Smart Media Journal / Vol.15, No.2 / ISSN:2287-1322
https://dx.doi.org/10.30693/SMJ.2026.15.2.72

NIST PQC 표준화 이후 해시 기반 전자서명 기술 동향

(Trends in Hash-Based Digital Signature Technology Since NIST PQC Standardization)

이재흥*

(Jae Heung Lee)

요 약

양자컴퓨팅의 발전으로 기존 전자서명(RSA, ECDSA 등)의 보안성이 위협받으면서 NIST는 2024년 FIPS 203, 204, 205를 통해 양자내성암호(Post-Quantum Cryptography, PQC) 표준을 확정하였다. 이 중 FIPS 205(SLH-DSA)는 해시 기반 전자서명 SPHINCS+를 채택하여 격자 기반 전자서명과 상호 독립적인 보안 기반을 마련하였다. 본 논문은 NIST PQC 표준화 이후 해시 기반 전자서명의 구조적 특징과 기술 동향을 분석하며, SPHINCS-α와 SPHINCS+C, FPGA와 GPU 기반 가속 연구 등 최신 개선 방향을 함께 고찰하였다. 또한 SPHINCS+, XMSS, LMS를 성능, 구조, 보안성 관점에서 비교한 결과 SPHINCS+는 높은 장기 보안성을 제공하나 효율성이 낮고, XMSS와 LMS는 작은 서명 크기와 빠른 속도 측면에서 실용적이나 상태 관리가 필요함을 확인할 수 있었다. 앞으로의 연구는 보안성과 효율성의 균형, 경량화, 파라미터 최적화 및 하드웨어 가속을 중심으로 전개될 것으로 전망된다.

■ 중심어 : 해시 기반 전자서명 ; 양자내성암호 ; SPHINCS+ ; SLH-DSA

Abstract

As the security of existing digital signatures (such as RSA and ECDSA) is threatened by the development of quantum computing, NIST finalized the Post-Quantum Cryptography (PQC) standard through FIPS 203, 204, and 205 in 2024. Among these, FIPS 205 (SLH-DSA) adopted SPHINCS+, a hash-based digital signature, to establish a security foundation that is independent from lattice-based digital signatures. This paper analyzes the structural characteristics and technological trends of hash-based digital signatures since the NIST PQC standardization, and examines the latest directions for improvement, such as SPHINCS-α and SPHINCS+C, and FPGA and GPU-based acceleration research. In addition, the results of comparing SPHINCS+, XMSS, and LMS in terms of performance, structure, and security show that SPHINCS+ provides high long-term security but low efficiency, while XMSS and LMS are practical in terms of small signature size and fast speed but require state management. Future research is expected to focus on balancing security and efficiency, lightweighting, parameter optimization, and hardware acceleration.

■ keywords : Hash-Based Digital Signature ; Post-Quantum Cryptography ; SPHINCS+ ; SLH-DSA

Ⅰ. 서 론

양자컴퓨팅의 발전으로 기존 공개키 암호 시스템의 보안성이 위협받고 있다. 현재 전자서명 분야에서 널리 사용되는 RSA[1]와 ECDSA[2]는 각각 소인수분해 문제와 이산대수 문제의 계산

* 정회원, 경기대학교 AI컴퓨터공학부
접수일자 : 2025년 11월 28일
게재확정일 : 2025년 12월 10일
교신저자 : 이재흥 e-mail : jhlee@kyonggi.ac.kr

적 어려움을 기반으로 하지만, 대규모 양자컴퓨터가 현실화될 경우 Shor 알고리즘과 같은 양자 알고리즘에 의해 이들 문제가 다항 시간 안에 해결될 수 있음이 알려지면서 보안성에 대한 우려가 커지고 있다[3][4]. 이로 인해 양자 환경에서도 안전하게 사용할 수 있는 양자내성암호(Post-Quantum Cryptography, PQC)의 개발이 국제적으로 중요한 과제로 떠오르고 있다.

