applied sciences

IMPACT FACTOR 2.7

CITESCORE 4.5

Article

Secure and Scalable Internet of Things Model Using Post-Quantum MACsec

Juhee Choi and Junwon Lee

Special Issue
Advanced Technologies in Data and Information Security III

Edited by
Dr. George Drosatos, Dr. Konstantinos Rantos and Dr. Konstantinos Demertzis

MDPI

https://doi.org/10.3390/app14104215

applied sciences

MDPI

Article

Secure and Scalable Internet of Things Model Using Post-Quantum MACsec

Juhee Choi [1] and Junwon Lee [2,*]

[1] Department of Smart Information and Telecommunications Engineering, Sangmyung University, Cheon-An, Cheonan-si 31066, Republic of Korea; jhplus@smu.ac.kr

[2] Samsung SDS, 125 Olympic-ro 35-gil, Songpa-gu, Seoul 05510, Republic of Korea

[*] Correspondence: junimirang@gmail.com

Abstract: For the secure deployment of network platforms tailored for IoT devices, the encryption of data transmission is equally as crucial as the process of authentication. In this context, we introduce the Secure and Scalable IoT network (SSI) network platform, designed to accommodate a diverse range of IoT devices. It provides scalability and implements effective many-to-many and end-to-end encryption across extensive regions. With the emergence of quantum computing, secure public key exchange mechanisms have become important. Among the various post-quantum cryptography (PQC) algorithms assessed, Nth Degree Truncated Polynomial Ring Units (NTRUs) have emerged as an optimally suited PQC algorithm for IoT devices constrained by limited computational capabilities. We have integrated NTRUs with SSI as a lightweight PQC solution. Moreover, SSI-PQM (Post-Quantum MACsec) enhances the SSI's initial authentication structure to minimize PQC-TLS session attempts and protect the SSI's important configuration information. When applying TLS with PQC for secret key exchange purposes, it was verified that this approach ensures stable performance in IoT environments. Upon the implementation of our proposed SSI-PQM on Raspberry Pi 3B+ based IoT devices, SSI-PQM exhibited acceptable performance at security levels from 80 to 128 and achieved a minimum speed improvement of 161% over RSA at security levels above 160. It can be concluded that SSI-PQM stands out as an effective Zero Trust-based IoT network platform, demonstrating its viability and efficiency in safeguarding data transmission against potential quantum computing threats.

Keywords: IoT; L2TP; MACsec; network overlay; Nth Degree Truncated Polynomial Ring Units; post-quantum cryptography; TLS; VXLAN

check for updates

Citation: Choi , J.; Lee, J. Secure and Scalable Internet of Things Model Using Post-Quantum MACsec. *Appl. Sci.* **2024**, *14*, 4215. https://doi.org/10.3390/app14104215

Academic Editors: Konstantinos Rantos, Konstantinos Demertzis and George Drosatos

Received: 13 April 2024
Revised: 12 May 2024
Accepted: 14 May 2024
Published: 16 May 2024

1. Introduction

IoT devices have become ubiquitous due to advances in seamless communication, wireless sensors, radio frequency identification (RFID), and cloud computing [1,2]. The application of these devices has led to tremendous expansion that has even extended into our daily lives, including mobile devices, medical devices, wearable devices, home appliances, automotive devices, and industrial equipment [3–6]. Moreover, the growth of smart devices has been significant in the automotive, healthcare, and retail fields [7]. The most critical feature of these devices is their ability to connect with humans, other devices, and systems without environmental constraints via an IoT network.

As IoT architecture evolved from a closed and centralized network to a distributed cloud over the Internet, the network architecture of these devices has been proposed to mitigate IoT security threats while minimizing the security function load on embedded devices [8]. However, these proposed models do not concurrently offer many-to-many and end-to-end encryption and network separation from client to server. A Secure and Scalable IoT (SSI) network platform has been developed to handle security risks while managing the computing resources used by the IoT devices [9]. The SSI provides a layer 2 VPN, which incurs a lower load compared to the TCP/IP-based VPNs and offers data link frame encryption. It combines L2TP and VXLAN for scalable layer 2 VPN, with the MACsec algorithm. However, a distinct public key encryption method for MACsec encrypted communication is not delineated.

Quantum computers and quantum information science exploit nature's fundamental properties to facilitate a fundamentally different computation paradigm [10]. The primary distinction between a classical and quantum computer lies in the basic unit of information—the "bit". Unlike a classical bit, which can exist in one of two states (0 or 1), a "qubit" (quantum bit) can exist in both states simultaneously, a principle known as superposition and a cornerstone of quantum theory. The field of quantum algorithm development has experienced significant progress and innovation in recent years [11]. The key quantum algorithms developed include "Shor's algorithm", "Grover's algorithm", the "Quantum Approximate Optimization Algorithm (QAOA)", and the "Harrow–Hassidim–Lloyd (HHL) algorithm". Significantly, Shor's algorithm poses a major threat to public-key cryptosystems due to its ability to solve the prime factorization problem [12]. The RSA-2048, a leading public-key cryptosystem, is vulnerable with the availability of 4000 qubits [13].

Since quantum computing can potentially break traditional cryptographic algorithms, post-quantum cryptography (PQC) is required to ensure security against any attack by a quantum computer. PQC aims to provide the necessary security measures that remain robust in the era of quantum computing. Among the several types of PQC algorithms, Nth Degree Truncated Polynomial Ring Units (NTRUs) are notable for their minimal computing resource consumption and swift encryption/decryption processes [14]. However, NTRUs typically require a longer duration for encryption and decryption compared to RSA and ECC, necessitating a comprehensive evaluation of their application in IoT devices [15]. Thus, it is not advisable to simply apply NTRUs to the IoT network architecture due to their above-mentioned drawbacks.

This paper introduces Post-Quantum MACsec-based SSI (SSI-PQM), which implements PQC in an IoT network platform by integrating NTRUs with SSI. It also employs MACsec for Layer 2 encryption communication. This study contributes to the literature by proposing an IoT network platform that efficiently provides post-quantum cryptography (PQC) in environments with low computing-power IoT devices and by validating the appropriate encryption level through experiments. We propose a unique approach that is distinct from the existing IoT network platforms for PQC:

- The confidentiality of L2TP and MACsec is enhanced, and the utilization of computing resources for PQC-TLS is optimized by improving the SSI's structure.
- A strategy to prevent unauthorized IoT device access to the IoT network platform has been proposed for Zerotrust.
- Experimental validation has confirmed that SSI-PQM, which incorporates the NTRU algorithm as PQC, is well-suited for IoT devices.

The rest of this paper is organized as follows: Section 2 addresses the vulnerabilities in the security of the existing MACsec in a quantum computing environment. The Section 3 elaborates on how MACsec and NTRUs are applied in this study. A detailed description of the methodology for integrating NTRUs with SSI is provided in Section 4. Section 5 evaluates the impact of the NTRU algorithm on computing performance in SSI-PQM. We conclude with a discussion in Section 6.

2. Problem Analysis

2.1. Secret Key Exchange in Quantum Computing Environment

Quantum computing utilizes quantum phenomena such as superposition and entanglement to perform computations. Quantum computers, which operate on these principles, use qubits that are capable of representing multiple states simultaneously, a concept known as superposition [16]. This fundamental difference enables quantum computers to solve certain problems much more efficiently than classical computers. For example, Shor's algorithm, introduced in 1994, solves the prime factorization problem in quantum settings, posing a significant threat to public-key encryption systems like RSA, which rely on the difficulty of factorizing large prime numbers [10,17]. In a similar vein, quantum environments might compromise other public key cryptosystems that depend on the solutions to discrete logarithm problems. Current Transport Layer Security (TLS) communications leveraging RSA, Diffie–Hellman (DH), and Elliptic Curve Diffie–Hellman (ECDH) are at a heightened risk of public key exposure with the emergence of quantum computers.

