마스킹 환경에서의 Crystals-KYBER 의 구현에 대한 고찰

김금태, 노종선
서울대학교

kkt1513@snu.ac.kr, jsno@snu.ac.kr

요 약

본 논문은 암호화 표준으로 선정된 Crystals-KYBER 알고리즘의 마스킹 구현이 어려운 점을 짚고, compression 함수와 소수 modulus 가 가지는 역할에 대해 논한다. 또한, 마스킹 기법이 부채널 공격에 대한 대응기법임을 설명하고 안전성을 유지하면서 효율적으로 구현할 수 있는 방법에 대해 논한다.

Ⅰ. 서 론

현대의 암호 시스템은 대부분 계산적 안전성에 기반하고 있다. 가장 널리 알려진 RSA 암호화[1] 방식은 아주 큰 두 소수의 곱으로 이루어진 수의 소인수 분해가 어렵다는 사실에 기반하며, 일정시간내에 비밀키를 찾아낼 확률이 현저히 낮다는 사실로 암호 알고리즘을 안전하게 만든다. 그러나 이 안전성은 어디까지나 현재까지의 기술을 바탕으로 한 상황이다.

1994 년 피터쇼어에 의해 제안된 Shor's algorithm[2]은 큰 정수를 소인수분해하는데 효과적인 알고리즘으로, 현대의 비트기반 컴퓨터에서는 현실적으로 오래걸리고 어려운 소인수분해 문제를 양자컴퓨터를 활용하여 빠른시간내에 찾아내는 알고리즘이다. 이 알고리즘의 제시로 인해 훨씬 더 어려운 문제에 기반한 알고리즘에 대한 필요성이 제시되었다.

이 문제에 발맞추어 2016 년 미국의 NIST 에서는 양자내성암호에 대한 표준화작업을 실시하였다[3]. 총 82 개의 제안된 알고리즘들은 3 단계의 선정작업 이후 Crystals-KYBER 가 최종 키 캡슐레이션 알고리즘으로 선정되었다. KYBER 는 MLWE 에 기반한 격자기반 암호 알고리즘으로, 각 암호문의 계수들은 modulus 3329 내에 존재한다.

안전성에 대한 기준으로 부채널 안전성 또한 제시되었는데, 해당 알고리즘이 부채널 분석에 대해서도 특정 레벨 이상의 부채널 안전성을 보여야 한다는 것이었다. 부채널 분석(Side Channel Analysis)이란 알고리즘이 동작하는 장비의 실제 구현과정에서 누출하는 정보인 전력소비량, 수행시간, 전자기파 등을 이용하여 비밀키에 접근하는 공격법을 말한다.

부채널 분석에 대응하는 대표적인 기법 중 하나인 마스킹(masking)은 민감한 데이터를 여러 조각들로 나누어 처리함으로써, 중간값 중 일부가 노출되더라도 다른 부분들에 대한 정보를 알 수 없도록 만들어 원본 비밀정보에 대한 안전성을 유지하는 방어법이다. 마스킹은 크게 arithmetic masking 과 Boolean masking 으로 구분할 수 있다. Arithmetic masking 은 원본 데이터 x 를 modulus q 상의 여러 개의 다른 조각들로 쪼개어 표현하는 방법으로, $x = x_1 + x_2 + \cdots + x_n mod\ q$ 으로 표현된다. 반면, Boolean masking 은 여러 조각들이 XOR 로 연결된 방법으로, $x = x_1 \oplus x_2 \oplus \ldots \oplus x_n$으로 표현된다.

Ⅱ. 본론

마스킹을 적용하여 본 알고리즘을 부채널 분석에 내성을 가지도록 만들게 되면, 필수적으로 연산시간이 오래걸리게 된다. 마스킹 되지 않은 상태에서 원본 데이터에 가할 수 있는 연산은, 마스킹된 환경에서 새로운 연산으로 재정의 되어야한다. 예시로, arithmetically masking 된 어떤 데이터 x 와 y 을 더하는 덧셈을 구현하고자하면, 각 조각들을 따로 더하는 $z_i = x_i + y_i$의 방식으로 구해야한다. 이 마스킹된 환경에서의 덧셈은 여전히 결과값도 마스킹된 값들로 나오게된다.

KYBER 는 이 마스킹 환경을 구축할 경우 다른 알고리즘들보다 더 추가적인 연산을 요하는데, 그 이유는 compression 함수와 소수 modulus 인 3329 때문이다. Compression 함수는 암호문의 효율적인 전송을 위하여 채널에 보내기 전 mod 3329 내의 각 계수를 mod 2^{10} 혹은 mod 2^{11} 의 값으로 압축시키는 과정이다. 이 compression 함수는 masking 된 환경에서 구현하기 까다로운데, 그 이유는 암호문의 각 계수가 arithmetically mask 되어있기 때문이다. 만약

Boolean masking 되어 있었다면, 단순히 뒤의 일부분을 자르는 방식으로 compression 을 수행할 수 있지만, arithmetic masking 은 필수적으로 carry 를 고려해 주어야 하며, 원본 데이터에 대한 연산을 어렵게 만든다. 따라서 compression 함수는 arithmetic masking 된 원본 데이터를 Boolean masking 으로 바꾸는 작업을 포함하게 되는데, 이를 A2B (arithmetic to Boolean) masking conversion 이라 한다.

2 의 거듭제곱을 modulus 로 사용하는 다른 알고리즘과는 다르게 KYBER 에서는 소수인 3329 를 사용하는데, 이는 compression 연산을 더욱 어렵게 만든다. 단순 bit shift 연산이 아닌 추가 precision 을 요구하며, KYBER 의 security level 에 따른 적절한 precision 이 이미 제시된 바 있다[4].

따라서 compression 함수를 문제없이 사용하고 성능 열화를 막기 위해서는 효율적인 A2B conversion 방법이 중요하다.

Ⅲ. 결론

양자내성암호 표준으로 선정된 KYBER 에 masking 기법을 적용할 경우 적지 않은 성능열화가 생긴다. 그러나 부채널 공격에 대응하기 위해서는 필수적으로 masking 기법이 들어가야하며, 합당한 연구의 방향은 security level 을 만족하는 마스킹 기법을 적용하면서, 최대한 성능열화를 막도록 하는 방향이 될것이다.

ACKNOWLEDGMENT

(국문) 본 연구는 삼성전자의 지원 (과제번호: MEM 210728_0001)을 받아 수행된 결과임

참 고 문 헌

[1] Rivest, Ronald L., Adi Shamir, and Leonard Adleman. "A method for obtaining digital signatures and public-key cryptosystems." *Communications of the ACM* 21.2 (1978): 120-126.

[2] Shor, Peter W. "Algorithms for quantum computation: discrete logarithms and factoring." *Proceedings 35th annual symposium on foundations of computer science*. Ieee, 1994.

[3] https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022

[4] Fritzmann, Tim, et al. "Masked accelerators and instruction set extensions for post-quantum cryptography." *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2022.1 (2021): 414-460.