용할 경우의 수식과 연산량을 계산하였다. □

3.3 기존 대응 방법과 비교

제안하는 DSA 서명 방법과 기존 DSA 서명 방법들의 부채널 공격에 대한 안전성을 비교하면 **Table.2.** 과 같다. 이 때, 제곱 연산에 대한 오류 주입 공격은 결국 난수의 부분 정보를 알기 위한 공격이므로 난수 공격에 포함 하였다.

Table.1. 는 original DSA와 Kim의 DSA, 그리고 제안하는 DSA 서명 알고리즘에 대한 연산량 비교 결과이다. 이 때, $|a|$는 a의 bit 길이를 의미하고, exp.는 지수승 연산으로 atomicity modulus exponentiation 방법을 사용하였고, mul.은 곱셈 연산인 montgomery modulus multiplication 방법을 사용하여 연산량을 계산하였다.

연산량 비교 결과 제안하는 DSA 서명 알고리즘의 연산량은 표준 DSA 알고리즘 대비 약 2배이며, 이는 Kim의 알고리즘에 비하여 약 34프로 개선된 수준이다. 즉, 전력 분석 공격에 대한 안전성도 가지면서 더욱 효율적인 알고리즘을 제안하였다.

Table 2. Safety Comparison of DSA Algorithms.

Method	Fault Injection		Power Analysis
	Secret Key	Nonce	
Original	weak	weak	weak
Nicodem	secure	weak	weak
Bae	secure	weak	weak
Jung	secure	weak	weak
Kim	secure	secure	weak
Proposed	**secure**	**secure**	**secure**

IV. 결 론

본 논문에서는 DSA 서명 알고리즘에 대한 부채널 공격 방법들과 기존 대응 방법 중 가장 안전한 Kim이 제안한 알고리즘을 살펴보았다. 또한 Kim의 알고리즘이 갖고 있는 문제점을 개선하여 안전성은 높이고 연산량은 줄이는 새로운 DSA 서명 알고리즘을 제안 하였다. 제안하는 알고리즘은 오류 주입 공격에 대해서 Kim의 알고리즘과 동일한 안전성을 가지면서 전력 분석 공격에도 안전하며, 연산량은 Kim의 알고리즘에 비하여 약 34% 감소하였다.

References

[1] D. Boneh. R. A. DeMillo and R. J. Lipton, "On the Importance of Checking Cryptographic Protocols for Faults," EUROCRYPT-1997, LNCS 1233, pp.37-51, 1997.

[2] P. Kocher, J. Jaffe, and B. Jun, "Timing Attacks on Implementations of DiffieHellman, RSA, DSS,and Others Systems," CRYPTO-1996, LNCS 1109, pp.104-113, 1996.

[3] P. Kocher, J. Jaffe, and B. Jun, "Introduction to differential power analysis and related attacks," White Paper, Cryptography Research, http://www.cryptography.com/dpa/technical, 1998.

[4] F. Bao, R. H. Deng, Y. Han, A. Jeng, A. D. Narasimhalu, and T. Ngair, "Breaking Public Key Cryptosystems on Tamper Resistant Devices in the Presence of Transient Faults," International Workshop on Security Protocols-1997, LNCS 1361, pp. 115-124, 1997.

[5] C. Giraud and E. Knudsen, "Fault Attacks on Signature Schemes," ACISP-2004, LNCS 3108, pp. 478-491, 2004.

[6] M. Nikodem, "DSA Signature Scheme Immune to the Fault Cryptanalysis," CARDIS-2008, LNCS 5189. pp. 61-73, 2008.

[7] M. Nikodem, "Error Prevention, Detection and Diffusion Algorithms for Cryptographic Hardware," RELCOMEX'07, pp. 127-134, June. 2007.

[8] D. Naccache, P. Nguyen, M. Tunstall and C.Whelan, "Experimenting with Faults, Lattices and the DSA," PKC-2005, LNCS 3386, pp. 16-28, 2005.

[9] J. Schmidt, and M. Medwed, "A Fault Attack on ECDSA," Workshop on Fault Diagnosis and Tolerance in Cryptography, pp. 93-99, Sep. 2009.

[10] C. Jung, D. Oh, D. Choi, H. Kim and J. Ha, "Cryptanalysis using Fault Injection and Countermeasures on DSA", Journal of The Korea Academia-industrial Cooperation Society, 11(8), pp. 3045-3052, Aug. 2010.

[11] T. Kim, T. Kim, S. Hong and Y. Park, "A new digital signature scheme secure against fault attacks," Journal of The Korea Institute of Information Security and Cryptology, 22(3), pp. 515-524, June. 2012.

[12] T. Messerges, "Power Analysis Attacks and Countermeasures for Cryptographic Algorithms," Ph.D Thesis, Univ. of Illinois at Chicago, pp. 541-548, 2000.

[13] J. Coron and Louis Goubin "On Boolean and Arithmetic Masking against Differential Power Analysis", CHES'00, LNCS 1965, pp. 231-237, 2000.

[14] "National institute of standards and technology," FIPS PUB 186-2: Digital Signature Standard, 2000.

[15] C.P. Schnorr, "Efficient Identification and Signatures for Smart cards", CRYPTO'89, LNCS 435, pp. 239-251, July. 2001.

[16] P. Nguyen and J. Stern, "Lattice Reduction in Cryptology: An Update", ANTS'00, LNCS 1838, pp. 85 - 112, 2000.

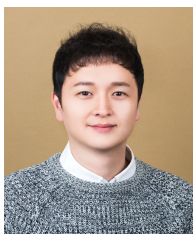
[17] P. Nguyen and I. Shparlinski, "The Insecurity of the Elliptic Curve Digital Signature Algorithm with Partially Known Nonces", Designs. Codes Cryptography. vol. 30, no.2, pp. 201 - 217, 2003.

[18] M. Bellare, S. Goldwasser, and D.

Micciancio, "Pseudo-Random Number Generation Within Cryptographic Algorithms: The DDS Case". CRYPTO'97, LNCS 1294, pp. 277 - 291, 1997.

[19] NIST CAVP: FIPS 186-2 DSA CAVS, "https://csrc.nist.gov/CSRC/media/Projects/Cryptographic-Algorithm-Validation-Program/documents/dss/186-2dsatestvectors.zip".

〈저자소개〉



이 훈 희 (HunHee Lee) 정회원
2013년 2월: 광운대학교 전파공학과 학사
2014년 9월~현재: 고려대학교 정보보호대학원 석사과정
〈관심분야〉 부채널 공격, 암호 알고리즘

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