An Efficient Implementation of NTRU Encryption in Post-Quantum Internet of Things

1st Y. M. Agus, 2nd M. A. Murti
School of Electrical Engineering
Telkom University
Bandung, Indonesia
ninoagus@protonmail.com,
arymurti@telkomuniversity.ac.id

3rd F. Kurniawan, 4th N.D.W. Cahyani
School of Computing
Telkom University
Bandung, Indonesia
febrian_k@protonmail.com,
nikencahyani@telkomunversity.ac.id

5th,* G.B. Satrya
School of Applied Science
Telkom University
Bandung, Indonesia
gbs@telkomuniversity.ac.id

***Abstract*—Cisco stated that in 2020 there would be 50 billion smart objects connected to the Internet. This adoption rate of digital infrastructure is five times faster than that of the electricity and telephony. The Internet of Things (IoT) or the Internet of Everything (IoE) goes even further beyond, not only it is affecting the way of exchanging data but also touching the physical lives. IoT comprises three things i.e., information technology, operational technology, and smart objects. On the other side, security challenges of an end to end device communication need to be addressed i.e., compliance & regulation, protocols, remediation, impact & risk, threat diversity, and new application. This research demonstrates the impacts & the risks along with the threat diversities of IoT. This research also provides proof of concepts of a security infrastructure for an end to end communication among the devices. Moreover, this research proposes and implements lightweight post-quantum cryptography in Raspberry Pi3 B+ end to end communication. The results suggest some critical points that should be considered for the future development of smart homes, smart factories, smart cities, smart health, etc.**

***Index Terms*—IoT, protocol, sensor nodes, secured communication, vulnerability.**

I. Introduction

The Internet of Everything (IoE) connects the unconnected to create business value e.g., people, data, process, and things [1]. Physical devices and things connected to the Internet and each other for intelligent decision making are also called the Internet of Things (IoT). Thus, IoT transforms data into experiences i.e., critical data information over the public networks, big data, etc. In IoT, data information security over the Internet should be addressed carefully because nowadays there are around 50 billion smart objects that have already been connected. Moreover, security engineers or developers should consider that the IoT devices have small resources and limited computation before developing their secure end to end communication.

Many directions of research and methods to overcome the IoT security issues have been proposed from different aspects. Initiated with the survey by [2] about the implementations of lattice-based cryptography on hardware and the survey by [3] about lattice-based public-key cryptosystem for IoT environment, the end-to-end network communication needs to be observed thoroughly. Some other researchers proposed the implementation of lattice-based cryptography e.g., smart card [4], 32-bit ARM Cortex-M4F [5], Xilinx Zynq-7000 [6], 8-bit AVR Processors [7], and CC2538 microcontroller [8]. However, they have not specifically mentioned the network topology that was used. Embarking with this preliminary literature, this research provides several state-of-the-art implementations of lattice-based cryptography for end-to-end communication.

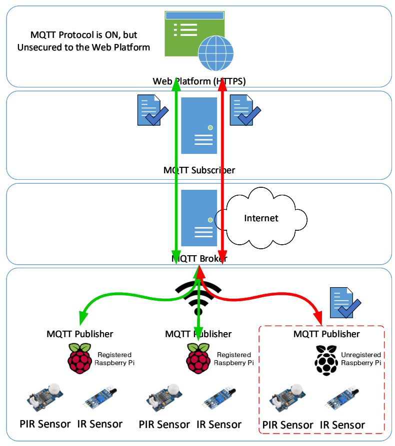

Figure. 1: Conventional IoT topology using MQTT [9]

The implementation of the IoT system normally consists of wireless sensors as the publisher, access point (AP) as connected point, Internet, and server as subscriber/cloud (as depicted in Fig. 1). This research was conducted by using three Raspberry Pi3 B+ (RPi), one AP, and one server enabled with HTTPS service. It assumed that two devices were registered as IoT devices and one device acted as an unregistered IoT device. From a particular point of view, the adversary might invade into the local network if the IoT system uses the default configuration even when the MQTT protocol is enabled. The IoT system should ban unregistered devices. Further explanation will be provided in the next sections.

This research was collaborated between School of Electrical Engineering, School of Computing, and School of Applied Science, Telkom University. This research was also funded by PPM, Telkom University.

