IMES DISCUSSION PAPER SERIES

Recent Trends on Research and Development of Quantum Computers and Standardization of Post-Quantum Cryptography

Kazutoshi Kan and Masashi Une

Discussion Paper No. 2021-E-5

IMES

INSTITUTE FOR MONETARY AND ECONOMIC STUDIES

BANK OF JAPAN

2-1-1 NIHONBASHI-HONGOKUCHO

CHUO-KU, TOKYO 103-8660

JAPAN

You can download this and other papers at the IMES Web site:

https://www.imes.boj.or.jp

IMES Discussion Paper Series 2021-E-5
June 2021

Recent Trends on Research and Development of Quantum Computers and Standardization of Post-Quantum Cryptography

Kazutoshi Kan* and Masashi Une**

Abstract

The security of widely used public-key cryptographic algorithms (e.g., RSA, elliptic-curve cryptography) is expected to deteriorate once large-scale and fault-tolerant quantum computers are developed. The potential threat is severe because such algorithms underlie the IT infrastructure in modern society, such as in the financial industry. Although the threat is unlikely to materialize in the foreseeable future, the National Institute of Standards and Technology (NIST) has been in the process of standardizing post-quantum cryptography (PQC), which is designed to be secure against quantum computers. NIST has been scrutinizing the security and performance of 15 candidate algorithms (seven finalists and eight alternates). Standardization should accelerate the migration to PQC around the world, not only within the U.S. government. In this paper, we discuss recent trends in the R&D of quantum computers and the security risks of public-key cryptographic algorithms. Then we review NIST's ongoing progress in standardizing PQC and the responses of other organizations in support of the migration. Finally, we discuss future challenges for the real-world implementation of PQC.

Keywords: Elliptic-curve cryptography; Post-quantum cryptography; Public-key cryptographic algorithm; Quantum computer; RSA; Standardization
JEL classification: L86, L96, Z00

* Director, Institute for Monetary and Economic Studies, Bank of Japan
(E-mail: kazutoshi.kan@boj.or.jp)
**Director, Institute for Monetary and Economic Studies, Bank of Japan
(E-mail: masashi.une@boj.or.jp)

This paper was written based on the information as of January 8 2021. The authors would like to thank Masaya Yasuda (Rikkyo University) and Yutaka Tabuchi (RIKEN) for their useful comments. The views expressed in this paper are those of the authors and do not necessarily reflect the official views of the Bank of Japan.

Table of Contents

I. Introduction

In recent years, research and development of quantum computers has been well underway all over the world. If ideal (i.e., large-scale and fault-tolerant) quantum computers are developed, the security of widely used public-key cryptographic algorithms (RSA and the elliptic-curve cryptography) would deteriorate as theoretically predicted. This threat would significantly impact society because the existing algorithms play a central role in cyber security.

To date, several companies (e.g., International Business Machines Corporation: IBM, Google LLC, Rigetti & Co, Inc.) have already made quantum machines available to users. The devices are limited in the number of quantum bits and the operations applied to quantum bits during the coherence time. Thus, at the time of writing, quantum computers have not posed a severe threat to public-key algorithms.

However, there is an unignorable possibility that technological innovations will lead to ideal machines in the future, the timing of which is unpredictable. Furthermore, the migration of cryptographic algorithms may take more than 10 years in practice. Thus, research and development of post-quantum cryptography (PQC, also referred to as quantum-resistant, quantum-safe, or quantum-proof), for security against ideal quantum computers, have been in progress in order to update the existing algorithms.[1]

The National Institute of Standards and Technology (NIST) has been in the process of standardizing PQC (see Section IV) since 2016. After the future migration to PQC algorithms was announced, NIST publicly called for candidate algorithms for new PQC standards in 2016. In the first-round process (Round 1), NIST selected 26 algorithms out of 69 candidates. In Round 2, which lasted until July 2020, NIST selected 15 algorithms (seven finalists and eight alternate candidates). NIST plans to publish the draft of a new PQC standard around 2022–

[1] Quantum key distribution (QKD) is an alternative countermeasure to the threat of quantum computers. By taking advantage of quantum mechanics, QKD enables users to detect eavesdropping third parties over a communication channel. QKD also provides a highly secure protocol in combination with information-theoretically secure cryptography that even an attacker with infinite computing power cannot break. This paper does not detail QKD because the implementations of QKD and PQC in a social infrastructure are fundamentally different. QKD requires the hardware devices on the network to be replaced in order to build a quantum channel that transmits quantum information as well as classical information. Meanwhile, PQC only uses the conventional (classical) channels, so it only requires software renewal.

2024 and then migrate the existing algorithms to those standardized in the IT systems used by the U.S. federal government. Once the standardization is completed, many system vendors are expected to implement standardized PQC algorithms for their products, and entities other than the U.S. government would be required to migrate to PQC rapidly.

The Cryptography Research and Evaluation Committees (CRYPTREC)[2] established by the Japanese government has evaluated the impact of quantum computers on current cryptographic algorithms and considered adoption of PQC in the future (Advisory Board for Cryptographic Technology [2020]). In 2019, CRYPTREC set up a task force to follow the research trends regarding quantum computers and discuss how to deal with PQC (see Section V).

Non-governmental sectors have been actively supporting NIST's standardization of PQC. For example, IETF[3] has been discussing the extension of the TLS 1.3 specification (see Section VI.A), which is the most important cryptographic protocol on the Internet. The extension uses the hybrid key exchange that also utilizes PQC algorithms. In addition, the Open Quantum Safe (OQS) project (see Section VI.B) aims to support the development and prototyping of PQC. The project implements the NIST candidate algorithms and evaluates their performance.

These activities indicate that the migration to PQC will be accelerated from the latter half of the 2020s, regardless of the development of ideal quantum computers. If PQC replaces the existing algorithms in many devices, financial institutions and payment service providers will have to apply PQC to their IT systems for online financial services. Thus, it will be necessary to consider how to prepare for the migration to PQC in the near future

In this paper, we first present an overview of the R&D trends in quantum computers in Section II and discuss the need to migrate to PQC in Section III.

---

[2] CRYPTREC (https://www.cryptrec.go.jp) was established to monitor the security of cryptographic algorithms and maintain the CRYPTREC Ciphers List. CRYPTREC also studies appropriate implementation and operational methodologies of the listed ciphers. CRYPTREC comprises the Advisory Board for Cryptographic Technology and its subcommittees, the Cryptographic Technology Evaluation Committee and the Cryptographic Technology Promotion Committee.

[3] The Internet Engineering Task Force (IETF) is an open standards organization whose main purpose is to develop and promote technical standards for Internet architecture and related operations. IETF publishes standardized technical specifications as a series of Request for Comments (RFCs).

Sections IV and V outline NIST's standardization process and discuss the task force under CRYPTREC, respectively. Section VI describes the activities of IETF and OQS.

II. R&D Trend of Quantum Computers and the Necessity of Early Migration to PQC

The development of quantum computers has garnered much attention in recent years. Specifically, in the context of cryptology, the gate-model quantum computers[4,5] ("quantum computers" hereafter) pose a significant threat to current public-key cryptographic algorithms. For the basic theory behind quantum computing on the basis of the gate-model, see Nielsen and Chuang (2010).

If the ideal quantum computer is ever built, the security of the current public-key cryptography would deteriorate (Takagi [2019]). In this section, we first explain the principles, capabilities, and limitations of quantum computing

---

[4] A gate-model quantum computer is modeled as a quantum circuit, which performs calculations by applying successive basic quantum circuit operations to qubits (a unit of information in the quantum computer; see Section II.A.1 for details) analogously to a classical computer, which is modeled as the classical circuit. The gate-model quantum computer can run algorithms for general purposes and theoretically covers at least all the operations of a classical computer. However, quantum computers are not simply upwardly compatible with classical computers due to unique restrictions imposed by the principles of quantum mechanics. Most importantly, creating an identical copy of an arbitrary unknown state of qubits is impossible (the no-cloning theorem). This results in differences in the software construction methodologies of quantum and classical computers. For the same reason, the computational efficiency of quantum computers does not necessarily exceeds that of classical computers, especially in terms of the number of qubits required for calculation. To date, quantum computers are viewed as an accelerator for computing that has an advantage in limited applications.

[5] Another type of quantum computer is called the quantum annealer, which is dedicated to solving combinatorial optimization problems. The annealer also takes advantage of quantum mechanical phenomena for calculations in common with the gate-model quantum computer. However, its calculation model based on quantum annealing, a metaheuristic procedure for finding the ground state of the Ising model, differs fundamentally from that of the gate-model computer. The commercial use of quantum annealers has been increasing, and their development advances relative to gate-model computers. In 2020, D-Wave Systems Inc. released a commercial machine with 5,640 qubits. In the realm of cryptanalysis, a number of studies have been published on the security evaluation of RSA using D-Wave's machines or simulators that mimic quantum annealing with classical computers. However, at the time of writing, the gate-model computer is the primary threat to the security of cryptography, so we do not discuss the impact of the quantum annealer in this paper.

and describe its threat to public-key cryptography. Next, we present an overview of the recent R&D trends of quantum computers and the background of active investment in this field. Lastly, we discuss the necessity of early migration to PQC.

A. Principles of Quantum Computing

In principle, the power of quantum computing stems from parallel computing taking advantage of three aspects of quantum states: *superposition*, *entanglement*, and *interference*. The computational process manipulates quantum states and obtains desired information by observing the states. A quantum bit (*qubit*) represents the quantum state that corresponds to the basic unit of information in the quantum computers.

1. Parallel computing using qubits

In classical computers, including supercomputers, a (classical) bit represents the basic unit of information and takes one of two states, 0 or 1. Thus, n classical bits represent any one of the 2^n possible states. Such deterministic states are called *classical states*.

In comparison, a qubit can take a state that represents both distinct classical states (i.e., 0 and 1) simultaneously. Such an ambiguous state is called the *quantum state* or the superposition of the classical states.[6] n qubits can represent 2^n distinct classical states (including the entangled states among them, which will be explained later) simultaneously. The quantum state converges to either of the classical states by observation with a certain probability in accordance with the original state.

Theoretically, a quantum computer can perform arbitrary (classical circuit) operations of a classical computer on qubits.[7] For instance, performing classical

---

[6] The quantum state must be distinguished from the mixed state of classical states, which is defined as the statistical ensemble of the classical states. For example, suppose a single qubit is predetermined to be either state 0 or 1, and a person does not know the exact state, only the probability that each state is observed (e.g., flipping a coin). In this case, the qubit takes a mixed state of the two classical states rather than a superposition because it physically contains information on either state 0 or 1, not both of them at the same time. In comparison, the quantum state can hold information for both 0 and 1. This has been verified by thorough physics experiments.

[7] In general, a quantum computer can perform a set of quantum circuit operations which are reversible and represented mathematically as unitary transformations. Any classical (reversible) circuit operations are proved to be constructed by combining the quantum circuit operations, which indicates the quantum computer is universal in the classical sense. More

circuit operations on n superposed qubits is equivalent to parallel computing which processes all 2^n classical states at once.

2. Efficient extraction of information from quantum states

To obtain desired information from qubits, the qubits must be observed. When a quantum state is observed, it converges to a classical state probabilistically, and the observer obtains the corresponding outcome. [8] At this moment, the quantum state loses the information on other classical states and the observer can never retrieve them again.

