PQ-DPoL 에 대한 양자 내성 전자서명 벤치마킹

강예준[1], 김원웅[1], 김현지[2], 장경배[2], 서화정[3]
[1]한성대학교 IT 융합공학과 석사과정
[2]한성대학교 정보컴퓨터공학과 박사과정
[3]한성대학교 융합보안학과 교수

etus1211@gmail.com, dnjsdndeee@gmail.com, khj1594012@gmail.com, starj1023@gmail.com, hwajeong84@gmail.com

Benchmarking of Post Quantum Digital Signature for PQ-DPoL

Yea-Jun Kang, Won-Woong Kim[1], Hyun-Ji Kim[2], Kyung-Bae Jang[2], Hwa-Jeong Seo[3]
[1]Division of IT Convergence Engineering, Han-Sung University
[2]Information Computer Engineering, Han-Sung University
[3]Dept. of Convergence Security, Han-Sung University

요 약

쇼어 알고리즘을 실행할 수 있는 양자 컴퓨터의 발전으로 인해 기존 ECC(Elliptic Curve Cryptography)를 사용하던 블록체인이 PQC(Post Quantum Cryptography)로의 전환을 고려하고 있다. 하지만 PQC 는 기존 암호에 비해 큰 사이즈, 느린 서명/검증 속도 등과 같은 문제점이 존재한다. 본 논문에서는 우리가 WISA'23 에서 새롭게 제안한 PQ-DPoL 합의 알고리즘에 NIST(National Institute of Standards and Technology)가 선정한 Crystal-Dilithium, Falcon 그리고 Sphincs+를 적용하여 비교 분석하였다. 실험 결과에 따르면, 매우 큰 서명 크기를 가지고 있기 때문에 블록에 담기는 트랜잭션의 수가 감소하므로 Sphincs+의 성능이 가장 떨어짐을 확인하였다. 또한 Dilithium 은 Falcon 과 비슷한 성능을 보여주었다. 그 중에서도 Falcon 이 가장 우수한 성능을 보여주었다. 이는 Falcon 의 공개키와 서명의 크기가 다른 알고리즘에 비해 작기 때문이다. 따라서 양자내성을 갖는 블록체인에는 Falcon 512 알고리즘이 가장 적합할 것으로 생각된다. 그러나 블록체인의 속도와 보안 강도는 Trade-off 관계이므로 보안성을 중요시하는 블록체인 네트워크에서는 Sphincs+가 적합할 수 있을 것으로 보인다. 따라서 블록체인 네트워크의 상황과 목적에 따라 적절한 알고리즘을 사용해야 될 것으로 생각된다.

1. 서론

Shor 알고리즘[1]을 실행할 수 있는 양자 컴퓨터로 인해 기존 ECC 를 사용하던 블록체인이 PQC 로의 전환을 시도하고 있다. 이는 기존 ECC 를 사용하던 블록체인이 양자 컴퓨터로부터의 위협에 대응하기 위해서이다.

블록체인에 PQC 를 적용할 때에는, 블록체인 네트워크의 규모와 요구하는 보안성 등을 고려하여 적절한 양자 내성 암호를 적용해야 한다. 하지만 PQC 는 기존 암호에 비해 큰 사이즈, 느린 서명/검증 속도 등으로 인한 블록체인 네트워크의 성능이 저하되는 문제점이 존재한다.

본 논문에서는 위임 기능을 적용하여 성능 저하를 완화시킨 PQ-DPoL 합의 알고리즘에 NIST 가 선정한 PQC 를 적용하여 성능을 측정하고 비교 분석한다.

본 논문의 구성은 다음과 같다. 2 장에서는 합의 알고리즘과 적용된 PQC 에 대해 설명한다. 3 장에서는 WISA'23 에서 제안한 우리의 합의 알고리즘에 대한 간단한 구현에 대해 다루고, 4 장에서는 PQC(Crystal-Dilithium, Falcon, Sphincs+)를 적용하였을 때의 블록체인의 성능 평가를 진행한다. 마지막으로 5 장에서 결론을 맺는다.