The contributions of this research are as follows:

1) Presenting the vulnerability in existing IoT protocol communication between publisher and subscriber.
2) Proposing a secure communication protocol by using NTRU encryption to ensure that the unregistered IoT devices can not connect.
3) Comparing the proposed protocol with AES, Fernet and RSA encryption.
4) Presenting guidance for secure end to end communication among IoT devices by considering that smart homes, smart factories, smart cities, smart health, or etc., are managing and exchanging critical messages.

The § II reviews existing works specifically in the practical implementation of IoT security while § III explains the proposed lightweight NTRU public-key encryption for IoT devices. Then, the specific aspect of unsecure and secure experiments are described in § IV. Finally, § V gives the conclusions and future recommendations of this research.

II. Related Works

Pöppelmann reviewed the implementations of Ring-LWE encryption and Bimodal Lattice Signature Scheme (BLISS) on an 8-bit Atmel ATxmega128 microcontroller [10]. All public and private keys were presumed to be stored in the flash of the microcontroller. Despite the reviews provided about the previous implementations of NTT and the improved approach that can significantly lower the runtime for polynomial multiplication, it has not been tested in a real network environment yet. The results showed that the implementation of Ring-LWE e.g., encryption takes 27 ms for the encryption and 6.7 ms for the decryption.

Buchmann et al. showed the practical potential of replacing the Gaussian noise distribution in the Ring-LWE based encryption scheme with a binary distribution *(R-BinLWEenc)* [11]. Due to the simple structure of R-BinLWEEnc, it is well suited for implementation on embedded devices. In C implementation, their scheme could enable public-key encryption even on very small and low-cost 8-bit (ATXmega128) and 32-bit (Cortex-M0) microcontrollers. However, the method has been implemented only in memory level, and not yet in the real case network implementation.

Guillen et al. analyzed the feasibility of employing the NTRU encryption scheme in resource-constrained devices such as those used for IoT endpoints [12]. They described four different NTRU encryption implementations on an ARM Cortex M0-based microcontroller, compared their results, and showed that NTRU encryption was suitable for use in battery-operated devices. However, they only implemented on the Infineon XMC1100 ARM Cortex-M0 32-bit microcontroller and didn't conduct in the real network implementation.

To achieve efficient leveled authentication, Liu et al. proposed a lightweight public-key encryption scheme that can produce very short ciphertexts without sacrificing its security [13]. They used Learning With Secretly Scaled Errors in Dense Lattice *(referred to as Compact-LWE)* problem on a small IoT device with an 8MHZ MSP430 16-bit processor and 10KB RAM. Even though they conducted the experiment with the 802.15.4 and 6LoWPAN protocols the authentication results still need to be developed i.e., 640ms *(for the first level authentication)*, 8373ms *(for the 16th level authentication)*. Again, this implementation didn't state clearly about the network topology.

Liu et al. studied the efficient techniques of lattice-based cryptography on the processors and presented the first implementation of ring-LWE encryption on ARM NEON and MSP430 architectures [14]. For ARM NEON architecture, a vectorized version of Iterative Number Theoretic Transform (NTT) was proposed for the high-speed computation of polynomial multiplication. While in MSP430 architecture, the study recommended an optimized SWAMS2 reduction technique consists of five different basic operations, including shifting, swapping, addition, as well as two multiplication-subtractions.

Xu summarized the advantages of lattice-based cryptography and the state-of-the-art implementations for IoT devices [15]. The study implemented lattice-based cryptography on FPGAs e.g., V6LX75T(128bit), S6LX9(128bit), S6LX25(128bit), and S6LX9(80bit). The results showed that lattice-based cryptography is practical even for resource-constrained devices. Regarding the computational speed, lattice-based cryptography is faster than traditional public-key cryptography such as RSA or even ECC. Nonetheless, in practice, lattice-based cryptography needs more network communication costs and consumes more resources.

Khalid et at. surveyed the practicality of deployment of Lattice-based Cryptography (LBC) [16]. In this context, the state-of-the-art LBC implementations on the constrained devices (including low-power FPGAs and embedded microprocessors), leading in terms of low-power footprint, small area, compact bandwidth requirements and high performance is fairly evaluated, and bench-marked. These implementations have been optimized in assembly using techniques specific to Cortex-M4. However, this implementation just on the local network, it should consider in the real internetworking implementation.