Given these circumstances, the development of cryptographic systems has become crucial because they safeguard against quantum computing attacks. PQC encompasses such systems. Efforts have been directed toward developing "multivariate cryptography", "code-based cryptography", "lattice-based cryptography", and "supersingular elliptic curve isogeny-based cryptography", all of which have been deemed secure in a quantum computing environment [18].

The advent of quantum computing necessitates the replacement of current cryptosystems across all devices, including those with limited computing power, such as IoT devices. Hence, the exploration of lightweight post-quantum cryptography is essential. NTRUEncrypt, employed in the current study, represents one such lightweight PQC. It is a lattice-based cryptography solution centered around the shortest vector problem in a lattice. NTRU operations, performed on objects within a truncated polynomial ring featuring convolution multiplication, enable significantly faster operations compared to RSA in OpenSSL, by a factor of 20× to 200× [19].

2.2. Vulnerabilities of the SSI Network Platform

The security risks vary across different IoT networks and devices, as depicted in Figure 1. These risks include latency in fast wireless networks such as 5G, DoS (Denial of Service (DoS) attacks targeting cloud application servers, increased power usage in low-power and limited-mobility devices, data interception in home IoT setups, and replay attacks across IoT systems. While traditional endpoint, network, and application systems share these security concerns, the unique constraints of IoT environments necessitate novel solutions.

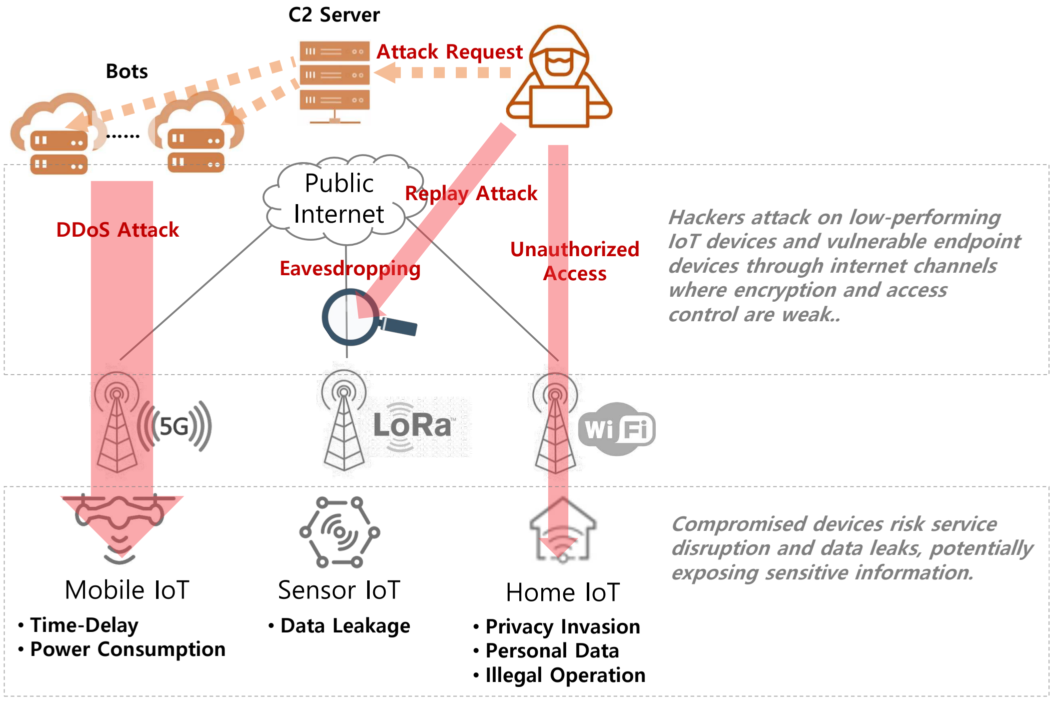

C2 Server
Attack Request
Bots
......
Public Internet
Replay Attack
DDoS Attack
Eavesdropping
Unauthorized Access
Hackers attack on low-performing IoT devices and vulnerable endpoint devices through internet channels where encryption and access control are weak..
5G
LoRa
Wi Fi
Compromised devices risk service disruption and data leaks, potentially exposing sensitive information.
Mobile IoT
• Time-Delay
• Power Consumption
Sensor IoT
• Data Leakage
Home IoT
• Privacy Invasion
• Personal Data
• Illegal Operation

Figure 1. Security threats in IoT services: security threats manifest differently based on the characteristics of the IoT service.

The SSI was proposed to implement MACsec for Layer 2 encryption communication, providing Connectivity Association Keys (CAKs) to authenticated IoT devices for symmetric key generation [9]. The CAK, used as the initial key to create the SAK for session encryption, usually remains the same for a long time; thus, it is crucial to protect it from exposure [20]. In SSI, IoT devices receive configuration information in plaintext after initial authentication for the L2TP session setup. This vulnerability may invite hackers to intercept, a risk not explicitly addressed by SSI. Moreover, vulnerabilities exist in the exchange of secret keys for MACsec encryption. The MACsec CAKs, distributed over the public internet using the Extensible Authentication Protocol Transport Layer Security (EAP-TLS) protocol, aim to safeguard against CAK exposure. However, the advent of quantum computing undermines the security assurances of current EAP-TLS implementations employing RSA, DH, and ECC cipher suites becasue PKI systems like RSA, DH, and ECC risk becoming obsolete with the advancement of quantum computing capabilities.

2.3. Security Threat Modeling Using STRIDE

Table 1 presents the IoT vulnerabilities, security threats, and countermeasures classified by STRIDE [21]. Most of the security vulnerabilities stem from insecure public key encryption, which uses RSA, DH, and ECC in an internet-exposed SSI network platform. To mitigate these security risks, it is crucial to apply data encryption using PQC and to enhance network access control in the SSI network. Moreover, Table 1 identifies specific vulnerabilities, such as authentication and insecure public key encryption in quantum computing. These vulnerabilities expose the network to several security threats, e.g., unauthorized access, response to false information, authentication failure, eavesdropping, data interception in home IoT setups, and replay attacks across IoT systems.

Table 1. STRIDE security threat Analysis: The SSI network platform has exposed authentication and session connection information for L2TP tunnel connections. However, vulnerabilities and security threats associated with SSI network platform can be effectively countered through data encryption using PQC.

STRIDE	SSI Vulnerabilities	SSI Security Threats	Countermeasures
Spoofing	Opened network for authentication Insecure public key encryption in quantum computing	Unauthorized access	Data Encryption with PQC
Tampering	Opened network for 1st authentication	Response: false information	Data Encryption with PQC
Repudiation	Opened network for 1st authentication	Authentication failure	Data Encryption with PQC
Information Disclosure	Opened network for 1st authentication Insecure public key encryption in quantum computing	Eavesdropping	Data Encryption with PQC
Denial of Service	Opened network for 1st authentication	Response: false information Unauthorized access	Network Access Control
Elevation of Privilege	Insecure public key encryption in quantum computing	Unauthorized access	Data Encryption with PQC

For each category of threats identified by STRIDE, Table 1 also lists the countermeasures, with a strong emphasis on data encryption using PQC as the primary method of safeguarding against these vulnerabilities. Additionally, for DoS threats, network access control is listed as a countermeasure. The primary reason for implementing robust security measures is to protect the SSI network platform against various vulnerabilities and threats.

