大和総研
Daiwa Institute of Research

Daiwa Institute of Research Ltd

Post-Quantum Cryptography

Proof of Concept Results and Crypto-Migration Strategy at Daiwa Securities Group

Daiwa Institute of Research Ltd, Digital Solutions R&D Dept.

July 24, 2026
Origin Publication Date (Japanese): March 31, 2026

Disclaimer

This document summarizes current information on Post-Quantum Cryptography (PQC) and the results of verifications conducted within our group. The content is subject to change depending on future standardization trends, product implementation status, and changes in the threat environment.

The information provided in this document is for informational purposes only. Product names, company names, and service names mentioned herein are included for the purpose of explaining the demonstration environment, and do not constitute a guarantee or recommendation by our company for the adoption, performance, or future compatibility of any specific product or service.

The technical trends and product compatibility status described in this document are generally based on information as of March 2026.

Unless otherwise noted, all figures and tables in this document were created by Daiwa Institute of Research Ltd.

Revision History

Version	Date	Reviser	Summary
V1.0	2026-07-24	Daiwa Institute of Research	First edition (Translated based on the Japanese version V1.0)

Table of Contents

Summary

With the advancement of quantum computers, currently widely used public-key cryptography such as RSA and elliptic curve cryptography may face decryption risks in the future. In particular, preparing for “Harvest Now, Decrypt Later” (HNDL) attacks—where encrypted data is collected now to be decrypted later—is crucial for financial institutions that require long-term confidentiality. Adapting to PQC is not only a medium- to long-term security issue but also a management issue. In addition, the U.S. National Institute of Standards and Technology (NIST) published its first PQC standards in 2024, and in Japan, transition studies are underway in the government and financial sectors with an eye toward around 2035.

The Daiwa Securities Group conducted a PQC proof-of-concept (PoC) experiment using the development environment of its online services. The verification results confirmed that for general web service applications in this demonstration environment, the impact on communication due to the PQC adaptation of key exchange (KEM) is limited. The increase in TLS handshake time during key exchange was limited, and no significant increase in CPU or memory usage was observed even with a PQC software-compatible load balancer, with the increase in communication volume remaining at about 5%. On the other hand, since an increase in communication volume due to the larger key and certificate sizes associated with the change to PQC will inevitably occur, individual impact assessments are necessary for systems with narrow communication bandwidths, poor communication quality, or strict low-latency requirements.

In practice, it is important not to view the PQC transition as a simultaneous overhaul, but to proceed in the following order: (1) inventory of cryptographic assets, (2) development of a cryptographic inventory, (3) risk assessment and prioritization, (4) roadmap formulation, and (5) phased transition. As of March 2026, the standardization of PQC signatures and certificates has not been completed. Discussions on how to use them in TLS and X.509 certificates are taking place in the Internet Engineering Task Force (IETF), and operational rules for public PKI are being discussed in the Certification Authority/Browser Forum (CA/B Forum). Therefore, in the first phase of the transition, it is realistic to start with key exchange, which is effective against HNDL attacks and has advanced implementation, and to proceed with PQC adaptation for signatures and certificates while monitoring standardization and product support. It is desirable to proceed in stages, prioritizing critical systems and focusing on responses at the edge termination and common cryptographic infrastructure, while keeping crypto-agility in mind.

1. Introduction

1.1 Background and Purpose

As research and development of quantum computers progresses globally, it has been pointed out that currently widely used public-key cryptography, such as RSA and elliptic curve cryptography, could be broken with the advent of quantum computers possessing sufficient performance.

A particularly serious threat recognized is “Harvest Now, Decrypt Later” (HNDL). This is an attack scenario in which an attacker collects and stores currently encrypted communication and stored data, and decrypts it after the practical application of quantum computers in the future. Even data that is securely protected today carries the risk of losing its confidentiality 10 or 20 years from now.

As a solution to this threat, the research, development, and standardization of Post-Quantum Cryptography (PQC) are underway. PQC is a cryptographic technology based on mathematical problems that are considered difficult for even quantum computers to solve efficiently, and it is a cryptography that operates on conventional computers.

Financial institutions handle large amounts of confidential information that must be kept secret over the long term, such as customer asset information, transaction histories, and personal information. The compromise of cryptography poses a direct business risk. Furthermore, regulations and guidelines, including Japan’s Financial Services Agency's “Policy Initiatives for Strengthening Cybersecurity in the Financial Sector,” require appropriate management of cryptographic technology and responses to the latest threat trends.

In a statement published by the G7 Cyber Expert Group in 2026, a cross-sectoral and coordinated timeline concept for the transition to PQC in the financial sector was presented. Recognizing the importance of completing the transition before the advent of quantum computers capable of breaking cryptosystems, the statement sets a target of around 2035 for the financial system as a whole, while aiming for prioritized action around 2030-2032 for the most critical systems, such as payment and settlement systems.

It is known that PQC requires larger key sizes compared to conventional cryptography, and evaluating its impact on performance and stability is an important factor in planning. This time, we verified the challenges and issues in introducing PQC under conditions close to the actual business environment of Daiwa Securities' online services. The target system is a core system that exchanges confidential information with customers via TLS communication, currently using RSA and elliptic

curve cryptography. This demonstration was conducted as a proof-of-concept (PoC) using a development environment to perform technical and operational evaluations before production application.

The purposes of this white paper are as follows:

- To organize the basic concepts, standardization, and institutional trends of Post-Quantum Cryptography (PQC) to build a common understanding within companies/organizations.
- To provide the results of the PQC proof-of-concept (PoC) conducted by the Daiwa Securities Group as reference material for the financial industry and the general public.
- To present a practical approach to PQC transition, including the development of a cryptographic inventory, risk assessment, and a cryptographic transition roadmap.
- To clarify the perspectives of transition plan formulation, vendor coordination, and performance impact assessment, providing them in a form applicable to practical operations.

1.2 Target Audience and Structure of this Document

This document is intended for a wide range of stakeholders in organizations considering or implementing a transition to Post-Quantum Cryptography (PQC). In particular, the content is geared toward critical infrastructure operators, including financial institutions, and organizations that need to protect highly sensitive information over the long term.

Specifically, we assume the following roles as our main readers:

[Technical Staff]

- **Infrastructure/Network Design and Operations Staff**: Considering PQC adaptation for TLS/SSL communications, load balancers, firewalls, etc.
- **Application Development Staff**: Selecting cryptographic libraries and considering implementation methods.
- **PKI Operations/Key Management Staff**: Grasping certificate transition plans and the support status of certificate authorities.

[Planning and Management Staff]

- **Information Security Staff**: Developing cryptographic inventories, assessing risks, and formulating transition roadmaps.
- **Risk Management/Audit Staff**: Checking regulatory compliance status and assessing third-party risks.
- **System Planning Staff**: Coordinating system renewal plans with PQC transition and considering budget allocation.

[Decision Makers]

- **IT Department Managers/Executive Management**: Understanding the necessity of PQC transition, making investment decisions, and formulating company-wide policies.

This document is not intended for cryptography experts, but is written as plainly as possible so that anyone involved in IT-related work can understand it. We have avoided delving deeply into the mathematical details of cryptography, focusing instead on presenting the practical knowledge and specific approaches necessary for the PQC transition. The structure allows even readers without prior knowledge of cryptographic technology to understand the overall picture and practical methods of PQC transition by reading sequentially from Chapter 2.

This document consists of five chapters and an appendix. The outline of each chapter is as follows:

Chapter 2, “Basic Knowledge of Post-Quantum Cryptography,” explains the basic concepts of PQC in plain language. It outlines the technical background, including the differences from Quantum Key Distribution (QKD), the timeline for predicting the emergence of quantum computers, the history of NIST standardization, and the adopted algorithms.

Chapter 3, “Proof-of-Concept Experiment of Post-Quantum Cryptography Conducted by the Daiwa Securities Group,” details the verification environment, verification items, and verification results for key exchange/key encapsulation and digital signatures, which were conducted in this demonstration experiment and basic verification using the verification environment. Based on the verification results, it specifically describes the potential impacts when introducing PQC.

Chapter 4, “Approach to Cryptographic Transition,” explains a practical approach in six steps for organizations implementing a PQC transition. It presents a series of processes considered necessary in practice, from inventorying cryptographic assets, creating and managing a cryptographic inventory, risk assessment and prioritization, roadmap formulation, consideration of specific transition methods, to impact assessment.

Chapter 5, “Conclusion,” provides a summary and future outlook.

This document is structured so that by reading sequentially from Chapter 1, readers can systematically understand everything from the necessity of PQC transition to specific practical methods.

However, depending on the reader's role and interests, the following reading methods are also possible:

- **Executive Management/Decision Makers**: Focus on Chapters 1, 2, and Section 1 of Chapter 5 to grasp the necessity of PQC transition, regulatory trends, and an overview of how to proceed with the transition.
- **Technical Staff**: Focus on Chapters 2 and 3 to deeply understand the technical details and the results of the demonstration experiment.
- **Planning/Security Staff**: Focus on Chapters 2 and 4 to understand institutional compliance and the practical aspects of the transition process.

Each chapter is designed to be read somewhat independently. We recommend that readers new to this topic read the entire document, while those seeking information from a specific perspective refer to the relevant chapters.

2. Basic Knowledge of Post-Quantum Cryptography

2.1 What is Post-Quantum Cryptography?

2.1.1 Definition of Post-Quantum Cryptography

Post-Quantum Cryptography (PQC) is a general term for a cryptographic scheme that can be implemented and operated on conventional (classical) computers and are expected to maintain security against attacks by quantum computers.

Currently, public-key cryptography widely used on the Internet and in e-commerce, such as RSA and elliptic curve cryptography, bases its security on mathematical problems that are considered "computationally difficult to solve," such as integer factorization and discrete logarithm problems. However, a quantum algorithm (Shor's algorithm) published by Peter Shor in 1994 theoretically demonstrated that if a quantum computer with sufficient performance is realized, these problems could be solved in a realistic amount of time. This is the basis for the possibility of current public-key cryptography being broken by quantum computers mentioned in Section 1.1.

PQC is a cryptographic scheme designed based on other mathematical problems that are considered difficult to solve efficiently even with a quantum computer. The important point is that PQC does not need to run on a quantum computer; it can basically be implemented on the conventional computers, servers, and smartphones we use today through software updates.

2.1.2 Differences from Quantum Cryptography

PQC is a different technology from "Quantum Cryptography." This distinction is important when considering the introduction of PQC, so it is clarified here.

Quantum Key Distribution (QKD), a method of quantum cryptography, is a technology that distributes secret keys using quantum mechanical properties such as photons, serving as an alternative key exchange method to public-key cryptography. Based on quantum mechanics, QKD can theoretically detect any eavesdropping and achieve key exchange with information-theoretic security. Information-theoretic security means mathematically provable security that does not depend on the attacker's computational power, guaranteeing extremely high security with no fear of algorithm compromise in the future.

On the other hand, conventional cryptographic methods, including PQC, are based on "computational security." This security is based on the premise that the amount of computation required to break the

cryptosystem is so enormous that it is practically impossible to execute within a realistic time and cost.

The main differences between the two are shown in the table below.

Aspect	PQC (Post-Quantum Cryptography)	QKD (Quantum Key Distribution)
Operating Environment	Conventional computers	Dedicated equipment
Basis of Security	Computational security	Information-theoretic security
Scope of Application	General Internet communication (TLS, etc.)	Limited (e.g., communication between specific sites)
Introduction Cost (Relative)	Low (Software updates)	High (Requires dedicated equipment and optical fibers)

Table 1: Main differences between PQC and QKD

Basically, PQC, which can be applied to existing Internet infrastructure (TLS/SSL communication, PKI certificate infrastructure, etc.), is considered to be the main countermeasure against the threats posed by the emergence of quantum computers. On the other hand, QKD is expected to play a complementary role in specific applications requiring particularly high security (e.g., communication between government agencies, core communication of critical infrastructure). Although QKD has a limitation in communication distance (generally about 100 km), it can be expanded over a wider area through a QKD network connecting QKD links. Realizing such a network infrastructure is expected to greatly expand the use cases of QKD, and demonstrations are underway in Europe and other countries. In Japan, there is a track record of a testbed (Tokyo QKD Network) by NICT (National Institute of Information and Communications Technology), and studies are progressing toward the construction of a broader area network.

2.1.3 Scope of PQC

Cryptographic technologies are broadly classified into “public-key cryptography” and “symmetric-key cryptography.” The threat of quantum computers affects both, but the magnitude of the impact differs.

Impact on Public-Key Cryptography

Public-key cryptography is widely used for the following purposes:

- Key sharing/Key exchange: Securely sharing a secret key at the start of communication (e.g., TLS/SSL handshake)
- Digital signatures: Authenticating the creator of an electronic document and detecting tampering (e.g., TLS/SSL certificates, code signing, electronic contracts)

These could potentially be efficiently broken by quantum computers using Shor's algorithm. To prepare for the emergence of quantum computers, public-key cryptography requires a complete replacement of the algorithms themselves, making it an area that particularly needs addressing.

Specifically, the following technical elements are targeted:

- Key exchange algorithms (DHE, ECDHE, etc.)
- Digital signature algorithms (RSA, ECDSA, etc.)

These are mainly used in situations such as:

- Authenticated key exchange for communication protection like TLS and IPSec
- Digital signatures for electronic contracts, etc.
- Code signing for software updates, secure boot, etc.
- Email encryption and signing (S/MIME)
- Authentication such as FIDO (digital signatures)
- Cryptocurrencies

Impact on Symmetric-Key Cryptography

Symmetric-key cryptography (e.g., AES) is a method where the sender and receiver share the same key for encryption and decryption. It is known that a quantum algorithm called Grover's algorithm reduces the computational complexity required for key search to the square root compared to classical exhaustive search, providing a quadratic speedup.

For example, the security of AES-128 (128-bit key) in a quantum computing environment drops to the equivalent of AES-64 (64-bit key) on a conventional computer.

Therefore, for symmetric-key cryptography, it is recommended to review the key length (e.g., from AES-128 to AES-256) in preparation for the emergence of quantum computers, but the priority is not high, and a complete replacement of the algorithm itself is considered unnecessary.

Scope of this Document

This document will cover the impact of migrating public-key cryptography to PQC, which is the most urgent response to the quantum threat, in Chapter 3.

2.1.4 Basis of PQC's Quantum Security

PQC is based on mathematical problems that are considered difficult to solve efficiently even with a quantum computer. It is mainly based on the following mathematical properties:

- Lattice-based cryptography: Based on problems like the shortest vector problem on high-

dimensional lattices.

- Code-based cryptography: Based on the decoding problem of error-correcting codes.
- Multivariate polynomial cryptography: Based on the problem of solving systems of multivariate polynomial equations.
- Hash functions: Based on the one-way property of hash functions.

Currently, no algorithms are known that can efficiently solve these mathematical problems even with a quantum computer. Therefore, it should be noted that the security of the cryptography is not "proven," but rather based on the fact that no efficient algorithm has been discovered at present.

As a result, evaluations may change depending on the progress of future quantum algorithm research, making the concept of "crypto-agility"—preparing to flexibly change cryptographic algorithms—crucial. This point will be detailed in Chapter 4.

2.2 Cryptographic Migration Timeline

When considering the migration to Post-Quantum Cryptography (PQC), determining the timeline of "by when we should respond" is extremely important. This section outlines the overall picture, starting from predictions of quantum computer practical application, the positioning of migration targets in international guidelines, and the phased approach to PQC migration.