이에 대응하기 위해 미국 국립표준기술연구소(National Institute of Standards and Technology, NIST)는 2016년부터 차세대 공개키 암호 체계를 선정하는 PQC 표준화 작업을 진행해왔다. 여러 차례의 공개 평가와 후보 검증 절차를 거친 후, 2024년 8월 FIPS 203, 204, 205 세 가지 표준이 최종 확정되었다.

- FIPS 203은 모듈 격자 기반 키 캡슐화 메커니즘(KEM)인 ML-KEM을 규정하며, 이는 CRYSTALS-Kyber를 기반으로 한다[5].
- FIPS 204는 모듈 격자 기반 전자서명 알고리즘인 ML-DSA를 규정하며, 이는 CRYSTALS-Dilithium을 기반으로 한다[6].
- FIPS 205는 해시 기반 전자서명 알고리즘인 SLH-DSA를 규정하며, 이는 SPHINCS+를 기반으로 한다[7].

특히 FIPS 205를 통해 해시 기반 전자서명 알고리즘이 PQC 환경에서 중요한 보안 요소가 되었음을 알 수 있다. 해시 기반 전자서명의 경우 해시 함수의 보안성에만 의지하는 구조 덕분에 이론적으로 안전성을 검증하기 쉬우며, 따라서 장기적 보안성 측면에서 어느 정도 유리하다고 볼 수 있다.

NIST PQC 표준화가 이뤄진 이후에도 해시 기반 전자서명에 대한 연구는 꾸준히 이어지고 있다. SPHINCS-α는 SPHINCS+의 기본 구조는 그대로 두되 일회성 서명 부분을 재설계하여 서명 크기를 줄였으며[8], SPHINCS+C는 서명 과정에서 생기는 중복 데이터와 필요 이상의 해시 경로 정보를 압축하여 서명 크기와 검증 비용을 낮추었다[9].

본 논문은 이와 같은 환경 변화를 바탕으로 NIST PQC 표준화 이후 해시 기반 전자서명 기술의 주요 흐름을 분석한다. Ⅱ장에서 NIST PQC 표준화 과정과 SPHINCS+의 핵심 구조를 살펴보고, Ⅲ장에서 표준화 이후 해시 기반 전자서명 기술 동향을 알아본다. Ⅳ장에서 주요 알고리즘을 대상으로 성능과 보안성을 비교, 분석하며, 마지막으로 Ⅴ장에서 전체 논의를 정리하며 결론을 제시한다.

Ⅱ. NIST PQC 표준화

1. NIST PQC 표준화 과정

기존 공개키 암호가 의존하는 소인수분해 문제와 이산대수 문제의 보안성이 양자컴퓨팅의 발전으로 위협받자 NIST는 2016년부터 양자내성암호(PQC)에 대한 표준을 수립하는 과정에 들어갔다. 이는 양자 공격 환경에서 사용할 수 있는 새로운 공개키 암호 시스템을 구축하는 것을 목표로 했으며, 여러 알고리즘을 공개적으로 모집하고 단계적으로 평가하는 과정을 통해 진행되었다.

아래 표 1은 NIST PQC 표준화 과정을 나타낸다[10].

표 1. NIST PQC 표준화 과정

구분	주요 내용
1단계 (2016-2019)	다양한 암호, KEM, 서명 알고리즘을 받아 기초 평가 수행
2단계 (2019-2021)	보안성, 효율성, 구현성 등을 중심으로 후보를 추려 서명, KEM 중심의 본선 체제로 전환
3단계 (2021-2022)	최종 후보군을 발표하고 표준 초안 작업에 착수

전자서명 분야에서는 격자 기반과 해시 기반 설계가 최종적으로 경쟁 구도를 이루었다. 평가 결과, 격자 기반에서는 CRYSTALS-Dilithium과 Falcon이, 해시 기반에서는 SPHINCS+가 마지막까지 남아 표준 후보로 선정되었다. 이후 Dilithium은 ML-DSA라는 이름으로 FIPS 204

로 표준화되었으며, SPHINCS+는 SLH-DSA라는 이름으로 FIPS 205로 표준화되었다. Falcon은 아직 표준으로 확정되지 않았으며, 향후 FIPS 206 (FN-DSA)로 제정될 예정이다.