3. Related Work

3.1. MACsec (802.1AE, MAC Security)

MACsec represents a Layer 2 security protocol that ensures the authenticity and integrity of data frames to provide robust protection against replay attacks [22]. It offers multiple advantages in terms of network security, such as reduced header size and support for physical port-level encryption and decryption. Since Layer 2 IoT networks, such as LoRa WAN (Long Range Wide Area Network), are characterized by limited bandwidth, the smaller size of headers improves transmission efficiency compared with IPSec. Additionally, physical port-level encryption and decryption in MACsec significantly enhances network security by protecting data right from their point of origin, ensuring confidentiality, integrity, and authenticity. On top of that, MACsec contributes to high-speed network connections by directly ensuring security at the data link layer. Although there are several implementation options, such as unicast, broadcast, and multicast communications, MACsec's security mechanisms are independent of upper-layer processes to avoid unnecessary modifications when user applications change.

In MACsec communication, hosts within the same CA share a common CAK. The CA leader periodically initiates an EAP request to other hosts to ensure all nodes within the CA use the same CAK. A CAK distributed among the hosts within the same CA is used to generate an Integrity Check Value-Key (ICK) and a Key Encrypting Key (KEK) through AES-ECB. The KEK, derived from the CAK, encrypts the Secure Association Key (SAK) using AES Key Wrap, while the ICK, also derived from the CAK, facilitates message authentication. The SAK encrypts the user data alongside the Security Tag (SecTAG) values in a sequence similar to that of the distributed SAKs, determined by the Association Number (AN) [23].

3.2. NTRUs (Nth Degree Truncated Polynomial Ring Units)

Quantum computers can quickly solve public-key cryptography systems like RSA that rely on problems like integer factorization and discrete logarithms. By finding the period r in a function $f(x) = a^x \bmod N$ involving modular arithmetic, where $f(x+r) = f(x)$, one can rapidly determine the factors of N. They can efficiently compute the quantum Fourier Transform in parallel processing, making it useful for discovering the period r.

However, the NTRU algorithm is based on solving problems such as finding the shortest vector in high-dimensional lattice structures or solving equations with added noise. These problems often involve searching every lattice point or finding solutions through random sampling, which results in exponential time complexity. Due to these characteristics, lattice-based encryption methods like NTRUs resist Shor's algorithm, which quantum computers utilize. NTRUs provide well-suited encryption and decryption processes with limited resources. Additionally, NTRUs' flexibility allows for customizable security levels, ensuring scalability and adaptability in various applications. There are two components in NTRUs: NTRUEncrypt, which is used for encryption, and NTRUSign, which is used for digital signatures. The patent for NTRUEncrypt was released into the public domain in 2017, while NTRUSign remains patented but is available under the GPL for software use [24].

Based on a complex mathematical problem, NTRUs generate two polynomials, $m(x)$ and $r(x)$, from an initial polynomial $f(x)$. These polynomials are then utilized in the encryption and decryption processes. Notable for its rapid processing speed and minimal memory requirements, NTRUs effectively balance efficiency and security, although increases in key length enhance computational complexity [25]. The NTRU algorithm includes a key generation phase that produces the polynomials $r(x)$ and $m(x)$, followed by encryption and decryption phases that utilize these polynomials. Predominantly, NTRUs enhance network communications across various domains to facilitate the secure transmission of data in services such as email, online banking, and file transfers.

Distinguished by their reliance on the lattice-based shortest distance problem, NTRUs offer superior cryptographic strength to counter quantum computer threats. Their swift speed primarily results from polynomial manipulations and the potential for complexity reduction through methods such as the Fast Fourier Transform to $n \log n$, thereby making NTRUs more efficient [14].

4. Methodology

4.1. Overview

SSI-PQM provides authentication, network access control, data encryption with PQC, route management, and network separation. It also offers configuration information for accessing SSI's communication node (CN) during initial authentication. Unlike the original SSI, the novel SSI scheme encrypts this information using PQC-TLS to ensure secure transmission instead of transmitting configuration information in plaintext. During the authentication of VXLAN, SSI-PQM delivers network separation information and encryption key details. Furthermore, the secret key for MACsec and the CAK transmitted are securely protected in externally exposed network segments via encrypted EAP communication utilized by PQC-TLS.

The MACsec was employed for data encryption as an end-to-end encryption algorithm characterized by low CPU usage and compatibility with various network protocols. For network separation, a Layer 2 overlay network based on L2TP and VXLAN was provided. L2TP sessions positioned in public internet segments use PQC-TLS so that data transmission is safe from quantum computer threats. Figure 2 illustrates the functionality of SSI-PQM modules. In the following subsections, the description of MACsec, L2TP, VXLAN, and NTRUs are provided. Next, we will elaborate in detail on the elements of the SSI-PQM platform and the entire operation process, including authentication.

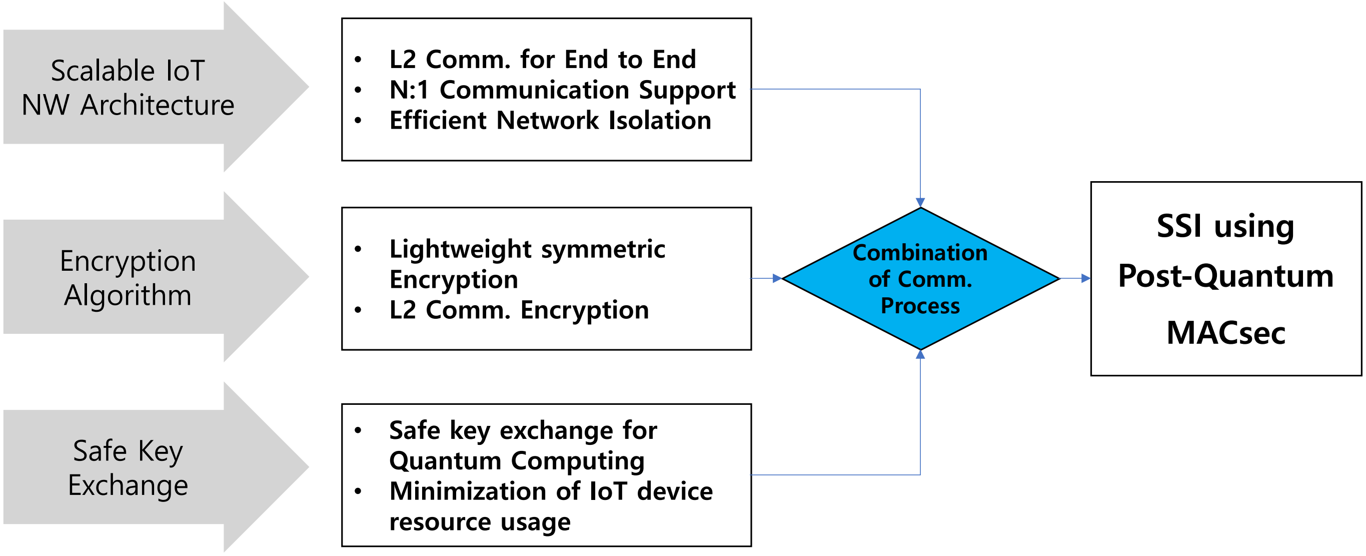

Scalable IoT NW Architecture
• L2 Comm. for End to End
• N:1 Communication Support
• Efficient Network Isolation
Encryption Algorithm
• Lightweight symmetric Encryption
• L2 Comm. Encryption
Safe Key Exchange
• Safe key exchange for Quantum Computing
• Minimization of IoT device resource usage
Combination of Comm. Process
SSI using Post-Quantum MACsec

Figure 2. Description of the function of SSI-PQM.