As this indicates, the amount of information obtained from qubits is limited in principle. For instance, in the case of a quantum state of n qubits in which each state shares the equal observation probability, only one of 2^n possible states can be obtained from a single observation. It should be noted that the amount of information obtained by a single observation from quantum bits equals that from classical bits. Moreover, the observed state is randomly determined and cannot be selected arbitrarily by the observer. Thus, if there is only one state corresponding to a correct answer out of 2^n states and any of the states are observed with equal probability, then the probability of obtaining the correct answer by observation is $1/2^n$. This means that the efficiency of naive quantum computation does not necessarily exceed that of classical computation. Thus, quantum computations using only superposition do not accelerate calculations.

Quantum computing accelerates computation by efficiently extracting information corresponding to the solution of a problem from qubits. Such efficient extraction is enabled by running quantum algorithms, which ingeniously amplify/attenuate the probability of observing the state corresponding to the desired/undesired information for arbitrary input quantum states. In the process of manipulating the observation probabilities, the (quantum) entanglement and the (quantum) interference play essential roles.[9]

---

generally, it is known that the quantum computer is also universal in the quantum sense.

[8] This paper assumes that an observation maps a quantum state onto the *computational basis*, represented by 0 or 1, i.e., an observation of a quantum state must yield a classical state. In general, however, a superposition does not necessarily converge to a classical state depending on the basis of observations. In this section, we describe the standard case of observations on the computational basis.

[9] Quantum entanglement is a physical phenomenon in which a quantum state of each qubit cannot be described independently from the state of others, i.e., the observed states of

The entanglement indicates that the entangled states correlate with each other in terms of their observed outcomes. The interference, which amplifies/attenuates the observation probabilities of multiple quantum states, takes advantage of the wave nature of the quantum states.

B. Capabilities of Quantum Computers and Threat to Cryptography

The capabilities of quantum computers, i.e., their speed and the class of computational problems they can solve, have yet to be fully clarified.[10] The nature of parallelism in quantum computing suggests that quantum computers have an advantage in computational problems where algorithms for finding the solution incorporate the parallel structures in some form. However, algorithms running on classical computers cannot always be parallelized straightforwardly. Thus, it is not entirely clear how quantum computers can speed up computation. To date, studies on *quantum algorithms*, i.e., algorithms running on quantum computers, are largely compilations of research on individual problems.[11] For some problems, quantum algorithms are known to outperform the fastest algorithms on classical computers in terms of computational complexity. For

---

entangled qubits correlate with each other. The Bell states are a simple example of quantum entanglement, showing a perfect correlation between two qubits in such a way that if an observation of one qubit returns 0, then the other qubit always returns 0 even though each outcome is randomly determined. The same holds for when 1 is observed.

The degree of quantum entanglement is measured by *entanglement entropy*, which is maximized in a perfect correlation such as in the Bell state. In the case of multiple entangled qubits, quantum gate operations or observations on part of these qubits immediately affect all other qubits, indicating that all qubits behave in an integrated manner during the calculation processes.

[10] The complexity class bounded-error quantum polynomial time (BQP) represents a set of decision problems that can be efficiently solved by a quantum computer in polynomial time with a probability of more than 2/3. BQP includes the complexity class P which represents a set of decision problems that can be efficiently solved by a classical computer. However, whether BQP includes the complexity class NP-complete has been an open question. NP-complete represents a set of decision problems that is believed to be incapable of solving efficiently by a classical computer. Here, in the computational complexity theory, a decision problem is a problem whose answer is either yes or no.

The decision problem naturally derived from the factoring problem is known to be solvable efficiently by a quantum computer (see Section II.A), so it is included in BQP. However, it is still unknown whether it is included in NP-complete.

[11] The Quantum Algorithm Zoo (https://quantumalgorithmzoo.org) shows a comprehensive list of quantum algorithms that provide speedup over the fastest known classical algorithms. At the time of writing, about 70 algorithms have been listed.

example, Grover's search algorithm (Grover [1996]), which finds a record that satisfies certain criteria within an unsorted database, provides quadratic speedup over its classical counterpart. Simon's algorithm (Simon [1994]) can solve a period-finding problem, which finds the period of a periodic function, exponentially faster than the most advanced classical algorithm.[12]

In the context of cryptology, quantum computers can *unfortunately* break standard public-key cryptography exponentially faster than the most advanced classical algorithm.[13] For concreteness, Shor's algorithm, proposed in 1994, efficiently solves the factoring problem and (elliptic curve) discrete logarithm problem (Shor [1994, 1997]).[14] The computational hardness of these problems guarantees the security of standard public-key cryptography, i.e., RSA and elliptic-curve cryptography. Thus, if an ideal quantum computer that can run Shor's algorithm is developed, the security of the dominant public-key algorithms (e.g., RSA, ECDSA, ECDH) would deteriorate as predicted.[15] This is the major threat that quantum computers pose to public-key cryptosystems.

C. Outlook and Obstacles towards an Ideal Quantum Computer

At the moment, it is unlikely that an *ideal* quantum computer capable of breaking public-key cryptography will appear in the foreseeable future. *Ideal* means large-scale and fault-tolerant; an ideal machine is equipped with a large number of qubits and conducts *quantum error correction*, which protects quantum

---

[12] Quantum algorithms for solving practical problems frequently incorporate the versatile parts of well-known algorithms as subroutines. Typical examples include the *phase estimation algorithm*, a part of Shor's algorithm, frequently applied in quantum chemistry. The *amplitude amplification algorithm*, a part of Grover's algorithm, and the Harrow-Hassidim-Lloyd (HHL) algorithm are applied in quantum machine learning. Quantum algorithms can be used to solve a number of problems faster than the most advanced classical algorithms, but few algorithms provide exponential speedup.

[13] The threat of quantum computers to symmetric-key cryptography seem to be moderate compared with that of public-key cryptography. However, caution is still recommended for certain aspects. For details, see Appendix.

[14] Shor's algorithm solves the factoring problem in polynomial time (k^3), where k is the number of bits in the binary representation of the input integer, providing exponential speedup. In comparison, the general number field sieve (GNFS), the fastest known classical algorithm, works in sub-exponential time in the size of the input integer.

[15] At the time of writing, there have been no cryptanalytic reports implementing Shor's algorithm that are applicable to an arbitrary input integer. Existing studies have constructed quantum circuits that utilize prior knowledge of prime numbers that correspond to the solutions (Cryptographic Technology Research Working Group [2019]).

information of qubits from errors during computation and noises from the environment.[16,17,18]

In contrast, the machines in practical use at the time of writing are not capable of error correction. These machines cannot obtain trustworthy output because the accuracy of the calculation decreases as the number of operations increases due to noises. This shortcoming is fatal in cryptanalysis, which requires a number of computational steps, for example, when running Shor's algorithm. In addition, the current machines handle at most 100 qubits, far fewer than the number of qubits required for practical cryptanalysis. Furthermore, the current machines cannot maintain the states of qubits including their quantum entanglement for a sufficient duration. For example, a qubit implemented by a superconducting circuit can maintain a state for about 100 microseconds (at the time of writing). This duration is not sufficient for cryptanalysis, which requires several hours.

In regards to the impact of quantum computing on cryptanalysis, the task force of CRYPTREC stated that "the magnitude of noises (generated during the

---

[16] The source of noises depends on the physical implementation of the devices. Superconducting qubits, which consist of superconducting electronic circuits, are faced with environmental noises such as thermal or electromagnetic radiation from the device, as well as cosmic rays (Vepsäläinen *et al.* [2020]), which do not have a significant impact. The qubits can also make errors due to the limited precision of quantum gate operations.

[17] Error correction in classical computers makes use of redundancy to protect one bit using multiple bits. For example, suppose that the information of a single bit indicating "0" is represented by three bits as "000." Then, even if the middle bit is erroneously flipped and becomes "010," it can be corrected to "000" by observing the remaining two bits. This classical error correction implicitly assumes that the state of an arbitrary bit can be read directly and duplicated.

On the contrary, this assumption does not hold in quantum computing, indicating that classical error correction cannot be straightforwardly applied to qubits. The qubits are destroyed by observation, so the errors in qubits cannot be identified by direct observation. In addition, the unknown state of qubits cannot be duplicated according to the principles of quantum mechanics (the no-cloning theorem). However, given these limitations, quantum error correction has been proved feasible theoretically (see footnote 23).

[18] To break modern cryptography, a quantum computer should be fault-tolerant, contain millions of qubits, and be capable of performing a series of calculations lasting from a few hours to tens of hours (National Academies of Sciences, Engineering, and Medicine [2019]). In particular, 4,098 logical qubits are required to break RSA with a 2,048-bit public key, indicating that 8 million physical qubits are required. This calculation takes about 30 hours. These estimates could vary significantly depending on the assumption of quantum error correction: A single logical qubit is assumed to be protected by thousands of physical qubits. Quantum gate operations are assumed to run at a frequency of 5 MHz.

calculation in quantum computers) is not at a level such that the machine can be used for factoring in cryptanalysis" and "the timing of the advent of an ideal machine capable of cryptanalysis is not clear" (Advisory Board for Cryptography Technology [2019]).

The main challenge of developing an ideal quantum computer is maintaining large-scale quantum entanglement for an extended duration, which would eliminate the impact of extremely small noises. Once this is achieved and quantum error correction can be implemented, the number of qubits will increase with the prolonged coherence time (the duration in which entangled qubits hold the information of their state).[19] However, it does not seem possible at this time, so some researchers argue that the ideal quantum computer will remain theoretical forever.[20]

Even if an ideal quantum computer emerges in the future, it will not be implemented as a straightforward extension of current mainstream technology. To develop an ideal machine, a number of innovations will be needed to significantly improve robustness to noises. Such technology would be vastly different in principle from the current one.

D. Intensive Investment in Quantum Computing and its Background

Although the technological innovations described in Section II.C are unforeseen, the possibility that they would occur cannot be ignored because investment in the R&D of quantum computing has been increasing in recent years. In addition to the current mainstream machines based on superconducting quantum circuits, many other devices have been developed on the basis of different principles.[21]

---

[19] It is considered difficult to expand the scale of a quantum computer with a superconducting circuit to 1,000 qubits or more due to hardware constraints such as the complexity of the wiring to control individual qubits (National Academies of Sciences, Engineering, and Medicine [2019]). To date, no scaling laws, such as *Dennard scaling* for classical computers, have been established for quantum computers (Tabuchi [2020]).

[20] Hirota (2020) discussed the scope of the threshold theorem (see footnote 23) underlying the feasibility of the quantum error correction. The paper pointed out that the current quantum error correction does not work if the noises increase nonlinearly with the number of entangled qubits, i.e., the threshold theorem does not hold for non-linear noises. This implies that the nature of the noise should be investigated further for the scalability of quantum computers.