Specific migration procedures will be detailed in Chapter 4. This section presents the basic concepts for an overview of the entire timeline.

2.2.1 Predictions for Quantum Computer Practical Application and Migration Timing

There is currently no definitive prediction for when quantum computers will reach a performance level capable of practically breaking widely used public-key cryptography. A quantum computer with such decryption capabilities is called a CRQC (Cryptographically Relevant Quantum Computer). Many public institutions and researchers estimate the emergence of CRQC to be in the 2030s at the earliest, but do not deny the possibility of it being brought forward due to technological breakthroughs or accelerated research investment.

Although there is uncertainty in predicting the timing of practical application, the threat of "Harvest Now, Decrypt Later" (HNDL) attacks is pointed out as a reason why the consideration of cryptographic migration cannot be postponed. HNDL is an attack scenario where an attacker collects and stores currently encrypted communication or stored data, and decrypts it later once quantum computers become practical. Even data that is securely protected today carries the risk of losing its confidentiality

10 or 20 years from now.

Customer asset information, transaction histories, and personal information handled by financial institutions are data that must maintain confidentiality over a long period, making them prime targets for HNDL attacks. Therefore, the timing of PQC migration must be judged based not only on "when quantum computers will emerge" but also on the following relationship:

Period of information confidentiality + Period required for system migration ≤ Period until the emergence of quantum computers

Example:

- Information requiring confidentiality for 10 years or more
- System migration takes about 5 years
- Assuming a quantum computer emerges in 15 years

In the above example, 10 years + 5 years = 15 years, indicating that the organization is already at the stage where it should start considering and preparing for migration.

2.2.2 Positioning of Migration Targets in International Guidelines

Given the risks posed by quantum computers, government agencies and standardization bodies in various countries and regions have begun to indicate directions for migrating to PQC. This section outlines the major trends in Japan and overseas.

Trends in Japan

In Japan, discussions toward PQC migration anticipating the quantum computer era have become active, mainly among government agencies and the financial industry, starting around 2025.

(1) Cross-Government Initiatives

The National Cybersecurity Office (NCO) established the "Liaison Conference of Relevant Ministries and Agencies on the Use of Post-Quantum Cryptography (PQC) in Government Agencies, etc."[i] on June 30, 2025, to conduct specific studies on PQC migration. On November 20, 2025, the "Interim Report on the Migration to Post-Quantum Cryptography (PQC) in Government Agencies, etc." was compiled and published.

The interim report states that the migration to PQC in government agencies, etc., should aim to be completed by 2035 in principle, and that a roadmap for migration will be formulated in FY2026 in cooperation with relevant ministries and agencies to promote smooth PQC migration in Japan.

(2) Trends of the Financial Services Agency (FSA)

The FSA held the "Study Group on the Response of Deposit-Taking Financial Institutions to Post-Quantum Cryptography"[ii] in July 2024 and published its report in November 2024.

The report points out that ensuring the confidentiality and integrity of information handled by financial institutions, such as customer and transaction data, is essential, and that the degradation of cryptographic security by quantum computers could have a serious impact on the reliability of the entire financial system.

Based on this recognition, the report indicates the importance of financial institutions understanding their own information assets and cryptographic usage status as a preliminary preparation for PQC migration. Specifically, it states that it is necessary to create a cryptographic inventory organizing the cryptographic algorithms and key lengths used in each scenario, such as communication, data storage, digital signatures, and authentication, and to consider responses based on this.

Furthermore, it states that when adapting to PQC, rather than migrating all systems uniformly, consideration based on the importance and impact of the information is required. In addition, since PQC is a new technology, the report concludes that it is important to proceed with responses in collaboration with the government/supervisory authorities, industry associations, vendors, etc., while taking into account future standardization trends and technological advancements.

(3) Trends of Other Ministries

PQC affects multiple layers, including not only the selection of cryptographic algorithms but also networks, authentication infrastructure, terminals/devices, operations, and procurement. Therefore, in Japan, initiatives are progressing in a way that connects policy, research and development, and evaluation aspects in parallel with cross-government studies.

For example, the Ministry of Internal Affairs and Communications (MIC) has presented explanatory materials including the positioning of PQC and research and development (activities of NICT), which are also related to evaluation activities in CRYPTREC.

(4) CRYPTREC Policy

CRYPTREC is a framework that conducts security evaluation and monitoring of cryptographic technologies used in e-government, etc., and investigates and studies appropriate implementation and operation. It is also proceeding with the development of guidelines and the organization of

research trends regarding PQC.

In April 2025, the "CRYPTREC Cryptographic Technology Guideline - Post-Quantum Cryptography - 2024 Edition"[iii] and the "Technical Report on Post-Quantum Cryptography"[iv] were published as CRYPTREC's FY2024 deliverables.

The cross-government interim report also clearly states the reference to CRYPTREC guidelines in migration studies, and mentions the intention to proceed with security and implementation performance evaluations of cryptographic methods published as NIST standards (e.g., FIPS 203/204/205) and aim to update the CRYPTREC cipher list.

Trends in the United States

In the United States, the National Institute of Standards and Technology (NIST) has led the standardization of PQC algorithms.

As a result of an international competition that began in 2016, the main algorithms for key encapsulation mechanisms and digital signature schemes were officially published as FIPS standards (FIPS 203/204/205) in August 2024.[v] With this, PQC is positioned to have transitioned from the research stage to the implementation and introduction stage.

Meanwhile, the National Security Agency (NSA) published the CNSA Suite 2.0 (Commercial National Security Algorithm Suite 2.0)[vi] as a cryptographic suite for national security systems, indicating the migration policy and timeline to quantum-resistant algorithms. CNSA 2.0 envisions completing the migration from conventional RSA and elliptic curve cryptography to PQC by around 2030, and PQC compliance is becoming a prerequisite for new systems and procurement.

Furthermore, the Cybersecurity and Infrastructure Security Agency (CISA) has published guidance and roadmaps to support PQC migration for government agencies and critical infrastructure operators.[vii] It features content directly linked to practical operations, such as developing cryptographic inventories, prioritizing based on impact, and formulating phased migration plans, emphasizing the need to treat PQC migration as a long-term management issue.

Thus, in the United States, the government is taking the lead in clearly setting forth the necessity and urgency of PQC migration for the entire society, including the private sector.

Trends in Europe

In Europe, the European Union Agency for Cybersecurity (ENISA) has taken the lead, publishing

a report as early as 2021 that organized the technical and operational challenges for the standardization process regarding PQC introduction.[viii] ENISA has indicated an approach that anticipates preparation and implementation starting in the mid-2020s and the completion of migration for critical infrastructure by the early 2030s.[ix] Following these studies, the European Commission published recommendations and a roadmap for PQC migration in 2024,[x] and ENISA is positioned as the core agency supporting technical studies, playing a role in promoting coordinated PQC migration across the EU.

In addition, the UK's National Cyber Security Centre (NCSC) also recommends that critical national infrastructure operators formulate early migration plans. The NCSC presents clear phasing for PQC migration (investigation/preparation, partial migration, full migration) and indicates specific milestones such as 2028, 2031, and 2035.[xi] This emphasizes that PQC migration is inseparable from medium- to long-term IT strategies and system renewal plans.

Implications from Overseas Trends

From the trends in overseas legal systems and guidelines, the following points can be read as common messages:

- PQC is positioned not as a technology to be considered in the future, but as a prerequisite technology for which migration preparations should already be starting.
- In parallel with standardization, policy demands accompanied by migration deadlines and procurement policies are advancing.
- Since migration takes a long time, it is important to start developing cryptographic inventories and conducting proof-of-concept experiments early.

These points indicate that financial institutions in Japan also need to position this as part of their medium- to long-term system strategies and risk management.

2.2.3 The Concept of Phased Cryptographic Migration

Cryptographic technology has a broad impact on the entire system, including communication protocols, authentication infrastructure, key management systems, application implementations, and operational procedures. Therefore, it is not realistic to implement a simultaneous migration to PQC in a short period.

Many guidelines assume a phased migration approach. This is a realistic response considering technical complexity, compatibility issues, operational load, and costs.

Generally, migration proceeds through four main stages.

Stage	Examples of Actions
Preparation Stage	Create a cryptographic inventory (understand where and what kind of cryptography is used) and conduct risk assessments. Establishing an internal organizational structure and securing a budget are also important at this stage.
Planning Stage	Determine the priority of migration targets based on the risk assessment, and formulate specific migration methods (e.g., whether to adopt a hybrid approach) and a roadmap. A hybrid approach is a method that uses both conventional cryptography and PQC together to balance security and compatibility during the migration period.
Migration Stage	Sequentially migrate to PQC based on priorities. When migrating, consider crypto-agility to prepare for future cryptographic updates.
Completion Stage	After the migration of all systems is complete, phase out conventional public-key cryptography. Also, continue to monitor trends regularly after migration, and if a cryptographic algorithm is compromised, respond by migrating to another cryptographic algorithm

Table 2: Phased Migration to PQC

Specific processes for realizing such a phased migration (how to develop a cryptographic inventory, risk assessment methods, how to formulate a roadmap, etc.) will be detailed in Chapter 4.

2.2.4 Example of a Timeline

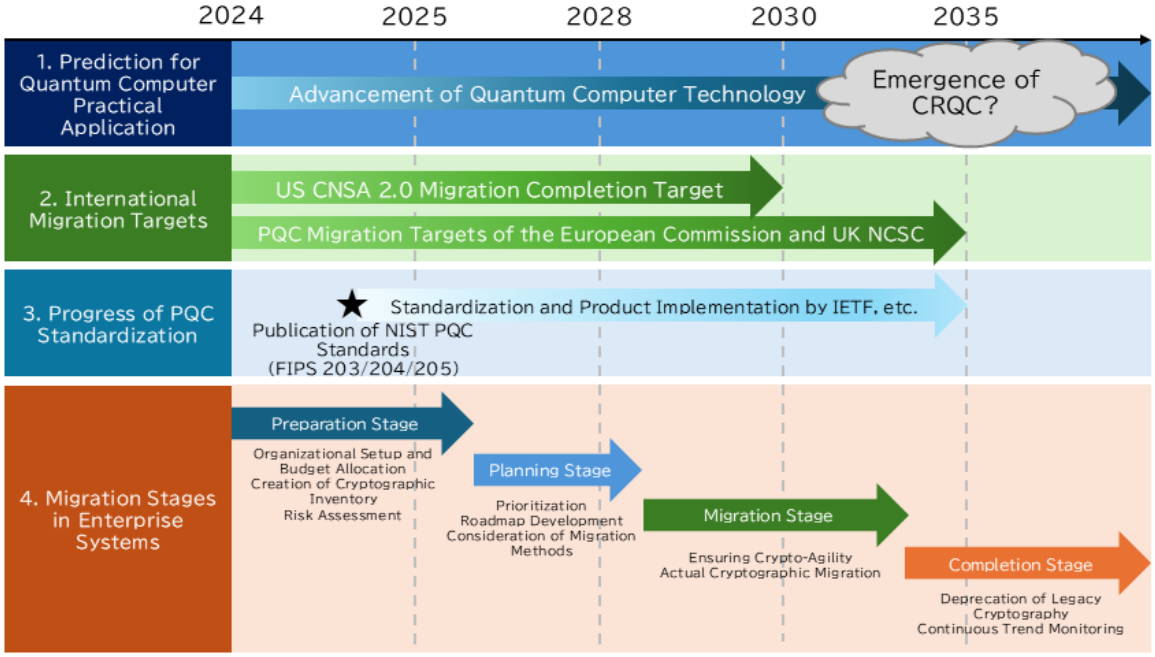

Figure 1: Example of Cryptographic Migration Timeline

Figure 1 organizes the elements discussed in this section chronologically. Time is on the horizontal axis, and the following four elements are placed on the vertical axis:

1. Predictions for quantum computer practical application: The range of time when the emergence of CRQC is predicted.
2. International migration targets: CNSA 2.0 (2030) and recommended timings by the European Commission and NCSC, etc.
3. Progress of PQC standardization: Completion of NIST standardization (2024) and future developments such as IETF standardization.
4. Migration stages in enterprise systems: The stages of Preparation -> Planning -> Migration -> Completion.

The important points to read from this figure are:

- There is little time left until international migration targets (2030-2035).
- Since migration takes a long time, starting preparations early is essential.

However, this timeline is a general guideline illustrated from various information, and adjustments are necessary according to the situation of each organization (system scale, confidentiality period of handled information, risk tolerance, etc.).

2.3 Trends in NIST Standardization

In the practical application of Post-Quantum Cryptography, the standardization by the National Institute of Standards and Technology (NIST) in the United States serves as a global guideline. Although NIST is a US government agency, NIST standards are adopted widely around the world, playing the role of a de facto international standard.

In December 2016, NIST initiated a Call for Proposals[xii] for new cryptographic algorithms in preparation for the quantum computer era. 82 proposals were gathered from around the world, and evaluations were conducted from three perspectives: security, cost (key size/computational cost), and algorithm/implementation characteristics.

The evaluation was conducted in a multi-round system, narrowing down candidates in each round, and the selection of cryptographic algorithms took place over about 8 years.

In August 2024, NIST officially published the first PQC standards.[xiii] The following three algorithms were standardized as Federal Information Processing Standards (FIPS) for the US government.

FIPS Standard	Algorithm Name	Purpose	Base Algorithm	Mathematical Basis
FIPS 203	ML-KEM	Key Exchange	CRYSTALS-Kyber	Lattice problem
FIPS 204	ML-DSA	Digital Signature	CRYSTALS-Dilithium	Lattice problem
FIPS 205	SLH-DSA	Digital Signature	SPHINCS+	Hash function

Table 3: NIST Standardized PQC Algorithms

Furthermore, to improve performance and implementability for these algorithms, and to ensure security in case vulnerabilities are found, NIST is continuing to standardize additional algorithms. FN-DSA is a digital signature based on lattice problems, but it features smaller sizes, such as signature length, compared to ML-DSA. The FIPS draft of FN-DSA has been created by NIST, and procedures toward publication are underway. For digital signatures, a new call for proposals was conducted to further improve size and performance and expand security diversity, and as of 2025, it has progressed to the second round. HQC is a code-based key encapsulation mechanism; it was selected for standardization in March 2025, and the process toward standardization is expected to proceed in the future.[xiv]

FIPS Standard	Algorithm Name	Purpose	Base Algorithm	Mathematical Basis
FIPS 206	FN-DSA	Digital Signature	Falcon	Lattice problem
(Undecided) FIPS 207?	(Undecided) HQC-KEM?	Key Exchange	HQC	Code-based problem

Table 4: PQC Algorithms to be Standardized by NIST in the Future

Importantly, NIST's FIPS only defines the specifications of the cryptographic algorithms. How they are actually used in systems is specified separately. For example, how to use ML-KEM in TLS communication is specified by RFCs defined by the IETF (Internet Engineering Task Force). How to handle ML-DSA signatures in X.509 certificates also requires separate standardization work. The standardization movements in the IETF will be explained in the column in Chapter 3.