2. 관련 연구

2.1 블록체인 합의 알고리즘

블록체인은 네트워크 내의 노드들이 Peer-to-Peer 방식으로 통신하여, 원장을 공유하는 분산 원장 기술이다[2]. 따라서 블록체인은 각 노드들이 원장을 소유하고 있는 탈중앙화 방식이며, 노드들이 직접 블록을 생성하고 트랜잭션을 검증한다.

합의 알고리즘은 블록체인 상의 노드들이 특정한 절차를 통해 데이터의 무결성을 보장하고, 동일한 의사결정을 하기 위해 사용된다. 각 합의 알고리즘 별로 블록 생성자와 검증자가 존재한다. 블록 생성자는 트랜잭션이 담긴 블록을 검증자들에게 전송한다. 검증자는 블록의 헤더 값이 유효한지 검증한다. 또한 검증자는 전송 받은 블록에 담긴 트랜잭션의 서명 값을 확인하여 검증을 수행한다.

이러한 합의 알고리즘에는 다양한 방식이 존재한다. 대표적으로 비트코인에서 사용되는 작업증명(Proof of Work)과 이더리움의 지분증명(Proof of Stake)[3]이 있다. 또한 PoS 에 위임 기능을 추가한 DPoS(Delegated Proof of Stake)[4]가 있으며, TEE 를 기반으로 하는 PoL(Proof of Luck)[5]과 PoET(Proof of Elapsed Time)[6] 등이 있다.

2.2 양자 내성 전자서명

2.2.1 CRYSTAL-DILITHIUM[7]

Crystal-Dilithium 은 에러를 삽입하여 결과값의 무작위성을 증가시키는 LWE(Learning With Error)를 기반으로하는 격자 기반 전자서명 알고리즘이다. LWE 는 양자 알고리즘으로도 효율적으로 풀리지 않으므로 Crystal-Dilithium 은 양자 컴퓨터에 내성을 가진다. Crystal-Dilithium 은 보안 수준에 따라 Dilithium-2, 3, 5 로 구분된다. Table 1 은 Crystal-Dilithium 의 공개키와 개인키 그리고 서명 크기를 바이트 단위로 나타낸다.

Table 1: Details of DILITHIUM (unit: Bytes).

	NIST level	Public key size	Private key size	Signature size
Dilithium 2	2	1,312	2,528	2,420
Dilithium 3	3	1,952	4,000	3,293
Dilithium 5	5	2,592	4,864	4,595

2.2.2 FALCON[8]

Falcon 은 Fast Fourier Lattice-based Compact Signatures over NTRU 의 약자로, 격자 기반 전자서명 알고리즘이다. NTRU 격자 상에서 Short Integer Solution 문제가 양자 알고리즘으로도 효율적으로 풀리지 않는다는 점에 기반을 두고 있다. Falcon 은 보안 수준에 따라 Falcon 512, 1024 로 구분되며, Table 2 는 Falcon 의 공개키 및 서명의 크기를 보여준다.

Table 2: Details of FALCON (unit: Bytes).

	NIST level	Public key size	Private key size	Signature size
Falcon 512	1	897	1,281	666
Falcon 1024	5	1,793	2,305	1,280

2.2.3 SPHINCS+[9]

SPHINCS+ 는 기존의 SPHINCS 를 속도와 서명 크기를 개선한 전자서명 알고리즘이다. 상태를 저장하지 않는 stateless 해시 기반 알고리즘이며, SHAKE256, SHA-256 그리고 Haraka 를 기반으로 하는 3 가지 종류의 알고리즘으로 나누어져있다. Table 3 은 Sphincs+의 공개키 및 서명의 크기를 바이트 단위로 나타낸다.

Table 3: Details of SPHINCS+ (unit: Bytes).

