大和総研
Daiwa Institute of Research

大和総研

# 耐量子計算機暗号

## 大和証券グループにおける実証結果と暗号移行アプローチ

大和総研　デジタルソリューション研究開発部

2026 年 3 月 31 日

**免責事項**

本書は、耐量子計算機暗号（PQC）に関する現時点の情報および当社グループ内での検証結果を整理したものであり、将来の標準化動向、製品実装状況、脅威環境の変化により内容は変更され得る。

本書の記載内容は情報提供を目的としており、本書中で言及している製品名、企業名、サービス名は、実証環境の説明を目的として記載したものであり、当社として特定の製品・サービスの採用、性能、将来対応を保証または推奨するものではない。

本書における技術動向・製品対応状況は、原則として 2026 年 3 月時点の情報に基づいている。

**改訂履歴**

| 版 | 日付 | 改訂者 | 概要 |
|---|---|---|---|
| V1.0 | 2026-03-31 | 大和総研 | 初版 |

# 目次

## サマリー

量子コンピュータの進展により、RSA 暗号や楕円曲線暗号など現在広く利用されている公開鍵暗号は将来的に解読リスクを持つ可能性がある。特に、暗号化データを今収集して将来解読する Harvest Now, Decrypt Later（HNDL：今収集し、後で解読）攻撃への備えは、長期秘匿性が求められる金融機関にとって重要であり、PQC 対応は中長期のセキュリティ課題であると同時に経営課題と言える。加えて、米国国立標準技術研究所（National Institute of Standards and Technology：NIST）は 2024 年に最初の PQC 標準を公表しており、日本でも政府・金融分野で 2035 年頃を見据えた移行検討が進んでいる。

大和証券グループでは、オンラインサービスの開発環境等を用いて PQC の実証実験を行った。検証の結果、本実証環境における一般的な Web サービス用途では、鍵交換（KEM）の PQC 対応によって通信に与える影響は限定的であることが確認できた。鍵交換での、TLS ハンドシェイク時間の増加は限定的であり、PQC ソフトウェア対応のロードバランサーでも CPU・メモリ使用率に顕著な増加は見られず、通信量増加も約 5%程度にとどまった。一方で、PQC に変更することによる鍵・証明書サイズ増に伴う通信量増加は確実に発生するため、通信帯域が狭いシステム、通信品質が悪いシステム、低遅延要件の厳しいシステムなどでは個別で影響評価が必要である。

実務上は、PQC 移行を一斉更改として捉えるのではなく、①暗号アセットの棚卸し、②暗号インベントリ整備、③リスク評価と優先順位付け、④ロードマップ策定、⑤段階的移行、の順で進めることが重要である。2026 年 3 月時点では、PQC の署名や証明書の標準化が完了しておらず、インターネット技術特別調査委員会（Internet Engineering Task Force：IETF）では TLS や X.509 証明書での利用方法、認証局/ブラウザフォーラム（Certification Authority/Browser Forum：CA/B フォーラム）ではパブリック PKI での運用ルールの議論が行われている。そのため、移行の第一段階では、HNDL 攻撃対策として効果があり、実装も先行している鍵交換から着手し、署名・証明書の PQC 対応は標準化と製品対応を見ながら進めるのが現実的である。重要システムを優先しつつ、クリプト・アジリティを意識した形で、エッジ終端での対応や共通暗号基盤での対応を中心に段階的に対応していくことが望ましい。

# 1. はじめに

## 1.1 背景・目的

量子コンピュータの研究開発が世界的に進展するにつれ、現在広く利用されている RSA 暗号や楕円曲線暗号等の公開鍵暗号が、十分な性能を持つ量子コンピュータの登場により解読される可能性が指摘されている。

特に深刻な脅威として認識されているのが「Harvest Now, Decrypt Later（HNDL：今収集し、後で解読）」である。これは、攻撃者が現在暗号化されている通信データや保存データを収集・保管しておき、将来的な量子コンピュータの実用化後に解読するという攻撃シナリオである。現時点では安全に保護されているデータであっても、10 年後、20 年後に秘匿性が失われるリスクが存在する。

この脅威に対する解決策として、耐量子計算機暗号（Post-Quantum Cryptography：PQC）の研究開発と標準化が進められている。PQC は、量子コンピュータでも効率的に解読できないと考えられる数学的問題に基づいた暗号技術であり、従来のコンピュータ上で動作する暗号方式である。

金融機関は、顧客の資産情報、取引履歴、個人情報など、長期にわたり秘匿性を保つべき機密情報を大量に取り扱っており、暗号の危殆化は直接的な事業リスクとなる。また、金融庁「金融分野におけるサイバーセキュリティ強化に向けた取組方針」をはじめとする規制・ガイドラインにおいても、暗号技術の適切な管理と最新の脅威動向への対応が求められている。

G7 サイバー・エキスパート・グループが 2026 年に公表した声明において、金融セクターにおける PQC への移行について、分野横断的かつ協調的なタイムラインの考え方が示されている。同声明では、暗号解読に利用可能な量子コンピュータの登場以前に移行を完了することの重要性を踏まえ、金融システム全体としては 2035 年頃を一つの目標時期としつつ、決済・清算等の最重要システムについては 2030～2032 年頃を目安に優先的な対応を進めることが目標とされている。

PQC は従来の暗号と比べて鍵のサイズが大きくなることが知られており、性能や安定性への影響の評価が計画を立てる上でも重要な事項となる。今回、大和証券のオンラインサービスという実際の業務環境に近い条件下で PQC 導入における課題性と課題の検証を行った。対