[21] For example, there are quantum computers based on photons (XANADU, PsiQuantum, Corp.), trapped ions (IonQ, Inc.), nitrogen-vacancy center in diamond, semiconductor quantum dots (Intel Corporation), solid-state nuclear magnetic resonance (NMR; Kitagawa laboratory at Osaka University), and topological quantum computing (Microsoft Corporation). The

Ongoing research on hardware and software covers vast fields ranging from the most fundamental hardware principles to industrial applications. The ecosystem formed by researchers and developers supports the research (National Academies of Sciences, Engineering, and Medicine [2019]).[22]

The recent increase in investment is motivated by a number of factors. First, an error correction methodology for quantum computing has been theoretically developed. The error correction was necessary for computing, but its quantum version was considered impossible during the early stage of development.[23] Second, *quantum supremacy* was achieved empirically in 2019, according to Arute *et al.* (2019). Quantum supremacy refers to the phenomena in which a quantum computer can solve a problem that cannot be solved by any classical computer, including a supercomputer, within a feasible time frame. The experimental achievement of quantum speedup is considered empirically proven.[24] Third, the Noisy Intermediate-Scale Quantum computer (NISQ) is expected to have promising industrial applications in chemistry, finance, and machine learning. NISQ cannot perform error correction for qubits, but it can be used in combination with classical computers by performing hybrid algorithms. Recent

superconducting circuit quantum computers are developed by many companies (Alibaba, D-Wave Systems Inc., Google LLC, IBM, Intel Corporation, and Rigetti & Co, Inc., etc.) and universities. Those approaches have different advantages and disadvantages. Thus, each approach can be promising as the future standard for different reasons.

[22] The Center for Research and Development Strategy (CRDS, an affiliated institution of Japan Science and Technology Agency), "[Research Trends in the Field of Quantum Technology] *Ryoushi gijyutsu bunya no kenkyu doukou ni tsuite* (in Japanese)," Material for the First Expert Meeting on Quantum Innovation Strategy, March 29, 2019 (https://www.kantei.go.jp/jp/singi/ryoshigijutsu_innovation/dai1/siryou3.pdf).

[23] Shor's code (Shor [1995]) and the stabilizer code (Gottesman [1997]) correct erroneous qubits without directly reading out their information. Furthermore, the threshold theorem states that the error correction can reduce the effect of errors to an arbitrary small level if the possibility of an error in a single operation falls below a certain threshold under the assumption of the noise property. Thus, less noisy operation leads to a fault-tolerant quantum computer. Relating to the assumption, the nature of noise deserves examining (see also footnote 20). Topological quantum computing using knot theory in algebra (mentioned in footnote 21) is also expected to be one of the approaches that lead to the fault-tolerant machine (https://cloudblogs.microsoft.com/quantum/2018/09/06/developing-a-topological-qubit/).

[24] In December 2020, quantum supremacy was also demonstrated using photons for the first time by a Chinese research team (Zhong *et al.* [2020]). The machine was equipped with 76 photons (76 qubits). The methodology was dedicated to the boson sampling, hard computation chosen to demonstrate quantum supremacy. This research is meaningful in demonstrating that quantum computers other than superconducting ones are promising for creating large-scale quantum entanglements.

developments of quantum computers have focused on NISQ. Fourth, users can easily access quantum computers via cloud services over the Internet, thus lowering the barrier of quantum computing. Fifth, quantum computers are energy-efficient.[25] Energy conservation has become increasingly important nowadays because training large-scale machine learning models consumes an enormous amount of energy. There are concerns that energy constraints would suppress the future development of machine learning (Thompson *et al*. [2020]). Quantum computers have the potential to bypass the constraint.

Considering the recent R&D trends focusing on NISQ, quantum computers would pose little threat to modern cryptography in the foreseeable future. However, the tail risk of innovations leading to an ideal quantum computer should not be ignored while the virtuous cycle of active R&D and its fruits continues to attract short-to-medium-term investments.

E. Necessity of Early Migration to PQC

Considering the above discussion , we argue that early migration to PQC should be prepared for the following four reasons.

First, cryptography migration is expected to take 10 years or more. The migration requires a large-scale system renewal, hardware and software replacement, and the involvement of various stakeholders in financial industry (Ito, Une, and Seito [2019]). The IT infrastructure for national defense could take 20 to 30 years to migrate (National Security Agency [2016]). If preparations for migration begin after innovations leading to ideal quantum computers have already been made, the migration may not be completed before the ideal machines are created. Thus, the migration should be started while there is still sufficient time.

Second, the migration must be started early in order to mitigate the threat of a *harvest attack*, which collects and stores encrypted data over a public communication channel and attempts to recover them after the attacker's

[25] According to the theory of thermodynamics and computer science, heat (the energy loss) occurs when the computer erases information (i.e., the entropy increases; Landauer's principle). Thus, the theoretical lower bound on the energy consumption in the reversible calculation, in which input can be reverse-calculated from the output, is zero because any information is not lost through the calculation. Now, except for the observation on qubits, all the quantum operations consist of reversible operations (i.e., unitary transformations), so the quantum computers can ideally compute without energy consumption except for the energy for running the device. In contrast, the classical computers' operations are mostly irreversible. Thus, the calculation always consumes energy in principle.

computational power has increased in the future. Encrypted data must be expired earlier than the advent of ideal quantum computers. For example, if data confidentiality must be retained for 10 years and ideal machines emerge 30 years later, the migration to PQC must be completed within 20 years (10 years before the advent of ideal machines).

Third, as discussed in Section II.D, the tail risk of the advent of ideal machines cannot be ignored considering the remarkable technological advances and increased R&D in quantum computing. In 2014, quantum computers equipped only five qubits. In September 2019, 53-qubit machines emerged and the quantum supremacy was considered to have been achieved. The recent machines equip about 70 qubits.[26] In such a way, the technological progress has been remarkable in the past five years supported by increased R&D investments in the quantum field.

Fourth, it takes time to fine-tune the implementation of cryptography and foster trustworthiness of the new technology such that it can be integrated into the social infrastructure. As described in Section III, the security of public-key cryptography cannot be proved purely theoretically. Moreover, a theoretical evaluation of the security is not sufficient for verifying the resistance to *side-channel attacks*[27] which depend on the implementation. Performance evaluations should also be required for limited computational resources such as IoT (Internet-of-Things) devices. Thus, we recommend introducing PQC into new IT services and systems first in order to identify and address issues found during practical use. Then PQC should be gradually disseminated as its trustworthiness as social infrastructure is fostered.

III. Overview of Post-Quantum Cryptography and Its Security Assessment

Various PQC algorithms have been proposed and are in the process of being evaluated for their security and performance. However, the history of such evaluation is relatively short, so the confidence in PQC algorithms is not on par

[26] At the time of writing, IBM has developed a 65-qubit machine (https://www.ibm.com/blogs/research/2020/09/ibm-quantum-roadmap/), and Google LLC has developed a 72-qubit one (https://ai.googleblog.com/2018/03/a-preview-of-bristlecone-googles-new.html). More recently, the development has focused on the practical usefulness in applications rather than on quantitative expansion of the number of qubits.

[27] The side-channel attack attempts to gain secret information from the behaviors of hardware devices that perform encryption and decryption, such as power consumption and electromagnetic leak. This unanticipated information flow is called *side channel*.

with that of the existing algorithms. In this section, we describe the basic methodology for evaluating the security of cryptographic algorithms and introduce PQC algorithms in accordance with underlying mathematical problems.[28]

A. Evaluation of Cryptographic Algorithms Based on Computational Security

The security of PQC and existing public-key cryptographic algorithms is evaluated on the basis of *computational security.* [29] An algorithm is considered computationally secure if an attacker with limited computational power cannot break it.

Computational security is classified into OW-CPA, IND-CPA, and IND-CCA, in the order of increasing security levels. OW-CPA (the one-wayness against chosen-plaintext attacks) represents a security level at which an attacker cannot recover a *full* plaintext corresponding to the target ciphertext, under the condition that *ciphertexts* corresponding to plaintexts of the attacker's choice are available. This security level is relatively weak in a sense that it enables the attacker to partially recover the plaintext. IND-CPA (the indistinguishability against chosen-plaintext attacks) represents a higher level of security at which an attacker cannot recover the plaintext *even partially* under the same conditions as OW-CPA. IND-CCA represents the security level at which an attacker cannot recover the plaintext even partially, under the condition that *plaintexts* corresponding to ciphertexts of the attacker's choice are available except for the target plaintext. IND-CCA is the highest security level for public-key algorithms in the context of computational security.[30] Recent public-key algorithms are required to satisfy

---

[28] For more details of PQC algorithms, see Shikata (2019), Nuida (2020), and Cryptographic Technology Research Working Group (2019).

[29] The security of cryptographic algorithms can be evaluated on the basis of *information-theoretic security* and *computational security*. Information-theoretic security refers to unconditional security, i.e., even an attacker who has *infinite* computational power cannot break such algorithms. However, it is mostly impractical because it requires a sender and a receiver to share a secret key (random number) with the same size as the message to be encrypted. Although computational security weakens the assumption of the attacker's computational power, it is relatively practical compared with information-theoretic security.

[30] Strictly speaking, IND-CCA is classified into IND-CCA1 and IND-CCA2 reflecting a slight difference in the attacker's ability. IND-CCA2 enables an attacker to conduct an adaptive chosen-ciphertext attack, i.e., an attacker can recover any ciphertext until it receives the target ciphertext. In contrast, IND-CCA1 does not allow this.

The security in terms of indistinguishability is formally presented as a game: (1) The

IND-CCA at the minimum. The Fujisaki-Okamoto transformation[31] (Fujisaki and Okamoto [1999]) converts an arbitrary algorithm with OW-CPA to one with IND-CCA.

The security of a cryptographic algorithm is verified in the following two steps. The first step is to prove the security level (e.g., IND-CCA) theoretically under the assumption that the underlying mathematical problem is computationally intractable. Here, such problems include the factoring problem and the discrete logarithm problem (DLP) on elliptic curves for commonly used algorithms, and the shortest vector problem (SVP), error-correcting code problem, and multivariate polynomial equations problem for PQC algorithms. Up until recently, attackers were assumed to use classical computers. Nowadays, they are assumed to use quantum computers in addition to classical ones in order to prove the algorithms' security against quantum computers. The theoretical proof of this step reduces the computational hardness of solving the mathematical problems.

The second step is to evaluate the computational hardness of the aforementioned mathematical problem by means of computer experiments. This empirical approach has been widely approved because proving the hardness theoretically is considerably difficult. Cryptanalysis competitions[32] are held among researchers to facilitate more rigorous evaluation by "making researchers work as hard as possible." The problems that remain unsolved by these researchers are believed to be unsolvable within a realistic time frame. An empirical evaluation is also important for determining security parameters such as the size of a public key.

---

attacker chooses two plaintexts $\{m_0, m_1\}$. (2) The challenger encrypts them into corresponding ciphertexts $\{c_0, c_1\}$ and returns one of them $c_* \in \{c_0, c_1\}$ to the attacker. (3) If the attacker cannot distinguish between $c_* = c_0$ and $c_* = c_1$, the algorithm is considered secure. In step (3), IND-CCA1 allows the attacker to decrypt arbitrary ciphertexts other than $\{c_0, c_1\}$. IND-CCA2 also allows the attacker to adaptively decrypt a ciphertext other than the correct one (e.g., c_1, if $c_* = c_0$).

[31] The Fujisaki-Okamoto transformation constructs a stronger public-key algorithm (IND-CCA-secure) from an original (not IND-CCA-secure) by combining it with a secure hash function and symmetric-key algorithm.

[32] One widely known competition for factoring problems is the RSA Factoring Challenge (held by RSA Security LLC, 1991-2007). The Lattice Challenge is a competition for the shortest vector problem (https://www.latticechallenge.org/) held by The Technical University of Darmstadt starting in 2008. The Fukuoka MQ Challenge is one for the multivariate polynomial equations problem organized by several institutions including Kyushu University starting in 2015 (https://www.mqchallenge.org/).