FIPS 204(ML-DSA)는 모듈 격자 구조를 활용한 전자서명으로 속도, 키 크기, 구현 난이도 사이의 균형이 좋다는 평가가 많다. 양자 내성을 갖춘 구조이기 때문에 기존 ECC 서명의 대안으로도 주목받고 있다[6].

반면 FIPS 205(SLH-DSA)는 해시 함수만으로 구성된 순수 해시 기반 서명으로, 만약 격자 기반 알고리즘에서 새로운 취약점이 발견되더라도 독립적인 안전성을 보장할 수 있다는 점에서 의미가 크다. 상대적으로 서명과 키 크기가 크고 속도가 느리다는 단점이 있지만, 장기 보존 문서나 아카이브와 같이 높은 신뢰성과 긴 수명이 필요한 환경에 적합하다는 평가를 받고 있다[7].

2. SPHINCS+의 구조적 특징

SPHINCS+는 2015년 EUROCRYPT에서 발표된 SPHINCS 알고리즘을 바탕으로 성능과 보안성을 개선한 무상태(stateless) 해시 기반 전자서명 알고리즘이다[11][12]. 이 알고리즘은 무상태 구조를 유지하면서 서명 크기, 검증 비용, 보안성 등 다양한 측면에서 SPHINCS보다 개선된 모습을 보인다. SPHINCS+는 아래의 세 가지 구성 요소를 중심으로 동작한다.

첫째, W-OTS+(Winternitz One-Time Signature)는 해시 체인을 이용하는 일회용 서명 방식으로, 메시지에 따라 서명에 일방향 함수를 적용하는 횟수를 다르게 하여 서명의 크기를 줄인다.

둘째, FORS(Forest of Random Subsets)는 메시지를 여러 개의 비트 블록으로 나누고, 각 블록을 별도의 작은 머클 트리로 인증하는 방식이다. 즉, 메시지의 비트 조각마다 별도의 인증 경로를 두는 형태라고 볼 수 있다.

셋째, 하이퍼트리는 여러 층의 머클 트리를 위아래로 연결한 구조인데, 각 층의 트리가 그 아래 단계 트리의 루트를 검증하는 역할을 한다. 이러한 다층 구조 덕분에 SPHINCS+는 많은 서명을 처리해도 별도의 상태 정보를 관리할 필요가 없다.

또한 SPHINCS+는 표 2에 있는 NIST PQC 보안 범주에 맞춰 다양한 매개변수 구성을 지원한다. 실제 적용 환경이나 요구 보안 수준에 따라 적당한 조합을 골라 사용할 수 있다.

표 2. NIST PQC 보안 범주

보안 범주	대칭키 보안	참조 공격(기준선)
1	~ 128비트 (AES-128)	128비트 블록 암호 대상 키 검색
2	~ 128비트 (SHA2/3-256)	256비트 해시 함수 대상 충돌 검색
3	~ 192비트 (AES-192)	192비트 블록 암호 대상 키 검색
4	~ 192비트 (SHA2/3-384)	384비트 해시 함수 대상 충돌 검색
5	~ 256비트 (AES-256)	256비트 블록 암호 대상 키 검색