III. Proposed Design

Fig. 1 in I , shows the deployment of conventional MQTT implementation for IoT topology or called as unsecured transaction method [9]. Most of the simulation and evaluation of MQTT just gave the assurance on the protocol level from sensor nodes to the MQTT broker. But from the adversary's point of view *(i.e., unregistered sensor nodes)*, the data sensor over the web-socket still can be diagnosed with plain text. Based on the preliminary research, Shor's algorithm was proposed to be used in quantum computing [17] and, to address this proposition, this research suggests lightweight post-quantum cryptography between publishers and subscribers to prevent the unregistered sensor nodes in IoT environments. The proposed IoT topology for securing and end to end communication from sensor nodes to the cloud or called a secured transaction method is illustrated in Fig. 2.

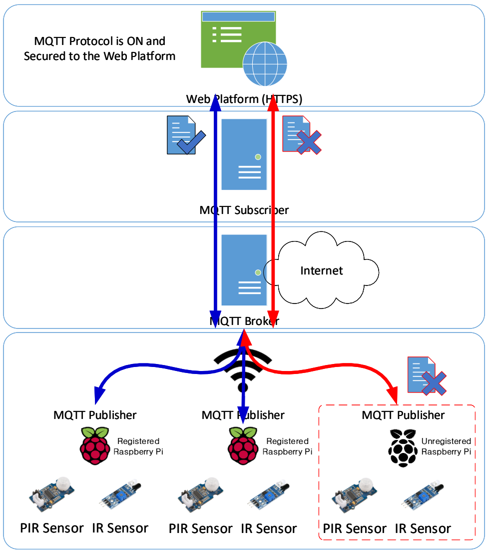

Figure. 2: Proposed secure transaction IoT topology

A. Unsecured Transaction

The implementation used MQTT protocol which is an ISO standard for a lightweight publish-subscribe network protocol. The MQTT is established with two network objects e.g., message brokers or subscribers, and several clients or publishers. Three RPi IoT end devices opted as representatives of Cortex-A53 (ARMv8) 64-bit SoC and this scenario used a real testbed for smart home as a representative of IoT systems. The end to end communication between publisher and subscriber by using the Internet was conducted for this vulnerability test.

B. Secured Transaction

In consideration of vulnerability issue and post-quantum computing, this research encourages the implementation of lightweight post-quantum cryptography i.e., NTRU for the communication between publisher and subscriber. Beginning with the outstanding research from [18], NTRU can be implemented in various computer systems. The proposed design not only implements the encryption in RPi devices but also considers the security in IoT system. This research also presents algorithms for the encryption and the decryption. Algorithm 1 illustrates the encryption in the publisher's side when sending the `message`. While Algorithm 2 illustrates the decryption in the subscriber's side when receiving the `message`.

IV. Security Analysis

TABLE I: Plaintext specification for AES-128, Fernet-128, RSA-2048, RSA-7680, NTRU-401, NTRU-593.

File Name	Size (bytes)	Plain String
plain_01.txt	1	8
plain_02.txt	2	72
plain_03.txt	3	556
plain_04.txt	4	1362
plain_05.txt	5	31513

Algorithm 1 IoT Encryption for the Publisher

INITIALISE *message*;
INITIALISE *publisher_ID*;
INITIALISE *ciphertext_array;*
Function: cryptosystem (message,key) ▷ AES, Fernet, RSA or NTRU
Function: MQTT (ciphertext_array)
while true **do**
    Ciphertext = base64(concatenate(ciphertext_array))
    final_data = concatenate(publisher_ID + splitter + Ciphertext)
    CALL ← MQTT(final_data)
end while

Algorithm 2 IoT Decryption for the Subscriber

INITIALISE *ciphertext_array = MQTT();*
INITIALISE plain_array;
Function: cryptosystem_decryption(ciphertext) ▷ AES, Fernet, RSA or NTRU
while true **do**
    plain_array = cryptosystem_decryption (base64 (ciphertext_array))
    if plain_array != "ERROR": **then**
        Store_data(plain_array)
    else
        Record_Error(Device_ID)
    end if
end while

A. Observation on Unsecured Transaction

The important attributes for this topology are message type (publisher and subscriber), QoS level, message length, topic length, topic type, and message. This implementation focuses more on the `message` itself by considering the Kerckhoffs' principle [19]. For an example, the plaintext from the data that was sent from the registered RPi is `"Current Temp, 30C"` as explained in Table I. Even when the MQTT is enabled, the `message` still can be revoked as a plaintext. This proof should be addressed to overcome the eavesdropping by unregistered RPi in public networks as can be seen in Fig. 3.