象システムは、顧客と TLS 通信により機密情報をやり取りする基幹システムであり、現在は RSA 暗号や楕円曲線暗号を利用している。本実証は開発環境を用いた概念実証（PoC）として、本番適用前に技術的・運用的な評価を行ったものである。

本ホワイトペーパーの目的は以下のとおりである。
・耐量子計算機暗号（PQC）の基本概念、標準化・制度動向を整理し、企業／組織内の共通理解を形成する
・大和証券グループで実施した PQC の実証実験（PoC）の結果を、金融業界をはじめ一般の方にも参考資料として提供する
・暗号インベントリの整備やリスク評価、暗号移行のロードマップを含む、PQC 移行の実践的なアプローチを提示する
・移行計画策定・ベンダー調整・性能影響評価の観点を明確化し、実務に適用できる形で提供する

## 1.2 想定読者と本書の構成

本書は、耐量子計算機暗号（PQC）への移行を検討・実施する組織における幅広い関係者を想定読者としている。特に、金融機関をはじめとする重要インフラ事業者や、機密性の高い情報を長期にわたり保護する必要がある組織の方々に向けた内容となっている。

具体的には、以下の役割を担う方々を主な読者として想定している。
【技術担当者】
・インフラ・ネットワーク設計・運用担当者：TLS/SSL 通信、ロードバランサー、ファイアウォール等の PQC 対応検討
・アプリケーション開発担当者：暗号ライブラリの選定、実装方法の検討
・PKI 運用・鍵管理担当者：証明書の移行計画、認証局の対応状況の把握

【企画・管理担当者】
・情報セキュリティ担当者：暗号インベントリの整備、リスク評価、移行ロードマップの策定
・リスク管理・監査担当者：規制対応状況の確認、第三者リスクの評価
・システム企画担当者：システム更改計画と PQC 移行の調整、予算確保の検討

【意思決定層】
・IT 部門管理職・経営層：PQC 移行の必要性理解、投資判断、全社方針の策定

本書は暗号技術の専門家向けではなく、IT 系の業務に携わる方であれば理解できる様に可

能な限り平易な記述を心がけている。暗号の数学的な詳細には深入りせず、PQC 移行に必要な実務的知識と具体的なアプローチの提示に重点を置いた。暗号技術の前提知識がない読者でも、第 2 章から順に読み進めることで、PQC 移行の全体像と実践方法を理解できる構成としている。

本書は全 5 章と付録から構成される。各章の概要は以下のとおりである。

第 2 章「耐量子計算機暗号の基礎知識」では、PQC の基本概念を平易に解説する。量子鍵配送（QKD）との違い、量子コンピュータ出現予測のタイムライン、NIST 標準化の経緯と採択されたアルゴリズムなど、技術的背景を概説する。

第 3 章「大和証券グループで行った耐量子計算機暗号の実証実験」では、本実証実験や検証環境を用いた基礎検証で行った、鍵交換・鍵カプセル化、デジタル署名それぞれについて、検証環境、検証項目、検証結果を詳述する。検証結果から PQC 導入時の影響として考えられる点を具体的に述べる。

第 4 章「暗号移行のアプローチ」では、組織が PQC 移行を実施する際の実践的なアプローチを 6 つのステップで解説する。暗号アセットの棚卸し、暗号インベントリの作成・管理、リスク評価と優先順位付け、ロードマップ策定、具体的な移行方法の検討、影響評価まで、実務で必要となると考えられる一連のプロセスを提示する。

第 5 章「おわりに」では、総括と今後の展望を述べる。

本書は、第 1 章から順に読み進めることで、PQC 移行の必要性から具体的な実践方法までを体系的に理解できる構成となっている。

ただし、読者の役割や関心に応じて、以下のような読み方も可能である。

・経営層・意思決定者：第 1 章、第 2 章、第 5 章の 1 節を中心に、PQC 移行の必要性、規制動向、移行の進め方の概要を把握する
・技術担当者：第 2 章、第 3 章を中心に、技術的詳細と実証実験の結果を重点的に理解する
・企画・セキュリティ担当者：第 2 章、第 4 章を中心に、制度対応と移行プロセスの実務を理解する

各章は、ある程度独立して読むことができるよう配慮している。初めて本テーマに触れる読者には通読を、特定の観点で情報を求める読者には該当章の参照を推奨する。

# 2. 耐量子計算機暗号の基礎知識

## 2.1 耐量子計算機暗号とは何か？

### 2.1.1 耐量子計算機暗号の定義について

耐量子計算機暗号（Post-Quantum Cryptography: PQC）とは、従来型（古典）コンピュータ上で実装・運用でき、量子コンピュータによる攻撃に対しても安全性を維持できると期待される暗号方式の総称である。

現在、インターネットや電子商取引で広く利用されている RSA 暗号や楕円曲線暗号などの公開鍵暗号は、素因数分解問題や離散対数問題という数学的に「効率的に解くのが困難」とされる問題に安全性の根拠を置いている。しかし、1994 年にピーター・ショア（Peter Shor）が発表した量子アルゴリズム（Shor のアルゴリズム）により、十分な性能を持つ量子コンピュータが実現すれば、これらの問題を現実的な時間で解けることが理論的に示された。これが、1.1 章で述べた「現在の公開鍵暗号が量子コンピュータにより解読される可能性」の根拠となっている。

PQC は、量子コンピュータでも効率的に解くことが困難と考えられる別の数学的問題（格子問題、符号問題、多変数多項式問題、ハッシュ関数の性質など）に基づいて設計された暗号方式である。重要な点は、PQC は量子コンピュータ上で動作する必要はなく、基本的に現在使用している従来型コンピュータ、サーバー、スマートフォン上でソフトウェア更新により実装できることである。