B. Type of PQC

PQC can be classified into (1) lattice-based cryptography, (2) code-based cryptography, (3) multivariate cryptography, (4) isogeny-based cryptography[33], or (5) hash-based cryptography, depending on the underlying mathematical problems. This section presents an overview of types (1) to (3), which cover the finalists that passed Round 2 of NIST's standardization.

1. Lattice-based cryptography

Lattice-based cryptography is a general term for public-key cryptography that uses a *lattice*, a regularly spaced array of points, in some form. Most lattice-based cryptographic algorithms are based on the SVP; that is, the problem of finding the point closest to the origin among a given set of points (lattice). This problem is considered hard to solve even for quantum computers in a large dimensional space. A number of methods have been proposed for solving the SVP,[34] but at the time of writing, none are particularly efficient in polynomial time. However, a number of harder problems were solved at the Lattice Challenge (see footnote 32). Thus, the computational hardness of the SVP has not yet been firmly evaluated.

Lattice-based cryptography also utilizes other problems related to lattices other than the SVP. Typical examples include the learning with errors (LWE) problem [35] (Regev [2004]) and the NTRU problem (Hoffstein, Pipher and

---

[33] Isogeny-based cryptography is based on the computational hardness of the path-finding problem in a supersingular isogeny graph in which each node and edge represent a supersingular elliptic curve and an isogeny mapping, respectively. Jao and De Feo (2011) proposed the Supersingular Isogeny Diffie-Hellman key exchange (SIDH) on the basis of the Diffie-Hellman key exchange. The Supersingular Isogeny Key Encapsulation (SIKE), which passed Round 2 as an alternative candidate, is based on the key exchange construction referred to in the SIDH (see Table 1 in Section IV).

[34] Algorithms for solving SVP efficiently include lattice basis reduction, lattice enumeration, and lattice sieving. Lattice basis reduction transforms a given lattice basis into a more optimal one consisting of shorter, nearly orthogonal vectors. The enumeration algorithm systematically enumerates all of the lattice points in a bounded region of the space. The sieving algorithm generates shorter lattice points from the list of points with heuristics and updates the list with shorter lattice points iteratively.

[35] The LWE problem is the computational problem of revealing secret vector $\boldsymbol{s}$ from the simultaneous equations $b_i = \langle \boldsymbol{s}, a_i \rangle + e_i$ defined over the integer residue ring $\mathrm{Z}/n\mathrm{Z}$. $\langle \cdot, \cdot \rangle$ denotes the inner product of the vectors. The LWE problem is used in lattice-based cryptography as follows: In encryption, for each bit k in plaintext m, choose subset S from

Silverman [1998]). The LWE problem is a computational problem of inferring the solution for a linear system of equations with errors for discrete variables. The computational hardness of these problems is reduced to that of the SVP. The Ring-LWE problem, the Module-LWE problem, and the Module-LWR (learning with rounding) are generalizations of the LWE problem.[36] These are also used in lattice-based cryptography.

2. Code-based cryptography

Code-based cryptography is based on the error correcting code problem, which originates from the error correction of a message transmitted through a noisy communication channel. The error correction enables the receiver to remove noises added to the original message and restore it uniquely. Code-based cryptography has been researched for about 40 years since McEliece introduced the application of the error correcting code to public-key cryptography (McEliece [1978]). The McEliece algorithm is designed to achieve OW-CPA but not IND-CPA. However, an appropriate transformation can convert an OW-CPA algorithm to an IND-CCA one. In addition, Classic McEliece and the Niederreiter algorithm (Niederreiter [1986]) have been proposed as PQC algorithms.

3. Multivariate polynomial cryptography

Multivariate polynomial cryptography uses the problem of solving multivariate quadratic polynomial equations in which variables only take discrete values (MQ problem).[37] The MQ problem can be solved by dividing the multivariate polynomials by each other and simplifying them (i.e., computing the Gröbner basis). However, such a method is known to be impractical due to its computational complexity. As of now, research on finding the Gröbner basis

---

$\{1, 2, \ldots, m\}$ randomly. Then encrypt k as $(\sum_{i\in S} a_i, \sum_{i\in S} b_i)$ and $(\sum_{i\in S} a_i, \lfloor q/2 \rfloor + \sum_{i\in S} b_i)$ if $k = 0$ and 1, respectively. In decryption, calculate $\boldsymbol{b} - \langle \boldsymbol{s}, \boldsymbol{a} \rangle$ for ciphertext $(\boldsymbol{a}, \boldsymbol{b})$ and output 0 if each value is closer to 0 than $\lfloor q/2 \rfloor$, and 1 otherwise.

[36] For more details about the relationship between the problems, see Peikert and Pepin (2019) and Shikata (2019).

[37] The problem of solving the simultaneous equation system $F(x) = a$ of randomly chosen multivariate polynomials is NP-hard. In cryptography, the easy-to-solve simultaneous equation system $F(x) = a$ is chosen as a secret key. Its randomized version $T \circ F \circ S(x) = P(x) = b$ with random affine transformations S and T is used as a public key. The published system is expected to be computationally hard. By exploiting this mechanism in cryptography, for given plaintext m, a sender encrypts it as ciphertext $\mathrm{c} \leftarrow P(m)$ and a receiver decrypts it as $m \leftarrow S^{-1} \circ F^{-1} \circ T^{-1}(\mathrm{c})$.

efficiently is almost complete.[38] However, we should not ignore the possibility of a more efficient method because the Gröbner basis is of much interest in basic science fields. In fact, a number of solutions to harder problems have been found in the Fukuoka MQ Challenge (see footnote 32).

IV. NIST PQC Standardization Process

NIST announced the conclusion of Round 2 of the PQC standardization process in July 2020. They narrowed down 26 candidates to 15 as objectives to be evaluated in the third round (Round 3). In this section, we give an overview of the history of the standardization process and the results of the technical review conducted in Round 2.

A. History of the Standardization Process

NIST approved RSA and ECDSA as the main public-key and digital signature algorithms respectively. However, given the recent evolution of quantum computers and the associated cryptanalytic risk, NIST concluded that an entirely new quantum resistant cryptography (i.e., PQC) needed to replace the current one. In December 2016, NIST announced the call for PQC algorithms (National Institute of Standards and Technology [2016a]) and a plan to migrate to new algorithms after establishing a new standard.

NIST and the National Security Agency projected that it could take as long as 20–30 years to complete the migration to the new algorithms due to the large scale and long lifecycles of government IT systems. There is also a risk that the advent of quantum computers will occur sooner than predicted. Thus, NIST intends to proceed with the migration as soon as possible.

NIST solicited proposals for candidate algorithms and established submission requirements and evaluation criteria. The call for proposals closed at the end of November 2017 (National Institute of Standards and Technology [2016b]). The types of algorithms to be standardized include public-key encryption, key-encapsulation mechanism (KEM)[39], and digital signature

---

[38] The F_5 algorithm (Faugère [2002]) is known to be efficient for computing the Gröbner basis. The computational complexity is evaluated with an invariant called the degree of regularity, which is determined from a set of multivariate polynomials as input (Bardet, Faugère, and Salvy [2004]).

[39] KEM is a mechanism for sharing a random bit string (a secret key). KEM enables high-speed and high-capacity communication on the basis of symmetric-key cryptography. A

algorithms. In Round 1, which lasted until January 2019, NIST selected 26 algorithms out of 69 candidates which met the submission requirements (National Institute of Standards and Technology [2019]). In Round 2, which lasted until July 2020, NIST selected 15 candidate algorithms (seven finalists and eight alternate candidates; see Table 1. National Institute of Standards and Technology [2020])[40].

NIST will continue the review process in Round 3 in 2020–2021 and publish a draft of the standards around 2022–2024. However, a number of researchers have expressed concerns that more time is needed to thoroughly evaluate the candidate algorithms.[41] In the second round status report, NIST stated that PQC diversity is desirable in order to avoid the risk of a single innovation breaking all of the standardized algorithms. Considering the differences in the stages of evaluation, NIST also expressed its intention to implement the candidate algorithms that are ready for standardization earlier. The seven finalists are the most promising algorithms and will most likely be ready for standardization at the end of Round 3. The eight alternative candidates are potential ones for future standardization. NIST draws an attention on how it will standardize the candidate algorithms while balancing security and diversity.

cryptographic primitive of KEM is the same as that of public-key encryption. While public-key encryption is constructed to encrypt an arbitrary type of data, KEM is designed to encrypt only the secret key of a symmetric-key algorithm.

40 In Round 1, NIST mainly evaluated the candidates in terms of security. In Round 2, NIST put more focus on performance (e.g., processing time for encryption and decryption). In Round 3, the final stage toward standardization, NIST plans to evaluate the security and performance in the implementation environment, e.g., resistance against side-channel attacks.

41 NIST held the second PQC standardization conference in August 2019 to discuss how to evaluate and select 26 second round candidates. At the conference, several researchers stated that a great deal of time would be required to evaluate the candidate algorithms exhaustively and thoroughly, and that NIST should refine the evaluation criteria in order to conduct fair evaluations in parallel.

Table 1. Candidate Algorithms for PQC Standardization

(Finalists: 7)

	Name of algorithm	Underlying mathematical problem	Institution which submitters joined
Public-key encryption/KEM	CRYSTALS-KYBER	Lattice problems (e.g., SVP)	Radboud University
	NTRU		University of Waterloo
	SABER		KU Leuven, etc.
	Classic McEliece	Error correcting code problem	Eindhoven University of Technology, etc.
Digital signature	CRYSTALS-DILITHIUM	Lattice problem (e.g., SVP)	IBM Research
	FALCON		Thales Communications & Security, etc.
	Rainbow	MQ problem	University of Cincinnati

(Alternate candidates: 8)

	Name of algorithm	Underlying mathematical problem	Institution which submitters joined
Public-key encryption/KEM	FrodoKEM	Lattice problem (e.g., SVP)	Microsoft Research
	NTRU Prime		Technische Universiteit Eindhoven, etc.
	BIKE	Error correcting code problem	Intel, etc.
	HQC		University of Limoges, etc.
	SIKE	Isogeny graph path-finding problem	University of Waterloo
Digital signature	GeMSS	MQ problem	Sorbonne University, etc.
	Picnic	Symmetric key decryption problem	Microsoft Research
	SPHINCS+	Hash function collision search problem	Eindhoven University of Technology

B. Evaluation Summary of Second-Round Candidate Algorithms

Out of the seven finalists, five are lattice-based algorithms. Three of them are algorithms for public-key encryption/KEM (CRYSTALS-KYBER, NTRU, and SABER), and the others are digital signature algorithms (CRYSTALS-DILITHIUM and FALCON). These lattice-based algorithms seem to be the most

promising due to their excellent performance in cryptographic processing, such as in encryption and decryption, their small public keys, and theoretical security proofs. Furthermore, lattice-based algorithms for encryption are important for constructing a fully homomorphic encryption scheme (FHE), which supports arbitrary computation on ciphertexts.

According to the status report, NIST will consolidate the candidates with similar methodologies into a single one, regardless of their performance, in order to ensure diversity. Thus, out of CRYSTALS-KYBER, NTRU, and SABER, only one will be selected as the public-key encryption/KEM standard. Either CRYSTALS-DILITHIUM or FALCON will be selected as the digital signature standard.