4.2. PQC-TLS Using NTRUs

Since NTRUs are favored for IoT because of their security and efficiency, they have been applied to SSI-PQM in the current study. IoT devices are usually characterized by limited processing power, small storage, and low energy consumption. Thus, NTRUs' lightweight encryption capability is suitable for an IoT system. NTRUs are secure against quantum attacks, protecting against the vulnerabilities common in traditional methods, namely, RSA and ECC. NTRU encryption and decryption processes are faster than many public key methods, which makes them ideal for resource-limited devices [26]. They offer high security with smaller keys, saving valuable storage space. NTRUs consume less power, which is important for battery-dependent IoT devices. Their adaptability to various protocols supports a wide range of applications of IoT. In summary, NTRUs stand out as an effective solution for IoT security, addressing the challenges of quantum computing and resource constraints [14].

4.2.1. Notations

The NTRU cryptosystem depends on three integer parameters (N, p, q) and four sets $\mathcal{L}_f, \mathcal{L}_g, \mathcal{L}_\phi, \mathcal{L}_m$ of polynomials of degree $N-1$ with integer coefficients. Note that p and q need not be prime, but p and q are relatively prime, and q is considerably larger than p. We work in the ring $R{=}\mathbb{Z}[X]/(X^N-1)$, and element $F \in R$ will be written as a polynomial or vector,

$$F = \int_{i=0}^{N-1} F_i x^i = [F_0, F_1, \ldots, F_{N-1}] \tag{1}$$

We use the symbol $\circledast$ to denote multiplication in R. This 'star multiplication' is given explicitly as a cyclic convolution product,

$$F \circledast G = H \; with \; H_k = \int_{i=0}^{k} F_i G_{k-i} + \int_{i=k+1}^{N-1} F_i G_{N+k-i} = \int_{i+j \equiv k \; (mod \; N)} F_i G_j \tag{2}$$

where a multiplication modulo q involves reducing the coefficients module q.

Now, we will discuss the key generation process. First, two polynomials $f \in \mathcal{L}_f$ and $g \in \mathcal{L}_g$ are chosen. The polynomial f must satisfy the additional requirement of having inverses modulo q and modulo p, which we will denote by F_q and F_p, respectively. Second, the quantity is computed as follows:

$$h \equiv F_q \circledast g \; (mod \; q) \tag{3}$$

Then, the public key is the polynomial h and private key is the polynomial f.

The encryption process consists of two steps. First, a message m is selected from the set of plaintexts $\mathcal{L}_m$.

Next, a polynomial $\phi \in \mathcal{L}_\phi$ is randomly chosen, and the public key h is used to compute

$$e \equiv p\phi \circledast h + m \; (mod \; q) \tag{4}$$

After the computation, e becomes the encrypted message, also known as the ciphertext.

Finally, we will review the method for decrypting encrypted messages. First, precompute the polynomial F_p from private key f. Next, obtain the coefficients of a as follows:

$$a \equiv f \circledast e \; (mod \; q) \tag{5}$$

where a belongs to the interval from $-q/2 \; to \; q/2$. After computing the following process, the final output becomes the recovered message.

$$F_p \circledast a \; (mod \; p) \tag{6}$$

4.2.2. Implementation

We have selected NTRUs as the key exchange algorithm to facilitate the sharing of a secret key within the SSI-PQM system. Figure 3 represents the integration of the NTRU encryption algorithm within the TLS 1.3 protocol, specifically designed to enable secure key exchanges between IoT devices and an Authentication Server or between CN and the CN controller (CN Cont.). It outlines a two-way process that focuses on the generation and decryption of a secret key through NTRUs, a public key encryption algorithm. This process is crucial for establishing a secure communication channel in a PQC environment.

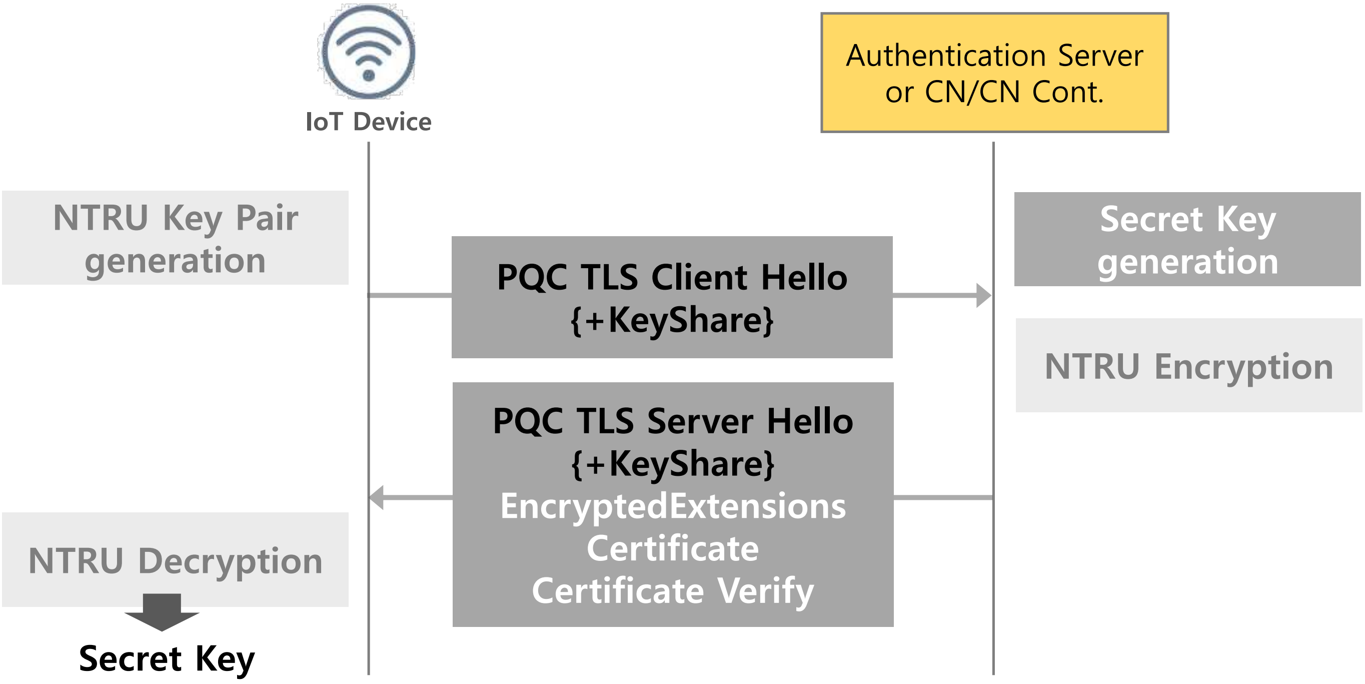

IoT Device
Authentication Server or CN/CN Cont.
NTRU Key Pair generation
PQC TLS Client Hello {+KeyShare}
Secret Key generation
NTRU Encryption
PQC TLS Server Hello {+KeyShare} EncryptedExtensions Certificate Certificate Verify
NTRU Decryption
Secret Key