### 2.1.2 量子暗号との違い

PQC は「量子暗号（Quantum Cryptography）」とは異なる技術である。この区別は、PQC 導入を検討する際に重要となるため、ここで明確にしておく。

量子暗号の一手法である量子鍵配送（Quantum Key Distribution: QKD）は、光子などの量子力学的性質を利用して暗号鍵を配送する技術であり、公開鍵暗号に代わる鍵交換の手段となる。QKD は量子力学に基づいて理論上、盗聴があれば必ず検知でき、情報理論的安全性を持つ鍵交換を実現することができる。情報理論的安全性とは、攻撃者の計算能力に依存せず、数学的に証明可能な安全性を意味し、将来もアルゴリズムの危殆化の恐れがないとい

う、非常に高い安全性が保証される。

一方、PQC を含む従来の暗号方式は「計算量的安全性」に基づいている。これは、暗号を解読するために必要な計算量が現実的な時間・コストでは実行不可能なほど膨大であることを根拠とする安全性である。

両者の主な違いについて表に示す。

| 観点 | PQC（耐量子計算機暗号） | QKD（量子鍵配送） |
|---|---|---|
| 動作環境 | 従来型コンピュータ | 専用機器 |
| 安全性の根拠 | 計算量的安全性 | 情報理論的安全性 |
| 適用範囲 | インターネット通信全般（TLS 等） | 限定的（拠点間の通信など） |
| 導入コスト（比較） | 低い（ソフトウェア更新） | 高い（専用機器と光ファイバーが必要） |

表 1 PQC と QKD の主な違い

基本的には、既存のインターネット基盤（TLS/SSL 通信、PKI 証明書基盤等）に対して適用が可能な PQC が、量子コンピュータ出現で生じる脅威への対策の主軸になると考えられる。一方で QKD は、特に高度なセキュリティが求められる特定用途（政府機関間通信、重要インフラの基幹通信等）での補完的な役割が期待される。QKD には通信距離の限界（一般には 100km 程度）という制約もあるが、QKD リンクをつなぐQKDネットワークで広域化も可能になる。このようなネットワークインフラとして実現することで QKD の利用シーンは大きく広がることが期待でき、欧州をはじめ各国で実証が進められている。日本でも NICT（情報通信研究機構）によるテストベッド（東京 QKD ネットワーク）の実績があり、一層の広域網の構築に向けての検討が進んでいる。

### 2.1.3 PQC の範囲

暗号技術は大きく「公開鍵暗号」と「共通鍵暗号」に分類される。量子コンピュータの脅威は、これら両方に影響するが、その影響の大きさは異なる。

**公開鍵暗号への影響**

公開鍵暗号は、以下の用途で広く利用されている。

- 鍵共有・鍵交換：通信の開始時に、安全に共通鍵を共有する（例：TLS/SSL のハンドシェイク）
- デジタル署名：電子文書の作成者の認証と改ざん検知（例：TLS/SSL 証明書、コード署名、電子契約）

これらは、Shor のアルゴリズムにより量子コンピュータで効率的に解読される可能性がある。公開鍵暗号は量子コンピュータの出現に備えるためにはアルゴリズム自体の全面的な置き換えが必要になり、特に対応が必要と言える。

具体的には、以下の技術要素が対象となる。

- 鍵交換アルゴリズム（RSA、ECDH 等）
- デジタル署名アルゴリズム（RSA、ECDSA 等）

これらは主に以下のような場面で用いられている。

- TLS や IPSec などの通信保護のための認証付き鍵交換
- 電子契約などのデジタル署名
- ソフトウェアアップデート、セキュアブートなどのコード署名
- 電子メールの暗号化、署名（S/MIME）
- FIDOなどの認証（デジタル署名）
- 暗号通貨

**共通鍵暗号への影響**

共通鍵暗号（AES 等）は、送信者と受信者が同じ鍵を共有して暗号化・復号を行う方式である。Grover のアルゴリズムという量子アルゴリズムにより、鍵探索に要する計算量は古典的な全探索に比べて平方根に低減され、二次の高速化が得られることが知られている。

例えば、AES-128（128 ビット鍵）の安全性は、量子コンピュータ環境下では従来型コンピュータの AES-64（64 ビット鍵）相当に低下する。

したがって、共通鍵暗号については、量子コンピュータの出現に備えて、鍵長の見直し（AES-128→AES-256 等）が推奨されるが、優先度は高くなく、アルゴリズム自体の全面的な置換は不要とされている。

**本書の対象範囲**

本書では、量子脅威への対応として最も緊急性が高い、公開鍵暗号の PQC 移行による影響について第 3 章で取り上げる。

### 2.1.4 PQC の量子安全性根拠

PQC は、量子コンピュータでも効率的に解くことが困難と考えられる数学的問題に基づい

ている。主に以下の数学的性質に基づいている。

- 格子ベース暗号：高次元格子上の最短ベクトル問題等に基づく。
- 符号ベース暗号：誤り訂正符号の復号問題に基づく。
- 多変数多項式暗号：多変数多項式方程式の求解問題に基づく。
- ハッシュ関数：ハッシュ関数の一方向性に基づく。

これらの数学的問題は、現時点では量子コンピュータでも効率的に解くアルゴリズムが知られていない。そのため、暗号の安全性は「証明」されている訳ではなく、「現時点で効率的なアルゴリズムが発見されていない」という点に留意が必要である。

そのため、今後の量子アルゴリズム研究の進展により、評価が変わる可能性もあるため、暗号アルゴリズムを柔軟に変更できる様に準備する「クリプト・アジリティ」の考え方が重要となる。この点については、第 4 章で詳述する。