NIST's evaluations of each finalist described in the second round status report are provided below (National Institute of Standards and Technology [2020]). The candidates were evaluated for theoretical security proofs, performance (encryption/decryption speed, key size), methodological simplicity, and intellectual property status.

1. Public-key encryption/KEM

a. CRYSTALS-KYBER (Lattice-based cryptography)

CRYSTALS-KYBER is a KEM algorithm based on the Module-LWE problem. The IND-CCA2 security is proved with the Fujisaki-Okamoto transformation in the quantum random oracle model (QROM).[42] The history of the Module-LWE problem is relatively short. No specific attacks are known to be more efficient than those applicable to the plain LWE problem. CRYSTALS-KYBER performs effectively for most applications and provides relatively straightforward adjustment of the performance/security trade-off by varying security parameters. In addition, the algorithm shares mathematical principles with CRYSTALS-DILITHIUM, which is also a finalist. Thus, CYSTALS-KYBER is one of the most promising KEM algorithms (See footnote 39).

b. NTRU (Lattice-based cryptography)

NTRU is a KEM algorithm based on the NTRU problems, which makes use of a

[42] In the random oracle model, a cryptographic hash function is modeled as an ideal *oracle* whose output is determined by a uniformly random function. In classical computing, the input and output of the hash function can take classical states, i.e., series of $\{0, 1\}$ bits. The quantum random oracle model extends it to quantum computing, enabling the superposition states to be input and output.

different security assumption (i.e., the NTRU assumption) from the Ring-LWE or Module-LWE problem. IND-CCA security is proved in QROM. Although NTRU performs sufficiently, it is not the most effective among the lattice-based algorithms. In particular, NTRU's key generation is slower than that of the algorithms based on the Ring-LWE and Module-LWE problems. While the performance gap is small between NTRU and CRYSTALS-KYBER and SABER, NTRU provides diversity to the collection of finalists due to the security assumption. Because of its longer history, NTRU has a lower risk of unexpected intellectual property claims. This is one reason that NTRU was selected as a second round finalist. NTRU may become more advantageous when new concerns regarding security or intellectual property arise in CRYSTALS-KYBER and SABER.

c. SABER (Lattice-based cryptography)

SABER is a KEM algorithm based on the Module-LWR problem. While IND-CCA is proven with the Fujisaki-Okamoto transformation, there is a mild concern that the theoretical proof is not complete. Namely, the reduction from the Module-LWE to Module-LWR problem is not concretely applicable to SABER. Regarding performance, the adoption of power-of-2 moduli in SABER enables efficient optimization of the rounding operation, the modular reduction, and polynomial multiplication. SABER is highly effective overall and is expected to be suitable for general-purpose applications. In Round 3, SABER's security against side-channel attacks will be closely evaluated. SABER is expected to be one of the most promising KEM schemes.

d. Classic McEliece (Code-based cryptography)

Classic McEliece is a KEM algorithm based on the McEliece algorithm (McEliece [1978]) built from a random binary Goppa code. IND-CCA2 security is proven in QROM with the Fujisaki-Okamoto transformation. This algorithm builds on the remarkable stability of the original McEliece algorithm studied for 40 years. Classic McEliece has a somewhat unusual performance profile; it has a very large public key but the smallest ciphertext of all competing KEM algorithms. This indicates that Classic McEliece is not suitable for general use in Internet protocols but could be appropriate for certain applications due to its very small ciphertext size. Overall, Classic McEliece's reliability can be attributed to its long history of research.

2. Digital signatures

a. CRYSTALS-DILITHIUM (Lattice-based cryptography)

CRYSTALS-DILITHIUM is a signature algorithm based on the Module-LWE problem. This algorithm shares mathematical principles with CRYSTALS-KYBER. Its key and signature sizes are small, and its key generation, signing, and verification operations are highly efficient. The algorithm has also shown to be effective in real-world experiments. The methodology of this algorithm is relatively simple compared with its main competitor FALCON. This is because CRYTSTALS-DILITHIUM uses the same modulus and ring for all security parameters. NIST expects that either CRYSTALS-DILITHIUM or FALCON will be standardized as the primary post-quantum signature scheme.

b. FALCON (Lattice-based cryptography)

FALCON is a signature algorithm based on the NTRU problem. The security is proven in both ROM and QROM. The algorithm performs effectively overall. Specifically, it offers the smallest public key and signature out of all of the second-round signature algorithms. Signing and verifying are also performed efficiently, although key generation is slower. The methodology is more complex to implement than CRYSTALS-DILITHIUM because FALCON requires tree data structures, extensive floating-point operations, and random sampling from several discrete Gaussian distributions.

c. Rainbow (Multivariate polynomial cryptography)

Rainbow is a multivariate polynomial signature algorithm with a layered construction based on the Unbalanced Oil-Vinegar (UOV) signature algorithm (Kipnis, Patarin and Goubin [1999]). The security is based on the MQ problem. Signing and verifying is fast, and the signature is very short. However, the public key is very large, so Rainbow is not suitable for general use. Nevertheless, the selection of Rainbow contributes to the diversity of the finalist signature algorithms. Rainbow is relatively suitable for applications in which frequent key distribution does not occur. In the status report, NIST noted a gap between actual performance and theoretical complexity, leading to the need for more detailed analysis.[43]

[43] Recently, an attack method has been proposed against Rainbow and GeMSS (listed in the alternative candidates). It has been raising concerns about the security of signature algorithms based on multivariate polynomial cryptography. In response to this, NIST has been discussing the future impact and responses at its forum. Researchers are discussing the idea of removing

V. Discussion for PQC in CRYPTREC

In 2019, a CRYPTREC task force discussed several potential issues relevant to the recent evolution of quantum computers, such as the impact on the security of CRYPTREC ciphers.[44]

A. Views on Migration to PQC

The task force noted the importance of preparing for the migration to PQC. According to Advisory Board for Cryptographic Technology (2019), members of the task force stated the following:

- System administrators should prepare for migration by taking into account the lifetime of data operated in an IT system while continuing to use the existing cryptographic algorithms. This is because it could take more than 10 years to complete the migration in case of a large-scale system.
- System administrators will be able to select one of the new algorithms, depending on the application in the future, as NIST adopts policies to maintain a certain level of diversity in PQC.
- It is necessary to pay attention to how widely each algorithm is used. In the past, NIST developed and standardized SHA-3 in preparation for the event that SHA-1 is compromised. However, SHA-3 has not prevailed widely.

These statements suggest continuing to evaluate various algorithms so that the most appropriate algorithm is selected for each application. [45]

Rainbow from the finalists and promoting SPHINCS+, a hash-based signature algorithm, in order to maintain candidate diversity (https://csrc.nist.gov/Projects/post-quantum-cryptography/Email-List).

[44] The CRYPTREC Ciphers List includes ciphers referred to in the procurement for the e-Government system in Japan (Cybersecurity Strategic Headquarters [2018]). FISC Security Guidelines on Computer Systems for Banking and Related Financial Institutions introduces the list as reference for the use of cryptography to prevent data leakage and manipulation (Center for Financial Industry Information Systems [2018]).

[45] The Cryptographic Technology Evaluation Committee published a report that explained Arute *et al.* (2019)'s claim and its impact on the CRYPTREC Ciphers List (Cryptographic Technology Evaluation Committee [2020b]). Arute *et al.* (2019) claimed to have experimentally confirmed quantum supremacy.

B. CRYPTREC Ciphers List

When adding a cryptographic algorithm to the CRYPTREC Ciphers List, the Advisory Board for Cryptographic Technology usually takes its practical use into account. Given that PQC algorithms have not been widely deployed yet, some members of the task force suggest establishing PQC guidelines for system administrators and developers. The Advisory Board for Cryptographic Technology discussed the documentation for PQC in June 2020 (Advisory Board for Cryptographic Technology [2020]).

To our knowledge, the Advisory Board for Cryptographic Technology has not published a concrete scope and contents of the document. According to Advisory Board for Cryptographic Technology (2019), some members of the task force suggested that the task force should continuously follow the activities of institutions such as NIST and evaluate the impact of quantum computers not only on public-key algorithms but also on symmetric-key algorithms. Taking into account these opinions, the document for PQC is likely to include topics relating to symmetric-key algorithms, as well as NIST's technical verification of the PQC candidate algorithms.

VI. Challenges in PQC Implementation

There are growing industry efforts toward the implementation of PQC candidate algorithms in NIST's standardization (Sikeridis, Kampanakis, and Devetsikiotis [2020], Schwabe, Stebila, and Wiggers [2020]). IETF has been discussing the extension of the TLS 1.3 specification in order to implement PQC.[46,47] In addition, OQS has been developing and evaluating prototypes of cryptographic libraries implementing the PQC candidate algorithms in order to integrate them into various cryptographic products.

A. Activities in IETF

The aim of the extension is to introduce a *hybrid design* to the TLS 1.3 specification. In the following, we will present an overview of the hybrid design and the

[46] Transport Layer Security (TLS) is a well-known cryptographic protocol that ensures the confidentiality and integrity of communication data and client/server authentication in various online financial services such as Internet banking. As of this writing, the latest version is 1.3 (Rescorla [2018]), whose technical specification is standardized as RFC 8446.

[47] IETF has also begun investigating the extension of SSH (Secure Shell, Kampanakis *et al.* [2020]). SSH is a protocol for authentication and cryptographic communication in remote login and remote file copying on the Internet. Its technical specification is standardized as RFC 4251.

extension.

1. Hybrid design

When a server performs cryptographic communication with a client, both parties must use the same set of cryptographic algorithms. If a server wants to use a new PQC algorithm instead of an existing one, the server needs to ask clients to update their network environments such that the new algorithm is available. However, it is not practical for all of potential clients to do so at the same time. Thus, how to ensure interoperability during the algorithm migration needs to be discussed.

The hybrid design has gained much attention as a method for carrying out the migration in such a heterogeneous environment. When updating or preparing a new network environment, a client and/or server make both the existing and new algorithms available in the environment (i.e., "hybrid-aware"). A hybrid-aware server can use the new algorithm when communicating with a hybrid-aware client. The server uses the existing algorithm when communicating with a client who is not hybrid-aware. If all potential clients become hybrid-aware, the server can stop using the existing algorithm before it is compromised.

However, the hybrid design requires additional costs for updating hardware and/or software. Additional operations for the algorithm negotiation also increase the processing time when establishing the connection. Thus, when implementing the hybrid design, it is necessary to assess its impact on performance and discuss how to mitigate it.

2. TLS 1.3 specification extension

a. Goals

Experts from IETF have been discussing the extension of the TLS 1.3 specification in order to introduce the hybrid design to the key exchange. [48,49] As of this writing, the latest version of the corresponding Internet-Draft lists the following goals

[48] Stebila, Fluhrer, and Gueron (2021) define the hybrid design for the key exchange as "using multiple key exchange algorithms simultaneously and combining the result with the goal of providing security even if all but one of the component algorithms is broken."

[49] In terms of the digital signature, the hybrid design is not always required in TLS. The server authentication requires the signature algorithm to be secure until the end of the corresponding session. Even if the algorithm becomes insecure after the session, the server authentication itself will not be affected (Paquin, Stebila, and Tamvada [2020]). As of this writing, no proposal has been made to extend the TLS 1.3 specification for the purpose of the hybrid design for signature algorithms.

