PQNetSim: PQC 네트워킹 부하 측정 시뮬레이터

성하경 강윤의 김현주* 이현우*

한국에너지공과대학교 (학부생), *한국에너지공과대학교 (교수)

PQNetSim: Simulator for Measuring PQC Networking Overhead

Hakyeong Seong, Yunui Kang, Hyunju Kim*, Hyunwoo Lee*

KENTECH (Undergraduate Student), *KENTECH (Professor)

요 약

양자컴퓨터의 발전은 기존 공개키 암호체계의 보안을 근본적으로 위협하고 있으며, 이에 대응하기 위한 양자 내성 암호(PQC)의 도입이 전 세계적으로 추진되고 있다. 특히 IoT 환경에서는 연산 자원이 제한되고 네트워크가 불안정하기 때문에 PQC의 적용이 더욱 어려운 상황이다. 본 연구는 PQC 알고리즘이 TLS와 DTLS 프로토콜에 적용될 때, 다양한 기기와 네트워크 조건에서 성능에 어떤 영향을 미치는지를 실증적으로 분석하였다. 이를 위해 기기 성능, 지연, 손실률, 대역폭을 조절할 수 있는 테스트베드인 PQNetSim을 구축하였다. 실험 결과, 저사양 기기와 손실(lossy) 네트워크 환경에서 PQC 적용으로 인해 연결 지연이 여러 환경에서 최소 59.914ms에서 373.674ms까지 증가하였고, 연결 실패 가능성도 증가하였다. 이는 PQC 성능 평가 시 연산 효율성뿐만 아니라 네트워크 적응성과 연결 안정성도 중요한 고려 요소임을 보여준다. 본 연구는 IoT 환경에서 PQC의 실용성을 검증할 수 있는 기반을 마련하고, 향후 다양한 알고리즘과 프로토콜 확장에 유연하게 대응할 수 있는 프레임워크를 제시한다.

I. 서론

양자컴퓨터의 등장은 기존 공개키 암호 체계를 근본적으로 위협한다. 특히 Shor 알고리즘은 RSA나 ECC처럼 현재 널리 사용되는 암호화 알고리즘을 효율적으로 무력화할 수 있다[1]. 이에 따라 양자 내성 암호(PQC, Post-Quantum Cryptography)의 도입은 미래 보안에서 필수로 간주되며, 미국을 포함한 여러 국가에서는 PQC 기반 암호체계의 전환과 적용이 이미 논의되고 있다. 특히 보안 프로토콜의 핵심인 TLS**에 PQC를 통합하려는 시도가 활발히 진행 중이다.

그러나 이러한 변화가 현업에 적용되기 위해서는 알고리즘 차원의 연구를 넘어, 실제 네트워크 환경과 기기 성능을 모두 고려한 실증적 검증이 병행되어야 한다. 이러한 맥락에서 IoT(사물인터넷)는 PQC 도입에 있어 반드시 고려되어야 할 중요한 영역이다. IoT 기기는 대부분 저전력·저사양 조건에서 운영된다. PQC는 기존 RSA나 ECC보다 더 큰 자원을 요구하기 때문에, IoT 환경에서 PQC를 TLS에 통합하여 프로토콜을 수행하였을 때, (1) 제한된 리소스로 인해 핸드셰이크 시간이 크게 증가하고, (2) 불안정한 네트워크에서는 핸드셰이크 자체가 실패할 수 있다.

해결 방안으로는 경량화된 PQC 알고리즘 선택, 사전 계산 기법 활용, 재전송 메커니즘 강화, 패킷 손실에 강한 프로토콜 구조 설계 등이 제안되고 있다[3]. 또한 UDP 기반의 TLS 프로토콜에 해당하는 DTLS*** 사용이 제안되고 있다[2].

현재 연산 측면에서 PQC 경량화에 대한 연구는 활발히 이루어지고 있으나, 불안정한 네트워크(lossy network)를 고려하여 PQC의 통신 성능을 정량적으로 분석한 연구는 매우 드물다. 본 연구에서는 PQC 알고리즘을 여러 Io

* 공동 교신

** Transport Layer Security

*** Datagram TLS

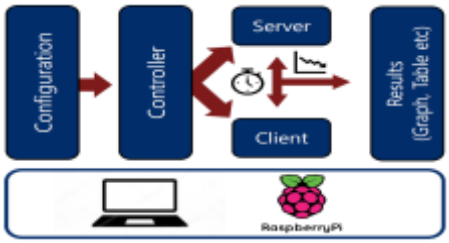

그림 1. PQNetSim 도식도

T 네트워크 환경에 적용하여, 지연, 손실, 대역폭 제약의 네트워크 조건에 따라 PQC의 네트워크 측면에 따른 분석을 수행하고자 한다.