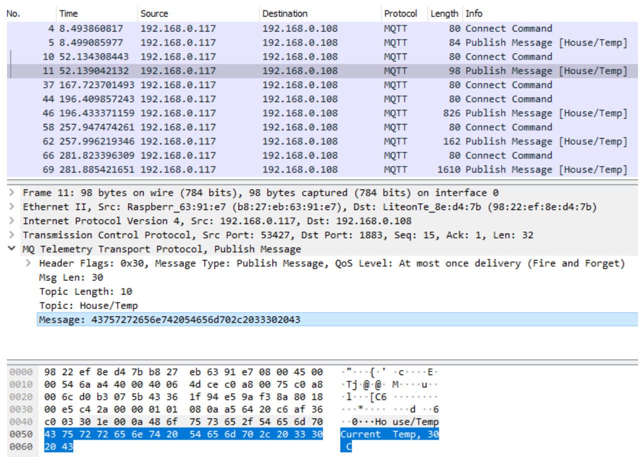

Figure. 3: Analyzing for the conventional topology

B. Observation on Secured Transaction

This implementation used algorithm 1 and 2 and was recorded. Fig. 4 shows that NTRU encryption was successfully established and proved to be more secure in the `message` part. The `message` was strongly encrypted for the scenarios. This preliminary development of the proposed design works properly for a maximum of `5 bytes` ASCII character. The `message` with longer character will be considered as future research recommendation. Even when the data length is `822 bytes`, which is around 8 times bigger than the conventional topology, the RPi processor still can overcome the time consumption. Other comparisons will be discussed in the next section.

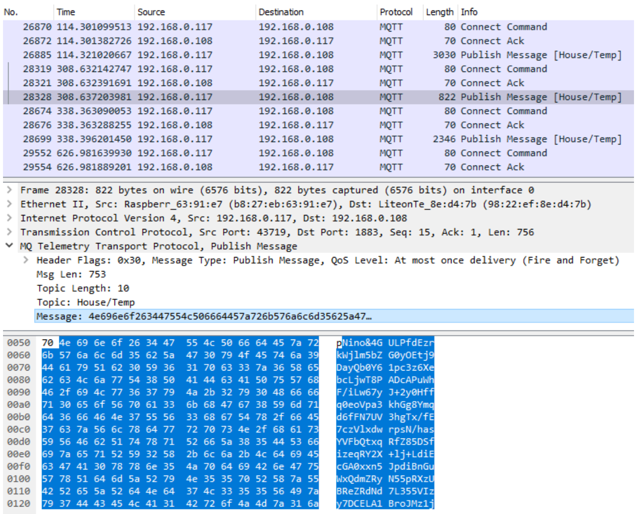

Figure. 4: Analyzing for the secured topology

TABLE II: Key Specification

Encryption	Role	File Name	Size (bytes)
AES-128	Private Key	aes128.key	16
Fernet-128 [20]	Private Key	fernet128.key	16
RSA-2048	Public Key	RSA2048_priv.pem	1678
	Private Key	RSA2048_pub.pem	450
RSA-7680	Public Key	RSA7680_priv.pem	5972
	Private Key	RSA7680_pub.pem	1404
NTRU-401	Public Key	ntru-key.raw	557
	Private Key	ntru-pubkey.raw	607
NTRU-593	Public Key	ntru-key.raw	821
	Private Key	ntru-pubkey.raw	891

C. Discussion of Experiment

The proposed design was verified by evaluating the existing benchmark encryption techniques e.g., AES-128, Fernet-128, RSA-2048, RSA-7680, NTRU-401, and NTRU-593. The detailed key specifications for those encryption algorithms are explained in Table II. The performance evaluation for those encryption algorithms is calculated by using the encryption time (as shown in Table III) and the decryption time (as shown in Table IV). For asymmetric encryption, the results showed that NTRU-401 and NTRU-593 outperform the existing encryption (RSA-2048 and RSA-7680). Contrary to the asymmetric encryption, AES has optimal encryption and decryption time. Considering the post-quantum computing, this research recommends adopting the NTRU encryption for general IoT systems. Even though AES-128 has the smallest encryption and decryption time, NTRU still has an acceptable delay (< `100ms`) i.e., `1.321ms`.