	NIST level	Public key size	Private key size	Signature size
Sphincs SHA256-128f simple	1	32	64	17,088
Sphincs SHA256-192f simple	3	48	96	35,664
Sphincs SHA256-256f simple	5	64	128	49,856

3. 구현

PQ-DPoL 합의 알고리즘은 WISA'23 에 제안한 우리의 합의 알고리즘으로, TEE(Trusted Execution Environment)[10]를 기반으로 동작하는 PoL(Proof of Luck) 합의 알고리즘에 PQC 와 위임 방식을 적용한 합의 알고리즘이다. PQ-DPoL 은 타원곡선암호 대신 PQC 를 적용함으로써 잠재적인 양자 공격으로부터 블록체인의 보안성을 보장할 수 있다. 하지만 양자 내성 암호의 특성상, 키와 서명의 길이가 크다는 단점이 존재한다. 이는 블록에 담을 수 있는 트랜잭션의 개수를 감소시켜 블록체인의 성능을 저하시킬 수 있다. 이와 같은 문제점을 해결하기 위해 위임 방식을 적용하였다.

Figure 1 은 PQ-DPoL 합의 알고리즘의 전체 동작과정을 보여준다. PQ-DPoL 은 크게 위임 단계와 합의 단계로 나뉜다. 먼저 위임 단계에서는 블록체인 내 노드들이 투표를 통해 위임자가 될 노드를 선출한다. 합의 단계에서는 득표수 순으로 위임된 노드가 블록을 생성한다. 이때, 생성된 블록에 포함되는 트랜잭션에는 양자 내성 전자서명과 공개키가 포함된다. 마지막으로 위임된 노드들이 블록을 검증한 후 검증된 블록을 체인에 추가한다.

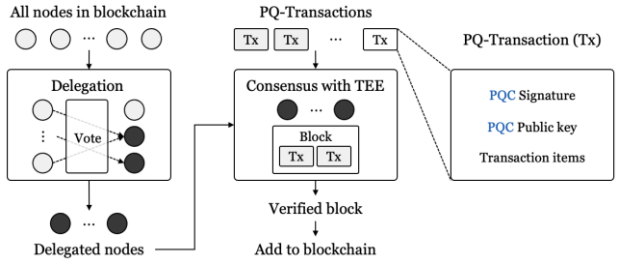

Figure. 1. Overview of PQ-DPoL

4. 성능평가

4.1 실험 환경

본 실험은 16GB RAM 의 Intel i3-6100U CPU 에서 수행되었다. 또한 실제 블록체인 네트워크와 유사한 환경을 구축하기 위해 이산 네트워크 시뮬레이터인 NS-3[11]상에서 C++ 언어를 통해 구현하였다. 또한 블록 사이즈는 환경의 제약으로 인해 50KB 로 설정하였다.

4.2 성능 평가 지표

본 논문에서는 블록체인의 성능을 평가하는 대표적인 지표인 TPS 와 Latency 를 사용한다.

TPS(Transaction Per Seconds)는 초당 처리할 수 있는 트랜잭션의 수를 의미한다. 가장 유명한 비트코인은 7TPS, 이더리움은 20TPS 그리고 이오스는 3000TPS 이다. 최근 많은 블록체인 네트워크들이 높은 TPS 를 달성하기 위한 연구를 수행하고 있다.

Latency 는 트랜잭션이 네트워크에 등장하는 시간부터 검증이 완료될 때까지 걸리는 시간을 의미한다. 즉, Latency 가 높으면 트랜잭션이 처리되는데 많은 시간이 요구됨을 의미한다. 따라서 높은 Latency 는 블록체인의 성능을 저하시킨다. 비트코인의 latency 는 10 분으로 매우 큰 편이고, 이더리움은 13 초이다. 또한 Ripple 은 약 4 초로 낮은 Latency 를 달성하였다.

4.3 성능 분석 및 비교

본 절에서는 4.2 에서 언급한 성능 지표인 TPS 와 Latency 를 기반으로 각 PQC 전자서명 알고리즘이 적용된 블록체인 네트워크의 성능을 분석하고 비교한다.