## 2.2 暗号移行タイムライン

耐量子計算機暗号（PQC）への移行を検討するにあたり、「いつまでに対応すべきか」という時間軸の判断は極めて重要である。本節では、量子コンピュータの実用化予測を起点に、国際的なガイドラインにおける移行目標の位置付け、および PQC 移行における段階的な考え方について、全体像を整理する。

なお、具体的な移行手順については第 4 章で詳述する。本節では、タイムライン全体を俯瞰するための基礎的な考え方を示す。

### 2.2.1 量子コンピュータの実用化予測と移行時期の考え方

量子コンピュータが現在広く利用されている公開鍵暗号を実用的に解読可能な性能に到達する時期については、現時点で確定的な予測は存在しない。このような暗号解読能力を持つ量子コンピュータは、CRQC（Cryptographically Relevant Quantum Computer：暗号解読に適した量子コンピュータ）と呼ばれる。多くの公的機関や研究者は、CRQC の出現時期を「早くても 2030 年代以降」としつつも、技術的ブレークスルーや研究投資の加速によって前倒しされる可能性を否定していない。

このように実用化時期の予測には不確実性があるものの、暗号移行の検討を先送りできない理由として指摘されているのが、Harvest Now, Decrypt Later（HNDL：今収集し、後で解読）攻撃の脅威である。HNDL とは、攻撃者が現在暗号化されている通信データや保存データを収集・保管しておき、将来的な量子コンピュータの実用化後に解読するという攻撃シナリオである。現時点では安全に保護されているデータであっても、10 年後、20 年後に秘匿

性が失われるリスクが存在する。

金融機関が扱う顧客の資産情報、取引履歴、個人情報などは、長期にわたり秘匿性を保つべき情報であり、この HNDL 攻撃の主要な標的となると考えられる。そのため、PQC 移行のタイミングは「量子コンピュータがいつ出現するか」だけでなく、以下の関係を踏まえて判断する必要がある。

情報の秘匿期間 ＋ システム移行に要する期間 ≤ 量子コンピュータの出現までの期間

例：

- 10 年以上の秘匿性が求められる情報
- システム移行に 5 年程度を要する場合
- 量子コンピュータが 15 年後に出現すると仮定

上記の例の場合、10 年 ＋ 5 年 ＝ 15 年となり、既に移行の検討・準備を開始すべき段階にあることがわかる。

### 2.2.2 国際的なガイドラインにおける移行目標の位置付け

量子コンピュータによる暗号リスクを踏まえ、各国・各地域の政府機関や標準化団体は、PQC への移行に関する方向性を示し始めている。本項では、日本および海外の主要な動向の概要を示す。

**日本の動向**

日本国内では、2025 年頃から政府機関および金融業界を中心に、量子コンピュータ時代を見据えた PQC 移行に向けた議論が活発に行われるようになった。

（1）政府横断の取り組み

国家サイバー統括室（NCO）は、2025 年 6 月 30 日に「政府機関等における耐量子計算機暗号（PQC）利用に関する関係府省庁連絡会議」を設置し、PQC 移行に関する具体的な検討を行っている。この中で、2025 年 11 月 20 日には、「政府機関等における耐量子計算機暗号（PQC）への移行について（中間とりまとめ）」[i]がまとめられ、公表された。

中間とりまとめでは、政府機関等における耐量子計算機暗号（PQC）への移行について、原則として、2035 年までに行うことを目指し、関係府省庁との連携の下、2026 年度に移行に向けた工程表（ロードマップ）を策定し、日本における円滑な PQC 移行を推進していく、とし

ている。

（2）金融庁の動向

金融庁は 2024 年 7 月に「預金取扱金融機関の耐量子計算機暗号への対応に関する検討会」を開催し、2024 年 11 月に「預金取扱金融機関の耐量子計算機暗号への対応に関する検討会 報告書」[ii]を公開した。

報告書では、金融機関が取り扱う情報は、顧客情報や取引情報など機密性・完全性の確保が不可欠であり、量子コンピュータによる暗号技術の安全性低下が金融システム全体の信頼性に重大な影響を及ぼし得る点が指摘されている。

こうした認識の下、報告書では、PQC への移行に向けた事前準備として、金融機関が自ら保有・利用する情報資産と暗号利用状況を把握することの重要性を示している。具体的には、通信、データ保存、電子署名、認証といった暗号の利用場面ごとに、使用している暗号アルゴリズムや鍵長等を整理した暗号インベントリを作成し、それを基に対応を検討していく必要があるとしている。

また、PQC への対応に当たっては、すべてのシステムを一律に移行するのではなく、情報の重要性や影響度等を踏まえた検討が求められるとされている。加えて、PQC は新しい技術であることから、今後の標準化動向や技術進展を踏まえつつ、政府・監督当局、業界団体、ベンダー等が連携しながら対応を進めていくことが重要であると報告書は整理している。

（3）その他省庁の動向

PQC は、暗号アルゴリズムの選定だけでなく、ネットワーク、認証基盤、端末・機器、運用、調達など複数レイヤに影響する。そのため国内では、政府横断の検討と並行して、政策面・研究開発面・評価面が接続する形で取り組みが進んでいる。

例えば総務省では、PQC の位置づけや研究開発（NICT の活動）を含めた説明資料が提示されており、CRYPTREC における評価活動とも関連している。

（4）CRYPTREC の方針

CRYPTREC は、電子政府等で利用される暗号技術の安全性評価・監視や、適切な実装・運用に関する調査検討を行う枠組みであり、PQC に関してもガイドライン整備や研究動向の整理を進めている。