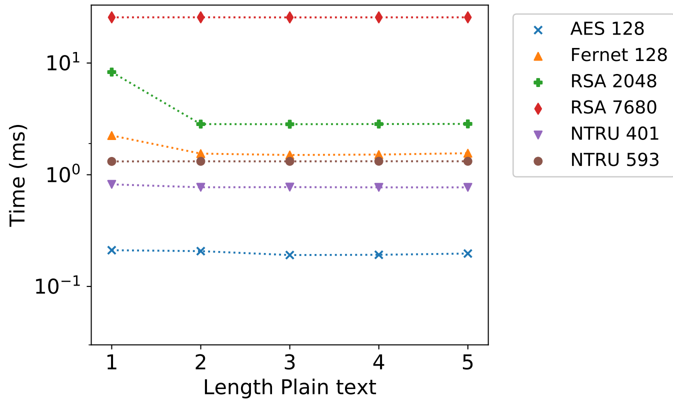

Figure. 5: Encryption Time

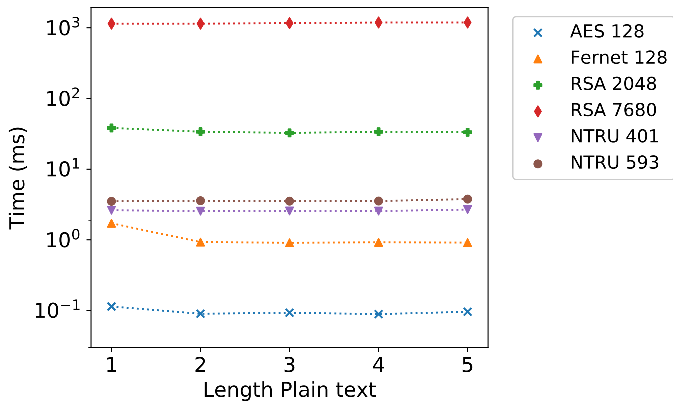

Figure. 6: Decryption Time

V. Conclusions and Further Research

This research was well-implemented and tested for the AES-128, Fernet-128, RSA-2048, RSA-7680, NTRU-401, and NTRU-593 respectively. This research also presents the algorithm of the proposed design i.e., NTRU-IoT encryption for the publisher and NTRU-IoT decryption for the subscriber. The comparisons between the encryption algorithms show that Lattice-based cryptography (i.e., NTRU encryption) can be efficiently implemented for securing an end to end communication in any IoT systems. Further security analysis might need to be considered e.g., different attack scenarios, different positions of adversary during sniffing, different character lengths of the message. Furthermore, future research also needs to magnify the differences between IoT device vendors e.g., NodeMCU, LoRA, APC 220 Radio, Arduino UNO, etc.

TABLE III: Encryption time *(in ms)* between *publisher* and *subscriber*

Plaintext (bytes)	AES-128	Fernet-128 [20]	RSA-2048	RSA-7680	NTRU-401	NTRU-593
1	0.227	2.359	8.271	25.668	0.774	1.317
2	0.189	1.481	2.895	25.673	0.773	1.318
3	0.195	1.477	2.918	25.676	0.771	1.318
4	0.192	1.468	2.853	25.655	0.774	1.321
5	0.203	1.472	2.872	25.675	0.773	1.320

TABLE IV: Decryption time *(in ms)* between *publisher* and *subscriber*

Plaintext (bytes)	AES-128	Fernet-128 [20]	RSA-2048	RSA-7680	NTRU-401	NTRU-593
1	0.111	2.189	38.828	1147.327	2.847	4.264
2	0.086	0.989	34.879	1150.524	3.322	4.614
3	0.087	1.032	32.893	1148.847	2.938	4.488
4	0.088	0.987	32.789	1151.937	3.087	4.554
5	0.162	1.014	33.332	1146.695	2.851	3.714

Conflict of Interest

The authors declare that they have no conflict of interest regarding the publication of this paper.

References

[1] Michael Geller and Pramod Nair. 5g security innovation with cisco. *Whitepaper Cisco Public*, pages 1–29, 2018.

[2] Hamid Nejatollahi, Nikil Dutt, Sandip Ray, Francesco Regazzoni, Indranil Banerjee, and Rosario Cammarota. Post-quantum lattice-based cryptography implementations: A survey. *ACM Comput. Surv.*, 51(6), January 2019.