II. PQNetSim 테스트베드

본 연구는 다양한 PQC 알고리즘과 네트워크 조건을 정밀히 모사할 수 있는 테스트베드 PQNetSim을 제안한다. 사용자는 원하는 조건을 입력하면, 시스템이 자동으로 서버-클라이언트를 구성하고 핸드셰이크 결과를 출력하며, IoT 환경의 손실 네트워크를 효과적으로 재현할 수 있다. 이를 위해 PQNetSim을 활용하면 다음 4가지의 조건을 변경할 수 있다.

- **기기**: 기기는 연산 능력과 네트워크 변화에 따른 성능 변화를 결정한다. 우리는 두 가지 실험 기기를 준비하였다. 서버는 범용 컴퓨터로 하고, 클라이언트로 i5 CPU 노트북(이하 i5)과 라즈베리파이 3 모델 B(이하 라즈베리)를 선택지로 구성하였다. 해당 라즈베리 파이는 1.2 GHz 쿼트코어·1 GB RAM 등 2016년급 사양으로 최신 SBC에 비해 모든 사양이 크게 부족하므로 저사양 IoT 기기 보드로 간주할 수 있다.

- **네트워크 제어**: 손실 네트워크 환경을 설정하기 위한 파라미터로 지연 시간, 손실률, 대역폭에 따른 효과를 보고자 하였다. 이를 위해 리눅스 시스템에서 사용 가능한 트래픽 컨트롤 도구인 tc를 사용하여 지연, 손실, 대역폭 제한 등 통제할 수 있게 하였다.

- **프로토콜**: 실제 현업에서는 PQC가 그 자체만으로가 아니라 프로토콜과 연계되기 때문에 사용성이 높은 프로토콜을 고려하여 통합 결과를 실험할 수 있게 하였다. 우리는 인터넷에서 널리 활용되는 TLS와 IoT에서 권장되는 보안 프로토콜인 DTLS를 대상으로 하였다.

- **PQC 알고리즘**: PQC 알고리즘은 근간이 되는 문제에 따라 종류가 다양하고, 이에 따른 키&서명 길이가 다르기 때문에, 현업에서의 영향은 알고리즘 종류에 따라 많이 다를 수 있다. 우리는 ML-KEM(Kyber1024)에 대해 우선 지원하였고, 여러 알고리즘에 대한 추가가 용이하도록 설계하였다.

* 지연시간 : 0 ms, 손실률 : 0%, 대역폭 : 기본값

Client	Server-Client Handshake 100 Times Average		
	TLS 1.3	DTLS 1.3	DTLS 1.3 + KEM
Raspberry PI	67.052	68.885	157.489
i5 CPU	23.219	39.154	71.254

표 1. 두 기기에서 연결 시간 비교 (단위:ms)

III. 실험 및 분석

본 연구팀은 PQNetSim을 활용하여 손실네트워크에서 DTLS에서 PQC를 적용했을 때의 성능 분석을 수행하였다. 성능 비교 수치로써 TCP/UDP 연결부터 TLS/DTLS 핸드셰이크까지의 시간을 측정하였다. 실용성을 고려하여 연결 시간이 2,000ms가 넘어가면 연결 실패로 정의하였다.

3.1 기기 성능에 따른 비교 (표 1 참조)

기기 성능의 차이가 PQC의 키 길이에 따른 네트워킹에서 어떤 영향을 끼치는지 보고자 하였고, 네트워크를 조작하지 않고 실험을 진행하였다. 이러한 기본 네트워크 환경에서 라즈베리와 i5환경에서 연결 시간을 측정하였다. 기본 네트워크 환경에서는 상대적으로 고사양인 i5가 저사양인 라즈베리에 비해 연결 시간이 세 경우에서 각각 2.89배, 1.76배, 2.21배 증가했다. PQC를 DTLS에 적용했을 때 증가율이 더 커진 까닭은 KEM 연산을 위한 기기의 성능 차이에서 유래한 것으로 보인다.

3.2 PQC 차이에 따른 비교 (표 2 참조)

1) 지연/손실 환경

DTLS는 UDP 기반으로 기본적으로 신뢰성을 보장하지 않는다. 이를 보완하기 위해 DTLS는 라이브러리 상에서 재전송 메커니즘을 포함하고 있다. 하지만 만약 패킷 손실이 발생할 경우, 재전송 횟수가 평균적으로 1-3회 발생하며, 이는 전체 연결 시간을 추가 지연시킨다. 반면 TLS는 TCP의 전송 계층이 재전송 메커니즘을 담당하며, 이는 커널 상에 구현되어 있다. 커널 상의 재전송 메커니즘이 보다

최적화된 상태로 구현되어 있는 것으로 예상되며, 이에 따라 보다 안정적인 연결이 이루어졌다[3].