Ⅲ. NIST PQC 표준화 이후 해시 기반 전자서명 기술 동향

1. SPHINCS+ 개선 알고리즘 동향

기존 해시 기반 전자서명, 특히 SPHINCS와 SPHINCS+에 대한 연구는 주로 새로운 구조 설계와 보안성 증명, 그리고 이론적 및 일반적 성능 개선에 초점이 맞추어져 있었다. 예를 들어 기존 연구들은 해시 기반 전자서명을 일회용 서명(One-Time Signature, OTS), 소수회용 서명(Few-Time Signature, FTS), 무상태(Stateless) 계열로 분류하고, 보안 가정과 서명 크기와 연산량 사이의 이론적 트레이드오프를 분석하는 데 비중을 두었다. 반면 SPHINCS+가 표준으로 채택된 이후에는 동일한 보안 모델을 유지하면서도 실제 배포 환경(서버, 임베디드, IoT, FPGA, GPU 등)에서 요구되는 처리량, 메모리, 에너지 제약을 만족시키는 방향으로 연구의 무게 중심이 옮겨지고 있다. SPHINCS-α나

SPHINCS+C처럼 파라미터와 압축 기법을 통해 서명 크기와 서명/검증 시간을 줄이려는 시도와 함께, FPGA 및 GPU 기반 가속 연구들은 구체적인 플랫폼에서의 처리량과 자원 효율을 정량적으로 개선하는 데 초점을 두고 있다는 점에서 이러한 흐름을 잘 보여준다.

SPHINCS-α는 기본 구조는 SPHINCS+와 동일하게 유지하면서 일회용 서명 구성 요소인 W-OTS+를 constant-sum 기반 메시지 인코딩을 사용하는 CS-WOTS+로 바꾸어 서명 크기를 줄인다[8].

SPHINCS+C 역시 SPHINCS-α와 마찬가지로 기본 구조는 SPHINCS+와 동일하게 유지하면서 서명 과정에서 발생하는 데이터 중복과 불필요한 해시 경로 정보를 압축하여 서명 크기와 검증 비용을 줄인다[9].

FPGA 기반 최적화 연구 중 하나로 SPHINCS+ 연산의 핵심인 W-OTS+ 체인 계산과 트리 인증 과정을 하드웨어 병렬 처리에 적합한 구조로 재구성한 연구가 있다[13]. 이로서 제한된 FPGA 자원 내에서도 처리량을 증가시키는 것이 가능함이 확인되었으며, 이는 임베디드 환경이나 전용 보안 모듈에서 SPHINCS+를 적용하는 데 중요한 근거가 된다.

GPU 기반 최적화 연구 중 하나로 대규모 해시 연산을 병렬화할 수 있는 CUDA 아키텍처의 장점을 활용하여 SPHINCS+의 서명 생성 속도를 높이는 연구가 있다[14]. GPU 자원을 더 효율적으로 활용하기 위해 배치 처리 방식을 도입하여 서버나 클라우드 환경에서도 SPHINCS+를 적용할 수 있음을 보여준다.

2. Stateful 해시 기반 서명: XMSS와 LMS

SPHINCS+가 무상태 서명 방식으로 표준화되었음에도 XMSS(RFC 8391)[15]와 LMS(RFC 8554)[16]가 여전히 다양한 실무 환경에서 활용되는 이유는 두 알고리즘이 갖는 효율성과 단순한 구조, 그리고 산업 현장에서의 높은 적용성 때문이다.

먼저 서명 효율성을 보면 XMSS와 LMS는 SPHINCS+에 비해 서명 크기와 처리 속도에서 확실한 이점을 가진다. XMSS의 서명 크기는 파라미터에 따라 보통 2.5KB 정도로 비교적 작은 편이며, SPHINCS+는 선택된 보안 수준과 파라미터 구성에 따라 약 8KB에서 최대 40KB 후반까지 서명 크기가 증가하는 것으로 알려져 있다. 또한 XMSS와 LMS는 해시 함수만을 기반으로 동작하기 때문에 서명 생성이나 검증 과정이 비교적 단순하고, 필요한 자원도 크지 않다.