3. Proof of Concept Conducted by Daiwa Securities Group

As stated in Chapter 1, the threat that requires early countermeasures against future compromise of cryptography by quantum computers is the HNDL (Harvest Now, Decrypt Later) attack, which is a threat to confidentiality. For this reason, among PQC (Post-Quantum Cryptography), key exchange is generally considered for migration ahead of digital signatures. In Internet Standards, the hybrid key exchange of X25519 and ML-KEM 768 has been adopted as the PQC standard. Major browsers already support this hybrid key exchange, and PQC application has begun where servers also support it. According to a survey by NICT[1], as of September 2025, 12% of accesses from NICT to external servers supported key exchange using X25519MLKEM768,[2] and this percentage is expected to have increased by now.

This proof of concept (PoC) was conducted using a basic verification environment built in the cloud and the development environment of Daiwa Securities' online services, which serves as a setting close to an actual system. The former conducted TLS performance verification in a simple client-server configuration, while the latter conducted TLS performance verification and load testing using a configuration with an F5 load balancer placed between the client and the business server as the verification environment. The target load balancer supports X25519MLKEM768 key exchange via software implementation.

Below, this chapter reports the results of the basic verification regarding key exchange, which is the first step of PQC migration in many cases, in Section 3.1, and the verification results using the online service development environment in Section 3.2. Section 3.3 reports on the basic verification of digital signatures for TLS authentication (certificates), which will be the next step in PQC migration.

[1] National Institute of Information and Communications Technology
[2] NICTER Blog, “Survey on PQC Usage on LiveNet”, https://blog.nicter.jp/2025/12/pqc-usage/, 4 December 2025

3.1 Basic Verification Using a Verification Environment (Key Exchange Algorithms)

3.1.1 Verification Environment

The terminals used for verification were Amazon EC2 instances. Please refer to the separate report for more detailed specifications. In addition, the following libraries were used to measure individual algorithms:

Library Name	Overview
Cryptography (Python)	Version 46.0.3. Used to measure the performance of RSA, X25519, and P-256.
Liboqs	Version 0.14.0. Used to measure the performance of each post-quantum cryptography algorithm. Also used liboqs-python as a wrapper.

Furthermore, the following libraries were used for specific communications:

Library Name	Overview
OpenSSL	Version 3.5.4. Used as a client and server for TLS communication.
OQS Provider	Version 0.10.1-dev. Used for TLS communication using some post-quantum cryptography.

Note that pwntools and paramiko were also used for the client-side implementation, but they were only used for communications unrelated to the handshake. TLS communication was performed using OpenSSL.

3.1.2 Notes

The versions of each library were the latest versions at the time the experiment started (as of November 2025).

At the time of writing, the latest versions are 46.0.5 for cryptography, 0.15.0 for liboqs, 3.6.1/3.5.5 for OpenSSL, and 0.11.0 for OQS Provider.

Also, although OpenSSL 3.6.0 was released at the time the experiment started, the LTS version 3.5.4 was selected.

3.1.3 Verification Items and Experiment Overview

First, to measure the performance of pure key exchange algorithms, we investigated the time required for key generation, encapsulation, and decapsulation (in the case of X25519, the generation of public keys and shared secrets).

The algorithms targeted for verification were X25519, ML-KEM 512, ML-KEM 768, and ML-KEM 1024. Note that although HQC was selected as a standardization target, it was excluded this time because the draft FIPS has not been published.

For each algorithm, the time taken for key generation, encapsulation, and decapsulation was measured 1,000 times, and calculated based on the data of 999 times, excluding the first time which is affected by overhead.

Next, we investigated the time required to establish a TLS handshake in actual communication. This communication was performed between EC2 instances in two different Availability Zones, with one acting as the client and the other as the server. At this time, the server was made to present the server certificate and certificate authority (CA) certificate prepared for the basic verification.

The key exchange algorithms targeted for verification were X25519, ML-KEM 512, ML-KEM 768, ML-KEM 1024, and X25519MLKEM768. In addition, 2048-bit RSA was used as the signature algorithm for the server and the CA. Note that the server's signature algorithm here refers to the algorithm used by the server for signing in the Certificate Verify. The CA's signature algorithm refers to the signature algorithm attached to the certificate.

The number of handshakes was set to 1,000, and the analysis was performed based on the communication data.

In the results presented later, the communication time is calculated by the difference in time recorded by the client and the server. The EC2 instances running the client and server were synchronized with the same NTP server (169.254.169.123), and the time recorded in the capture file was corrected based on the deviation from the NTP server.

3.1.4 Results

3.1.4.1 Performance Measurement Results

First, the sizes of the public key, private key, ciphertext, and secret are as follows:

	PUBLIC KEY	PRIVATE KEY	CIPHERTEXT	SECRET
X25519	32	32	32	32
ML-KEM 512	800	1632	768	32
ML-KEM 768	1184	2400	1088	32
ML-KEM 1024	1568	3168	1568	32

Table 5: Sizes of Public Key, Private Key, Ciphertext, and Secret (Unit: Bytes)

Regarding the performance of algorithms, the average, standard deviation, quartiles, and robust coefficient of variation of the time taken for key generation were as follows:

	AVERAGE	STANDARD DEVIATION	1ST QUARTILE	MEDIAN	3RD QUARTILE	ROBUST CV
X25519	0.034287	0.002860	0.032659	0.033418	0.034522	0.055748
ML-KEM 512	0.010792	0.001629	0.010579	0.010670	0.010873	0.027554
ML-KEM 768	0.014644	0.002009	0.013911	0.014133	0.014519	0.043055
ML-KEM 1024	0.017122	0.001838	0.016500	0.016808	0.017057	0.033169

Table 6: Average, Standard Deviation, Quartiles, and Robust Coefficient of Variation of Time Taken for Key Generation of Key Exchange Algorithms (Units for average, standard deviation, and quartiles are milliseconds)

Also, the time distribution was as follows:

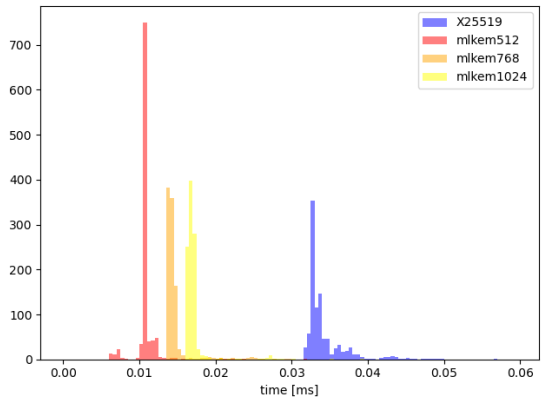

Figure 2: Distribution of Time Taken for Key Generation of Key Exchange Algorithms

Next, the average, standard deviation, quartiles, and robust coefficient of variation of the time taken for encapsulation were as follows:

	AVERAGE	STANDARD DEVIATION	1ST QUARTILE	MEDIAN	3RD QUARTILE	ROBUST CV
X25519	0.064064	0.005161	0.061432	0.061660	0.064597	0.051322
ML-KEM 512	0.012838	0.002468	0.012215	0.012394	0.012560	0.027836
ML-KEM 768	0.015550	0.002440	0.014796	0.014929	0.015205	0.027396
ML-KEM 1024	0.017671	0.001962	0.017026	0.017260	0.017483	0.026506

Table 7: Average, Standard Deviation, Quartiles, and Robust Coefficient of Variation of Time Taken for Encapsulation (Units for average, standard deviation, and quartiles are milliseconds)

Also, the time distribution was as follows:

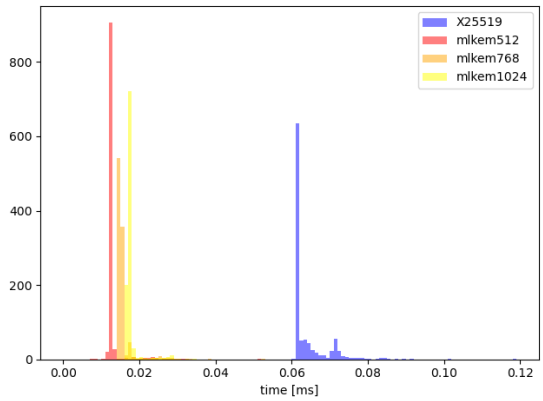

Figure 3: Distribution of Time Taken for Encapsulation

Thirdly, the average, standard deviation, quartiles, and robust coefficient of variation of the time taken for decapsulation were as follows:

	AVERAGE	STANDARD DEVIATION	1ST QUARTILE	MEDIAN	3RD QUARTILE	ROBUST CV
X25519	0.032776	0.003067	0.031598	0.031854	0.032543	0.029651
ML-KEM 512	0.010223	0.002208	0.009601	0.009748	0.010252	0.066680
ML-KEM 768	0.013716	0.002795	0.013071	0.013263	0.013540	0.035399
ML-KEM 1024	0.016848	0.001990	0.016273	0.016511	0.016726	0.027406

Table 8: Average, Standard Deviation, Quartiles, and Robust Coefficient of Variation of Time Taken for Decapsulation (Units for average, standard deviation, and quartiles are milliseconds)

Also, time distribution was as follows:

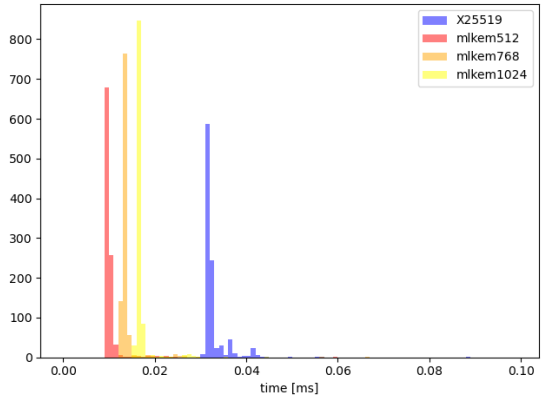

Figure 4: Distribution of Time Taken for Decapsulation

Finally, the average time taken for the handshake is as follows. For the legend of the stacked graph below, please refer to the separate report, and for detailed values, refer to the CSV.

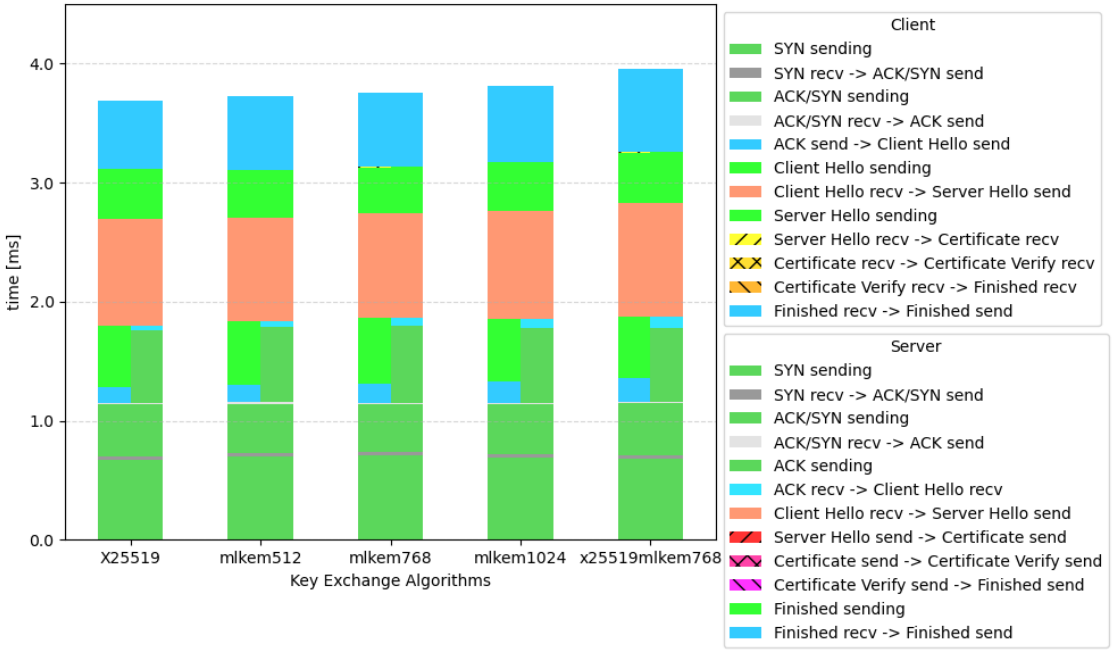

Figure 5: When the Signature Algorithm of the Server and CA is 2048-bit RSA

3.1.4.2 Evaluation of Measurement Results

The average time taken for key generation, encapsulation, and decapsulation is on the order of tens of microseconds. In contrast, the time required to establish a TLS handshake is on the order of milliseconds, so it can be said that the difference in time taken for cryptographic processing is within the margin of error for normal communication purposes.

If we dare to evaluate further, the time taken for key generation and encapsulation of ML-KEM can be said to be faster and more stable than X25519 based on the results of the average and robust coefficient of variation. On the other hand, regarding decapsulation, it is considered that ML-KEM 512 tends to fluctuate upwards based on the shape of the distribution in Figure 4 and the results of the robust coefficient of variation.

In addition, the load of all algorithms is extremely low from the perspective of computational resources such as CPU usage and memory usage, and we believe that the load of cryptographic processing will not be a problem if the specifications are equal to or higher than the terminals used this time.

The sizes of the public key, private key, and ciphertext of ML-KEM are more than 20 times those of X25519.

In TLS handshake applications, the public key is sent in the Client Hello and the ciphertext is sent in the Server Hello to the communication partner, so if the public key and ciphertext are large, the number of packets required for transmission will increase accordingly.

Since the de facto standard for the TCP maximum segment size is 1460 bytes (or 1440 bytes), ML-KEM 1024 exceeds this, increasing the number of TCP packets required for Client Hello and Server Hello. However, two TCP packets were sufficient to transmit the Client Hello and Server Hello.

Finally, the time taken for the entire handshake does not show an extreme difference, being 3.69 milliseconds for X25519 and 3.72 milliseconds for ML-KEM 512. On the other hand, the hybrid X25519MLKEM768 took 3.96 milliseconds, which is a difference of 0.27 milliseconds compared to X25519.

This difference mainly stems from 0.06 milliseconds from sending ACK to sending Client Hello, 0.06 milliseconds from receiving Client Hello to sending Server Hello, and 0.13 milliseconds from receiving Finished to sending Finished.

Note that this cannot be said to be a very large difference for normal use.

3.2 Demonstration Experiment Using the System

3.2.1 Environment Settings

The target system was a system called the Tsumitate Platform (hereinafter referred to as TPF). The rough communication path is as follows:

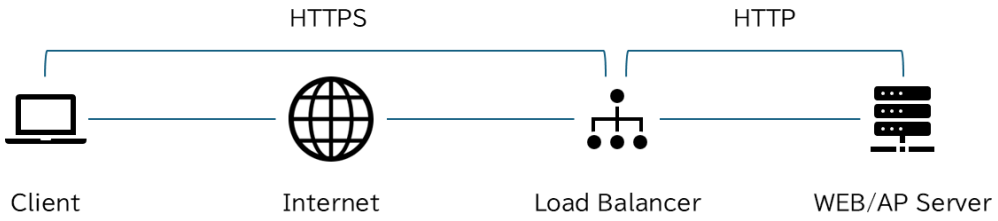

Figure 6: Overview of the system's communication path

The client is the EC2 instance described in “3.1 Basic Verification Using a Verification Environment (Key Exchange Algorithms)” and uses the following software:

OpenSSL	Version 3.5.4. Used as a client for TLS communication.
Nginx	Version 1.28.0. Built using OpenSSL 3.5.4. Used as a reverse proxy for JMeter to specify the key exchange algorithm in the load test.
Apache JMeter	Version 5.6.3. Used in the load test.