(Stebila, Fluhrer, and Gueron [2021]):

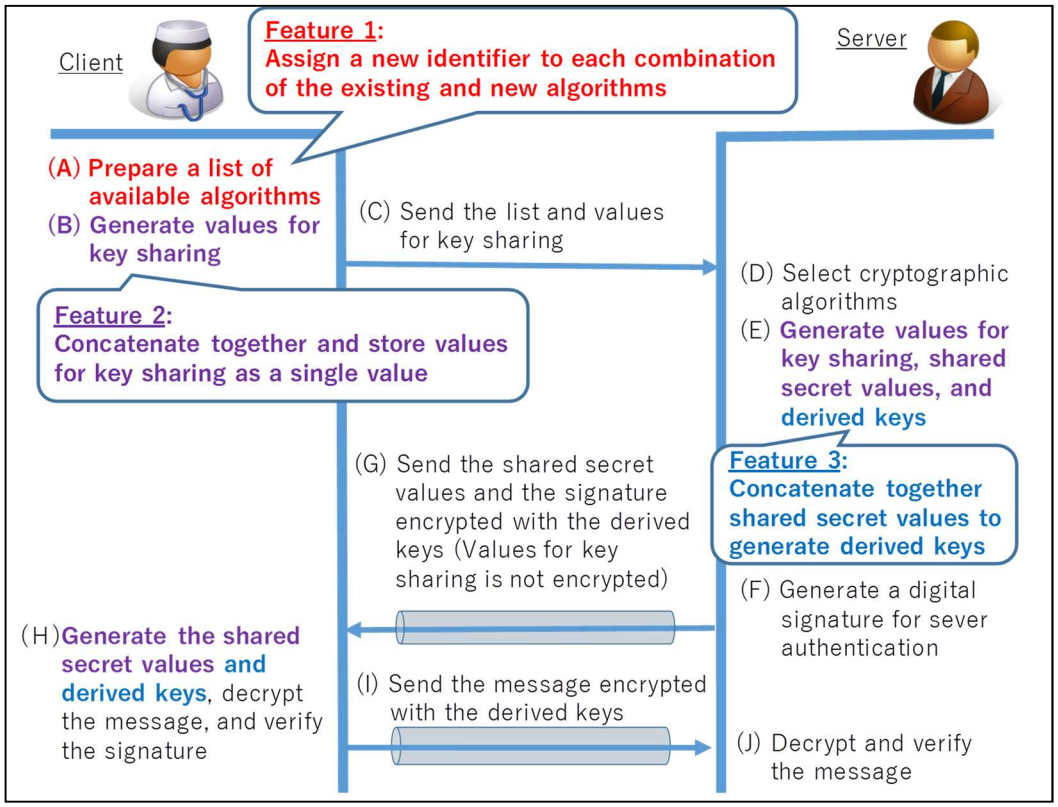
- Security: When combining multiple algorithms (with different cryptographic security properties) to share multiple secret keys, one of the secret keys should remain secure as long as one of the algorithms remains unbroken.
- Backwards compatibility: A hybrid-aware server can share a key with a client who is not hybrid-aware (i.e., a client who can only use the old algorithm).
- High performance: The use of hybrid key exchange is not be prohibitively expensive in terms of computational performance.
- Low latency: The use of hybrid key exchange should not substantially increase latency when establishing the connection.
- No extra round trips: The hybrid key exchange should not lead to extra round trips.
- Minimal duplicate information: The hybrid key exchange should not require sending multiple public keys of the same type.

b. Features and issues for specification extension

The proposed specification for the hybrid key exchange has the following three main features (see Figure 1).

- Feature 1: A new identifier is assigned to each combination of existing and new algorithms used for the hybrid key exchange.
  - ✓ At the beginning of the TLS 1.3 session, a client provides a server with a list of identifiers representing available algorithms for key exchange, as well as values for key sharing. The server selects one of those algorithms and generates a value for key sharing. Then the server sends the corresponding identifier and the value back to the client. However, there is no identifier for the combination of existing and new algorithms for the hybrid key exchange.
- Feature 2: A value for key sharing in the new algorithm is concatenated with one in the existing algorithm.
  - ✓ Messages of the current TLS 1.3 specification have no area to store the value for key sharing in the new algorithm. In order to avoid changing existing message structures, the two values for key sharing are directly

Figure 1: Overview of the hybrid key exchange in the TLS 1.3 session



concatenated together and stored as a single value in the existing area.

- Feature 3: Two shared secret values calculated in the key schedule are concatenated together.
  - ✓ Messages are encrypted with a symmetric-key algorithm in the TLS 1.3 session. Secret keys for the algorithm (i.e., derived keys) are derived from the key schedule that takes a shared secret value as an input parameter. In a hybrid key exchange, two shared secret values are generated by using the existing and new algorithms. However, there is no data area for the value in the new algorithm. In order to avoid modification of the key schedule specification, the shared secret values are directly concatenated together.

However, some implementation issues remain even with these extensions (Stebila, Fluhrer, and Gueron [2021]). One of such issues is the size of public keys and ciphertexts. PQC algorithms tend to have larger public keys and/or ciphertexts than the existing algorithms. For example, some algorithms have larger public keys than the upper limit of the data area in the message structure

specified in TLS 1.3.[50] The following three solutions have been proposed. The first is to revise the upper limit on the size of the data area for storing public keys and ciphertexts. The second is to add an extended area for storing them. The third is to store public keys on an external server and provide their location data (e.g., URL) as reference information for public keys.

B. Open Quantum Safe

The OQS project is an industry-academia collaborative project aimed at supporting the prototyping and evaluating the performance of PQC algorithms.[51] One of the project's developments is LIBOQS, an open-source cryptographic library that runs NIST's candidate algorithms. In the following, we give an overview of LIBOQS and the performance evaluation using LIBOQS.

1. LIBOQS

LIBOQS was released and is constantly updated on GitHub. As of this writing, the latest version of LIBOQS (0.5.0, released on March 10, 2021) includes seven finalists and six alternatives from NIST's candidate algorithms.

LIBOQS is incorporated into several software libraries that run cryptographic protocols, such as TLS, SSH, and S/MIME.[52] For example, OQS-OpenSSL and OQS-BoringSSL, which integrate LIBOQS into OpenSSL and BoringSSL, respectively, were also developed under the OQS project.[53] These enable the implementation of hybrid key exchange and server authentication with PQC algorithms in TLS communication. PQCrypto-VPN has been developed in such a way that integrates LIBOQS into OpenVPN.[54]

---

[50] In Classic McEliece, the size of a public key is greater than or equal to 261,120 bytes. This is larger than the maximum size (65,535 bytes) of the data area for storing the key.

[51] Many researchers are involved in OQS (https://openquantumsafe.org/, December 11, 2020), including researchers from the University of Waterloo, University of London, University of Delaware, University of Rathbone, University of New Mexico, as well as researchers from IBM, Amazon.com, Inc., Microsoft Corporation, and Cisco Systems, Inc.

[52] Secure/Multipurpose Internet Mail Extension (S/MIME) enables message confidentiality, sender authentication, and message authentication by using data encryption and digital signatures. Its message format has been standardized as RFC 5751 (SMIME Version 4.0 Message Specification) in 2019.

[53] OpenSSL is a widely known cryptographic library that runs SSL and TLS. BoringSSL has been developed on the basis of OpenSSL. OQS-OpenSSL has a function that runs S/MIME using PQC algorithms and issues the X.509-compliant digital certificate.

[54] A virtual private network (VPN) is a secure network that obtains the same security properties

Some hardware security modules have also been integrated with LIBOQS.[55] In 2019, Utimaco GmbH and evolutionQ Inc. announced that they implemented LIBOQS on hardware security modules produced by Utimaco GmbH.[56,57]

2. Performance evaluation

Many studies have been published on the performance evaluation of cryptographic libraries incorporating LIBOQS. [58] Most of them measured the time required to complete a TLS 1.3 handshake. In the following, we will explain recent results of the performance evaluation for hybrid key exchange and server authentication.

a. Processing time of hybrid key exchange

Paquin, Stebila, and Tamvada (2020) evaluated the processing time of the hybrid key exchange in OQS-OpenSSL (version 1.1.1) under various network conditions. They implemented ECDH with a 256-bit public key as the existing algorithm. For the PQC algorithms, they selected three NIST candidates, CRYSTALS-KYBER, FrodoKEM, and SIKE.[59] They emulated a network between a client and server and conducted experiments with several variations on network conditions such

---

as a private network by using cryptographic techniques. VPN also refers to the method of establishing such a network.

[55] A hardware security module has functions to securely generate and store cryptographic keys and perform cryptographic algorithms using such keys. It also has the capability to detect or prevent non-invasive attacks (e.g., side-channel attacks) and invasive attacks (e.g., probe attacks).

[56] The press release indicates that evolutionQ Inc. verified that LIBOQS runs properly (https://hsm.ultimaco.com/news/utimaco-evolutionq-set-standards-by-taking-post-quantum-crypto-open-source/, December 15, 2020).

[57] The cloud is also one application of PQC. In November 2020, IBM announced the research and development of quantum-safe cryptography for improving the security of data processing in the cloud against quantum computers (https://newsroom.ibm.com/2020-11-30-IBM-Cloud-Delivers-Quantum-Safe-Cryptography-and-Hyber-Protect-Crypto-Services-to-Help-Protect-Data-in-the-Hybrid-Era, December 15, 2020). Cryptographic tools developed by OQS are included in the research scope.

[58] The latest results of the performance evaluation can be accessed on the OQS website (https://openquantumsafe.org/benchmarking/visualization/handshakes.html, April 30, 2021).

[59] Parameters of the candidate algorithms were set to those corresponding to security level 1 of NIST's criteria. According to National Institute of Standards and Technology (2016b), any attack that breaks algorithms with this security level must require computational resources comparable to or greater than those required for key search on a block cipher with a 128-bit key.

as a round-trip time and packet loss rate.

The round-trip time represents the geographic distance from a client to a server at different locations. Paquin, Stebila, and Tamvada (2020) chose four conditions for a round-trip time based on those actually measured on the Internet. For example, the round-trip time was set to about 200 milliseconds for the least optimal condition (i.e., the longest distance).

The packet loss rate represents the quality of the data transmission channel. To observe how the packet loss affects performance, they conducted experiments while varying the rate from 0% to 20% for each round-trip time.

Their results indicated that the processing time of the hybrid key exchange using CRYSTALS-KYBER did not increase significantly compared with that of ECDH alone, regardless of the network conditions. In terms of the hybrid key exchange with FrodoKEM, the processing time increased compared with that of ECDH alone when the packet loss rate is equal to or more than 5%. As the packet loss rate increased, the difference in the processing time also increased.[60] In terms of the hybrid key exchange with SIKE, the processing time did not increase significantly compared with that of ECDH alone, except for relatively short distances (a round-trip time of about 30 milliseconds or less). In the short-distance experiments, the processing time in SIKE was longer than that of ECDH alone.[61]

b. Processing time of server authentication

Paquin, Stebila, and Tamvada (2020) measured the processing time of the TLS 1.3 handshake in OQS-OpenSSL implemented with PQC algorithms for the server authentication. The experiments were set up in the same way as those mentioned in Section VI.B.2.a. They combined ECDH and CRYSTALS-KYBER for the hybrid key exchange. They selected two NIST candidates as digital signature algorithms,

---

60 With the parameter settings of FrodoKEM in the experiments, the public key and ciphertext sizes are about 9,600 and 9,700 bytes, respectively (more than 150 times larger than those of ECDH). This is why packet losses are more likely to cause errors in the key exchange.