2025 年 4 月には、CRYPTREC の 2024 年度成果物として「CRYPTREC 暗号技術ガイドライン（耐量子計算機暗号）2024 年度版」[iii]および「耐量子計算機暗号の研究動向調査報

告書」[iv]が公開された。

政府横断の中間とりまとめでも、移行検討において CRYPTREC のガイドライン参照が明記されており、また NIST 標準として公開された暗号方式(例:FIPS 203/204/205)について安全性・実装性能評価を進め、CRYPTREC 暗号リスト更新を目指す趣旨が述べられている。

**米国の動向**

米国では、米国国立標準技術研究所(National Institute of Standards and Technology:NIST)が PQC アルゴリズムの標準化を主導してきた。

2016 年に開始された国際コンペティションの結果、2024 年 8 月に鍵カプセル化方式およびデジタル署名方式の主要アルゴリズムが FIPS 標準(FIPS 203/204/205)として正式に公開された[v]。これにより、PQC は「研究段階」から「実装・導入段階」へ移行したと位置付けられている。

一方、国家安全保障局(National Security Agency:NSA)は、国家安全保障システム向けの暗号スイートとして CNSA Suite 2.0(Commercial National Security Algorithm Suite 2.0)[vi]を公表し、量子耐性アルゴリズムへの移行方針とタイムラインを示している。CNSA 2.0 では、2030 年前後までに従来の RSA 暗号や楕円曲線暗号から PQC への移行を完了することが想定されており、新規システムや調達においては PQC 対応が前提となりつつある。

さらに、サイバーセキュリティ・社会基盤安全保障庁(Cybersecurity and Infrastructure Security Agency:CISA)は、政府機関および重要インフラ事業者向けに、PQC 移行を支援するためのガイダンスやロードマップを公表[vii]している。暗号インベントリの整備、影響度に基づく優先順位付け、段階的な移行計画の策定など、実務に直結する内容が特徴であり、PQC 移行を「長期的な経営課題」として扱う必要性を強調している。

このように米国では、政府が中心となり、民間を含めた社会全体に対して PQC 移行の必要性と緊急性を明確に打ち出している。

**欧州の動向**

欧州では、欧州ネットワーク・情報セキュリティ機関(European Network and Information Security Agency:ENISA)が中心となり、2021 年という早期から PQC 導入に関する標準化プロセスに対する技術的・運用的課題を整理した報告書[viii]を公表してい

る。ENISA は、2020 年代中盤からの準備・実装開始と、2030 年代前半までの重要インフラの移行完了を見据えた考え方を示している[ix]。こうした検討を受けて、欧州委員会は 2024 年に PQC 移行に関する勧告やロードマップを公表[x]しており、ENISA は技術的検討を支える中核機関として位置付けられ、EU 圏全体での協調的な PQC 移行を推進する役割を担っている。

また、英国の国家サイバーセキュリティセンター（National Cyber Security Centre:NCSC）も、重要国家インフラ事業者に対し、早期の移行計画策定を推奨している。NCSC は PQC 移行に関する明確なフェーズ分け（調査・準備、部分移行、全面移行）を提示[xi]しており、2028 年、2031 年、2035 年といった具体的な節目を示している。これにより、PQC 移行が中長期の IT 戦略・システム更改計画と不可分であることが強調されている。

**海外の動向から得られる示唆**

海外の法制度・ガイドラインの動向から、以下の点が共通したメッセージとして読み取れる。

- PQC は「将来検討すべき技術」ではなく、既に移行準備を開始すべき前提技術と位置付けられている
- 標準化と並行して、移行期限や調達方針を伴う政策的な要請が進んでいる
- 移行には長期間を要するため、暗号インベントリの整備や実証実験を早期に開始することが重要である

これらの点は、日本の金融機関においても、中長期のシステム戦略・リスク管理の一環として位置付ける必要性があることを示していると言える。

### 2.2.3 段階的な暗号移行という考え方

暗号技術は、通信プロトコル、認証基盤、鍵管理システム、アプリケーション実装、運用手順など、システム全体に広範な影響を及ぼす。そのため、PQC への移行を短期間で一斉に実施することは現実的ではない。

多くのガイドラインにおいて、段階的な移行アプローチが前提とされている。これは、技術的な複雑性、互換性の問題、運用負荷、コストなどを考慮した現実的な対応である。

一般的には、大きく 4 つの段階を経て移行が進められる。

| 段階 | 対応内容の例 |
|---|---|
| 準備段階 | 暗号インベントリの作成(どこでどのような暗号が使われているかの把握)やリスク評価を行う。<br>この段階では、組織内の体制整備や予算確保も重要となる。 |
| 計画策定段階 | リスク評価に基づき移行対象の優先順位を決定し、具体的な移行方法(ハイブリッド方式の採用可否など)やロードマップを策定する。<br>ハイブリッド方式とは、従来暗号と PQC を併用することで、移行期間中の安全性と互換性を両立する方式である。 |
| 移行段階 | 優先順位に基づいて順次 PQC に移行する。移行にあたっては将来的な暗号の更新に備えるクリプト・アジリティを考慮する。 |
| 完了段階 | 全システムの移行完了後、従来の公開鍵暗号を段階的に廃止する。また、移行後も定期的に動向を確認して、暗号が危殆化した場合には、別の暗号へ移行するなど対応する。 |

表 2 PQC の段階的移行

このような段階的移行を実現するための具体的なプロセス(暗号インベントリの整備方法、リスク評価の手法、ロードマップの策定方法など)については、第 4 章で詳述する。