The load balancer provided by F5 for the demonstration experiment is as follows:

BIG-IP	Version 17.5.1.3. Supports key exchange with X25519MLKEM768. Note that it does not support digital signatures of post-quantum cryptography such as ML-DSA.

3.2.2 Notes

At the time of writing, Nginx 1.28.1 is the latest version.

Also, the support for post-quantum cryptography in BIG-IP 17.5.1.3 is at the software level. In this version, X25519 is also processed by software.

3.2.3 Verification Items and Overview

In this demonstration experiment, the differences between using X25519 and X25519MLKEM768 were verified for the following items regarding performance:

- Time required to establish a handshake
- Communication delay at the first connection
- Load on the load balancer
- Whether long-term communication (session maintenance) can be performed stably

In addition, a load test was conducted from the perspective of integration with the existing application (WEB server).

Regarding the time required to establish a handshake, a full handshake was performed 1000 times from the client to the load balancer, and the analysis was based on the communication data. Note that 2048-bit RSA is used as the signature algorithm for the server and the certificate authority (DigiCert).

For session maintenance, a session ticket was obtained in advance, and a handshake by session resumption was performed 1000 times, with the analysis based on the communication data.

In addition, the load test was conducted using the same scenario as a load test conducted in the past to investigate the presence or absence of errors caused by changing the cryptography.

3.2.4 Results

3.2.4.1 Establishment of Handshake

First, the average time required to establish a TLS session by a full handshake was as follows:

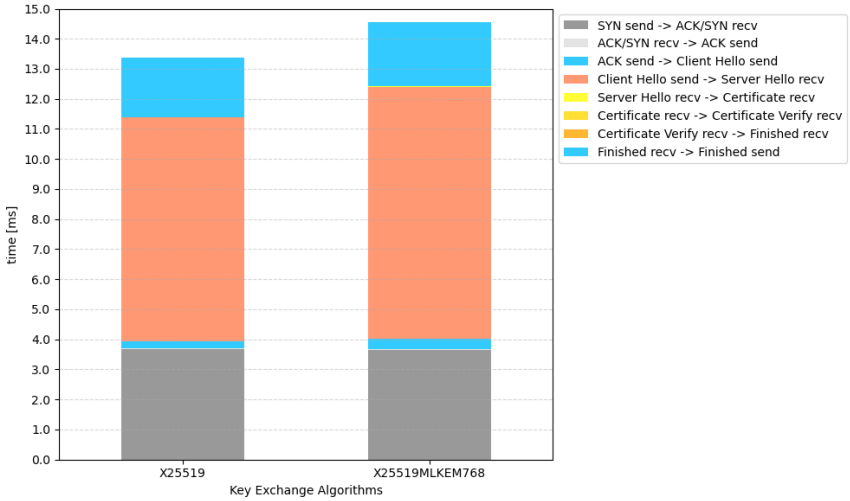

Figure 7: Time required to establish a TLS session by a full handshake

The specific values and standard deviations are as follows:

	X25519		X25519MLKEM768	
SYN SEND -> ACK/SYN RECV	3.667381	(±0.902909)	3.645872	(±1.001504)
ACK/SYN RECV -> ACK SEND	0.029023	(±0.003387)	0.029067	(±0.003430)
ACK SEND -> CLIENT HELLO SEND	0.243639	(±0.212927)	0.342445	(±0.017377)
CLIENT HELLO SEND -> SERVER HELLO RECV	7.440411	(±12.904842)	8.387439	(±13.851253)
SERVER HELLO RECV -> CERTIFICATE RECV	0.003316	(±0.010422)	0.005497	(±0.094986)
CERTIFICATE RECV -> CERTIFICATE VERIFY RECV	0.0	(±0)	0.0	(±0)
CERTIFICATE VERIFY RECV -> FINISHED RECV	0.0	(±0)	0.0	(±0)
FINISHED RECV -> FINISHED SEND	1.988683	(±0.04957)	2.142754	(±0.063122)
TOTAL	13.372453		14.553073	

Table 9: Time required to establish a TLS session by a full handshake (Unit: milliseconds)

Three points can be pointed out regarding this result:

1. The time required to establish a handshake is more than 3.6 times longer compared to the results of the basic verification.
2. The coefficient of variation of the time from sending Client Hello to receiving Server Hello is extremely large, at 1.73 for X25519 and 1.65 for X25519MLKEM768.
3. In X25519, the coefficient of variation of the time from sending ACK to sending Client Hello is relatively large at 0.87.

Since the basic verification was communication between EC2 instances, whereas the communication with TPF was via the Internet, it is possible that communication quality affected these events.

Therefore, we first consider the magnitude of the variation in point 3.

The magnitude of the variation is considered to be caused by outliers. Therefore, we investigated the time taken from sending Client Hello to receiving Server Hello in each handshake.

In many handshakes, the time taken from sending Client Hello to receiving Server Hello was in the range of 4 to 25 milliseconds. In contrast, it was found that all handshakes taking 30 milliseconds or

more were in the range of 90 to 105 milliseconds. Checking the actual communication, TCP retransmissions occurred in these cases. The number of retransmissions was 22 for X25519 and 26 for X25519MLKEM. Hereafter, they are excluded as outliers.

Also, regarding the time from sending ACK to sending Client Hello in X25519, only one case took 6.95 milliseconds. Since the others were 0.5 milliseconds or less, this is also excluded as an outlier.

The average time excluding those with TCP retransmissions and outliers is as follows:

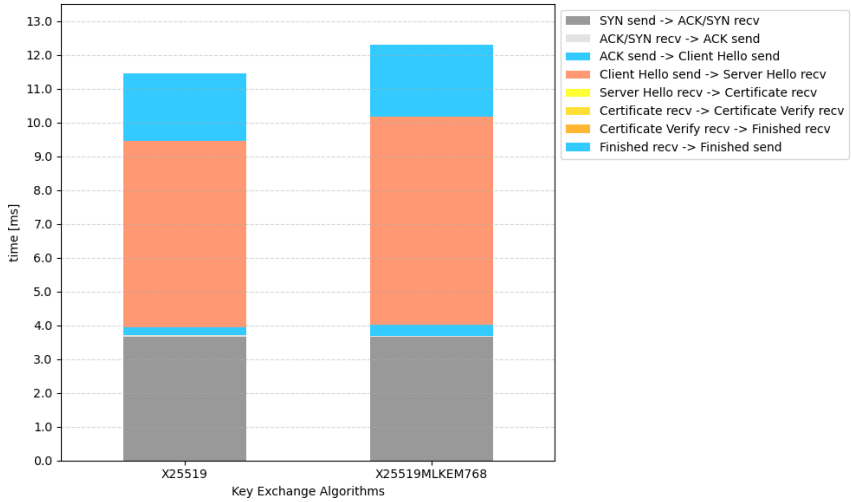

Table 10: Time required to establish a TLS session by a full handshake (excluding outliers)

The specific values and standard deviations of the time from sending Client Hello to receiving Server Hello and the time required for establishment at this time are as follows (Unit: milliseconds):

	X25519		**X25519MLKEM768**	
CLIENT HELLO SEND -> SERVER HELLO RECV	5.515653	(±1.288781)	6.135297	(±1.289622)
TOTAL	11.438157		12.301078	

Table 11: Time required to establish a TLS session by a full handshake (excluding outliers, excerpt) (Unit: milliseconds)

The standard deviation of the time taken from sending Client Hello to receiving Server Hello has become considerably smaller. Regarding this point, first, from the comparison of the coefficient of variation from sending the SYN packet to receiving the ACK/SYN packet, it is inferred that the communication time in the case of communication with TPF is 2 to 3 times more likely to vary compared to the case of basic verification. And the coefficient of variation of the time taken from sending Client Hello to receiving Server Hello in the case of communication with TPF is 2.3 or 2.6 times that of the basic verification. From these, the average and standard deviation after removing the data where TCP retransmission occurred are considered to be reasonable values.

Next, we examine the effect of communication speed by comparing the results after removing TCP retransmissions and outliers with the results of the basic verification. Comparing the average time of each part of the handshake with TPF with the case of basic verification, it is generally in the range of 3 to 3.5 times. And from the time from sending the SYN packet to receiving the ACK/SYN packet, the communication speed is considered to be roughly 2.7 times. Therefore, it seems plausible at first glance that the time required to establish a handshake is 3.1 times longer.

However, the time from receiving Finished to sending Finished is also more than 3.1 times longer. Since there was no evidence of communication in this process, it means that the cryptographic processing also took more time than in the basic verification.

Regarding this point, it is thought that the depth of the certificate chain being one level deeper than in the basic verification, the contents of the server certificate and intermediate CA certificate being more complex, and the increase in the handshake context size subject to signature and authentication in Certificate Verify and Finished due to the increase in certificate size lead to an increase in processing time.

Finally, changing the key exchange algorithm caused a difference of 0.86 milliseconds in the average time required to establish a handshake. This is 3.2 times that of the basic verification.

3.2.4.2 Session Maintenance

Session resumption could be performed without any problems using either algorithm. The average time taken to establish a session by session resumption is as follows:

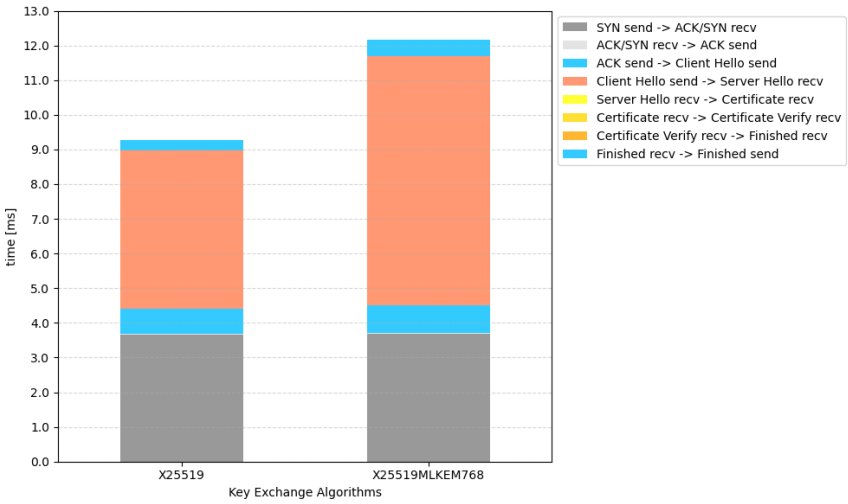

Figure 8: Time taken to establish a session by session resumption

The specific values and standard deviations are as follows (Unit: milliseconds):

	X25519		X25519MLKEM768	
SYN SEND -> ACK/SYN RECV	3.664061	(±0.945719)	3.681401	(±0.880288)
ACK/SYN RECV -> ACK SEND	0.029180	(±0.003419)	0.029436	(±0.003322)
ACK SEND -> CLIENT HELLO SEND	0.702639	(±0.046091)	0.792402	(±0.039641)
CLIENT HELLO SEND -> SERVER HELLO RECV	4.586722	(±7.365329)	7.197391	(±14.812765)
SERVER HELLO RECV -> CERTIFICATE RECV	0	(±0)	0	(±0)
CERTIFICATE RECV -> CERTIFICATE VERIFY RECV	0	(±0)	0	(±0)
CERTIFICATE VERIFY RECV -> FINISHED RECV	0	(±0)	0	(±0)
FINISHED RECV -> FINISHED SEND	0.279162	(±0.134476)	0.453690	(±0.139647)
TOTAL	9.261764		12.154321	

Table 12: Time taken to establish a session by session resumption (Unit: milliseconds)

In X25519 and X25519MLKEM768 during a full handshake before removing TCP retransmissions and outliers, the difference in the average time from sending Client Hello to receiving Server Hello was 1.18 milliseconds. In contrast, it expanded to 2.89 milliseconds in session resumption. In addition, the coefficient of variation of the time taken from receiving Client Hello to receiving Server Hello is also extremely large, at 1.61 for X25519 and 2.06 for X25519MLKEM768.

Therefore, as with the full handshake, we remove the data where TCP retransmissions occurred.

The number of TCP retransmissions that occurred when the key exchange algorithm was 7 for X25519 and 30 for X25519MLKEM768. This difference is considered to appear in the difference in the average.

Also, as mentioned above, the number of TCP retransmissions in a full handshake was 22 for X25519 and 26 for X25519MLKEM768. We consider the validity of the fact that TCP retransmissions actually increased when X25519MLKEM768 was adopted in session resumption, where the communication volume is expected to decrease.

In the range from Client Hello sent by the client to Finished sent by the server, the total size of the TCP payload sent by the client and server and the number of packets sent (estimated) were as follows:

	Full Handshake		Session Resumption	
	X25519	X25519MLKEM768	X25519	X25519MLKEM768
CLIENT	455	1631	998	2174
SERVER	3661	4749	225	1313
TOTAL PACKETS	4	6	2	3
PACKET LOSS RATE	0.55%	0.43%	0.35%	1%

Table 13: Total size of TCP payload sent by client and server (Unit: bytes) and number of packets sent (estimated)

Figure 9 shows when TCP retransmissions occurred during the full handshake and session resumption experiments. Note that handshakes during each experiment were performed every 5 seconds:

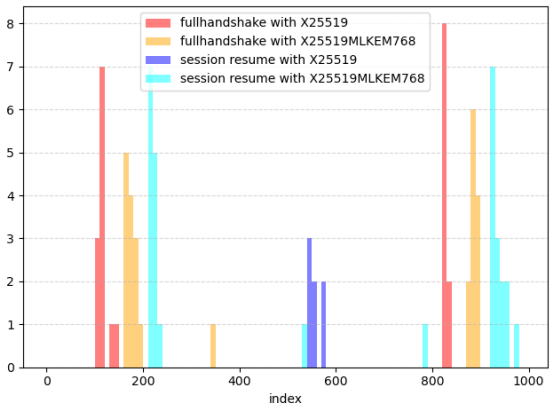

Figure 9: Timing of TCP retransmissions during full handshake and session resumption experiments

As can be seen from Timing of TCP retransmissions during full handshake and session resumption experimentsFigure 9, intensive TCP retransmissions occurred twice in the full handshake with X25519 and X25519MLKEM and the session resumption with X25519MLKEM768. Also, since the intervals between the peaks are about the same, there is a possibility that they occur periodically. Assuming periodicity, the number of TCP retransmissions in session resumption with X25519 might have doubled depending on the timing of the experiment. Furthermore, it is suggested that the session resumption with X25519MLKEM768 swung upward at the second peak.

From these facts, the 30 TCP retransmissions in the case of a handshake by session resumption and the key exchange algorithm being X25519MLKEM768 is somewhat high compared to other experimental results, but it seems to be within a possible range considering that it is communication via the Internet.