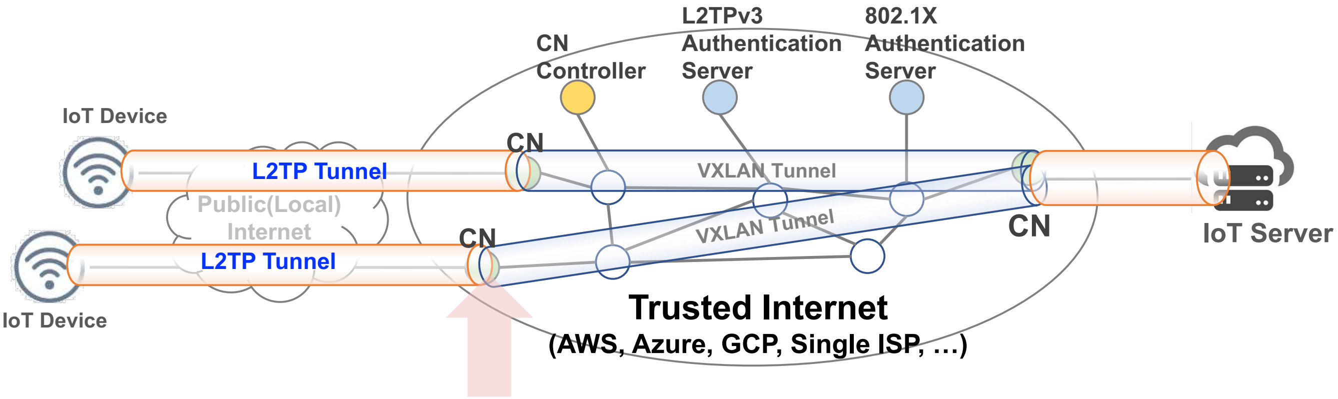
Figure 3. PQC-TLS is applied to the configuration of information provision and EAP communication within the SSI-PQM system. TLS 1.3 is utilized to minimize the overhead associated with TLS communication.

On the IoT device side, the initial step involves generating an NTRU Key Pair. This key pair is crucial for the PQC mechanism to enable the IoT device to securely decrypt the secret key sent by the server. The decryption process using NTRUs ensures that only the intended recipient IoT device can access and utilize the secret key, thereby maintaining the confidentiality and integrity of the transmitted data. We highlight the secure key generation step in Figure 3, which is designed to safeguard the communication from potential eavesdroppers or attackers who might exploit vulnerabilities in classical cryptographic algorithms.

Conversely, the server side in Figure 3 shows the encryption of the secret key using NTRUs before its transmission to the IoT device. This encryption ensures that the secret key is protected throughout its journey from the server to the IoT device for further secure communication. The figure also displays the components of the PQC-TLS handshake process, including the "PQC-TLS Client Hello {+KeyShare}" and "PQC-TLS Server Hello {+KeyShare}, EncryptedExtensions, Certificate, Certificate Verify", illustrating the comprehensive steps involved in establishing a secure TLS session. The visual representation emphasizes NTRUs' role in enhancing security during key exchanges in the quantum computing era, ensuring protection against advanced cryptographic attacks.

4.3. Network Architecture of SSI-PQM

SSI-PQM features a network architecture similar to that of traditional SSI but introduces enhancements to improve the confidentiality of secret keys and control unauthorized access. The network diagram of SSI-PQM is depicted in Figure 4, where it is compared with the conventional SSI diagram. It introduces additional functionalities provided by the authentication server and CN to bolster the security against vulnerabilities inherent in the original SSI system. Authorized IoT devices gain access to the CN by receiving authentication information from the L2TP authentication server and updating access control settings accordingly. The Attribute Value Pairs (AVPs) received after 802.1X and the configuration information for L2TP authentication are encrypted using the AES algorithm with a secret key securely shared via PQC-TLS. This ensures that they are not exposed externally. Furthermore, the CN controller identifies authorized IoT devices, thereby preventing abnormal access attempts.



i. CN protects L2TP config information and CAK of MACsec.
ii. CN allows access to IoT devices that have been identified and authorized.

Figure 4. SSI-PQM using global IaaS: The SSI platform, constructed using the global IaaS, maintains the same Layer 2 network and MACsec encryption. L2TP and EAP communications over the public internet are secured using PQC-TLS.

Figure 5 illustrates the two-layer end-to-end session formation process by SSI-PQM. The SSI framework undergoes a four-step process to establish end-to-end and many-to-many communications. SSI-PQM retains all of the functionalities of the SSI along with enhancing the confidentiality of config information and CAK and strengthening the access control of the IoT devices. Additionally, SSI-PQM applies the following enhancements to steps #1 to #3 of the SSI process:

Step #1: IoT Device Authentication and Config Info. Provision—When an IoT device attempts authentication, the server redirects communication to the nearest CN. Subsequently, the IoT device initiates a PQC-TLS handshake process with the CN. After successfully completing the handshake, the IoT device proceeds to authenticate via the CN. This step allows the IoT device to receive the necessary configuration information for L2TP from the authentication server.

Step #2: L2TP Tunnel with Communication Node (CN)—The CN receives the IP address of the authorized IoT device, along with the provisioned Tunnel ID and Session ID from the L2TP authentication server. When the IoT device attempts to establish an L2TP session, the CN grants access based on the provisioning information previously received from the authentication server.

Step #3: 802.1X Authentication/Authorization—The IoT devices access the CN via the L2TP tunnel and undergo 802.1X authentication using EAP. Upon receiving authorization from the 802.1X authentication server, the CN acquires AVPs through encrypted communication.

Figure 5 illustrates the detailed process of our proposal. Although NTRUs have reduced the encryption and decryption times compared with RSA, they have led to an increase in overall computing resource usage due to the enhanced security level. This trend could potentially degrade the performance of the IoT devices. Our SSI-PQM system integrates PQC-TLS into L2TP and MACsec to securely store critical information such as secret keys. One of the previous studies reported that NTRUs demonstrated shorter encryption and decryption times than RSA [27]. The authors observed that as the security level increased, so did the demand for computing resources, which in turn could have decreased the performance of IoT devices. In this study, we have designed a system that securely shares a secret key for encrypting essential information in L2TP and MACsec within a single PQC TLS session. Table 2 details the process by which SSI-PQM minimizes the transmission of PQC-TLS's public key and the encryption/decryption of the session key. PQC-TLS is applied once to share the pre-master secret between the CN and the IoT device. This pre-master secret is then relayed to the 802.1X authentication server. Subsequently, the pre-master secret is transmitted back to the CN, where it is reused for generating a Pre-Shared Key (PSK) for encrypted communication using EAP-PSK.

