NIST PQC 암호 최적화 구현 동향

Trends in Optimized Implementations of NIST PQC Cryptography

엄시우*, 송민호**, 김상원**, 서화정**†
* 한성대학교 대학원 정보컴퓨터공학, **한성대학교 대학원 융합보안공학

ABSTRACT

양자컴퓨터의 빠른 발전은 현재의 안전하다고 알려진 공개키 암호 알고리즘을 위협하고 있다. NIST에서는 양자컴퓨터로부터 안전한 양자내성암호를 개발하고자 공모전을 개최하였으며, 2022년 4개의 최종 알고리즘이 선정되었다. 양자내성암호는 기존 암호 알고리즘에 비해 많은 암호화 시간을 필요로 하며 또한 많은 메모리를 요구한다. 이러한 문제로 인해, 암호화 시간을 최소화 하거나 제한된 환경에서도 활용할 수 있도록 메모리를 최소화 하는 방법등 여러 최적화 연구가 진행되고 있다. 본 논문에서는 최근 PQC 최적화 연구에 대해서 조사하여 소개하고 이를 통해 최신 PQC 최적화 구현 기법에 대해 확인한다.

I. NIST PQC

현재 안전하다고 알려져 사용되고 있는 공개키 암호 알고리즘(RSA, ECDSA)은 양자컴퓨터의 발전으로 안정성이 위협받고 있다. 이로 인해 미국 국립표준기술연구소(National Institute of Standards and Technology, NIST)는 양자컴퓨터로부터 안전한 양자내성암호(Post Quantum Cryptography, PQC) 표준화를 위한 공모전을 개최하였다[1]. Crystals-Kyber는 LWE의 변형인 Module RLWE를 기반으로 하는 KEM 알고리즘이다. Crystals-Dilithium는 Kyber와 마찬가지로 LWE의 변형인 Module RLWE를 기반으로 하는 전자 서명 알고리즘이다. Falcon은 NTRU 문제에 기반한 전자 서명 알고리즘이다. Sphincs+는 해시 트리 기반의 전자 서명 알고리즘이다.

II. HI-Kyber: A novel high-performance implementation scheme of Kyber based on GPU

GPU를 활용하여 Kyber 최적화 구현인 HI-Kyber를 소개하고 있다. HI-Kyber는 Kyber 알고리즘의 핵심 연산인 NTT, INTT, SHA3, Encode, Decode 등에 대해 전역 메모리 접근 횟수를 최소화하고 전체 알고리즘의 성능을 향상 시키는 효율적인 커널 융합 스킴을 제안하고 있다. 또한 핵심 연산중 하나인 NTT 연산의 병목 현상을 해결하기 위해 SLM, SDFS-NTT, EDFS-NTT 세가지 계산 체계를 제안하고 있다. SLM(Sliced Layer Merging) 체계는 NTT 계수의 여러 레이어를 재사용함으로써 글로벌 메모리의 메모리 접근 횟수를 줄이는 것을 목표로 하는 체계이다. SDFS-NTT(Sliced depth-first search NTT)와 EDFS-NTT(Entire depth-first search NTT)는 깊이 우선 탐색 기법을 활용하는 새로운 접근 방식은 제안하였다. 효율적인 연산을 위해 데이터를 NTT 계수의 연속적인 부분을 포함하는 슬라이스로 나누고 병렬로 처리할 수 있도록 구성하였다. 이를 통해 메모리 접근은 최소화하고 레지스터의 활용을 최대화하여 성능 향상을 보여주고 있다. 이러한 기법을 통해 결과적으로 각각의 계산 체계는 일반적인 구현에 비해 7.5%, 28.5% 41.6%의 성능 향상을 보여주고 있다. 또한 전체적인 성능은 기존의 GPU 최적화 구현 대비 3.52배 높은 성능을 보여주고 있으며, AI tensor core를 활용한 구현의 성능 대비 1.78배 높은 성능을 보여주었다.