### 2.2.4 タイムラインの例

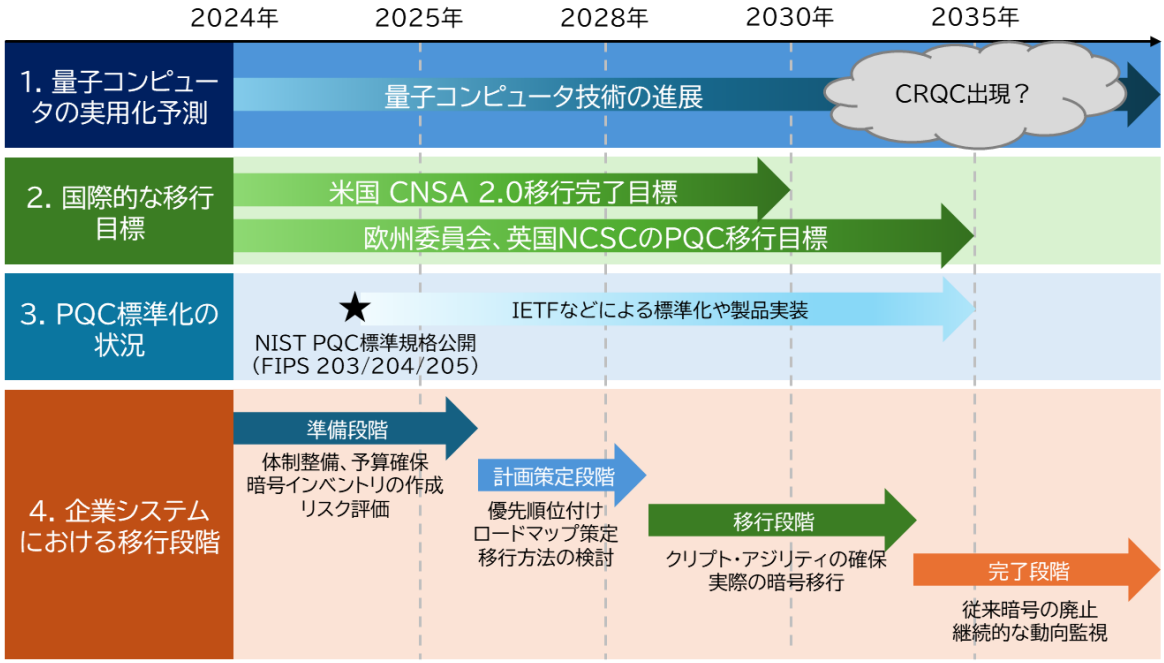


図 3 暗号移行タイムラインの例

図 3 は、本節で述べた要素を時系列で整理したものである。横軸に時間、縦軸に以下の 4 つの要素を配置している。

1. 量子コンピュータの実用化予測：CRQC の出現が予測される時期の幅
2. 国際的な移行目標：CNSA 2.0（2030 年）や欧州委員会や NCSC の推奨時期など
3. PQC 標準化の進展：NIST 標準化の完了（2024 年）と IETF の標準化など今後の展開
4. 企業システムにおける移行段階：準備段階→計画策定準備→移行段階→完了段階の段階

この図から読み取るべき重要なポイントは、以下のとおりである。

- 国際的な移行目標（2030 年～2035 年）まで時間的余裕は少ない
- 移行には長期間を要するため、早期の準備開始が不可欠

ただし、本タイムラインは様々な情報から例示した一般的な目安であり、各組織の状況（システム規模、扱う情報の秘匿期間、リスク許容度など）に応じて調整が必要である。

## 2.3 NIST 標準化の動向

耐量子計算機暗号の実用化において、米国国立標準技術研究所（National Institute of Standards and Technology：NIST）の標準化の一つの世界的な指針となっている。NIST は米国政府機関だが、NIST の標準暗号は世界中の多くで採用されており、事実上の国際標準としての役割を果たしている。

NIST は 2016 年 12 月、量子コンピュータ時代に備えた新しい暗号アルゴリズムの公募（Call for Proposals）を開始[xii]した。世界中から 82 件の提案が集まり、安全性、コスト（鍵のサイズ／計算コスト）、アルゴリズム／実装特性の 3 つの観点から評価が行われた。

評価は複数ラウンド制で実施され、各ラウンドで候補を絞り込む方式が採られ、約 8 年間にわたって暗号アルゴリズムの選定が行われた。

2024 年 8 月、NIST は最初の PQC 標準を正式に公表した[xiii]。米国政府の情報処理標準規格 FIPS（Federal Information Processing Standards）として、以下の 3 つの方式が標準化された。

| FIPS 規格 | アルゴリズム名 | 用途 | 基になった方式 | 数学的根拠 |
|---|---|---|---|---|
| FIPS 203 | ML-KEM | 鍵交換 | CRYSTALS-Kyber | 格子問題 |
| FIPS 204 | ML-DSA | デジタル署名 | CRYSTALS-Dilithium | 格子問題 |
| FIPS 205 | SLH-DSA | デジタル署名 | SPHINCS+ | ハッシュ関数 |