구현 측면에서 보면 XMSS와 LMS는 해시 함수 기반이라 연산 부담이 적고 내부 구조도 단순하다. 리소스가 제한된 환경에서도 쓸 만한 성능이 나오기 때문에 보안 부팅, 펌웨어 서명, IoT 기기 인증 같은 곳에서 계속 쓰이고 있다. 예전에 문제로 지적되던 상태 관리 이슈도 요즘은 프레임워크나 도구들이 나오면서 많이 나아졌다.

SPHINCS+는 무상태 구조라서 장기 보안이 확실하다는 게 강점이다. 하지만 그렇다고 XMSS나 LMS가 필요 없어지는 건 아니다. 실제 현장에서는 처리 속도나 구현 난이도가 중요한 경우가 많아서, 상태 기반 방식이 더 적합한 상황도 여전히 있다. 결국 세 방식은 각자 다른 설계 철학을 가지고 있고, 어떤 걸 선택할지는 시스템 요구사항이나 운영 환경에 따라 달라진다.

Ⅳ. 성능 및 보안성

1. 평가 지표 및 방법론

해시 기반 전자서명 기법의 성능과 보안성을 비교하기 위해 표 3과 같이 다양한 지표를 사용하여 평가하였다.

표 3. 평가 지표

구분	평가 지표	설명
보안성	비트 보안 강도(Bit Security)	NIST PQC 보안 범주 기준

성능	서명 크기(Signature Size)	전체 서명 데이터의 크기
	키 크기(Key Size)	공개키 및 비밀키의 크기
	서명 속도(Signing Time)	서명 생성 시간
	검증 속도(Verification Time)	서명 검증 시간
구조적 특성	상태 관리(Stateful/Stateless)	키 재사용 가능성 및 상태 관리 부담

2. 주요 알고리즘별 성능 비교

아래 표 4는 주요 해시 기반 전자서명 알고리즘의 상태 관리 방식, 서명 크기, 공개키/개인키 크기, 서명 및 검증 속도를 비교한다. 수치는 대표 파라미터 세트 기준이며, 구현 환경에 따라 달라질 수 있다.

SPHINCS+의 경우 NIST PQC 보안 범주 1(128-bit 보안 강도)에 해당하는 SHAKE-128s 및 SHAKE-128f 파라미터를 사용하였으며, SPHINCS-C와 SPHINCS-α는 각각 제안된 연구 결과에 기반한 값이므로 공식 표준의 일부가 아니다. 따라서 두 변형에서 제시된 서명 크기와 속도 향상 수치는 각각의 원 논문 및 실험 환경에서 보고된 결과에 따른 상대적 비교 값이다.

XMSS와 LMS/HSS의 서명 크기와 키 크기는 RFC 8391 및 RFC 8554에서 권고하는 SHA-256 기반 파라미터 세트 중 일반적으로 사용되는 구성을 기준으로 작성하였다. 특히 LMS/HSS는 트리 높이, Winternitz 파라미터 w, LMOTS 파라미터 선택에 따라 서명 크기와 개인키 크기가 크게 달라지므로, 여기서는 대표 구성을 선택하여 범위 형태로 제시하였다.

서명 생성 및 검증 속도는 특정 하드웨어 플랫폼, 구현 최적화(예: AVX2 사용 여부), 해시 함수 구현 방식 등에 따라 유의미한 차이가 발생할 수 있다. 따라서 표 4의 “빠름/느림”과 같은 상대적 성능 비교는 동일한 구현 조건에서 관찰되는 일반적 경향을 요약한 것으로, 절대적 성능 순위를 의미하는 것은 아니다.