When the effect of TCP retransmissions is removed as in the full handshake, the average time was as follows:

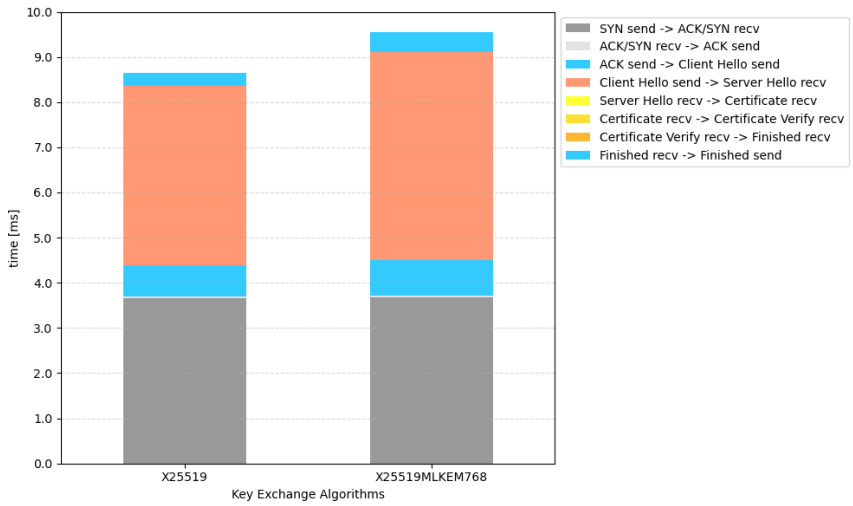

Figure 10: Time taken to establish a TLS session by session resumption (excluding outliers)

The time taken for each part (excerpt) at this time is as follows:

	X25519		X25519MLKEM768	
ACK SEND -> CLIENT HELLO SEND	0.702700	(±0.046217)	0.792405	(±0.040093)
CLIENT HELLO SEND -> SERVER HELLO RECV	3.973420	(±0.914449)	4.604470	(±1.364555)
FINISHED RECV -> FINISHED SEND	0.278153	(±0.134399)	0.447444	(±0.137059)
TOTAL	8.646324		9.557866	

Table 14: Time taken to establish a TLS session by session resumption (excluding outliers, excerpt) (Unit: milliseconds)

Unlike the full handshake, the time taken from sending the ACK packet to sending Client Hello has increased. On the other hand, the time from sending Client Hello to receiving Server Hello and the time from receiving Finished from the server to the client returning Finished have been decreased.

Since the key exchange itself is performed, the increase in time due to hybridization cannot be suppressed even if session resumption is used. However, whichever key exchange algorithm is selected, it can be said that the handshake as a whole can benefit from the shortening by session resumption to the same extent.

3.2.4.3 Load Test Results

The load test for this system was conducted with a scenario of logging in and performing certain operations on the web application. The degree of load ranged from what the system assumes to several times that amount.

Under high load, there were situations where the CPU usage rate on the application server side was close to 100%, but even in such situations, the CPU usage rate of the load balancer never exceeded 8%. The specific resource data is as follows.

First, the specific CPU usage rates during and outside the experiment time on the load test day were as follows:

	X25519	X25519MLKEM768	OUTSIDE EXPERIMENT TIME
AVERAGE	6.69	6.63	6.81
MINIMUM	5.64	5.94	4.75
1ST QUARTILE	6.54	6.42	6.64
MEDIAN	6.83	6.63	6.85
3RD QUARTILE	6.88	6.85	7.07
MAXIMUM	7.38	7.23	7.74
STANDARD DEVIATION	0.47	0.41	0.43

Table 15: CPU usage rate of the load balancer during and outside the experiment time

Also, the TMM memory usage rate was as follows:

	X25519	X25519MLKEM768	OUTSIDE EXPERIMENT TIME
AVERAGE	6.89	6.89	6.89
MINIMUM	6.89	6.89	6.89
1ST QUARTILE	6.89	6.89	6.89
MEDIAN	6.89	6.89	6.89
3RD QUARTILE	6.89	6.89	6.89
MAXIMUM	6.90	6.90	6.90
STANDARD DEVIATION	0.004	0.004	0.002

Table 16: TMM memory usage rate of the load balancer during and outside the experiment time

In terms of memory and CPU usage rates, the results showed that there was no significant difference not only between X25519 and X25519MLKEM768 but also compared to the situation outside the experiment time where the handshake itself was not performed. Note that the swap area was not used.

From these results, it can be said that under high load, the system side reaches its limit first, so switching to post-quantum cryptography does not make the load balancer a bottleneck. In addition, it is considered that the performance difference between X25519 and X25519MLKEM768 does not become apparent under such circumstances.

Also, since the load test could be executed without any problems as long as it was within the load range assumed by the system, it was judged that the service would not stop functioning due to post-quantum cryptography.

Next, comparing Total Bytes In/Out and Total Throughput based on the resource data of the load balancer in terms of communication volume, changing from X25519 to X25519MLKEM768 resulted in an increase of about 5%.

However, during the load test, only the information necessary to execute the scenario was sent and received in JSON format, and other resources such as images necessary in practice were not acquired. Therefore, the increase is expected to be even lower for practical communication.

3.3 Basic Verification Using a Verification Environment (Signature Algorithms)

3.3.1 Verification Environment

The environment is the same as the one used for the key exchange algorithms.

3.3.2 Verification Items

First, to measure the performance of pure signature algorithms, we investigated the time required for key generation, signature generation, and signature verification.

The algorithms targeted for verification were 2048-bit RSA, 3072-bit RSA, P-256, and ML-DSA 44. Note that Falcon was excluded this time because, although it was selected as a standardization candidate, the final version of FIPS had not been approved as of January 2026. SLH-DSA was also excluded as it is not suitable for regular use in terms of performance.

For each algorithm, the time taken for key generation, signature generation, and signature verification was measured 1,000 times, and the results were calculated based on 999 data points, excluding the first time which is affected by overhead.

Next, we investigated the time required to establish a TLS handshake in actual communication. This communication was conducted between EC2 instances in two different availability zones, with one acting as the client and the other as the server. At this time, the server was made to present the server certificate and certificate authority (CA) certificate prepared for the basic verification.

The key exchange algorithms targeted for verification were X25519, ML-KEM 512, ML-KEM 768, ML-KEM 1024, and X25519MLKEM768. In addition, 2048-bit RSA, 3072-bit RSA, P-256, and ML-DSA 44 were used as the signature algorithms for the server and the CA. Note that the server's signature algorithm here refers to the algorithm used by the server for signing in the Certificate Verify message. The CA's signature algorithm refers to the signature algorithm attached to the certificate.

The number of handshakes was set to 1,000, and the analysis was performed based on the communication data.

In the results presented later, the communication time is calculated by the difference between the times recorded by the client and the server. The EC2 instances running the client and server were synchronized with the same NTP server (169.254.169.123), and the times recorded in the capture files were corrected based on the deviation from the NTP server.

For reference, we also include some results when MAYO, CROSS, UOV, and SNOVA, which are currently being considered as standardization candidates, were adopted as the server's signature algorithm.

3.3.3 Results

3.3.3.1 Performance Measurement Results

First, the sizes of the keys and signatures are as follows. For those with a range, it depends on the actual measured values in this basic verification:

	Public Key	Secret Key	Signature
2048-BIT RSA	294	1213 - 1219	256
3072-BIT RSA	422	1791 - 1795	384
P-256	91	138	69 - 72
ML-DSA 44	1312	2560	2420

Table 17: Sizes of Verification Key, Signature Key, and Signature (in bytes)

Next, regarding the performance of the pure algorithms, the average, standard deviation, quartiles, and robust coefficient of variation of the time taken for key generation were as follows. For the distribution of time, please refer to the separate report:

	Average	Standard Deviation	1st Quartile	Median	3rd Quartile	Robust CV
2048-BIT RSA	40.100951	24.470661	22.610428	34.956447	50.986659	0.811760
3072-BIT RSA	141.447273	87.240103	75.312573	121.211471	185.203632	0.906606
P-256	0.035991	0.003434	0.034602	0.034802	0.035105	0.014453
ML-DSA 44	0.021171	0.002716	0.020317	0.020650	0.020974	0.031840

Table 18: Average, standard deviation, quartiles, and robust CV of the time taken for key generation of signature algorithms (units for average, standard deviation, and quartiles are milliseconds)

Thirdly, the average, standard deviation, quartiles, and robust coefficient of variation of the time taken for signature generation were as follows. For the distribution of time, please refer to the separate report:

	Average	Standard Deviation	1st Quartile	Median	3rd Quartile	Robust CV
2048-BIT RSA	0.284450	0.046486	0.263387	0.268292	0.285657	0.083008
3072-BIT RSA	0.743106	0.094973	0.698908	0.703430	0.754942	0.079658
P-256	0.027631	0.003128	0.026358	0.026473	0.027143	0.029653
ML-DSA 44	0.056068	0.029422	0.035173	0.048201	0.069191	0.705732

Table 19: Average, standard deviation, quartiles, and robust CV of the time taken for signature generation (units for average, standard deviation, and quartiles are milliseconds)

Finally, the average, standard deviation, quartiles, and robust coefficient of variation of the time taken for verification were as follows. For the distribution of time, please refer to the separate report:

	Average	Standard Deviation	1st Quartile	Median	3rd Quartile	Robust CV
2048-BIT RSA	0.037349	0.003374	0.035497	0.036482	0.037973	0.067869
3072-BIT RSA	0.057842	0.004449	0.055404	0.055832	0.057474	0.037067
P-256	0.080259	0.005438	0.076950	0.078023	0.080582	0.046544
ML-DSA 44	0.021976	0.001927	0.021339	0.021473	0.021886	0.025451

Table 20: Average, standard deviation, quartiles, and robust CV of the time taken for signature verification (units for average, standard deviation, and quartiles are milliseconds)

For the handshake results, please refer to the separate report and CSV.

3.3.3.2 Evaluation of Measurement Results

Regarding the time required for key generation, it can be seen that RSA has a considerably larger average and robust coefficient of variation compared to the others. That is, generation takes time, and that time has a relatively wide range. However, since keys are not generated for every TLS handshake at present, this is not considered a significant problem from the perspective of TLS communication.

The time taken for signature generation and verification affects the time taken for the server to generate the Certificate Verify and for the client to verify the Certificate/Certificate Verify.

The time taken for signature generation with RSA is more than 10 times that of P-256, and more than 5 times that of ML-DSA 44. However, since it is less than 1 millisecond, it cannot be said to be fatally slow for the client unless there is a demand to establish communication extremely quickly. On the other hand, for the server, changing the signature algorithm can save computational resources, albeit slightly. If handshakes are performed frequently, there is room for consideration.

Also, from the shape of the distribution and the results of the robust coefficient of variation, it is considered that the time taken for signature generation with ML-DSA 44 tends to fluctuate upwards slightly.

The time taken for signature verification is the slowest for P-256, but it is on the order of tens of microseconds, and the robust coefficient of variation is not that large, so it can be said that there is no significant difference regardless of which algorithm is selected for normal use.

If we dare to evaluate it further, ML-DSA 44 can be said to be faster and more stable than conventional cryptography based on the results of the average and robust coefficient of variation.

Since the public key is included in the server certificate and the signature is sent in the Certificate Verify, if these sizes are large, the number of TCP packets sent by the server will increase accordingly.

Regarding the number of TCP packets used by the server to send from Server Hello to Finished in the basic verification, an excerpt of the results is as follows.

Note that these values depends not only on the signature algorithm but also on the contents of the certificate and the presence or absence of intermediate certificates. For the size of the certificates used in the basic verification, please refer to the separate report.

CA	**2048-BIT RSA**			**ML-DSA 44**	
Server	2048-bit RSA	3072-bit RSA	ML-DSA 44	2048-bit RSA	ML-DSA 44
X25519	2	2	5	6	9
ML-KEM 512	3	3	5	7	9
ML-KEM 768	3	3	5	8	9
ML-KEM 1024	3	4	6	8	10
X25519MLKEM768	3	3	5	8	9

Table 21: Number of TCP packets used to send from Server Hello to Finished (Excerpt)

If the communication quality is not ideal, TCP retransmissions may occur. If the number of TCP packets increases, the possibility of TCP retransmissions occurring during the handshake increases. If the communication after the handshake is established is lightweight and you want to keep latency as low as possible, you need to be careful. In addition, it is necessary to pay attention to the limitation of the congestion window.

Regarding the time required to establish a handshake, we reached the following conclusions:

1. Using the same signature algorithm for the server and the CA makes the handshake establishment faster.
2. An increase in the total size of the ciphertext sent in the Server Hello and the certificate sent in the Certificate may actually make the handshake establishment faster in some cases.
3. Comparing the use of 3072-bit RSA and ML-DSA 44 as the server's signature algorithm, using ML-DSA 44 results in faster handshake establishment.[3]

Note that these are conclusions under an ideal communication environment of communication between EC2 instances. When routing through the Internet, there is a possibility of packet loss, so the difference between ML-DSA 44 and 3072-bit RSA could be reversed depending on the packet loss rate and congestion control.

[3] This comparison shows the results when the certificate authority uses P-256 as its signing algorithm and when it uses the same signing algorithm as the server.

3.4 Discussion

In this section, based on the measurement results obtained in Sections 3.1 to 3.3, we organize the impact of applying PQC to TLS from the perspectives of “performance,” “communication volume,” and “operation/compatibility,” and present three practical implications.

(1) Cryptographic processing time itself is unlikely to become a bottleneck

In the basic verification (Section 3.1), we confirmed that the key generation, encapsulation, and decapsulation of X25519 and ML-KEM (512/768/1024) all take around tens of microseconds, which is sufficiently small compared to the entire TLS handshake (millisecond order). Therefore, in general Web applications, network latency and the presence or absence of retransmissions are more likely to affect perceived performance than the “slowness of cryptographic calculations.”

On the other hand, in the demonstration via the Internet (Section 3.2), it was also confirmed that variations due to communication quality (outliers) and the occurrence of TCP retransmissions can increase the variance of the handshake time. Therefore, in evaluating the introduction of PQC, it is important to observe not only the average value but also the variance/outliers (e.g., 95th/99th percentiles) and the retransmission rate.

However, it should be noted that in environments with limited computational resources, such as IoT devices and embedded systems, the cryptographic processing itself may become a bottleneck.

(2) The main impact of PQC introduction appears in the increase in communication volume

ML-KEM has larger public key and ciphertext sizes compared to X25519 (32 bytes), which increases the data sent in Client Hello/Server Hello during the TLS handshake. In the basic verification, we organized that ML-KEM 1024 and X25519MLKEM768 easily exceed the MSS and can be a factor in increasing the number of packets.

However, in general Web usage, it is thought that in many cases, an increase in packets does not directly lead to fatal problems due to initial window enlargement, etc. On the other hand, in environments with unstable connections, environments with bandwidth limitations, or systems that require low latency and low variance, the increase in packets may push up the retransmission rate, resulting in the manifestation of a “sometimes slow” phenomenon.

(3) Is PQC migration at endpoint devices realistic?

In this demonstration environment, the load balancer (F5 BIG-IP) supported X25519MLKEM768 key exchange, and we were able to verify it without making major modifications to the existing application (Web/AP) side. In addition, no significant increase in the load balancer's CPU usage was observed in the load test, suggesting that it is unlikely to become a bottleneck within the assumed load range.

On the other hand, the device at the time of the demonstration verification did not support PQC signatures such as ML-DSA, and the plan is for the PQC conversion of signatures and certificates to lag behind key exchange. Therefore, practically speaking, a realistic policy is to prioritize key exchange (KEM), use conventional methods (RSA/ECDSA) for signatures and PKI for the time being, and proceed in stages while monitoring the support status.