表 4 NIST 標準化された PQC アルゴリズム

また、これらのアルゴリズムに対して性能や実装性の向上、万が一脆弱性が見つかった場合に備える安全性のため、NISTでは追加のアルゴリズムについても標準化を継続して行っている。FN-DSA は格子問題に基づくデジタル署名であるが、署名長などのサイズが ML-DSA よりも小さいという特長があり、NIST による FIPS のドラフトが作成され公開に向けた手続きが進められている。デジタル署名では一層のサイズや性能の改善、安全性のダイバーシティ拡大に向けて新たな公募が実施され、2025年時点では第2ラウンドまで進んでいる。HQC は符号問題に基づく鍵交換方式で、2025 年 3 月に標準化候補に選定され、今後標準化に向けたプロセスが進められるとみられる[xiv]。

| FIPS 規格 | アルゴリズム名 | 用途 | 基になった方式 | 数学的根拠 |
|---|---|---|---|---|
| FIPS 206 | FN-DSA | デジタル署名 | Falcon | 格子問題 |
| （未決定）<br>FIPS 207? | （未決定）<br>HQC-KEM? | 鍵交換 | HQC | 符号問題 |

表 5 今後 NIST 標準化される PQC アルゴリズム

重要な点として、NIST による FIPS は「暗号アルゴリズムの仕様」のみを定めており、実際のシステムでの使い方は別途規定される。例えば、TLS 通信で ML-KEM をどう使うかは IETF（インターネット技術特別調査委員会）が定める RFC で規定される。X.509 証明書で ML-DSA 署名をどう扱うかも同様に別の標準化作業が必要となる。IETF での標準化の動きについては、3 章のコラムにて解説する。

# 3. 大和証券グループで行った実証実験

1 章で述べたように量子コンピュータによる将来的な暗号解読に対して早期に対策が必要な脅威は HNDL 攻撃であり、秘匿に関する脅威である。このため、PQC の中でもデジタル署名よりは鍵交換が一般には先行して移行を検討する対象となる。インターネット標準においては X25519 と ML-KEM 768 のハイブリッド鍵交換が PQC 標準として採用され、主だったブラウザは既にこのハイブリッド鍵交換に対応しており、サーバーが対応している場合には PQC の適用が始まっている。NICT[1]の調査では 2025 年 9 月の時点で NICT から外部サーバーへのアクセスの 12%が X25519MLKEM768 による鍵交換に対応という報告[2]があり、現在ではその割合は増えていると予想される。

今回の実証実験はクラウドに構築した基礎検証の環境および実システムに近い設定として、大和証券のオンラインサービスの開発環境を用いて実施した。前者はクライアント-サーバーの単純な構成での TLS の性能検証を実施し、後者はクライアントと業務サーバーとの間に F5 社のロードバランサーをおいた構成を検証環境として用いて TLS の性能検証および負荷試験を実施した。対象のロードバランサーは X25519MLKEM768 の鍵交換にソフトウェア実装で対応している。

以下、本章では多くのケースで PQC 移行の最初のステップとなる鍵交換に関する基礎検証の結果を 3.1 で、オンラインサービスの開発環境を用いての検証結果を 3.2 で報告する。3.3 では PQC 移行の次ステップとなる TLS の認証（証明書）のデジタル署名についての基礎検証について報告する。

---

[1] 情報通信研究機構

[2] NICTER Blog, “ライブネットにおける PQC 利用状況の調査”, https://blog.nicter.jp/2025/12/pqc-usage/, 4 December 2025

## 3.1 検証環境を使った基礎検証(鍵交換アルゴリズム)

### 3.1.1 検証環境

検証に使用した端末は Amazon EC2 のインスタンスである。具体的な仕様は別紙報告書を参照のこと。また、個別のアルゴリズムの測定については以下のライブラリを使用した:

| ライブラリ名 | 概要 |
|---|---|
| Cryotography (Python) | バージョンは 46.0.3。<br>RSA、X25519、P-256 の性能を測定するために使用した。 |
| liboqs | バージョンは 0.14.0。<br>各耐量子計算機暗号アルゴリズムの性能を測定するために使用した。<br>また、ラッパーとして liboqs-python を使用した。 |

更に、具体的な通信においては以下のライブラリを使用した:

| ライブラリ名 | 概要 |
|---|---|
| OpenSSL | バージョンは 3.5.4。<br>TLS 通信のためのクライアントやサーバーとして使用した。 |
| OQS Provider | バージョンは 0.10.1-dev。<br>一部の耐量子計算機暗号を用いた TLS 通信のために使用した。 |

尚、クライアント側の実装のために pwntools や paramiko も使用しているが、ハンドシェイクとは無関係な通信にのみ用い、TLS 通信は OpenSSL を使用して行っている。

### 3.1.2 留意事項

尚、各ライブラリのバージョンは実験開始当時(2025 年 11 月時点)の最新版である。

執筆時点に於いて、cryptography は 46.0.5、liboqs は 0.15.0、OpenSSL は 3.6.1/3.5.5、OQS Provider は 0.11.0 が最新のバージョンとなっている。

また、OpenSSL は実験開始当時に 3.6.0 も公開されていたが、LTS である 3.5.4 を選択した。

### 3.1.3 検証項目及び実験概要

まず、純粋な鍵交換アルゴリズムの性能の測定として、鍵の生成、カプセル化・デカプセル化(X25519 の場合は公開鍵・共有秘密の生成)にかかる時間を調査した。

検証対象としたアルゴリズムは、X25519、ML-KEM 512、ML-KEM 768、ML-KEM 1024 である。尚、HQC は標準化対象に選定されたものの、FIPS 案が公開されていないこ

とから今回は対象外とした。

各アルゴリズムについて、鍵の生成、カプセル化、デカプセル化それぞれについてかかる時間を 1000 回分計測し、オーバーヘッドの影響がある初回を除いた 999 回分のデータを基に算出した。

次に、実際の通信において、TLS ハンドシェイクの確立までにかかる時間を調査した。この通信は、二つの異なるアベイラビリティゾーンにある EC2 インスタンス間で、一方をクライアント、他方をサーバーとして行う。このとき、サーバーには基礎検証用に用意したサーバー証明書及び認証局証明書を提示させる。