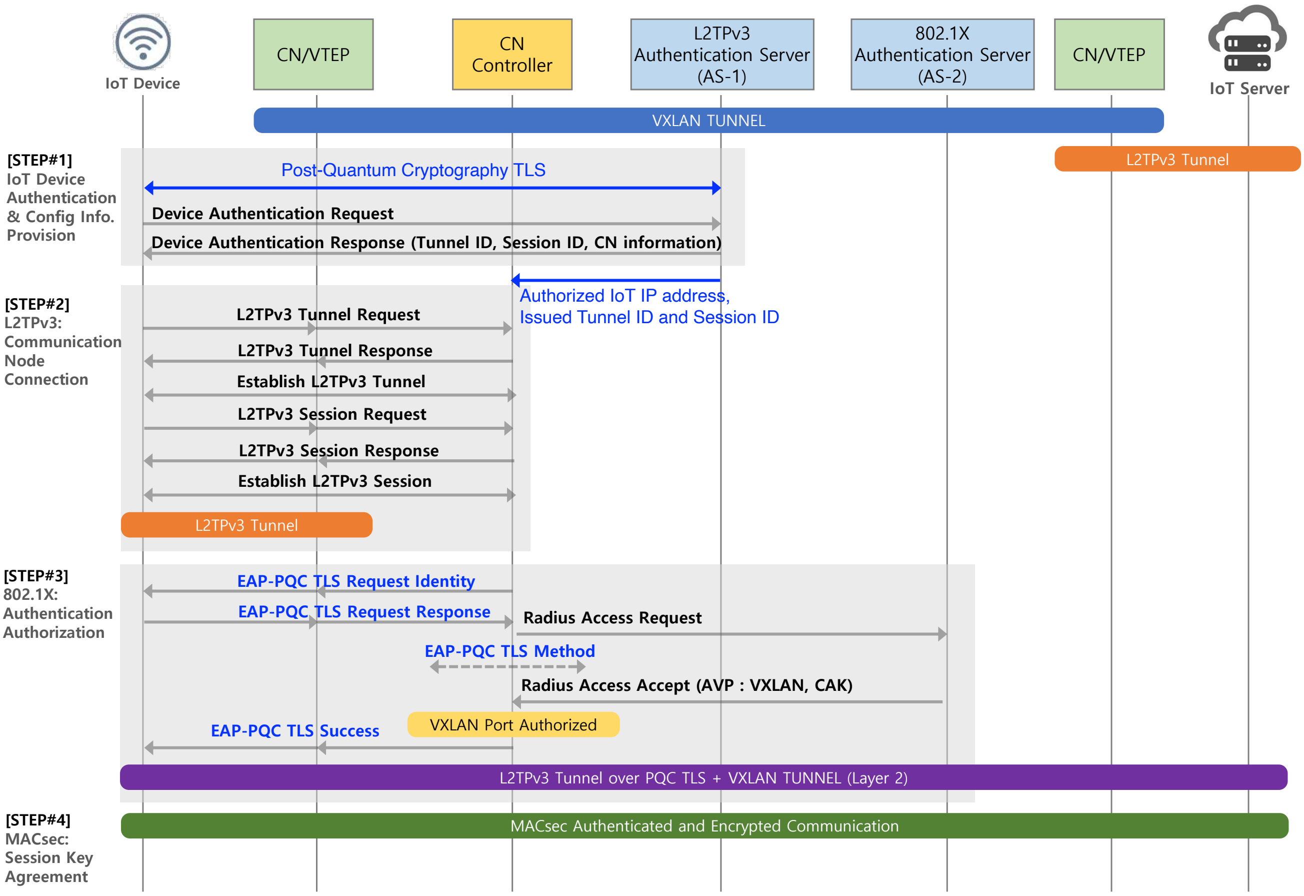

IoT Device
CN/VTEP
CN Controller
L2TPv3 Authentication Server (AS-1)
802.1X Authentication Server (AS-2)
CN/VTEP
IoT Server
VXLAN TUNNEL
L2TPv3 Tunnel
[STEP#1] IoT Device Authentication & Config Info. Provision
Post-Quantum Cryptography TLS
Device Authentication Request
Device Authentication Response (Tunnel ID, Session ID, CN information)
Authorized IoT IP address, Issued Tunnel ID and Session ID
[STEP#2] L2TPv3: Communication Node Connection
L2TPv3 Tunnel Request
L2TPv3 Tunnel Response
Establish L2TPv3 Tunnel
L2TPv3 Session Request
L2TPv3 Session Response
Establish L2TPv3 Session
L2TPv3 Tunnel
[STEP#3] 802.1X: Authentication Authorization
EAP-PQC TLS Request Identity
EAP-PQC TLS Request Response
Radius Access Request
EAP-PQC TLS Method
Radius Access Accept (AVP : VXLAN, CAK)
EAP-PQC TLS Success
VXLAN Port Authorized
L2TPv3 Tunnel over PQC TLS + VXLAN TUNNEL (Layer 2)
[STEP#4] MACsec: Session Key Agreement
MACsec Authenticated and Encrypted Communication

Figure 5. SSI-PQM platform process: The process from IoT device authentication by AS-1 → L2TP tunneling over PQC-TLS → 802.1X authentication over PQC-TLS → MACsec Key Agreement. The process ultimately leads to encrypted communication between the IoT device and the IoT server through the SSI platform.

Table 2. The process of applying the pre-master secret used in PQC-TLS to EAP-PSK.

1. PQC -TLS Communication Setup in Step#1
   - (a) IoT device and CN initiate PQC-TLS handshake.
   - (b) CN generates a pre-master secret randomly:

$$\text{Secret Key} = \text{KDF}(\text{PreMasterSecret}, \text{“session key derivation”}) \tag{7}$$

   - (c) CN securely distributes the pre-master secret to the IoT device.
   - (d) IoT device and CN derive session keys from the pre-master secret:

$$\text{Secret Key} = \text{KDF}(\text{PreMasterSecret}, \text{“session key derivation”}) \tag{8}$$

2. L2TP Authentication and 802.1X Authentication Preparation in Step#1
   - (a) IoT device attempts L2TP authentication with CN using the established PQC-TLS session.
   - (b) **if** L2TP authentication succeeds **then**
     - Proceed with 802.1X Authentication Preparation in Step#1.
       - i. CN prepares IoT device's account information, MAC address, and pre-master secret.
       - ii. CN transmits the prepared information to the 802.1X authentication server.
   - (c) **else**
     - Terminate PQC-TLS communication and abort the process.
3. EAP-PSK-based 802.1X Authentication in Step#3
   - (a) IoT device initiates 802.1X authentication with CN, using EAP-PSK protocol.
   - (b) PSK is derived from the previously exchanged pre-master secret:

$$\text{PSK} = \text{KDF}(\text{PreMasterSecret}, \text{“PSK derivation”}) \tag{9}$$

5. Experimental Environment and Results

5.1. Experimental Environment

To evaluate our proposal, we assessed the impact of applying the NTRU algorithm to PQC-TLS on IoT device performance. The performance gap was measured as the security level increased in terms of CPU usage and operation time. Experiments were conducted on an IoT device, which consists of Raspberry Pi 3B+, featuring a 64-bit ARM Cortex-A53 Quad-Core Processor and 1GB Memory to mimic the SSI experiment environment under identical conditions. In addition, a server was equipped with a 2.4 GHz 8-Core Intel Core i9 and 64 GB Memory to utilize the NTRU cryptographic algorithm across six security levels: 80, 112, 128, 160, 192, and 256 bits. Considering the issuance of certificates to the devices or the direct generation on the IoT devices, experiments were carried out on both the server and IoT devices to test the PQC key pair generation. Furthermore, we also measured the performance of the encryption of the secret key on the server and the decryption of the same on the IoT device within the PQC-TLS framework.

Table 3 compares the key sizes of the NTRUs and RSA across various security levels [15]. By measuring the computing resources required for key generation, encryption, and decryption at each security level for the NTRUs, we aimed to validate the effectiveness of SSI-PQM as an IoT network platform. Our findings were then compared to previous research that reported that NTRUs exhibit superior performance compared to RSA at similar security levels, with respect to key size [28].

Table 3. Key size comparison between NTRUs and RSA: NTRU Key Size = N $\log_2(q)$.