3.5 Column: Latest Trends in Post-Quantum Cryptography (PQC) Standardization

DigiCert Inc. VP of Industry Standards, Timothy Hollebeek

DigiCert Japan G.K. Senior Product Marketing Manager, Masato Hayashi

— IETF and Public Trust PKI —

While NIST is a US organization primarily responsible for developing cryptographic standards, the Internet Engineering Task Force (IETF) is a volunteer organization that sets common technical standards for the Internet, enabling global interoperability. The IETF is currently discussing and developing Internet implementations of PQC standards. Regarding PQC, discussions are progressing mainly in three working groups: **PQUIP, LAMPS, and TLS**.

Furthermore, regarding the PQC standardization of public PKI, which is provided as a mechanism for an unspecified number of people to prove that “the other party is genuine” on the Internet, encrypt communication, and detect tampering, the CA/B Forum, a voluntary membership organization operated mainly by Certificate Authorities (CAs) that issue digital certificates and browsers that display public TLS certificates, is responsible. Standardization efforts are being carried out through multiple working groups to discuss and develop standards.

For a public CA such as DigiCert to issue certificates using PQC algorithms, RFC standardization of PQC digital signatures at the IETF and additions to the Baseline Requirements at the CA/B Forum are required. Additionally, devices such as FIPS 140-3 compliant HSMs, which NIST (the setter of PQC standards) defines as the security evaluation criteria for cryptographic modules, are necessary. However, anticipating these standardizations, DigiCert is offering test certificates in various environments and use cases.

IETF PQUIP Working Group

PQUIP (Post-Quantum Use In Protocols) provides cross-sectional coordination and guidance for PQC migration within the IETF, organizing common PQC issues,

terminology, and migration knowledge advanced by each protocol WG. It provides visibility into PQC adoption efforts across working groups and acts as a receptacle for areas where no WG exists.

IETF LAMPS Working Group (X.509 / PKI)

The **LAMPS (Limited Additional Mechanisms for PKIX and S/MIME)** Working Group is responsible for standards related to X.509 certificates. The main documents related to PQC are as follows. (The "Internet-Draft" status of an IETF document indicates that a draft has been created, discussions are ongoing during its 6-month validity period, and new additions/corrections are being made. When this discussion matures and consensus is reached, it is published as an RFC standard. Drafts that are not discussed or amended for 6 months expire.)

- **RFC 9881**: ML-DSA Algorithm Identifiers for X.509 (Published)
- **draft-ietf-lamps-x509-slhdsa**: SLH-DSA Algorithm Identifiers for X.509 (IETF Internet-Draft)
- **draft-ietf-lamps-pq-composite-sigs**: Composite ML-DSA Signatures for X.509 (IETF Internet-Draft)
- **draft-ietf-lamps-pq-composite-kem**: Composite ML-KEM for X.509 (IETF Internet-Draft)

IETF TLS Working Group

The TLS Working Group is responsible for technical standards related to TLS. Regarding this standard, hybrid key exchange using **X25519MLKEM768** has already been deployed in production environments by Chrome and Cloudflare (**draft-ietf-tls-ecdhe-mlkem**). This addresses the **"Harvest Now, Decrypt Later"** attack.

- **draft-ietf-tls-mldsa** (Hollebeek, Schmieg, Westerbaan) (IETF Internet-Draft)

  Specifies the authentication method using ML-DSA in TLS 1.3, defines SignatureScheme values for **ML-DSA-44, ML-DSA-65, and ML-DSA-87**, and specifies the signature verification method.

- **draft-reddy-tls-composite-mldsa** (IETF Internet-Draft)

Defines composite ML-DSA authentication for TLS 1.3.

The Debate: Hybrid vs. Pure PQC vs. Dual

One of the major debates currently underway at the IETF is the approach to PQC migration.

- **Pure PQC**: Uses only PQC algorithms.
- **Hybrid/Composite**: Combines legacy cryptography and PQC in a single certificate; security is maintained as long as at least one component remains secure. (defined in RFC 9794).
- **Dual Certificates**: Maintains certificate chains for conventional cryptography and PQC in parallel.

Comparison of Scenarios for PQC Migration

	Pure PQC	Hybrid	Composite	Dual
Structure	1 certificate (Uses only PQC. legacy crypto is excluded.)	1 certificate (legacy crypto key in Subject Public Key Info, PQC key in extension area)	1 certificate (Both legacy and PQC keys in Subject Public Key Info)	2 independent certificates
Main Purpose	Long-term security and efficiency	Resistance to unknown vulnerabilities	Strong identity proof (strict as both cryptos must be checked)	Highest compatibility
Backward Compatibility	None (Non-compliant devices cannot communicate)	Medium to High (Handled via extension area)	Low (Systems must understand the new composite algorithm. Older systems that don't understand will error.)	Highest (Operates as before)
Data Size	Relatively small as it's only PQC	Large (Partially combines 2 sets of keys and signatures)	Largest (Fully combines 2 sets of keys and signatures)	Only the selected 1 set (Efficient)
Implementation Complexity	Low (Simple)	Medium (Increased computational	High (Requires redefinition of	Medium (Server-side management load)

		processing)	standards)	
Challenges	Legacy system incompatibility.	Communication / computational capacity issues. Cutoff of some systems.	Communication / computational capacity issues. Patches for new communication standards and devices. Cutoff of some systems (more than hybrid)	Operational management doubles. Vulnerability to Downgrade Attacks Targeting Weaker Crypto.
Concept	Complete Migration	Defense in Depth	Strict Security	Separation of use, coexistence

In the IETF LAMPS Working Group, **composite specifications** combining PQC and conventional algorithms (RSA, ECDSA, Ed25519, X25519, etc.) are being developed.

However, recommended policies vary by country and region. In some cases, the hybrid method is recommended during the transition period, while in others, such as the US **CNSA 2.0** guidance, **direct migration to pure PQC** is recommended for national security systems.

Furthermore, in the process of standardizing composite signatures, it became clear that systems need to be able to understand the “Subject Public Key Info” of certificates, which did not exist conventionally, more than initially expected. The method of implementing these standardizations and network traffic size changes across the entire Internet stack, and evaluating them, is significantly complex and requires much discussion. On the other hand, verification and trust in pure PQC algorithms are improving. As a result, there are many discussions about direct migration to pure PQC. However, regarding the migration of the Internet, which consists of many devices, it is necessary to minimize cutoffs while ensuring security, so the composite of PQC and RSA/ECC remains the center of migration discussions.

Meanwhile, as a method unaffected by sudden changes in public PKI standards (e.g., migration of financial systems like ATM networks and proprietary financial networks), X9, which sets cryptographic and security standards for the financial industry at ANSI (American National Standards

Institute), is preparing for PQC implementation, establishing and implementing proprietary PKIs that get away from the influence of the every CA/B Forum decision.

Discussions at the CA/B Forum

Discussions on PQC are also spreading to the CA/B Forum, but there are differences in temperature among working groups regarding implementation in the Baseline Requirements.

In the S/MIME and Code Signing Working Groups, ballots regarding PQC have been passed (e.g., SMC-013) or are in the proposal stage, and additions to each Baseline Requirements are actively progressing.

However, regarding additions to the public TLS Baseline Requirements, while it is a well-known fact that experiments with TLS hybrid key exchange have been progressing, it remains limited to key exchange. The reality is that the exit towards realizing true PQC TLS communication that performs signature authentication is not yet in sight. The background to this is cautious opinions due to the magnitude of the impact on public TLS as a whole. For example, if a PQC algorithm is broken earlier than expected or an implementation flaw is found, once it is included as an implementation, it becomes the Achilles' heel of public TLS. Naturally, there are some discussions regarding migration approaches and implementation methods, such as delays in FIPS 140-3 certification for HSMs, smart cards, and USB tokens, and challenges to communication performance due to PQC.

Under the current situation where the shortening of the validity period of TLS certificates, which is currently underway, requires a transition period of several years, the absence of a mechanism for migrating cryptographic algorithms towards PQC is an industry-wide challenge.

Under such conditions, adding PQC to the root programs/root stores of browsers and OSs will become the foundation of trust worldwide, so many prerequisites, such as ensuring technical compatibility, must be met, and it is expected to be a conservative update in the future.

PQC Consideration Status in Other Standards

As a trend that is likely to directly affect the financial industry, preparations are underway in the EU as a multi-layered structure integrating policy, legal regulations, technical standards, and industrial support. While referring to the decisions of the IETF and CA/B Forum, a characteristic is that the binding force is clear.

First, the Quantum Europe Strategy (2025) presented a vision for PQC. Then, the NIS2 Directive (Revised Network and Information Systems Directive) legislated for critical and essential operators,

including financial institutions, and the EU Quantum Act, scheduled to be enacted in 2026, plans to promote the development of technologies, products, and human resources within the region to put it into practice. (Both are directives, so legislation in each EU country is required.) Furthermore, as a technical standard layer to materialize these implementations, ETSI and ENISA define specific implementation requirements, creating a structure that supports the implementation decisions of supervisory authorities and operators.

For example, ETSI defines the PQC compliance of long-term signature formats (AdES: Advanced Electronic Signature) that serve as criteria for judging legal validity, updates to the requirements for Certificate Authorities (QTSP: Qualified Trust Service Provider) responsible for issuing certificates for these signatures, and the PQC compliance of the EUDI Wallet mandated for each country by eIDAS 2.0. ENISA sets the recommended cryptographic methods and technical guidance for this. The EUDI Wallet is strongly tied to financial KYC, so its trends are drawing attention.

Regarding the Latest Information (Reference Links)

We have briefly explained the current discussions as much as possible. The following links can be referred to for future progress. For the IETF, anyone can ask questions or make requests by registering for the mailing list while checking the latest information.

- IETF PQUIP Working Group:

  https://datatracker.ietf.org/wg/pquip/about/

- IETF LAMPS Working Group:

  https://datatracker.ietf.org/wg/lamps/about/

- IETF TLS Working Group:

  https://datatracker.ietf.org/wg/tls/about/

- NIST PQC Project:

  https://csrc.nist.gov/projects/post-quantum-cryptography

- CA/B Forum:

  https://cabforum.org/

- PKI Consortium PQC Resources:

  https://pkic.org/

- ANSI X9 Resources:

  https://x9.org

- Quantum Europe Strategy Resources:

  https://data.europa.eu/en/news-events/news/quantum-europe-strategy-building-future-technology-security-and-trust

4. Approach to Cryptographic Migration

This chapter presents a practical approach for organizations to systematically proceed with migration to Post-Quantum Cryptography (PQC).

4.1 How to Proceed with Cryptographic Migration

Since migration to PQC affects a wide range of areas, including TLS, certificates, signatures, and key management, it is important to establish a promotional structure (governance) as a company-wide initiative spanning multiple years.

Furthermore, PQC migration cannot be handled solely by the IT and security departments. It is desirable to set up a cross-functional team that includes the following roles to facilitate swift decision-making.

Role	Main Responsibilities
General Manager	Head of the team, responsible for internal/external explanations and formulating the migration roadmap.
Promotion Lead	Conducts cryptographic inventory, formulates the migration framework, and manages overall progress.
Infrastructure/Network Lead	Handles cryptographic migration and verification for TLS/VPN/Load Balancers.
App Development Lead	Handles cryptographic migration and verification for cryptographic libraries, signatures, and protocols within applications.
PKI/Key Management Lead	Considers certificate lifecycles and key protection policies.
Procurement Lead	Checks the PQC readiness of the supply chain and reviews contracts/SLAs.
Support Lead	Investigates domestic and international laws/regulations regarding PQC and informs the team.

Table 22: Roles in the PQC Migration Team (Example)

In parallel with setting up a cross-functional team, it is advisable to establish cryptographic policies to be applied to new/upgraded systems at this stage to prevent duplicate efforts.

Area of Consideration	Proposed Policy
Design Principles	Include design requirements for crypto-agility, allowing cryptography to be switchable.
Communication	Adopt a hybrid approach (traditional cryptography + ML-KEM, etc.) to ensure compatibility with existing equipment.
Signatures/Certificates	Establish a policy for adopting PQC for signatures used in long-term code and documents.
Exception Management	If PQC support is difficult during the new/upgrade phase, define and manage deadlines and response policies.

Table 23: Proposed Cryptographic Policies for New/Upgraded Systems

Since cryptography assumes mutual support between the communication source and destination, there may be cases where PQC migration cannot be performed due to the readiness of external contractors, clouds, or common industry infrastructure. Therefore, it is desirable to confirm the PQC support plans and migration policies of relevant vendors and manage them within the cross-functional team.

Because PQC migration takes a long time, costs will be incurred not only for the cryptographic migration itself but also for project promotion, cryptographic inventory, risk assessment, and verification. Therefore, it is advisable to secure an estimated budget for each phase.

Phase	Main Estimation Points
Project Promotion	Coordination with internal and external stakeholders, overall progress management.
Cryptographic Inventory	Investigation costs, man-hours for interviews (include costs for investigation tools if used).
Risk Assessment	Consideration man-hours, formulation of the migration roadmap.
Verification	Performance impact, compatibility, confirmation of operational procedures.
Migration	Equipment upgrades, software updates, cryptographic library updates, testing.

Table 24: Estimation Points by Phase

4.2 Inventory of Cryptographic Assets

The purpose of taking an inventory of cryptography is to understand where cryptography is used and to gather the necessary materials for determining migration targets and prioritization.

At this stage, it is important not to force complete exhaustiveness from the beginning. It is advisable to start with critical areas exposed to the outside, identify undetected areas, and fill them in gradually.

Considering the magnitude of the impact of the emergence of quantum computers, one approach is to prioritize investigating areas where public-key cryptography is used, and then expand the investigation to symmetric-key cryptography used for data storage, etc. At this time, rather than identifying investigation targets haphazardly, it is effective to organize the places where public-key cryptography might be used within the system and prioritize which areas to check first.

Figure 11 summarizes the areas within the system where public-key cryptography is used. For example, you might first check devices and services on the path connecting to the external internet (communication between browsers/apps and CDNs, WAFs, FW, LBs, public servers, authentication infrastructure, etc.), and then expand the investigation to VPN devices, connections to other companies' data centers, cloud connections, and internal communications within your own data center.

Particularly for externally exposed communication paths, functions related to authentication, and connections spanning multiple organizations or locations, it is necessary to keep in mind that the investigation may take time because there will be multiple parties to confirm with.

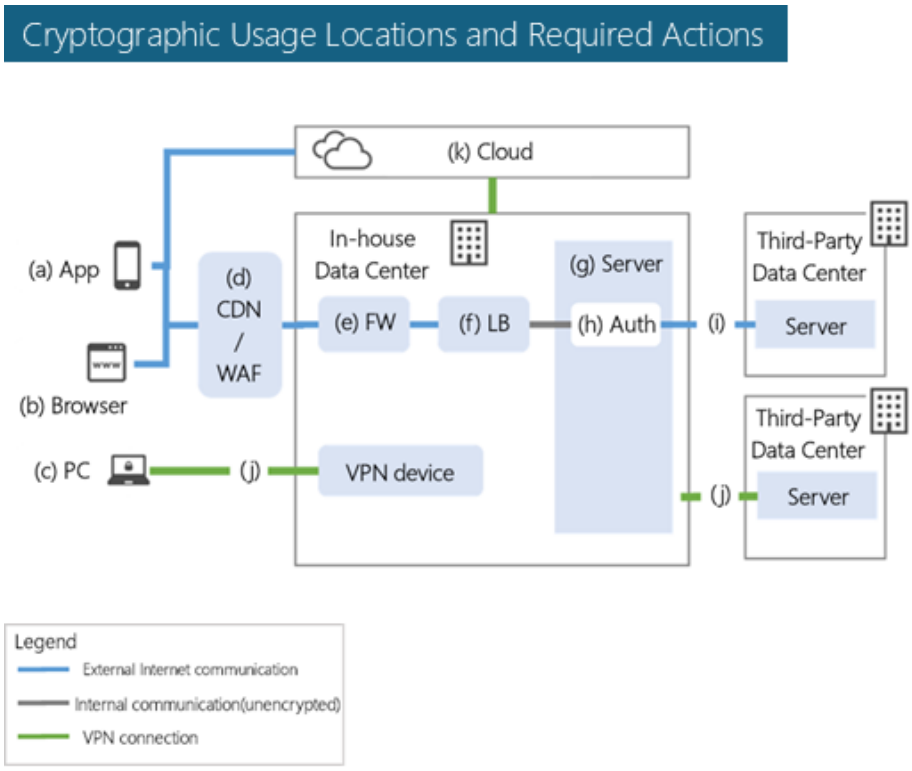

Figure 11: Organizing locations where public-key cryptography is used

When taking an inventory of cryptography, the UK's NCSC recommends taking both top-down and bottom-up approaches. This involves starting with a top-down investigation to grasp the overall picture from business flows and service configurations such as Figure 11, and then conducting a bottom-up analytical investigation from source code, configuration information, certificate deployments, communication logs, packet captures, etc., to complement any omissions in the top-down approach.