표 4. 주요 알고리즘별 성능 비교 (128-bit 보안 범주)

알고리즘 (파라미터)	상태 관리	서명 크기 (bytes)	공개키 (bytes)	개인키 (bytes)	서명 속도 (상대)	검증 속도 (상대)
SPHINCS+ (SHAKE-128s)	Stateless	7856	32	64	느림 (128f, XMSS 대비)	보통
SPHINCS+ (SHAKE-128f)	Stateless	17088	32	64	128s보다 빠름	128s보다 빠름
SPHINCS+C (128s)	Stateless	6304	32	64	128s와 거의 동일	128s보다 다소 느림
SPHINCS-α (SHAKE-128s)	Stateless	6880	32	64	128s 대비 3% 빠름	128s보다 다소 느림
XMSS (SHA2_10_256)	Stateful	2500	64	1373	빠름	빠름
XMSS (SHA2_20_256)	Stateful	2820	64	2573	SHA2_10_256보다는 느리지만 빠름	빠름
LMS/HSS (LMS_SHA256_M32_H10 + LMOTS_SHA256_N32_W8)	Stateful	약 1.2–1.5 KB	56	수십~수백 KB (상태 정보 포함)	XMSS와 유사하게 빠름	빠름
LMS/HSS (LMS_SHA256_M32_H20 + LMOTS_SHA256_N32_W8)	Stateful	약 1.4–1.8 KB	56	수십~수백 KB (상태 정보 포함)	H10보다는 느리지만 빠름	빠름

표 4는 다양한 해시 기반 전자서명 알고리즘의 구조적 특징과 파라미터 선택이 성능에 어떤 영향을 주는지를 보여준다. 먼저 SPHINCS+ 계열은 무상태(stateless) 구조로 인해 장기적 보안성과 키 관리의 단순성을 제공하지만, 다층 트리를 기반으로 하기 때문에 XMSS나 LMS/HSS에 비해 서명 생성 속도가 상대적으로 느리다. 특히 SPHINCS+-128s는 가장 작은 키 크기를 유지하면서도 서명 크기가 7.8KB 수준이며, SPHINCS+-128f는 서명 크기 증가와 맞바꾸는 대신 더 빠른 서명 및 검증 속도를 제공한다.

SPHINCS+C와 SPHINCS-α는 SPHINCS+ 구조를 유지하면서 서명 크기(각각 6304B, 6880B)와 속도를 개선한 연구 변형이다. 단, 공식 표준이 아니므로 표의 값은 해당 논문의 실험 결과를 기반으로 한다.

XMSS와 LMS/HSS는 상태 기반 방식이라 사용 횟수를 추적해야 한다. 관리가 번거롭지만, 해시 기반이라 서명과 검증이 빠르고 SPHINCS+보다 서명도 작다. XMSS는 트리 높이에 따라 2.5 - 2.8KB, LMS/HSS는 파라미터에 따라 1.2 - 1.8KB 정도다. LMS/HSS 개인키는 상태 정보 때문에 수십~수백 KB까지 늘어날 수 있다.

SPHINCS+는 상태 관리가 필요 없지만 서명이 크고 느리다. XMSS/LMS는 빠르고 서명이 작

지만 상태 관리가 필요하다. 따라서 알고리즘 선택은 환경의 자원 제약, 상태 관리 가능 여부, 보안 요구사항에 따라 달라진다.

3. 보안성 분석

해시 기반 전자서명 기법의 보안성은 해시 함수의 충돌 저항성(collision resistance), 제2 역상 저항성(second preimage resistance), 역상 저항성(preimage resistance)에 기반한다. 이 특성들은 양자 환경에서도 안정적으로 유지되며, 각 알고리즘은 다음 공격 모델에 대응하도록 설계되었다.

표 5. 주요 공격 유형과 대응 기법

공격 유형	대응 기법	적용 알고리즘
양자 탐색 공격 (Grover 공격)	256-bit 이상 해시 출력 길이 사용	SPHINCS+, XMSS, LMS
적응적 선택 메시지 공격 (ACMA 공격)	Per-message salt, 랜덤 시드	SPHINCS+, SPHINCS-α
사이드 채널 공격	서명 랜덤화, 상수 시간 구현	SPHINCS+, XMSS, LMS
인덱스 재사용 공격	Stateless 구조 또는 안전한 상태 관리	SPHINCS+, XMSS, LMS