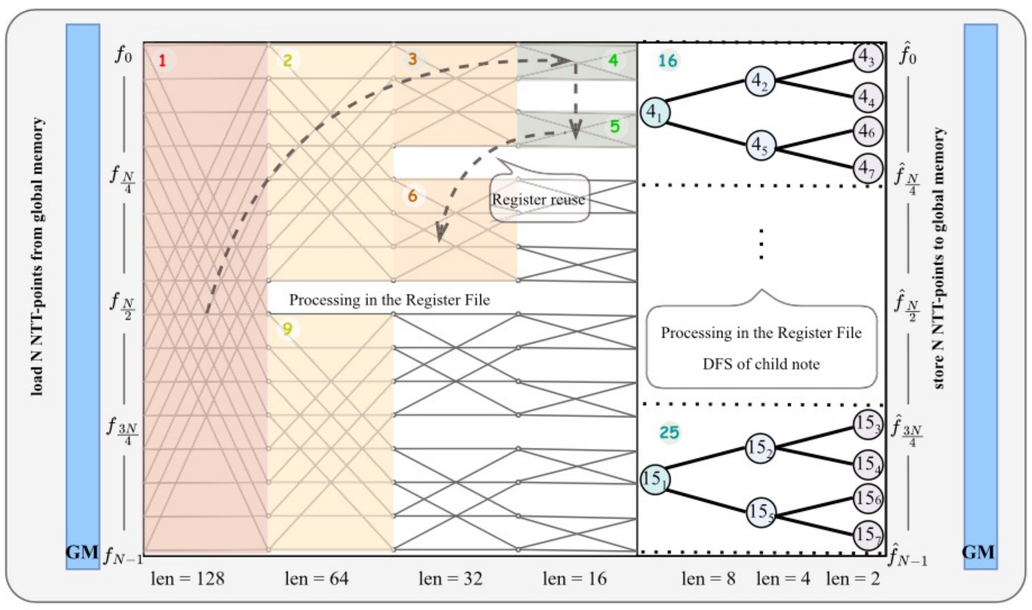

Fig. 7. Entire depth-first search scheme for NTT.

III. Revisiting Keccak and Dilithium Implementations on ARMv7-M

ARMv7-M에서 기존의 Keccak 최적화 구현을 분석하고 기존의 Keccak 구현을 개선하여 최적화 구현을 하였다. Keccak의 최적화 구현을 위해 Pipelining 메모리 접근 최적화와 Lazy rotations 최적화 기법을 제안하였다. Pipelining 메모리 접근 최적화는 Cortex-M3에서 메모리 로드 명령어(ldr)와 연산 명령어를 교대로 사용하는 것이 아니라 메모리 로드 명령어를 그룹화하여 한번에 실행하고 연산 명령어가 실행하도록 함으로써 Pipeline 지연을 최소화 하는 방법이다. Lazy rotation은 인라인 배럴 시프터를 활용하며 rotations 과정을 미루고 생략함으로써 최적화를 하는 방법이다. 결과적으로 이러한 기법을 통해 ARMv7-M에서 12.84%의 성능 향상을 보여주었으며, 이외에도 NTT 연산 최적화도 제안하였다.

```
.macro xor5    result,b,g,k,m,s
    ldr        \result, [r0, #\b]
    ldr        r1, [r0, #\g]
    eors       \result, \result, r1
    ldr        r1, [r0, #\k]
    eors       \result, \result, r1
    ldr        r1, [r0, #\m]
    eors       \result, \result, r1
    ldr        r1, [r0, #\s]
    eors       \result, \result, r1
.endm
```

Listing 2: Original ARMv7-M assembly code from [BDH+] to compute half a parity lane. Loads from memory are not fully grouped and thus not optimally pipelined on M3 and M4 processors.

```
.macro xor5  result,b,g,k,m,s
    ldr      \result, [r0, #\b]
    ldr      r1,  [r0, #\g]
    ldr      r5,  [r0, #\k]
    ldr      r11, [r0, #\m]
    ldr      r12, [r0, #\s]
    eors     \result, \result, r1
    eors     \result, \result, r5
    eors     \result, \result, r11
    eors     \result, \result, r12
.endm
```

Listing 3: ARMv7-M assembly code after optimization to compute half a parity lane. Loads from memory are now fully grouped and thus optimally pipelined on M3 and M4 processors.

IV. 결 론

본 논문에서는 2022년 NIST PQC 공모전에서 최종 선정된 4개의 PQC 알고리즘의 최신 최적화 구현에 대해서 조사하였다. 기존의 PQC 알고리즘의 최적화 구현은 다항식 곱셈을 최적화하기 위한 NTT의 최적화와 같이 특정 플랫폼에서 NTT 최적화 구현 연구에 취중되어 있었다면, 최근의 최적화 구현 연구는 NTT 최적화 구현 연구와 함께 Keccak과 같은 내부에서 활용되는 해시 함수의 최적화, AI 가속기의 활용 그리고 GPU의 높은 병렬 연산 기능 활용 등 여러 기술과 플랫폼을 활용하는 연구가 진행되고 있는 것을 확인하였다. 이러한 연구 결과는 국내에서 진행되고 있는 KPQC 공모전에 제출된 알고리즘과 Additional Digital Signature 공모전에 제출된 알고리즘에 적용함으로써 여러 PQC 알고리즘의 성능 향상에 이바지 될 수 있으며, 또다른 새로운 연구 방향을 제시할 수 있을 것으로 기대된다.