61 The shorter the distance, the shorter the time for data transmission and the longer cryptographic processing takes out of the entire processing time. Paquin, Stebila, and Tamvada (2020) observed that the time required for encryption in SIKE (about 23 milliseconds) was much longer than that of ECDH (about 0.07 milliseconds). Thus, the difference in processing time occurred in short-distance experiments.

CRYSTALS-DILITHIUM and Picnic,[62] as well as ECDSA (a 256-bit public key) as the benchmark.

The processing time of the server authentication using CRYSTALS-DILITHIUM did not increase significantly compared with when ECDSA was used. In contrast, the processing time when using Picnic increased significantly compared with that of ECDSA. The difference in the processing time between Picnic and ECDSA tended to increase as the packet loss rate increased.[63]

Sikeridis, Kampanakis, and Devetsikiotis (2020) also evaluated the processing time of the TLS 1.3 handshake in OQS-OpenSSL. For the PQC signature algorithms, they selected five NIST candidates: CRYSTALS-DILITHIUM, FALCON, Rainbow, Picnic, and SPHINCS+.[64] They used RSA with a 3,072-bit public key and ECDSA with a 384-bit public key as the benchmark. In the experiments, they implemented ECDH with a 256-bit public key as the key exchange algorithm and prepared several variations of geographic distances between a client and server, e.g., a round-trip time of about 11 milliseconds (assuming Internet communication within the United States), about 230 milliseconds (assuming the communication between the United States and the Asian region), etc.

The results indicated that the processing time of server authentication using CRYSTALS-DILITHIUM did not increase significantly compared with the benchmark algorithms in the experiments where both the client and server were located in the United States.[65] This was the case for FALCON. On the other hand, the processing time using Rainbow was more than three times longer than the benchmark algorithms due to the larger public key and server certificate. This was the case for Picnic and SPHINCS+, which had a larger signature size.

---

[62] The algorithms implemented in OQS-OpenSSL were those from the NIST Round 2 submissions. Their parameters were set to those corresponding to security level 1 of NIST's criteria.

[63] The signature size of Picnic is about 34,000 bytes (more than 530 times larger than ECDSA). Similarly to FrodoKEM, packet losses are more likely to cause errors in transmitting signature data compared with ECDSA.

[64] The algorithms implemented were those from the NIST Round 2 submissions. Their parameters were set to those corresponding to security level 1 of NIST's criteria.

[65] In server authentication, a client verifies not only the server certificate but also the intermediate CA certificate. Paquin, Stebila, and Tamvada (2020) did not introduce the intermediate CA certificate in their study; a client only verified the server certificate in their experiments.

VII. Concluding Remarks and Future Prospects

A. Security of Cryptographic Algorithms in the Future

Existing algorithms such as RSA and the elliptic-curve cryptography are believed to be secure against attackers using classical computers. However, they have been theoretically demonstrated insecure against attackers with ideal quantum computers running Shor's algorithm.

On the contrary, PQC algorithms are expected to remain secure against attackers with ideal quantum computers; however, the security of some classes of new PQC algorithms have yet to be thoroughly investigated, even against attackers only using classical computers. Cryptographic researchers may discover a new and powerful cryptanalysis such as side-channel attacks exploiting the implementation or mathematical structures of the algorithms in the future. Thus, the security of PQC algorithms needs to be further evaluated against attackers with classical computers as well as those with quantum computers.

PQC algorithms are based on advanced mathematics, making it difficult for non-experts to understand the implication of studies regarding security evaluation. We expect cryptographers and researchers to provide their insights on security evaluation, for example, through the activities of CRYPTREC.

Organizations that intend to use PQC algorithms should regularly monitor trends in security evaluation. Although not mentioned in this paper, most PQC algorithms have many security parameters, and appropriate combinations of the parameters in practical use should also be verified.

B. Challenges on Migration to PQC

In the migration to PQC algorithms in an IT system, system administrators should improve the robustness of the system by combining PQC with existing algorithms which have been thoroughly studied for classical computing. NIST has selected multiple candidate algorithms that make use of various difficult-to-solve mathematical problems so that one technological innovation does not compromise all of the candidates. When developing a new IT system with a PQC algorithm, a system administrator should follow NIST's policy and design the system in such a way that enables the algorithm to be easily replaced in the future.

In addition to the U.S. government, major vendors and standardization bodies have been preparing for the migration to PQC algorithms. The U.S. government, which is the largest user of cryptography in the world, has

established a tentative timeline to complete the PQC standardization in the latter half of the 2020s, as of this writing. After completion, standardized algorithms will likely prevail rapidly regardless of the development of an ideal quantum computer. In such a case, financial institutions and payment service providers, including the Bank of Japan, will have to consider using PQC algorithms in order to preserve interoperability among their IT systems. Thus, the migration to PQC algorithms should be taken into account when building or upgrading an IT system in the future.

The hybrid design should be discussed further. OQS evaluates the performance of the TLS 1.3 handshake using both existing and NIST candidate algorithms in the key exchange. Their experiments cover most candidate algorithms, and some classes of algorithms have been shown to be effective. IETF is also discussing the extension of the TLS 1.3 specification to enable hybrid key exchange, so a cryptographic software library with the hybrid design will likely become available.

If new IT systems with PQC algorithms are developed during the algorithm migration, there will be a heterogeneous mixture of systems with the new and old algorithms. As the migration proceeds, the old systems will be gradually replaced while the new ones are deployed with backward compatibility. During the migration of the digital signatures algorithm, it is necessary to extend the validity period of digital documents signed with existing algorithms. The use of time-stamping services is one way to carry out this extension (Ito, Une, and Seito [2019]). In any case, we should assume that it will take a long time from the preparation to the completion of the migration.

C. Impact on Crypto-Assets

In recent years, public blockchains, which are used for crypto-assets such as Bitcoin, have become increasingly important in society. These systems will also likely to be required to migrate to PQC algorithms in order to ensure security in the future.

Bitcoin mainly utilizes two algorithms: SHA-256 and ECDSA. SHA-256 is a hash function used for combining multiple transaction data into a single block in the form of a Merkle tree. SHA-256 is also used in the Proof-of-Work to agree on the correctness of the linkage between blocks among miners. At present, an ideal quantum computer is estimated to achieve at most cubic acceleration in the SHA-256 collision search. Given this estimation, the threat of quantum computers to

SHA-256 can be mitigated simply by extending a hash size in the same way as symmetric-key algorithms. In contrast, ECDSA used for digital signatures on transactions must be replaced with PQC algorithms in order to counter this threat.

Participants of crypto-assets must agree in advance on new algorithms to be implemented and the timing of the migration. In addition, existing signatures and hash values need to be protected in order to ensure the integrity of past transactions and blocks. One way to preserve the integrity is to take and securely store a hash value from the entire blockchain at the beginning of the migration to PQC algorithms.

Methods to ensure both the privacy protection and transaction integrity have been studied; however, the achieved security levels may differ among these properties. For example, we can suppose the following situation in which privacy protection is only ensured against attackers with classical computers while transaction integrity is ensured against attackers with both classical and quantum computers. When implementing such a method, the social impacts of the difference in these security levels for each property should be considered.

References

Advisory Board for Cryptographic Technology, "Angou Gijutsu Kentoukai 2019 Nendo Houkokusho (Advisory Board for Cryptographic Technology FY 2019 Annual Report)," Cryptography Research and Evaluation Committees, 2020 (available at https://www.cryptrec.go.jp/report/ cryptorec-rp-1000-2019.pdf, in Japanese).

———, "Ryoushi Konpyu-ta Jidai Ni Muketa Angou No Arikata Kentou Tasuku Fo-su, Dai 3 Kai (Task Force on Cryptography in the Era of Quantum Computers, the 3rd Meeting)," Cryptography Research and Evaluation Committees, 2019 (available at https://www.cryptrec.go.jp/report/cryptrec-mt-1430-2019.pdf, in Japanese).

Arute, Frank, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Bukett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowlr, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Ladhuis, Mike Lindmark, Erik Hucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xio Mi, Kristel Michielsen, Masoud Mohsni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis, "Quantum Supremacy Using a Programmable Superconducting Processor," *Nature*, 574, 2019, pp. 505-510 (available at https://www.nature.com/articles/s41586-019-1666-5.pdf).

Bardet, Magali, Jean-Chales Faugère, and Bruno Salvy, "On the Complexity of Gröbner Basis Computation of Semi-Regular Overdetermined

Algebraic Equations," Proceedings of International Conference on Polynomial System Solving, 2004, pp. 71-74 (available at http://magali.bardet.free.fr/Publis/ltx43BF.pdf).

Center for Financial Industry Information Systems, *Kinyuu Kikan Tou Konpyu-ta Shisutemu No Anzen Taisaku Kijun, Kaisetsusho Dai 9 Han (FISC Security Guidelines on Computer Systems for Banking and Related Financial Institutions, the 9th edition)*, Center for Financial Industry Information Systems, 2018 (in Japanese).

Cryptographic Technology Evaluation Committee, "CRYPTREC Report 2019; Report of the Cryptographic Technology Evaluation Committee (Revision 1)," Cryptography Research and Evaluation Committees, 2020a (available at https://www.cryptrec.go.jp/report/cryptrec-rp-2000-2019r1.pdf, in Japanese).

———, "Genzai No Ryoushi Konpyu-ta Ni Yoru Angou Gijutsu No Anzensei Heno Eikyou (An Impact of Current Quantum Computers on the Security of Cryptographic Technology)," Cryptography Research and Evaluation Committees, 2020b (available at https://www.cryptrec.go.jp/topic/cryptrec-er-0001-2019.html, in Japanese).

Cryptographic Technology Research Working Group, "Tai Ryoushi Keisanki Angou No Kenkyu Doukou Chousa Houkokusho (Report on Research Trends of Post-Quantum Cryptography)," Cryptography Research and Evaluation Committees, 2019 (available at https://www.cryptrec.go.jp/report/cryptrec-tr-2001-2018, in Japanese).

Cybersecurity Strategic Headquarters, "Common Standards for Information Security Measures for Government Agencies, FY2018," Cybersecurity Strategic Headquarters, 2018 (available at https://www.nisc.go.jp/eng/pdf/kijyun30-en.pdf).

Faugère, Jean-Charles, "A New Efficient Algorithm for Computing Gröbner Bases without Reduction to Zero (F_5)", Proceedings of the 2002 ACM International Symposium on Symbolic and Algebraic Computation, Association for Computing Machinery, 2002, pp. 75-83.

Fujisaki, Eiichiro, and Tatsuaki Okamoto, "Secure Integration of Asymmetric and Symmetric Encryption Schemes," Proceedings of CRYPTO 1999, Lecture Notes in Computer Science, 1666, Springer, 1999, pp. 537-554.

Gottesman, Daniel, "Stabilizer Codes and Quantum Error Correction," arXiv:quant-ph/9705052, 1997.

Grover, Lov K., “A Fast Quantum Mechanical Algorithm for Database Search,” Proceedings of the 28th Annual ACM Symposium on Theory of Computing, Association for Computing Machinery, Association for Computing Machinery, 1996, pp. 212-219.