양자 탐색 공격(Grover 공격)에 대한 기본적인 대응책은 해시 함수의 출력 길이를 충분히 길게 만드는 것이다. Grover 알고리즘을 가정하면 n비트 보안을 얻기 위해 대략 2n비트 정도의 해시 출력이 필요하다고 보는 것이 일반적이며, 따라서 256비트 출력 해시를 사용하면 양자 공격을 고려하더라도 실질적으로 128비트 수준의 보안 강도를 확보할 수 있다. SPHINCS+, XMSS, LMS는 모두 이런 관점에서 256비트급 해시를 사용할 수 있는 파라미터 구성을 제공하므로, Grover 유형의 양자 탐색 공격에 대해 구조적으로 충분한 여유를 두고 설계되어 있다고 볼 수 있다.

SPHINCS+와 SPHINCS-α는 적응적 선택 메시지 공격(Adaptive Chosen Message Attack, ACMA)에 메시지마다 독립적인 솔트와 시드로 대응한다. 그렇기 때문에 공격자가 반복적으로 서명을 요청해도 서명 사이의 연관 정보를 알아내는 것은 불가능하다. SPHINCS+는 논문을 통해 ACMA에 대해 안전하다는 것이 증명되었다.

사이드 채널 공격은 알고리즘 설계만으로는 막을 수 없어 구현 단계 대책이 필요하다. SPHINCS+는 서명 생성 과정에 난수를 포함시켜 같은 메시지를 서명하더라도 매번 다른 서명 결과를 생성한다. 덕분에 전력 사용량이나 처리 시간 같은 반복 패턴이 줄어들며, 타이밍 분산이나 마스킹 등을 더하면 더 안전해진다.

마지막으로 인덱스 재사용 공격은 상태 저장형 서명에서 특히 위험하다. XMSS나 LMS 같은 방식은 서명마다 다른 인덱스를 써야 하는데, 같은 인덱스를 두 번 이상 쓰면 보안이 무너질 수 있다. 그래서 안전하게 운영하려면 인덱스 상태를 계속 기록하고 롤백이나 동시 접근으로 인한 중복 사용을 막는 관리 체계가 필요하다. 반면 SPHINCS+ 같은 상태 비저장 구조는 애초에 인덱스 관리 부담 자체가 없도록 설계되었다. 이런 방식은 근본적으로 이 유형의 공격 경로를 아예 회피한다.

V. 결 론

본 논문에서는 NIST PQC 표준화 이후 해시 기반 전자서명 분야에서 나타난 주요 변화를 살펴보았다. 먼저 SPHINCS+의 구조적 특징과 파생 기법들을 살펴보았고, 최근 제안된 SPHINCS-α와 SPHINCS+C, 그리고 FPGA와 GPU 기반 구현 연구들이 어떻게 효율성을 높이려 하고 있는지 정리하였다. SPHINCS+, XMSS, LMS를 성능, 구조, 보안성 관점에서 비교한 결과 SPHINCS+는 높은 장기 보안성을 제공하나 효율성이 낮고, XMSS와 LMS는 작은 서명 크기와 빠른 속도 측면에서 실용적이나 상태 관리가 필요함을 확인할 수 있었다. 앞으로의 연구는 보안성과 효율성의 균형, 경량화, 파라미터 최적화 및 하드웨어 가속을 중심으로 전개될 것으로 전망된다.

REFERENCES

[1] R. L. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,” *Communications of the ACM*, vol. 21, no. 2, pp. 120-126, Feb. 1978.

[2] D. Johnson, A. Menezes, and S. Vanstone, “The Elliptic Curve Digital Signature Algorithm (ECDSA),” *International Journal of Information Security*, vol. 1, no. 1, pp. 36 - 63, Aug. 2001.

[3] P. W. Shor, “Algorithms for quantum computation: discrete logarithms and factoring,” *Proceedings of 35th Annual Symposium on Foundations of Computer Science*, pp. 124 - 134, 1994.