Investigation Method	Investigation Perspective	Investigation Details
Top-down	Business flows / Services	Investigate business flows and services from design documents, etc., to find out which paths data flows through and where cryptography is used.
Bottom-up	Source code / Communication logs	Analyze the use of cryptography by investigating application source code or capturing and analyzing communication packets. Dedicated tools exist for collection and analysis.

Table 25: Cryptographic Investigation Methods

4.3 Creation and Management of a Cryptographic Inventory

A cryptographic inventory is a ledger where the results of the investigated cryptographic assets can be continuously updated and referenced.

When creating a cryptographic inventory, it is desirable to create it in a format that the entire team can understand and that includes information that can be used in the subsequent risk assessment and prioritization. Conversely, including a large amount of unnecessary information in the cryptographic inventory can make the investigation difficult or complicate the risk assessment.

The following are items considered to be the minimum necessary to be included in a cryptographic inventory.

Item	Perspective
System/Service Related Info	Name, managing department, person in charge, business system or non-business system?
Location of Cryptography Use	On Web servers, on devices, etc.
Purpose of Cryptography Use	Communication confidentiality, signatures, authentication, data storage.
(For Communication)	Protocol (TLS, IPsec, SSH, etc.), peer information.
Algorithm/Parameters	Algorithm: Public-key cryptography (RSA/ECDSA), Symmetric-key cryptography (AES), etc. Parameters: 2048-bit, etc.
Details of Protected Data	(e.g., Confidentiality of login information for securities accounts, electronic contract data with customers).
Last Updated Date	

Table 26: Elements of a Cryptographic Inventory

To prevent the cryptographic inventory from being neglected or failing to be updated, it is desirable to determine the administrator, update triggers, and inspection cycle of the cryptographic inventory at the beginning. The following is an example.

Item	Example
Administrator	The promotion lead manages and controls it.
Update Triggers	PQC migration, system upgrades, library updates, etc.
Inspection Cycle	At least annually, the managing department of each system checks for any discrepancies in the recorded content.

Table 27: Cryptographic Inventory Management Rules (Example)

Regarding the cryptographic inventory, there are products that can integrate and manage it with a CMDB (Configuration Management Database), and if utilized well, it is considered possible to implement it efficiently, including configuration management. However, if the structure and operations for managing the cryptographic inventory are not established, the introduction of the product itself will become the goal. Therefore, it is considered that you should first think about the structure and operations using traditional tools like spreadsheets, and then consider introducing a product after determining that there are missing points or points that can be made more efficient.

4.4 Risk Assessment and Prioritization

Since it is not realistic to migrate all systems and applications to PQC at once, prioritization is essential. Based on the created cryptographic inventory, consider which functions of which systems and applications should be migrated to PQC first.

Regarding the priority of PQC migration, in addition to the two axes often used in risk assessment: Impact (the magnitude of damage if data is leaked) and Probability (how easy it is for an attacker to collect the data), it is considered good to evaluate from two additional perspectives: Confidentiality Period (the length of time confidentiality must be maintained) and Migration Difficulty (whether it is technically and operationally possible to migrate at present, or whether it cannot be easily migrated due to constraints).

However, items with deadlines set externally, such as regulatory compliance or migration schedules for industry infrastructure (Zengin System, SWIFT, etc.), require prioritized action over the evaluation.

4.5 Creation of a Cryptographic Migration Roadmap

The priorities considered in the previous section must be aligned with annual plans, system upgrade plans, and vendor response plans, and materialized as a roadmap that can be agreed upon and updated by the entire organization.

In January 2026, the G7 Cyber Expert Group published a statement on promoting a coordinated roadmap for the financial sector.[xv] This statement recommends a dual-track approach that sets different timelines for critical and non-critical systems.

4.5.1 Dual-Track Approach and Migration Target Dates

By setting migration target dates based on both the dual-track approach and the risk assessment discussed in the previous section, a method can be considered to prioritize the protection of high-risk systems and realize an efficient migration overall.

Setting Migration Deadlines in the Dual-Track Approach

Based on current guidance from multiple international organizations, the G7 Cyber Expert Group provides the following migration periods as examples depending on the criticality of the system.

Target System	Migration Completion Period
Critical Systems	2030–2032
Non-Critical Systems	2035

Table 28: Examples of Migration Periods (G7 Cyber Expert Group)

The determination of whether a system is critical is made by comprehensively evaluating the impact on business continuity when viewed as a whole system, the impact on the financial system, customer impact, regulatory requirements, etc. For less critical systems, it is conceivable to migrate them after 2032, utilizing the experience gained from the migration of critical systems.

Note that the migration periods and timelines described in this document are reference information compiled from domestic and international public documents and guidance, and do not indicate official confirmed plans or deadlines for our group.

4.5.2 Roadmap Management Structure and Continuous Efforts

Roadmap Management Structure

It is desirable to prepare the following management structure for the roadmap.

Management Structure	Recommendations
Responsible Department	The promotion lead of the cross-functional team established in Section 4.1 manages it comprehensively.
Regular Reviews	Semi-annually or annually (linked with management reports and system upgrade plans).
Ad-hoc Reviews	Respond flexibly to major advancements in quantum technology, major changes in standardization trends, changes in regulatory/industry infrastructure migration schedules, etc.
Reviews with Relevant Depts.	Quarterly or semi-annual reviews within the cross-functional team, reporting to management.

Table 29: Roadmap Management Structure

The roadmap is not fixed once created; it is important to continuously monitor external trends and review it regularly. In particular, since the progress of quantum technology and the status of standardization are difficult to predict even at present, it is desirable to establish a structure that can flexibly adjust the plan.

Continuous Efforts (Activities Supporting the Roadmap)

Even during the migration period, it is desirable to undertake the following efforts to prepare for roadmap reviews.

Continuous Efforts	Recommendations
Governance and Risk Management	Incorporate quantum risks into existing organizational governance. (e.g., Quarterly reports to management, making it a confirmation item in internal audits).
Management of External Dependencies	Monitor trends in quantum technology, standardization, tools, and threats. (e.g., Tracking standardization trends of NIST/IETF, confirming PQC support of major vendors).
Dialogue with Stakeholders	Identify issues, share knowledge, and promote common solutions. (e.g., Information exchange in industry groups (ISAC, etc.), coordinating migration plans with contractors).

Table 30: Continuous Efforts for the Roadmap

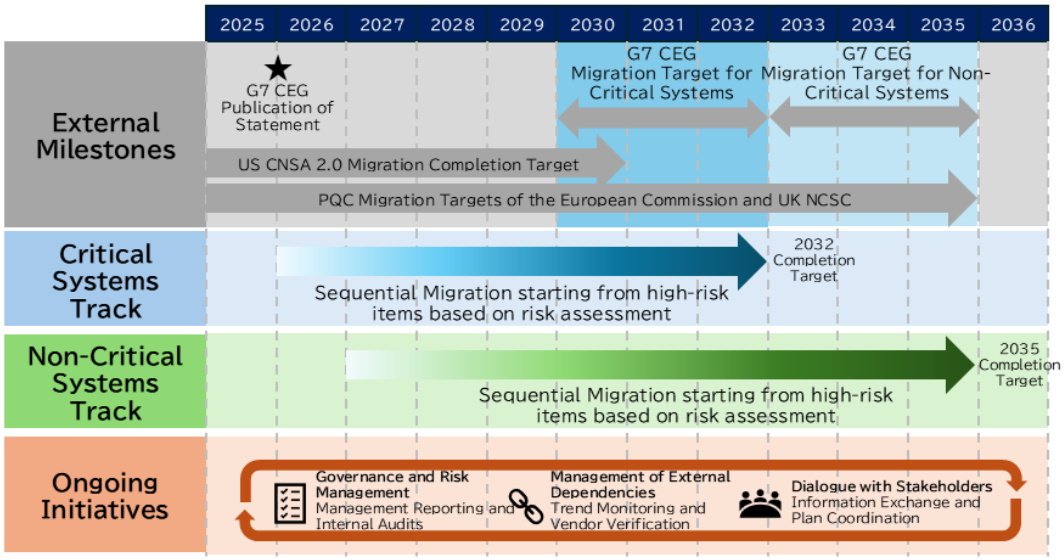

Figure 12: PQC Migration Roadmap

4.6 Consideration of Specific Cryptographic Migration Methods

Up to the previous section, we discussed everything from securing the structure to creating the cryptographic migration roadmap. In this section, we describe the basic thinking for cryptographic migration based on the situation surrounding PQC as of March 2026.

4.6.1 Basic Migration Policy: Prioritize Key Exchange

As an internet implementation trend, PQC support for key exchange (Key Encapsulation Mechanism: KEM) in TLS communication is leading the way. This is because the migration of key exchange is relatively easy and highly effective as an immediate countermeasure against the “Harvest Now, Decrypt Later (HNDL)” risk.

On the other hand, PQC support for digital signatures requires issuance/verification infrastructure (Certificate Authorities, certificate chains, browser/OS trust anchors, audit/operational procedures) and is expected to take time. Furthermore, the representation formats (key formats, etc.) for PQC keys and certificates are still undergoing standardization adjustments. Therefore, a phased approach—**first stage: key exchange, second stage: signatures**—is realistic.

4.6.2 Migration Patterns with Crypto-Agility in Mind

As mentioned in Chapter 2, PQC bases its security on computational security, meaning no efficient algorithm has been discovered at present even for quantum computers. Therefore, an architecture with crypto-agility should be considered, allowing for flexible migration to another PQC in the unlikely event that a vulnerability is discovered in the adopted PQC.

For this reason, when selecting a migration method, it is necessary to evaluate not only “which method to adopt” but also “to what extent the scope of impact of cryptographic changes can be localized (= suppressing the ripple effect of changes).” Specifically, a design that minimizes individual application modifications by consolidating cryptographic scheme changes at the endpoint (edge) or common cryptographic infrastructure (library/runtime) where they can be centrally managed is desirable.

Below, typical migration patterns are organized in descending order of crypto-agility, including perspectives on the minimal impact of changes and centralized management.

Pattern	Overview	Application Scenarios	Evaluation
A. Absorption at the Edge Endpoint	Enable PQC at TLS endpoints such as CDN/LB/WAF.	Web/API endpoints, common paths.	High Crypto-Agility. No application modification required, fast deployment, easy to control the scope of application.
B. Absorption in Common Libraries / Crypto Infrastructure	Centrally manage methods in the organization's common cryptographic infrastructure.	Microservices, API server groups.	High Crypto-Agility. Even in environments where edge termination is difficult, implementation is unified and control is easy to apply.
C. Application Modification in Exception Areas	Incorporate PQC into the application.	Proprietary communications, special requirements, areas that cannot be absorbed at the endpoint.	Low Crypto-Agility. Since it is independent of the endpoint, it can handle special requirements.

Table 31: Crypto-Agility Evaluation by Migration Pattern

4.6.3 Adoption of Hybrid Mode

Hybrid mode is an implementation method during the transition period that uses traditional cryptography (e.g., ECDHE) and PQC (e.g., ML-KEM) simultaneously, maintaining confidentiality even if one of them is broken.

In TLS 1.3, the standardization and implementation of a hybrid mode combining ECDHE and ML-KEM (X25519MLKEM768) are progressing. When hybrid mode is adopted, the cryptography used is selected based on the client's support status.

On the other hand, pure PQC, which uses only PQC, does not have many adoption examples as of March 2026. Therefore, it is realistic to adopt hybrid mode during the transition period and decide whether to use pure PQC in the future.

4.6.4 Supply Chain and Upgrade Strategy

PQC migration affects not only your own organization but the entire supply chain. It is necessary to confirm the support status of third parties from the following perspectives and incorporate them into the roadmap.

Examples of confirmation targets and perspectives are shown below.

Confirmation Target	Confirmation Perspective
Cloud Services	PQC support roadmap, timing of support for TLS termination services.
Software Vendors	PQC support plans for middleware (Web servers, DBs, etc.), license conditions.
Network Equipment Vendors	Firmware update plans for LBs, FW, VPN devices, etc.
Certificate Authorities (CA)	Plans for issuing PQC certificates by public CAs, upgrade plans for private CAs.
Industry Infrastructure	Migration schedules for Zengin System, SWIFT, etc.

Table 32: Confirmation Targets and Perspectives in the Supply Chain

Also, consistency between the migration roadmap and system upgrades is important.

Strategy	Decision Criteria / Action
Address during Upgrade	When the scheduled upgrade comes before the roadmap target.
Address without waiting for Upgrade	When the roadmap target is before the upgrade, or is uncertain.
Exception Management	If migration during an upgrade is difficult from a technical, supply chain, or economic perspective, clearly state the risk on the roadmap.

Table 33: Migration Strategy and Decision Criteria

4.7 Evaluation of Performance and Stability Impact After Migration

To prevent unexpected performance degradation or stability issues from occurring after the introduction of PQC, we organize the perspectives to consider in advance and the performance indicators to check.

4.7.1 Perspectives to Consider in Advance

Compared to traditional cryptography, PQC has the following characteristics.

Characteristic	Reason and Possible Impact
Large Key/Certificate/Signature Sizes	For example, while the public key size of X25519 is 32 bytes, the public key size of ML-KEM is about 1 to 2 kilobytes. This increases communication volume in TLS handshakes, etc.
Difference in Cryptographic Schemes	Due to the change in cryptographic schemes, hardware acceleration that could be used with traditional cryptography may not be usable.
Operational Constraints	PQC in TLS assumes TLS 1.3. If TLS 1.2 or earlier is used, migration to 1.3 is required.

Table 34: Comparison of Characteristics between PQC and Traditional Cryptography

As described in Chapter 3, in the proof-of-concept experiment by the Daiwa Securities Group, when measuring the time required to establish a TLS handshake using the traditional cryptography X25519 and the PQC ML-KEM, it was confirmed that although the time required for key generation, encapsulation, and decapsulation was shorter for PQC, the overall time increased slightly due to the increase in communication volume itself.

While it can be said that the impact is small for normal usage such as websites and cannot be felt by users, it is desirable to consider this before considering migration policies in special businesses/environments such as high-frequency trading systems or environments with bandwidth limitations.