Security Level (bits)	Example Values	RSA Key Size (bits)	Notes
80	$N = 251, p = 3, q = 2048$	1024	Constrained environments
112	$N = 401, p = 3, q = 2048$	2048	Balanced security and performance
128	$N = 439, p = 3, q = 2048$	3072	Standard security level
160	$N = 487, p = 3, q = 2048$	4096	Higher security
192	$N = 593, p = 3, q = 2048$	7680	Enhanced security
256	$N = 743, p = 3, q = 2048$	15,360	Maximum security

Since the client, corresponding to the IoT device, is responsible for PQC key pair generation and the decryption of the ciphertext, we measured the time taken for these processes. Our implementation refers to the Python version of NTRUs by pointed sphere [14,29].

5.2. Results

Table 4 displays the time taken for PQC key generation, encryption, and decryption across different security levels. At first glance, it is found that the CPU performance significantly impacts the time required for PQC key generation.

In the context of key generation with a security level above 128, the NTRU algorithm demonstrated superior performance compared to RSA. However, in terms of encryption and decryption processes, RSA exhibited superior performance. Nevertheless, the time delay that occurred during NTRU encryption and decryption processes was negligible in comparison to the overall communication performance. In the SSI-PQM process, the NTRU-based public key encryption was utilized once in Step #1 for the transmission of the pre-master secret. Consequently, the time delay experienced during NTRU encryption and decryption did not affect the overall communication between the servers and IoT devices.

From these evaluation results, we identified that PQC-TLS corresponding to security levels 80 and 128 is the most suitable with respect to the performance of the IoT device. The performance of NTRUs significantly surpassed that of RSA, which can be attributed to faster generation of public key pairs for communications that require a security level above 160. These results lead to improvements ranging from at least 161% to as much as 8255%.

Table 4. Comparison of the time (second) spent on public key generation, encryption, and decryption using the NTRU algorithm in PQC.

Security Level (bits)	Minimum Latency for TLS (s.) (①+②+③)		① Key Generation (IoT)		② Encryption (Server)		③ Descryption (IoT)	
	NTRUs	RSA	NTRUs	RSA	NTRUs	RSA	NTRUs	RSA
80	123.4110	2.437	105.2439	2.3780	0.6336	0.0001	17.5335	0.0589
112	273.1801	58.4968	251.3992	58.1634	0.8043	0.0002	20.9766	0.3332
128	319.3766	245.786	293.9652	244.7936	0.7778	0.0004	24.6336	0.9920
160	409.9317	658.7048	381.2517	656.5520	0.9558	0.0006	27.7242	2.1522
192	612.2242	8214.6037	571.8158	8202.1173	1.0726	0.0017	39.3358	12.4847
256	978.7406	80,798.9095	916.3855	80,710.4963	1.2568	0.0070	61.0983	88.4062

As the security level increases, the time required for generating RSA public keys rises exponentially. An RSA public key is based on the product of two large prime numbers, and the difficulty of finding large primes increases exponentially with their size. To generate these primes, random numbers must be created in the desired range. This process requires significant computational effort due to the scarcity and increased length of larger prime numbers, which increases the challenge of their discovery. In contrast, key generation in

NTRUs involves generating random polynomials and performing arithmetic in modular polynomial rings. NTRUs are based on solving the shortest vector problem (SVP) in a lattice and specific polynomial equations, which remain computationally efficient even with longer key lengths. Moreover, NTRU key generation enhances security based on the probabilistic hardness assumptions underlying lattice problems and relies on random polynomial selection to ensure unpredictability to prevent attackers from predicting or reversing keys.

Moreover, for the effective operation of SSI-PQM, it is necessary to establish a separate certificate issuance system for generating PQC public keys and then providing them to the IoT devices. Since the PQC decryption performed by the IoT device is executed only once for secret key exchange, it has a insignificant impact on the encryption of communication using MACsec.

Table 5 lists the comparison of the security function with the previous IoT platforms in terms of IoT network security aspects. It includes authentication, access control, network separation (or secure routing), encryption, detection, and SDN [30–32]. Our proposed scheme supports all the security aspects except detection. Detection can be easily applied without affecting the response time by mirroring traffic through the CN.

Table 5. Security function comparison with previously reported IoT platforms.

Function	**SSI-PQM**	**SSI (2021) [9]**	**Linda et al. (2018) [33]**	**Kumar et al. (2019) [34]**	**McCormack et al. (2020) [35]**	**Irshad et al.(2023) [36]**
PQC	Yes	No	No	No	No	Yes
layer 2 Communication	Yes	Yes	No	No	No	No
Net Separation	Yes	Yes	Yes	No	No	Yes
Authentication	Yes	Yes	No	Yes	No	Yes
Access Control	Yes	Yes	Yes	No	Yes	Yes
End-to-End Enc.	Yes	Yes	No	Yes	No	Yes
Many-to-Many Enc.	Yes	Yes	No	Yes	No	Yes
L2 Encryption	Yes	Yes	No	No	No	Yes
SDN	Yes	Yes	Yes	No	Yes	No
Detection	No	No	No	No	Yes	Yes

6. Conclusions and Outlooks

SSI-PQM effectively counters security threats, such as quantum computing-based Man-In-The-Middle (MITM) attacks, route tampering, and privacy breaches in open IoT networks, by employing an overlay network, end-to-end encryption, authentication, and PQC. Furthermore, the applications of MACsec and NTRU algorithms for encryption proved to be efficient for IoT devices due to their low CPU usage, while the Layer 2 overlay network facilitates the unrestricted use of various communication protocols. The experimental results showed satisfactory performance across security levels ranging from 80 to 128, and a minimum speed enhancement of 161% compared to RSA for security levels exceeding 160 was secured.

With the rise of SaaS-based IoT and edge computing, the significance of security in industrial IoT, smart home and city, and healthcare is in increasing demand. Following COVID-19, an uptick in telecommuting and remote work collaboration is expected to enhance the deployment of SSI-PQM. The current study can broaden its application spectrum with a VPN architecture to provide a network environment conducive to collaboration with offices, even remotely. Next, we will proceed with our research on applying Quantum Key Distribution (QKD) to SSI as an alternative to PQC-TLS, aiming to maintain the security of Layer 2 communications.

Author Contributions: Writing—original draft, Juhee Choi and Junwon Lee; Writing—review & editing, Juhee Choi and Junwon Lee. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by a 2023 Research Grant from University Innovation Support Project of Sangmyung University.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/junimirang/SSI-PQM (accessed on 13 May 2024).

Acknowledgments: This research was supported by a 2023 Research Grant from University Innovation Support Project of Sangmyung University.