Table 4 는 PQ-DPoL 합의 알고리즘에서 각 서명 알고리즘의 성능을 보여준다. Dilithium 과 Falcon 그리고 Sphincs+의 키 생성, 서명 생성, 그리고 서명 검증 시간을 측정한다. 키 생성과 서명 그리고 서명 검증 시간 모두 Dilithium 이 가장 빠른 속도를 보여준다. Falcon 의 서명과 검증 시간은 Sphincs+ 에 비하여 빨랐으나, 키 생성에서는 가장 느린 속도를 보여준다. 마지막으로 Sphincs+의 서명과 검증은 매우 느린 속도를 보여준다. 이는 해시 기반 암호인 Sphincs+가 강한 안전성을 보장하는 대신 속도가 느리기 때문이다.

Table 4: Execution speed comparison of PQC (unit: second).

	Key Gen	Sig Gen	Sig Verify
Dilithium 2	0.000045	0.000163	0.000047
Dilithium 3	0.000098	0.000196	0.000077
Dilithium 5	0.000119	0.000228	0.00011
Falcon 512	0.011444	0.000512	0.000075
Falcon 1024	0.033799	0.000838	0.000142
Sphincs SHA256-128f simple	0.000581	0.012447	0.001029
Sphincs SHA256-192f simple	0.000880	0.021434	0.001475
Sphincs SHA256-256f simple	0.002116	0.043421	0.001500

Table 5 와 Table 6 은 블록체인 네트워크의 규모와 각 알고리즘 종류에 따라 TPS 와 Latency 를 보여준다. 블록체인 네트워크의 규모는 노드의 개수에 따라 달라지며, n개의 노드가 존재할 때 해당 네트워크의 규모는 2^n임을 의미한다.

성능 측정 결과에 따르면, 블록체인의 네트워크 규모가 증가함에 따라 TPS 가 감소하였다. 따라서 위임을 통해 합의에 참여하는 노드 수를 감소시킴으로써 TPS 를 감소율을 완화시켰다. 또한 Latency 는 Falcon 512 가 가장 낮고, 그 다음으로 Dilithium 2 가 낮은 수치를 보였으며, Sphincs SHA256-128f simple 은 가장 높은 Latency 를 나타냈다.

TPS 와 Latency 관점에서 종합적으로 분석한 결과, Falcon 알고리즘이 가장 우수한 성능을 보였다. 이는 Falcon 의 공개키와 서명의 크기가 다른 알고리즘에 비해 작기 때문이다. 또한 Falcon 의 키 생성 속도는 매우 느린 편에 속하지만, 서명과 검증 속도가 가장 빠른 Dilithium 과 비슷한 성능을 보여주기 때문이다.

Sphincs+은 하드웨어 환경의 한계로 인해 Sphincs+ SHA256-128f simple(Sphincs+의 여러 매개변수 중 가장 간단한 버전)만을 측정하였다. Sphincs+는 상대적으로 낮은 성능을 보였다. 이는 Sphincs+의 서명 크기가 상당히 큰 편이어서 블록에 포함시킬 수 있는 트랜잭션의 수가 제한적이고, Sphincs+의 서명 생성 및 검증 속도가 다른 알고리즘들에 비하여 상당히 느리기 때문이다. 따라서 양자 내성을 갖는 블록체인에 적용하기는 어려울 것으로 판단된다.

마지막으로 Dilithium 의 TPS 는 Falcon 과 비슷한 성능을 보여주었다. 그러나 TPS 와 Latency 측면에서 Falcon 512 가 우수한 성능을 보였다. 다시 말해서, 양자 내성을 갖는 블록체인을 구현하기 위해서는 Falcon 512 전자서명을 사용하는 것이 적합할 것으로 생각된다.

Table 5: TPS of DPoL(n: the number of nodes).

n	1	2	3	4	5
Dilithium 2	102.9905	25.8397	6.5516	1.637	0.4088
Dilithium 3	68.539	15.8406	4.104	1.0632	0.2669
Dilithium 5	36.3322	9.9926	2.3893	0.6356	0.1567
Falcon 512	307.0817	72.2645	18.4524	4.57	1.111
Falcon 1024	116.2432	30.5676	7.4315	1.9117	0.4669
Sphincs SHA256-128f simple	4.9496	1.3056	0.3473	0.0908	0.0227

Table 6: Latency of DPoL(n: the number of nodes).