4.7.2 Performance Indicators to Evaluate

Before and after PQC migration, the following indicators may be affected to some extent. Therefore, especially for critical systems, it is desirable to measure the indicators and compare them before and after migration.

Indicator	Examples of Measurement Items
Latency	TLS handshake time (initial / session resumption), API response time (average, 95th percentile).
Throughput	TPS (Transactions Per Second), number of concurrent connections, data transfer volume.
Resources	CPU usage, memory usage, HSM processing capacity.
Stability	TLS handshake failure rate, retransmission/timeout occurrence rate, availability.

Table 35: Performance Indicators to Evaluate for PQC

4.7.3 Policies for Performance Improvement

If performance degradation is a concern during the consideration phase before PQC migration, or if performance degradation becomes apparent in performance measurements after PQC migration, the following countermeasures need to be considered.

Countermeasure	Examples of Countermeasures
Hardware Enhancement	Scale-out/up of LBs, Web servers, etc. Hardware acceleration support (firmware updates, equipment replacement). Additional introduction of HSMs.
Adjustment of Security Parameters	Change to lightweight security parameters (e.g., ML-KEM768 → 512). Requires careful judgment due to the trade-off with security.
Network Optimization	Tuning such as MTU adjustment of network equipment. Offloading cryptography from servers to edge endpoints (introduction of TLS accelerators).

Table 36: Examples of Countermeasures when Performance Degradation Becomes Apparent

5. Conclusion

This document has summarized the basic knowledge of Post-Quantum Cryptography (PQC), trends in domestic and international systems and guidelines, the Proof of Concept (PoC) conducted in our group's online service development environment, and practical approaches for cryptographic migration.

With the U.S. NIST publishing the first PQC standards (FIPS 203/204/205) in August 2024, and the establishment of an inter-ministerial liaison meeting in Japan, we have entered the preparation phase for migration.

In our group's demonstration experiment, we confirmed that while there is a slight impact on communication delay due to the increased communication volume caused by the larger key and certificate sizes in PQC, there is no significant impact. On the other hand, it should be noted that this demonstration experiment was a PoC in a development environment, and the evaluation may change depending on the characteristics of each system.

PQC migration is a medium- to long-term initiative that goes beyond simply replacing cryptographic algorithms and affects communications, authentication infrastructure, key management, procurement, and operations. At present, practical points common to many guidelines recommend inventorying cryptographic assets and maintaining an inventory, prioritizing based on risk assessment, and phased migration (based on system importance, starting from key exchange to signatures and PKI).

Since the situation surrounding PQC is a highly dynamic field, it is necessary to continuously monitor the latest trends in quantum computers, standardization, and implementation. We hope that this document will serve as a starting point for considering, explaining, and implementing PQC responses in many organizations, including the financial sector, and contribute to a smooth transition for the entire industry.

Appendix 1. Glossary

Term	Description
PQC	Post-Quantum Cryptography. A general term for cryptographic schemes expected to maintain security against attacks by quantum computers. It is characterized by being able to be implemented and operated on conventional computers.
Quantum Cryptography	A general term for cryptographic technologies utilizing quantum mechanical properties. Includes QKD, etc. Unlike PQC, it requires dedicated equipment.
QKD	Quantum Key Distribution. A technology that distributes cryptographic keys using quantum mechanical properties such as photons. Theoretically, any eavesdropping can be detected, realizing key exchange with information-theoretic security.
HNDL	Harvest Now, Decrypt Later. An attack scenario where an attacker collects and stores currently encrypted communication data or stored data, and decrypts it after the practical application of quantum computers in the future.
CRQC	Cryptographically Relevant Quantum Computer. A quantum computer with the performance to decrypt currently widely used public-key cryptography in a practical amount of time.
Information-Theoretic Security	Security that is mathematically provable regardless of the attacker's computational power. QKD is considered to have this security.
Computational Security	Security based on the premise that the amount of computation required to break the cryptosystem is so enormous that it is impossible to execute within a realistic time and cost. Conventional cryptographic schemes, including PQC, are based on this security.
Public-Key Cryptography	A cryptographic scheme that uses a different key pair (public key and private key) for encryption and decryption. Widely used for key sharing/exchange and digital signatures. RSA and elliptic curve cryptography (ECDSA, etc.) are typical conventional ciphers, but they may be broken by a quantum computer with sufficient performance using Shor's algorithm, prompting discussions on migrating to PQC.
Symmetric-Key Cryptography	A method where the sender and receiver share the same key for encryption and decryption. AES is a typical example. Grover's algorithm provides a quadratic speedup for key searches by quantum computers, but this can be addressed by doubling the key length (e.g., AES-128 to AES-256), and a complete replacement of the algorithm itself is considered unnecessary.
KEM	Key Encapsulation Mechanism. A scheme to establish a shared key by safely encapsulating (encrypting) the shared key using the receiver's public key, and the receiver extracting (decapsulating) it with their private key. Used for key exchange in

Term	Description
	TLS handshakes, etc. ML-KEM is standardized in PQC.
Key Exchange	The process of sharing a common key with a communication partner in a way that is safe from eavesdropping at the start of communication. Performed during the TLS/SSL handshake and used to encrypt subsequent communication data with symmetric-key cryptography. It is the main target of HNDL attacks and is considered the first priority for PQC migration.
TLS	Transport Layer Security. A protocol that provides data encryption, authentication, and tamper detection in Internet communications. Widely used in HTTPS, etc. PQC key exchange support assumes TLS 1.3.
TLS Handshake	A series of processes where the client and server agree on cryptographic methods, authenticate, and exchange keys when starting TLS communication. Migration to PQC may cause an increase in the number of packets and changes in processing time due to the increased size of keys and certificates.
TLS Session Resumption	A mechanism to re-establish a session by omitting certificate verification, etc., using a session ticket of a previously established TLS session. It can shorten the connection establishment time compared to a full handshake.
PKI	Public Key Infrastructure. A general term for the mechanism to issue, manage, and revoke digital certificates. Certificate Authorities (CAs) play a central role, serving as the trust foundation for TLS communications, code signing, etc. In migrating to PQC, supporting PQC for signatures and certificates is a challenge.
CA	Certificate Authority. An organization that issues and manages digital certificates. There are public CAs that provide services to the general public and private CAs used within organizations. Issuing PQC-compliant certificates requires RFC standardization in IETF and Baseline Requirement updates in the CA/B Forum.
Certificate Chain	The chain of trust from the end-entity certificate (server certificate, etc.) through intermediate CAs to the root CA. Verified to confirm the legitimacy of the server in TLS communication. The depth of the certificate chain and the certificate size affect the handshake processing time.
HSM	Hardware Security Module. Dedicated hardware for securely generating, storing, and managing cryptographic keys.
NIST	National Institute of Standards and Technology. A U.S. government agency that has led the standardization of PQC algorithms. It started a public call in 2016 and published the first PQC standards as FIPS 203/204/205 in August 2024. NIST's standard cryptography schemes are adopted worldwide as de facto international standards.
FIPS	Federal Information Processing Standards. A set of standards issued by NIST.

Term	Description
ML-KEM	A key encapsulation mechanism standardized in NIST FIPS 203. Based on CRYSTALS-Kyber, it has security based on lattice problems.
ML-DSA	A digital signature scheme standardized in NIST FIPS 204. Based on CRYSTALS-Dilithium, it has security based on lattice problems
SLH-DSA	A digital signature scheme standardized in NIST FIPS 205. Based on SPHINCS+, it has security based on the one-way characteristics of hash functions. Since it does not rely on lattice problems, it is considered a backup algorithm for ML-DSA.
FN-DSA	A digital signature scheme currently being standardized by NIST as FIPS 206. Based on Falcon, it has security based on lattice problems.
HQC	Hamming Quasi-Cyclic. A key encapsulation mechanism based on code-based problems (decoding problems of error-correcting codes). Selected as a NIST standardization candidate in March 2025, it is considered a backup algorithm because it has a different mathematical basis from ML-KEM.
Cryptographic Inventory	A ledger that centrally manages the usage status of cryptography, such as cryptographic algorithms, protocols, certificates, and key management status used within an organization.
Crypto-Agility	A design concept that allows flexible migration to another cipher while minimizing the impact on the entire system if a cryptographic algorithm is compromised or a new vulnerability is discovered. Since PQC's security is based on the fact that no efficient attack algorithm is currently known, it is important as a design principle to prepare for future changes.
Dual-Track Approach	A PQC migration approach that sets different migration timelines for critical and non-critical systems. Recommended by the G7 Cyber Expert Group, it realizes an efficient overall migration while prioritizing high-risk systems.
Hybrid Mode	An implementation method during the transition period that uses a combination of conventional cryptography (e.g., X25519) and PQC (e.g., ML-KEM) simultaneously. Based on the concept of defense-in-depth, where confidentiality can be maintained even if one is broken. In TLS 1.3, X25519MLKEM768 is being standardized and implemented, and major browsers already support it.
Pure PQC	A method that uses only PQC algorithms. Since it does not combine with conventional cryptography, the communication data volume is relatively small and implementation is simple, but compatibility with old systems is lost. As of March 2026, there are not many adoption examples, but discussions on adoption may occur when PQC becomes common.
Composite Certificate	A certificate that combines both conventional cryptographic and PQC keys/signatures

Term	Description
	within a single certificate. A method where security is not compromised unless both signature schemes are broken. Standardization is progressing in the IETF LAMPS working group, but implementation complexity is an issue, such as the need for the system side to understand the new certificate format.
Dual Certificate	A method that maintains two independent certificates, one for conventional cryptography and one for PQC, in parallel, and uses them selectively depending on the other party's support status. It avoids the compatibility issues that arise with composite certificates, but there are concerns about the doubled certificate management load on the server side and vulnerability to downgrade attacks.
X25519	An Elliptic Curve Diffie-Hellman key exchange algorithm using Curve25519. Currently widely used in TLS communications. The public key size is very small (32 bytes) and fast, but it is a target for migration to PQC due to the risk of decryption by quantum computers.
X25519MLKEM768	A hybrid key exchange method combining X25519 and ML-KEM 768. Standardization and implementation in TLS 1.3 are progressing. It can be said to be the most practical PQC response method at present to counter HNDL attacks.
IETF	Internet Engineering Task Force. An organization that develops global Internet technical standards. It is advancing discussions and standardization on the Internet implementation of PQC, mainly through three working groups: PQUIP, LAMPS, and TLS.
PQUIP	Post-Quantum Use In Protocols. A working group in IETF that provides cross-sectional coordination and guidance for PQC migration.
LAMPS	Limited Additional Mechanisms for PKIX and S/MIME. An IETF working group responsible for standards related to X.509 certificates.
CA/Browser Forum (CA/B Forum)	An organization operated mainly by Certificate Authorities that issue digital certificates and browsers that display public TLS certificates. It is responsible for PQC standardization in public PKI and discusses and formulates the addition of PQC to the Baseline Requirements.
CNSA 2.0	Commercial National Security Algorithm Suite 2.0. A cryptographic suite for national security systems published by the NSA (U.S. National Security Agency). It outlines the migration policy and timeline to quantum-resistant algorithms, assuming completion of the migration by around 2030.
CRYPTREC	A Japanese framework that conducts security evaluation and monitoring of cryptographic technologies used in e-government, etc., and investigates and considers appropriate implementation and operation.
ENISA	European Union Agency for Cybersecurity. It has published reports on the introduction

Term	Description
	of PQC, indicating a concept that anticipates the start of preparation and implementation from the mid-2020s and the completion of migration for critical infrastructure by the early 2030s.
NCSC	National Cyber Security Centre (UK). It recommends early formulation of migration plans for critical national infrastructure operators in the UK.
G7 Cyber Expert Group	A group of cybersecurity experts from the G7 (Group of Seven) countries. In 2026, it published a statement outlining the concept of a cross-sectoral and coordinated timeline for the transition to PQC in the financial sector.
Tokyo QKD Network	A QKD testbed network built and operated by NICT (National Institute of Information and Communications Technology). It has a track record of QKD demonstrations in Japan and serves as a foundation for studies toward building a wide-area QKD network.

Appendix 2. References and Related Links

It has been confirmed that each link is accessible as of March 26, 2026.

---

[i] Cabinet Secretariat: Transition to Post-Quantum Cryptography (PQC) in Government Agencies and Other Organizations (Interim Report), 2025-11-20, https://www.cas.go.jp/jp/seisaku/pqc/pdf/report_202511.pdf

[ii] Financial Service Agency: Publication of the Report of the Study Group on Measures for Deposit-Taking Financial Institutions to Address Post-Quantum Cryptography, 2025-11-26, https://www.fsa.go.jp/news/r6/singi/20241126.html

[iii] CRYPTREC: CRYPTREC Cryptographic Technology Guideline - Post-Quantum Cryptography - 2024 Edition, 2025-03, https://www.cryptrec.go.jp/report/cryptrec-gl-2007-2024.pdf

[iv] CRYPTREC: Technical Report on Post-Quantum Cryptography, 2025-03, https://www.cryptrec.go.jp/report/cryptrec-tr-2001-2024.pdf

[v] NIST: NIST Releases First 3 Finalized Post-Quantum Encryption Standards, 2024-08-13, https://www.nist.gov/news-events/news/2024/08/nist-releases-first-3-finalized-post-quantum-encryption-standards

[vi] NSA: NSA Releases Future Quantum-Resistant (QR) Algorithm Requirements for National Security Systems, 2022-09-07, https://www.nsa.gov/Press-Room/News-Highlights/Article/Article/3148990/nsa-releases-future-quantum-resistant-qr-algorithm-requirements-for-national-se/

[vii] CISA: Quantum-Readiness: Migration to Post-Quantum Cryptography, 2023-08-21, https://www.cisa.gov/resources-tools/resources/quantum-readiness-migration-post-quantum-cryptography

[viii] ENISA: Post-Quantum Cryptography: Current state and quantum mitigation, 2021-05-03, https://www.enisa.europa.eu/publications/post-quantum-cryptography-current-state-and-quantum-mitigation

[ix] ENISA: Post-Quantum Cryptography - Integration study, 2022-10-18, https://www.enisa.europa.eu/publications/post-quantum-cryptography-integration-study

[x] European Commission: Recommendation on a Coordinated Implementation Roadmap for the transition to Post-Quantum Cryptography, 2024-04-11, https://digital-strategy.ec.europa.eu/en/library/recommendation-coordinated-implementation-roadmap-transition-post-quantum-cryptography

[xi] NCSC: Timelines for migration to post-quantum cryptography, 2025-03-20, https://www.ncsc.gov.uk/guidance/pqc-migration-timelines

[xii] NIST: Call for Proposals (historical reference), 2017-01-03, https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Post-Quantum-Cryptography-Standardization/Call-for-Proposals

[xiii] NIST: Announcing Approval of Three Federal Information Processing Standards (FIPS) for Post-Quantum Cryptography, 2024-08-13, https://csrc.nist.gov/News/2024/postquantum-cryptography-fips-approved

[xiv] NIST: Post-Quantum Cryptography, 2017-01-03, https://csrc.nist.gov/Projects/post-quantum-cryptography/news

[xv] Financial Service Agency: Announcement of the Statement on the Promotion of a Coordinated Roadmap for the Transition to Post-Quantum Cryptography in the Financial Sector by the G7 Cyber Expert Group, 2026-1-15, https://www.fsa.go.jp/inter/etc/20260115/pqc.html