Hirota, Osamu, “Quantum Noise Analysis for Quantum Computer, Error Model in Large Scale Quantum Many Body System and Examples,” IEICE Technical Report, Vol. IEICE-120, No. 105, Institute of Electronics, Information and Communication Engineers, 2020, pp. 37-42 (in Japanese).

Hoffstein, Jeffrey, Jill Pipher, and Joseph H. Silverman, “NTRU: A Ring-Based Public Key Cryptosystem,” Algorithmic Number Theory, 3rd International Symposium, ANTS-III, Lecture Notes in Computer Science, 1423, Springer, 1998, pp. 267-288.

Hosoyamada, Akinori, “Ryoushi Konpyu-Ta Ga Kyoutsuukagiangou No Anzensei Ni Oyobosu Eikyou No Chousa Oyobi Hyouka (Evaluation of Impacts of Quantum Computers on the Security of Symmetric-Key Ciphers),” Cryptography Research and Evaluation Committees, 2020 (available at https://www.cryptrec.go.jp/exreport/cryptrec-ex-2901-2019.pdf, in Japanese).

Ito, Tadahiko, Masashi Une, and Takenobu Seito, “Ryousi Konpyu-ta Ni Yoru Kyoui Wo Misueta Angou No Ikou Taiou (A Study on Transition of Cryptographic Algorithms Under Threats from Quantum Computers),” IMES Discussion Paper No. 2019-J-15, Institute for Monetary and Economic Studies, Bank of Japan, 2019 (available at https://www.imes.boj.or.jp/research/papers/japanese/19-J-15.pdf, in Japanese).

Jao, David, and Luca De Feo, “Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies.” Proceedings of International Workshop on Post-Quantum Cryptography 2011, Lecture Notes in Computer Science, 7071, Springer, 2011, pp. 19-34.

Kampanakis, Panos, Douglas Stebila, Markus Friedl, Torben Hansen, and Dimitrios Sikeridis, “Post-Quantum Public Key Algorithms for the Secure Sell (SSH) Protocol, Draft-Kampanakis-Curdle-PQ-SSH-00,” Internet-Draft, Internet Engineering Task Force, 2020 (available at https://tools.ietf.org/pdf/draft-kampanakis-curdle-pq-ssh-00.pdf).

Kaplan, Marc, Gaëtan Leurent, Anthony Leverrier, and María Naya-

Plasencia, “Breaking Symmetric Cryptosystems Using Quantum Period Finding,” Proceedings of CRYPTO 2016 (Part II), Lecture Notes in Computer Science, 9815, Springer, 2016, pp. 207-237.

Kipnis, Aviad, Jacques Patarin, and Louis Goubin, “Unbalanced Oil and Vinegar Signature Schemes,” Proceedings of EUROCRYPT 1999, Lecture Notes in Computer Science, 1592, Springer, 1999, pp. 206-222.

McEliece, Robert J., “A Public-Key Cryptosystem Based on Algebraic Coding Theory,” The Deep Space Network Progress Report, DSN PR 42-44, 1978, pp. 114-116.

National Academies of Sciences, Engineering, and Medicine, *Quantum Computing: Progress and Prospects*, The National Academies Press, Washington, DC, 2019, (available at https://doi.org/10.17226/25196).

National Institute of Standards and Technology, “Announcing Request for Nominations for Public-Key Post Quantum Cryptographic Algorithms,” Federal Register, Vol. 81, No. 244, National Archives and Records Administration, 2016a, pp. 92787-92788 (available at https://www.govinfo.gov/content/pkg/FR-2016-12-20/pdf/2016-30615.pdf).

———, “Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process,” National Institute of Standards and Technology, 2016b (available at https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf).

———, “Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process,” National Institute of Standards and Technology, 2019 (available at https://doi.org/10.6028/NIST.IR.8240).

———, “Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process,” National Institute of Standards and Technology, 2020 (available at https://doi.org/10.6028/NIST.IR.8309).

National Security Agency, “Commercial National Security Algorithm Suite and Quantum Computing FAQ,” MFQ-U-OO-815099-15, National Security Agency, 2016.

Niederreiter, Harald, “Knapsack-Type Cryptosystems and Algebraic Coding Theory,” *Problems of Control and Information Theory*, 15(2), Akadémiai Kiadó, 1986, pp. 159-166.

Nielsen, Michael A., and Isaac L. Chuang, *Quantum Computation and Quantum*

Information: 10th Anniversary Edition, Cambridge University Press, 2010.
Nuida, Koji, *Tai Ryoushi Keisanki Angou (Post-Quantum Cryptography)*, Morikita Publishing Co., Ltd., 2020 (in Japanese).
Paquin, Christian, Douglas Stebila, and Goutam Tamvada, "Benchmarking Post-Quantum Cryptography in TLS," Proceedings of Conference on Post-Quantum Cryptography 2020, Lecture Notes in Computer Science, 12100, Springer, 2020, pp. 72-91.
Peikert, Chris, and Zachary Pepin, "Algebraically Structured LWE Revisited," Proceedings of Theory of Cryptography Conference 2019, Lecture Notes in Computer Science, 11891, Springer, 2019, pp. 1-23.
Regev, Oded, "New Lattice-Based Cryptographic Constructions," *Journal of the ACM*, 51(6), Association for Computing Machinery, 2004, pp. 899-942.
Rescorla, Eric, "The Transport Layer Security (TLS) Protocol Version 1.3," Request for Comments: 8446, Internet Engineering Task Force, 2018.
Schwabe, Peter, Douglas Stebila, and Thom Wiggers, "Post-Quantum TLS without Handshake Signatures," Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, Association for Computing Machinery, 2020, pp. 1461-1480.
Shikata, Junji, "Ryousi Konpyu-ta Ni Taisei No Aru Angou Gijutsu No Hyoujunka Doukou: Beikoku Seifu Hyoujun Angou Ni Tsuite (Recent Trends on Standardization of Cryptography Secure against Quantum Computers by the U.S. Government)," IMES Discussion Paper No. 2019-J-4, Institute for Monetary and Economic Studies, Bank of Japan, 2019 (available at https://www.imes.boj.or.jp/research/papers/japanese/19-J-04.pdf, in Japanese).
Shor, Peter W., "Algorithms for Quantum Computations: Discrete Logarithms and Factoring," Proceedings of 35th Annual Symposium on Foundations of Computer Science, IEEE, 1994, pp. 124-134.
———, "Scheme for Reducing Decoherence in Quantum Computer Memory," *Physical Review A*, 52(4), 1995, pp. 2493-2496.
———, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM Journal on Computing*, 26(5), Society for Industrial and Applied Mathematics, 1997, pp. 1484-1509.
Sikeridis, Dimitrios, Panos Kampanakis, and Michael Devetsikiotis, "Post-Quantum Authentication in TLS 1.3: A Performance Study," Proceedings of Network and Distributed System Security 2020, Internet Society, 2020

(available at https//www.ndss-symposium.org/wp-content/uploads/2020/02/24203.pdf).

Simon, Daniel R., "On the Power of Quantum Computation," Proceedings of 35th Annual Symposium on Foundations of Computer Science, IEEE, 1994, pp. 116-123.

Stebila, Douglas, Scott Fluhrer, and Shay Gueron, "Hybrid Key Exchange in TLS 1.3, Draft-Ietf-Tls-Hybrid-Design-02," Internet-Draft, Internet Engineering Task Force, 2021 (available at https://tools.ietf.org/pdf/draft-ietf-tls-hybrid-design-02.pdf).

Tabuchi, Yutaka, "Ryoushi Konpyu-ta Ha-dowhea A-kitekucha (Choudendo Soshi) No Kentou (Research on Hardware Architecture of Quantum Computer with Superconductivity Device)," Presentation at the First Workshop of Special Interest Group on Quantum Software, Information Processing Society of Japan, 2020 (in Japanese).

Takagi, Tsuyoshi, *Angou To Ryoushi Konpyu-Ta – Tai Ryoushi Keisanki Angou Nyuumon – (Cryptography and Quantum Computers: Introduction to Post-Quantum Cryptography)*, Ohmsha, Ltd., 2019 (in Japanese).

Thompson, Neil C., Kristjan Greenewald, Keeheon Lee, and Gabriel F. Manso, "The Computational Limits of Deep Learning," arXiv:2007.05558, 2020.

Vepsäläinen, Antti P., Amir H. Karamlou, John L. Orrell, Akshunna S. Dogra, Ben Loer, Francisca Vasconcelos, David K. Kim, Alexander J. Melville, Bethany M. Niedzielski, Jonilyn L. Yoder, Simon Gustavsson, Joseph A. Formaggio, Brent A. VanDevender, and William D. Oliver, "Impact of Ionizing Radiation on Superconducting Qubit Coherence," *Nature*, 584, 2020, pp. 551-556.

Zhong, Han-Sen, Hui Wang, Yu-Hao Deng, Ming-Cheng Chen, Li-Chao Peng, Yi-Han Luo, Jian Qin, Dian Wu, Xing Ding, Yi Hu, Peng Hu, Xiao-Yan Yang, Wei-Jun Zhang, Hao Li, Yuxuan Li, Xiao Jiang, Lin Gan, Guangwen Yang, Lixing You, Zhen Wang, Li Li, Nai-Le Liu, Chao-Yang Lu, and Jian-Wei Pan, "Quantum Computational Advantage Using Photons," *Science*, 370(6523), 2020, pp. 1460-1463.

Appendix. Threat of Quantum Computing to Symmetric-Key Cryptography

Symmetric-key cryptography is applied to the encryption of the main body of data instead of key distribution. Large-scale fault-tolerant quantum computing will pose a moderate threat to symmetric-key cryptography rather than to public-key cryptography. Grover's algorithm achieves quadratic speedup for running the full search of secret keys and the pre-image of cryptographic hash functions. Brasard-Hoyer-Tapper (BHT) algorithm is one application of Grover's algorithm that enables cubic speedup for finding collisions in the hash functions. The symmetric-key cryptography or the hash functions are mostly expected to remain secure against the speedup of attacks stemming from those algorithms simply by extending the sizes of keys and hash values. However, it is notable that recent research has improved the efficiency of algorithms for breaking symmetric-key cryptography (Kaplan *et al*. [2016]).

The Cryptographic Technology Evaluation Committee of CRYPTREC commissioned an external researcher to evaluate the impact of quantum computing to the security of symmetric-key cryptography (in CRYPTREC Ciphers List) and published the report (Hosoyamada [2020]).[66] Referring to this, the committee published their official report for the 2019 fiscal year (Cryptographic Technology Evaluation Committee [2020a]). The report approved the evaluation of Hosoyamada (2020) and concluded that imminent threats to the symmetric-key cryptography with modes of operations and the hash functions are not likely to emerge. At present, CRYPTREC does not need to take any specific action against the threats; however, the report warns that future risks are unpredictable. Thus, CRYPTREC will continue to monitor the threat of novel attacks and the development of quantum computers.

[66] Hosoyamada (2020) concluded that symmetric-key algorithms with a 192-bit or 256-bit key should be implemented to ensure long-term data confidentiality in order to address the threats of quantum computers. This conclusion is based on the research that a full search for k bit keys can be performed in time $O(2^{k/2})$ using Grover's algorithm.