n	1	2	3	4	5
Dilithium 2	0.1165	0.4644	1.8316	7.33	29.3506
Dilithium 3	0.1313	0.5681	2.1929	8.4643	33.7204
Dilithium 5	0.1651	0.6004	2.5111	9.4387	38.2889
Falcon 512	0.0958	0.4071	1.5943	6.4375	26.4582
Falcon 1024	0.129	0.7907	2.0184	7.8461	32.1206
Sphincs SHA256-128f simple	0.404	1.5317	5.7581	22.0108	87.7281

5. 결론

본 논문에서는 효율적인 양자 내성 블록체인을 구현하기 위해 양자 내성 전자서명을 적용한 우리의 합의 알고리즘(WISA'23)에 대해 추가적인 PQC 를 적용하고 그에 대한 성능을 측정 및 분석하였다.

본 논문에서는 블록체인의 성능 지표 (TPS, Latency)를 통해 PQC 가 블록체인의 성능에 미치는 영향을 분석하였다. TPS 와 Latency 관점에서 Falcon 이 가장 우수한 성능을 보였다. 이는 Falcon 의 공개키와 서명의 크기가 다른 알고리즘에 비해 작기 때문에, 블록에 담기는 트랜잭션의 수가 증가하기 때문이다. 이와 반대로 큰 서명 크기를 갖는 Sphincs+의 성능은 매우 떨어졌다. Dilithium 은 Falcon 과 비슷한 성능을 보여주었으나, TPS 와 Latency 를 분석한 결과, Falcon 512 가 가장 효율적이며 적합한 알고리즘임을 확인하였다. 또한, 블록체인 네트워크에서 Sphincs+를 사용하는 것은 매우 비효율적이다.

그러나 블록체인 네트워크의 속도와 보안성은 Trade-off 관계이므로 블록체인 네트워크의 상황과 목적에 따라 적절한 PQC 알고리즘을 선택해야 할 것으로 생각된다.

6. Acknowledgment

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2018-0-00264, Research on Blockchain Security Technology for IoT Services, 50%) and this work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-00627, Development of Lightweight BIoT technology for Highly Constrained Devices, 50%).

참고문헌

[1] Shor, Peter W. "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer." SIAM review 41.2 (1999): 303-332.

[2] Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." Decentralized business review (2008).

[3] Wood, Gavin. "Ethereum: A secure decentralised generalised transaction ledger." Ethereum project yellow paper 151.2014 (2014): 1-32.

[4] Yang, Fan, et al. "Delegated proof of stake with downgrade: A secure and efficient blockchain consensus algorithm with downgrade mechanism." IEEE Access 7 (2019): 118541-118555.

[5] Milutinovic, Mitar, et al. "Proof of luck: An efficient blockchain consensus protocol." proceedings of the 1st Workshop on System Software for Trusted Execution. 2016.

[6] Chen, Lin, et al. "On security analysis of proof-of-elapsed-time (poet)." Stabilization, Safety, and Security of Distributed Systems: 19th International Symposium, SSS 2017, Boston, MA, USA, November 5–8, 2017, Proceedings 19. Springer International Publishing, 2017.

[7] Ducas, Léo, et al. "Crystals-dilithium: A lattice-based digital signature scheme." IACR Transactions on Cryptographic Hardware and Embedded Systems (2018): 238-268.

[8] Fouque, Pierre-Alain, et al. "Falcon: Fast-Fourier lattice-based compact signatures over NTRU." Submission to the NIST's post-quantum cryptography standardization process 36.5 (2018): 1-75.

[9] Bernstein, Daniel J., et al. "The SPHINCS+ signature framework." Proceedings of the 2019 ACM SIGSAC conference on computer and communications security. 2019.

[10] Sabt, Mohamed, Mohammed Achemlal, and Abdelmadjid Bouabdallah. "Trusted execution

environment: what it is, and what it is not." 2015 IEEE Trustcom/BigDataSE/Ispa. Vol. 1. IEEE, 2015.
[11] Carneiro, Gustavo. "NS-3: Network simulator 3." UTM lab meeting April. Vol. 20. No. 1. 2010.