Attacks on RSA 조사

[제출물] 공격 정의, 동작원리가 기술된 문서

[문제 1] 각 RSA의 공격에 대한 내용을 기술하고 동작과정을 설명하시오.

(1) Factorization

: Fermat number(페르마 소수)를 사용하면 소인수분해가 매우 쉬워져서, N을 소인수분해하는 데 걸리는 시간이 매우 짧아져 p, q를 알 수 있고, 개인 키를 얻을 수 있다. p/q의 값이 1에 근사하면 RSA 암호화 방식은 Fermat factorization 공격에 취약

(2) Chosen-ciphertext(CCA, 선택 암호문 공격)

: RSA가 갖는 homomorphism을 이용한 공격. RSA와 같은 키로 생성된 서로 다른 암호문 두 개를 곱하면, 평문 두 개의 곱을 암호화한 것과 결과가 같다.

$$(r \times m^*)^e = (r)^e \times (m^*)^e$$

textbook RSA에서 주로 사용된다.

$$C = M^e \mod N$$

$$C' = C \times r^e \mod N$$

$$(C')^d = (C \times r^e)^d = (M^e \times r^e)^d = M^{e \times d} \times r^{e \times d} = M \times r$$

C를 알고 있다고 가정하면 r^e를 만들어 C'를 만들고 C'에 대한 서명(d)를 한다.

이것으로부터 M*r을 얻고 r은 임의로 만든 값이므로 M을 구할 수 있다.

- 평문 M을 암호화한 것이 C라고 하면 C' = C*2^e mod n
- C'를 복호화키 d로 복호화하면 2*m mod n을 구할 수 있다.

(3) Encryption exponent

$$C = M^e \mod n$$

(M: 평문, C: 암호문.)

n, e를 통해 암호화를 진행

(4) Decryption exponent

$$M = C^d \ mod \ n$$

(M: 평문, C: 암호문.)

$$c^d \equiv (m^e)^d \equiv m \pmod{n}$$

를 통해 복호화

(5) Plaintext Attack

: 암호문에 대응하는 일부 평문을 아는 상황. 평문을 다 알지는 못하지만 일부 알려진 평문과 암호문의 관계로부터 키와 전체 평문을 추론하는 과정. 공격자가 키를 알아낸다면 같은 키로 암호화된 모든 암호문은 복호화 가능

(6) Modulus

- 공격자가 두 수신자의 동일한 메시지 m의 두 암호문 c1, c2와 수신자의 공개키 e1, e2가 서로소임을 알고 있을 때,

1. re_1 + se_2 = 1을 계산

2. r이 음수라고 가정(r과 s 중 하나는 반드시 음수)

3. 확장 유클리드 알고리즘을 통해 c_1^{-1}을 계산한다.

4. $(c_1^{-1})^{-r} \cdot c_2^s \equiv (m^{-e_1})^{-r} \cdot m^{e_2 \cdot s} \equiv m^{re_1+se_2} \equiv m \pmod{n}$

(7) Implementation Attack(Timing Attack, Power Attack)

- Timing Attack(소요 시간 공격): 계산을 하는데 소요되는 시간을 측정하는 것을 기반으로 하는 공격 CPU나 메모리가 이동하는 시간을 측정하여 전체 비밀키가 무엇인지 알아낼 수 있다. 소요 시간의 통계적 분석 과정을 거쳐서 알아낼 수 있다.

- Power Attack(전력 분석 공격): '0', '1'을 처리할 때 소비되는 전력이 서로 다르다는 점을 이용하여, device에서 암호 알고리즘이 실행되는 동안 소비 전력을 분석하여 비밀 키에 대한 정보를 얻어내는 방법