Conflicts of Interest: Author Junwon Lee was employed by the company Samsung SDS. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Ahmid, M.; Kazar, O.; Barka, E. Internet of Things Overview: Architecture, Technologies, Application, and Challenges. In *Decision Making and Security Risk Management for IoT Environments*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 1–19.
2. Bommu, S.; Babburu, K.; N, S.; Thalluri, L.N.; Gopalan, A.; Mallapati, P.K.; Guha, K.; Mohammad, H.R. Smart City IoT System Network Level Routing Analysis and Blockchain Security Based Implementation. *J. Electr. Eng. Technol.* **2023**, *18*, 1351–1368. [CrossRef] [PubMed]
3. Rana, P.; Patil, B. Cyber security threats in IoT: A review. *J. High Speed Netw.* **2023**, *29*, 105–120. [CrossRef]
4. Sheng, H.; Zhu, Q.; Tao, J.; Zhang, H.; Peng, F. Distribution network reconfiguration and photovoltaic optimal allocation considering harmonic interaction between photovoltaic and distribution network. *J. Electr. Eng. Technol.* **2024**, *19*, 17–30. [CrossRef]
5. Wang, C.; Wang, Z.; Guan, W.; Wang, W.; Xu, L.; Li, L.; Huang, S.; Wang, W. Trustworthy Health Monitoring Based On Distributed Wearable Electronics With Edge Intelligence. *IEEE Trans. Consum. Electron.* **2024**, *70*, 2333–2341. [CrossRef]
6. Liu, L.; Feng, J.; Wu, C.; Chen, C.; Pei, Q. Reputation Management for Consensus Mechanism in Vehicular Edge Metaverse. *IEEE J. Sel. Areas Commun.* **2023**, *42*, 919–932. [CrossRef]
7. Ukil, A.; Bandyoapdhyay, S.; Puri, C.; Pal, A. IoT healthcare analytics: The importance of anomaly detection. In Proceedings of the 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA), Crans-Montana, Switzerland, 23–25 March 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 994–997.
8. Hassija, V.; Chamola, V.; Saxena, V.; Jain, D.; Goyal, P.; Sikdar, B. A survey on IoT security: Application areas, security threats, and solution architectures. *IEEE Access* **2019**, *7*, 82721–82743. [CrossRef]
9. Lee, J.; Lee, H. Secure and Scalable IoT: An IoT Network Platform Based on Network Overlay and MAC Security. In Proceedings of the IFIP International Conference on ICT Systems Security and Privacy Protection, Oslo, Norway, 22–24 June 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 287–301.
10. Nielsen, M.A.; Chuang, I.L. Quantum computation and quantum information. *Phys. Today* **2001**, *54*, 60.
11. Aithal, P. Advances and new research opportunities in quantum computing technology by integrating it with other ICCT underlying technologies. *Int. J. Case Stud. Business Educ.* **2023**, *7*, 314–358. [CrossRef]
12. Berberich, J.; Fink, D. Quantum computing through the lens of control: A tutorial introduction. *arXiv* **2023**, arXiv:2310.12571.
13. Banegas, G.; Bernstein, D.J.; Van Hoof, I.; Lange, T. Concrete quantum cryptanalysis of binary elliptic curves. *Cryptol. Eprint Arch.* **2020** . [CrossRef]
14. Hoffstein, J.; Pipher, J.; Silverman, J.H. NTRU: A ring-based public key cryptosystem. In Proceedings of the International Algorithmic Number Theory Symposium, Portland, OR, USA, 21–25 June 1998; Springer: Berlin/Heidelberg, Germany, 1998; pp. 267–288.
15. Hermans, J.; Vercauteren, F.; Preneel, B. Speed records for NTRU. In Proceedings of the Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, 1–5 March 2010; Springer: Berlin/Heidelberg, Germany, 2010; pp. 73–88.
16. Grover, L.K. Synthesis of quantum superpositions by quantum computation. *Phys. Rev. Lett.* **2000**, *85*, 1334. [CrossRef] [PubMed]
17. Shor, P.W. Algorithms for quantum computation: Discrete logarithms and factoring. In Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, USA, 20–22 November 1994; IEEE: Piscataway, NJ, USA, 1994; pp. 124–134.
18. Ghosh, S. Quantum-Resistant Security Framework for Scada Communication in Industrial Control Systems. Ph.D. Thesis, Dalhousie University, Halifax, NS, Canada, 2023.
19. CyaSSL+NTRU High-Performance SSL. Available online: https://www.wolfssl.com/documentation/flyers/cyassl_ntru.pdf (accessed on 23 March 2024).
20. McGrew, D.; Viega, J. The Galois/counter mode of operation (GCM). *Submiss. Nist Modes Oper. Process* **2004**, *20*, 10.
21. Shahan, R.; Phil Meadows, B.L. IoT Security Architecture. Available online: https://docs.microsoft.com/en-us/azure/iot-fundamentals/iot-security-architecture (accessed on 23 March 2024).
22. Carnevale, B.; Fanucci, L.; Bisase, S.; Hunjan, H. Macsec-based security for automotive ethernet backbones. *J. Circuits Syst. Comput.* **2018**, *27*, 1850082. [CrossRef]

23. Lee, J.W.; Park, S.H.; Gum, K.H.; Chung, T.M. Design of secure arp on MACsec (802.1 AE). In Proceedings of the 5th International Conference on Ubiquitous Information Technologies and Applications, Sanya, China, 16–18 December 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 1–4.
24. Schanck, J. Practical Lattice Cryptosystems: NTRUEncrypt and NTRUMLS. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
25. Hülsing, A.; Rijneveld, J.; Schanck, J.; Schwabe, P. High-speed key encapsulation from NTRU. In Proceedings of the International Conference on Cryptographic Hardware and Embedded Systems, Taipei, Taiwan, 25–28 September 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 232–252.
26. Kadam, V.R.; Naidu, P.S. Lightweight Cryptography to Secure Internet of Things (IoT). *Int. Res. J. Eng. Technol.* **2020**, *7*, 5.
27. Harjito, B.; Tyas, H.N.; Suryani, E.; Wardani, D.W. Comparative Analysis of RSA and NTRU Algorithms and Implementation in the Cloud. *Int. J. Adv. Comput. Sci. Appl.* **2022**, *13*, 247960097. [CrossRef]
28. Nandanavanam, A.; Upasana, I.; Nandanavanam, N. NTRU and RSA cryptosystems for data security in IoT environment. In Proceedings of the 2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE), Bengaluru, India, 9–10 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 371–376.
29. NTRU_Python. Available online: https://github.com/pointedsphere/NTRU_python (accessed on 4 November 2021).
30. Alaba, F.A.; Othman, M.; Hashem, I.A.T.; Alotaibi, F. Internet of Things security: A survey. *J. Netw. Comput. Appl.* **2017**, *88*, 10–28. [CrossRef]
31. Farris, I.; Taleb, T.; Khettab, Y.; Song, J. A survey on emerging SDN and NFV security mechanisms for IoT systems. *IEEE Commun. Surv. Tutorials* **2018**, *21*, 812–837. [CrossRef]
32. Hassan, W.H. Current research on Internet of Things (IoT) security: A survey. *Comput. Netw.* **2019**, *148*, 283–294.
33. Shif, L.; Wang, F.; Lung, C.H. Improvement of security and scalability for IoT network using SD-VPN. In Proceedings of the NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium, Taipeim Taiwan, 23–27 April 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–5.
34. Kumar, S.; Hu, Y.; Andersen, M.P.; Popa, R.A.; Culler, D.E. JEDI Many-to-Many End-to-End Encryption and Key Delegation for IoT. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019 ; pp. 1519–1536.
35. McCormack, M.; Vasudevan, A.; Liu, G.; Echeverría, S.; O'Meara, K.; Lewis, G.; Sekar, V. Towards an Architecture for Trusted Edge IoT Security Gateways. In Proceedings of the 3rd {USENIX} Workshop on Hot Topics in Edge Computing (HotEdge 20), Santa Clara, CA, USA, 30 April 2020.
36. Irshad, R.R.; Hussain, S.; Hussain, I.; Nasir, J.A.; Zeb, A.; Alalayah, K.M.; Alattab, A.A.; Yousif, A.; Alwayle, I.M. IoT-Enabled Secure and Scalable Cloud Architecture for Multi-User Systems: A Hybrid Post-Quantum Cryptographic and Blockchain based Approach Towards a Trustworthy Cloud Computing. *IEEE Access* **2023**, *11*, 105479–105498. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.