2) 대역폭(BW, Bandwidth) 제한 환경

TLS는 ACK 메커니즘에 의해 DTLS보다 핸드셰이크 상 주고받는 메시지량이 더 많다. 따라서 DTLS는 낮은 대역폭 환경에서 TLS에 비해 빠른 연결이 가능하다. 그래서 DTLS는 상대적으로 단순하여 저사양 기기에 보다 적합하다. 그렇지만 패킷 손실이 많을 경우 재전송 메커니즘 구현의 최적화 정도에 따라 불안정한 모습을 보인다.

3.3 PQC 적용에 따른 비교 (표 2 참조)

DTLS에 PQC 알고리즘(ML-KEM, Kyber 1024)을 적용한 결과, 최소 59.914 ms에서 373.674 ms까지 연결 지연이 추가되었다. PQC 메시지 크기가 기존보다 훨씬 크기 때문에 패킷 손실이 더 치명적이고, 재전송을 위해 소모되는 시간이 많아진 것을 볼 수 있다.

본 연구에서는 PQC 적용 시 기존 DTLS에 비해 성공률이 10% 이상 하락할 가능성이 있다고 추정된다. 이는 향후 PQC 기반 보안 연결에 있어 성공률을 별도의 성능 지표로 고려할 필요성을 제기한다.

IV. 결론

본 연구는 PQNetSim를 구축하여 TLS와 DTLS 환경에서 ML-KEM의 성능을 기기 성능별, 네트워크 환경 영향을 포함하여 정량적으로 측정하였다. 또한, 실제로 PQC 적용 시 손실 네트워크 환경에서 연결 성능이 급격히 저하되는 것을 실험적으로 확인하였다.

이는 추후 PQC 표준화 과정에서 연산 성능뿐 아니라 네트워크 적응성, 성공률 지표 등의 추가 고려 요소가 필요함을 의미한다. 또한 PQNetSim는 향후 새로운 PQC 알고리즘이 등장하더라도 민첩하게 적용 및 테스트할 수 있어, IoT를 포함한 실행 기기의 맞춤형 알고리즘 선택 기준 마련에도 활용 가능하다.

향후 ML-DSA(Dilithium)와 QUIC 프로토콜까지 실험 대상에 포함시켜, 프로토콜-알고리즘-기기-네트워크 조합에 따른 최적화 전략

	Server-Client Handshake 100 Times Average (Client : Raspberry PI)		
	TLS 1.3	DTLS 1.3	DTLS 1.3 + KEM
Baseline	67.052	68.885	157.489
Delay 100ms	302.438	366.563	458.616
Delay 200ms	523.007	685.531	836.457
Delay 500ms	1179.227	1665.760	1777.028
5% 손실	104.213	306.711	203.833
10% 손실	273.533	803.538	877.263
20% 손실	415.257	1584.206	1874.212
BW 1Mbps	64.048	66.598	137.512
BW 500Kbps	77.533	71.849	145.934
BW 300Kbps	127.231	98.125	158.039
BW 100Kbps	349.274	270.388	417.312
300ms + 100Kbps	975.056	1243.086	1448.246
300ms + 100Kbps + 5%	1078.770	1578.542	1746.202
300ms + 100Kbps + 10%	1152.845	1540.026	1913.700
300ms + 100Kbps + 20%	1404.429	2474.259	2485.080

표 2. 다양한 네트워크 환경(딜레이, 패킷 손실, 대역폭 제한)에서의 연결 시간 (단위:ms)

을 수립하는데 도움을 줄 테스트베드로 확장할 계획이다. 또한 사용용이성을 증대하기 위해 PQNetSim 편리한 인터페이스를 제공할 것이다. 이렇게 PQNetSim는 IoT 기기의 PQC 실용화를 위한 핵심 기준과 지침을 제시하는 데 기여할 수 있을 것이다.

Acknowledgment

이 논문은 2025년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.RS-2023-00225201, 국방 무인이동체 역이용 방지 제어권 보호기술 개발)

[참고문헌]

[1] Bernstein & Lange, *Post-quantum Cryptography, Nature,* vol. 549, no. 7671, 2017.

[2] Wirges & Dettmar, *Performance of TCP and UDP over NB-IoT,* Proc. IEEE IoTaIS, 2019.

[3] Restuccia et al., *Low-Power IoT Communication Security: On the Performance of DTLS and TLS 1.3, Proc.* IFIP PEMWN, 2020.