[4] L. K. Grover, “A fast quantum mechanical algorithm for database search,” *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, pp. 212 - 219, May 1996.

[5] National Institute of Standards and Technology (NIST), FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard, U.S. Department of Commerce, Aug. 2024.

[6] National Institute of Standards and Technology (NIST), FIPS 204: Module-Lattice-Based Digital Signature Standard, U.S. Department of Commerce, Aug. 2024.

[7] National Institute of Standards and Technology (NIST), FIPS 205: Stateless Hash-Based Digital Signature Standard, U.S. Department of Commerce, Aug. 2024.

[8] K. Zhang, H. Cui, and Y. Yu, “Revisiting the Constant-Sum Winternitz One-Time Signature with Applications to SPHINCS+ and XMSS,” *Proc. of CRYPTO 2023, Part V*, pp. 455 - 483, Aug. 2023.

[9] A. Hülsing, M. Kudinov, E. Ronen, and E. Yogev, “SPHINCS+C: Compressing SPHINCS+ with (Almost) No Cost,” *Proc. of the 44th IEEE Symposium on Security and Privacy (S&P 2023)*, pp. 1435 - 1453, San Francisco, CA, USA, May 2023.

[10] G. Alagic et al., “Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process,” *NIST Internal Report (NISTIR) 8413*, NIST, Gaithersburg, MD, Jul. 2022.

[11] D. J. Bernstein, D. Hopwood, A. Hülsing, T. Lange, R. Niederhagen, L. Papachristodoulou, M. Schneider, P. Schwabe, and Z. Wilcox-O’hearn, “SPHINCS: Practical stateless hash-based signatures,” *Proc. of Advances in Cryptology - EUROCRYPT 2015*, LNCS 9056, pp. 368 - 397, Apr. 2015.

[12] J.-P. Aumasson, D. J. Bernstein, W. Beullens, C. Dobraunig, M. Eichlseder, S. Fluhrer, S.-L. Gazdag, A. Hülsing, P. Kampanakis, S. Kölbl, T. Lange, M. M. Lauridsen, F. Mendel, R. Niederhagen, C. Rechberger, J. Rijneveld, P. Schwabe, and B. Westerbaan, SPHINCS+ Submission to the NIST Post-Quantum Project, v.3.1, 2022.

[13] Q. Berthet, A. Upegui, L. Gantel, A. Duc and G. Traverso, “An Area-Efficient SPHINCS+ Post-Quantum Signature Coprocessor,” *Proc. of the 2021 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pp. 180 - 187, June 2021.

[14] J. Wu, Y. Yu, Z. Chen, H. Yang, C. Li and Z. Liu, “CBPSPX: A CUDA-Based Batch Parallel Optimization of Post-Quantum Signature SPHINCS+,” *IEEE Internet of Things Journal*, vol. 12, no. 18, pp. 37898 - 37911, 2025.

[15] J. Buchmann, E. Dahmen, and A. Hülsing, “XMSS - A practical forward secure signature scheme based on minimal security assumptions,” *Proc. of Post-Quantum Cryptography (PQCrypto 2011)*, LNCS 7071, pp. 117 - 129, Nov. - Dec. 2011.

[16] D. McGrew, M. Curcio, and S. Fluhrer, “Leighton-Micali Hash-Based Signatures,” *RFC 8554*, Internet Engineering Task Force (IETF), April 2019.

저 자 소 개

이재흥(정회원)

2001년 서울대학교 컴퓨터공학부 학사 졸업.

2003년 서울대학교 전기·컴퓨터공학부 석사 졸업.

2013년 서울대학교 전기·컴퓨터공학부 박사 졸업.

2016년～2023년 대전대학교 정보보안학과 부교수.

2024년～ 경기대학교 AI컴퓨터공학부 부교수.

<주관심분야 : 정보 보안, 시스템 보안, 인공 지능, 분산 처리 시스템>