[3] Rajat Chaudhary, Gagangeet Singh Aujla, Neeraj Kumar, and Sherali Zeadally. Lattice based public key cryptosystem for internet of things environment: Challenges and solutions. *IEEE Internet of Things Journal*, 2018.

[4] Ahmad Boorghany, Siavash Bayat Sarmadi, and Rasool Jalili. On constrained implementation of lattice-based cryptographic primitives and schemes on smart cards. *ACM Trans. Embed. Comput. Syst.*, 14(3), April 2015.

[5] Ruan de Clercq, Sujoy Sinha Roy, Frederik Vercauteren, and Ingrid Verbauwhede. Efficient software implementation of ring-lwe encryption. In *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*, DATE '15, page 339–344, San Jose, CA, USA, 2015. EDA Consortium.

[6] Konstantin Braun, Tim Fritzmann, Georg Maringer, Thomas Schamberger, and Johanna Sepúlveda. Secure and compact full ntru hardware implementation. In *2018 IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, pages 89–94. IEEE, 2018.

[7] Zhe Liu, Hwajeong Seo, Sujoy Sinha Roy, Johann Großschädl, Howon Kim, and Ingrid Verbauwhede. Efficient ring-lwe encryption on 8-bit avr processors. In Tim Güneysu and Helena Handschuh, editors, *Cryptographic Hardware and Embedded Systems – CHES 2015*, pages 663–682, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[8] Nan Li, Dongxi Liu, and Surya Nepal. Lightweight mutual authentication for iot and its applications. *IEEE Transactions on Sustainable Computing*, 2(4):359–370, 2017.

[9] Gaston C Hillar. *MQTT Essentials-A Lightweight IoT Protocol*. Packt Publishing Ltd, Birmingham, UK, 2017.

[10] Thomas Pöppelmann, Tobias Oder, and Tim Güneysu. High-performance ideal lattice-based cryptography on 8-bit atxmega microcontrollers. In Kristin Lauter and Francisco Rodríguez-Henríquez, editors, *Progress in Cryptology – LATINCRYPT 2015*, pages 346–365, Cham, 2015. Springer International Publishing.

[11] Johannes Buchmann, Florian Göpfert, Tim Güneysu, Tobias Oder, and Thomas Pöppelmann. High-performance and lightweight lattice-based public-key encryption. In *Proceedings of the 2nd ACM International Workshop on IoT Privacy, Trust, and Security*, IoTPTS '16, page 2–9, New York, NY, USA, 2016. Association for Computing Machinery.

[12] O. M. Guillen, T. Pöppelmann, J. M. Bermudo Mera, E. F. Bongenaar, G. Sigl, and J. Sepulveda. Towards post-quantum security for iot endpoints with ntru. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2017*, pages 698–703, March 2017.

[13] Dongxi Liu, Nan Li, Jongkil Kim, and Surya Nepal. Compact-lwe: Enabling practically lightweight public key encryption for leveled iot device authentication. *IACR Cryptology ePrint Archive*, 2017:685, 2017.

[14] Z. Liu, R. Azarderakhsh, H. Kim, and H. Seo. Efficient software implementation of ring-lwe encryption on iot processors. *IEEE Transactions on Computers*, pages 1–1, 2017.

[15] Rui Xu, Chi Cheng, Yue Qin, and Tao Jiang. Lighting the way to a smart world: lattice-based cryptography for internet of things. *arXiv preprint arXiv:1805.04880*, 2018.

[16] A. Khalid, S. McCarthy, M. O'Neill, and W. Liu. Lattice-based cryptography for iot in a quantum world: Are we ready? In *2019 IEEE 8th International Workshop on Advances in Sensors and Interfaces (IWASI)*, pages 194–199, June 2019.

[17] P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, Nov 1994.

[18] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. Ntru: A ring-based public key cryptosystem. In Joe P. Buhler, editor, *Algorithmic Number Theory*, pages 267–288, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

[19] Claude E Shannon. Communication theory of secrecy systems. *Bell system technical journal*, 28(4):656–715, 1949.

[20] Y. M. Agus, M. D. Falih, and G. B. Satrya. On the possibilities of cybercrime in iot devices. *International Journal of Recent Technology and Engineering (IJRTE)*, in press.