検証対象とした鍵交換アルゴリズムは、X25519、ML-KEM 512、ML-KEM 768、ML-KEM 1024、X25519MLKEM768 である。また、サーバー及び認証局の署名アルゴリズムとして 2048-bit RSA を使用している。尚、ここでいうサーバーの署名アルゴリズムとは、Certificate Verify でサーバーが署名に使用するアルゴリズムを指す。認証局の署名アルゴリズムとは、証明書に付されている署名アルゴリズムを指す。

ハンドシェイクの回数は 1000 回とし、その通信データを基に分析を行った。

後に提示する結果において、通信時間はクライアント及びサーバーで記録された時間の差分によって算出している。尚、クライアントとサーバーを動かしている EC2 インスタンスは同一の NTP サーバ(169.254.169.123)と同期させ、キャプチャファイルに記録された時間は NTP サーバーとのずれを基に補正している。

### 3.1.4 検証結果

#### 3.1.4.1 性能の計測結果

まず、公開鍵・秘密鍵・暗号文・シークレットのサイズは以下の通りである:

| | 公開鍵 | 秘密鍵 | 暗号文 | シークレット |
|---|---|---|---|---|
| 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 |

表 1:公開鍵・秘密鍵・暗号文・シークレットのサイズ(単位はバイト)

純粋なアルゴリズムの性能について、鍵の生成にかかった時間の平均、標準偏差、四分位及びロバストな変動係数は以下の通りであった：

| | 平均 | 標準偏差 | 第一四分位 | 中央値 | 第三四分位 | ロバストな変動係数 |
|---|---|---|---|---|---|---|
| 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 |

表 2：鍵交換アルゴリズムの鍵の生成にかかった時間の平均、標準偏差、各四分位及びロバストな変動係数（平均、標準偏差、各四分位の単位はミリ秒）

また、時間の分布は以下の通りであった：

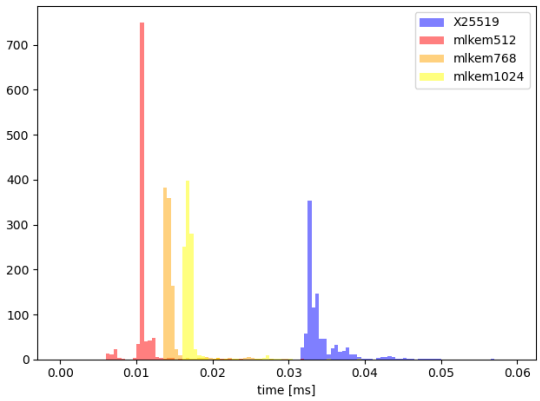


図 1：鍵交換アルゴリズムの鍵の生成にかかった時間の分布

次に、カプセル化にかかった時間の平均、標準偏差、四分位及びロバストな変動係数は以下の通りであった：

| | 平均 | 標準偏差 | 第一四分位 | 中央値 | 第三四分位 | ロバストな変動係数 |
|---|---|---|---|---|---|---|
| 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 |

表 3：カプセル化にかかった時間の平均、標準偏差、各四分位及びロバストな変動係数（平均、標準偏差、各四分位の単位はミリ秒）

また、時間の分布は以下の通りであった：

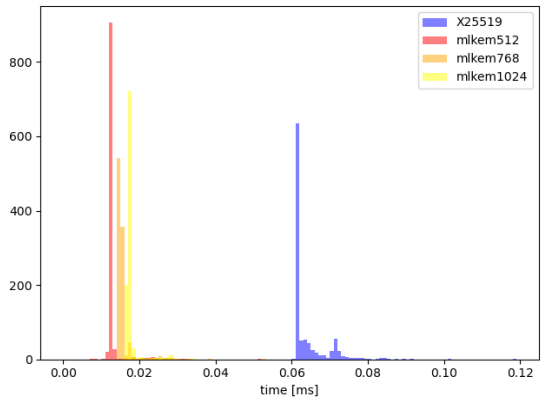


**図 2：カプセル化にかかった時間の分布**

最後に、デカプセル化にかかった時間の平均、標準偏差、四分位及びロバストな変動係数は以下の通りであった：

| | 平均 | 標準偏差 | 第一四分位 | 中央値 | 第三四分位 | ロバストな変動係数 |
|---|---|---|---|---|---|---|
| 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 |

**表 4：デカプセル化にかかった時間の平均、標準偏差、各四分位及びロバストな変動係数（平均、標準偏差、各四分位の単位はミリ秒）**

また、時間の分布は以下の通りであった：

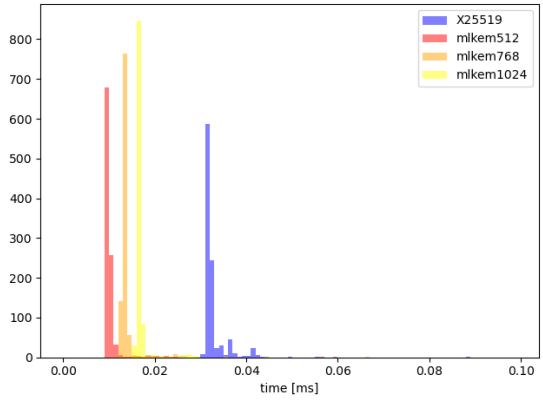


**図 3：デカプセル化にかかった時間の分布**

最後に、ハンドシェイクにかかった時間の平均は以下の通りである。下記積み上げグラフの凡例は別紙報告書、詳細な値については CSV を参照のこと。

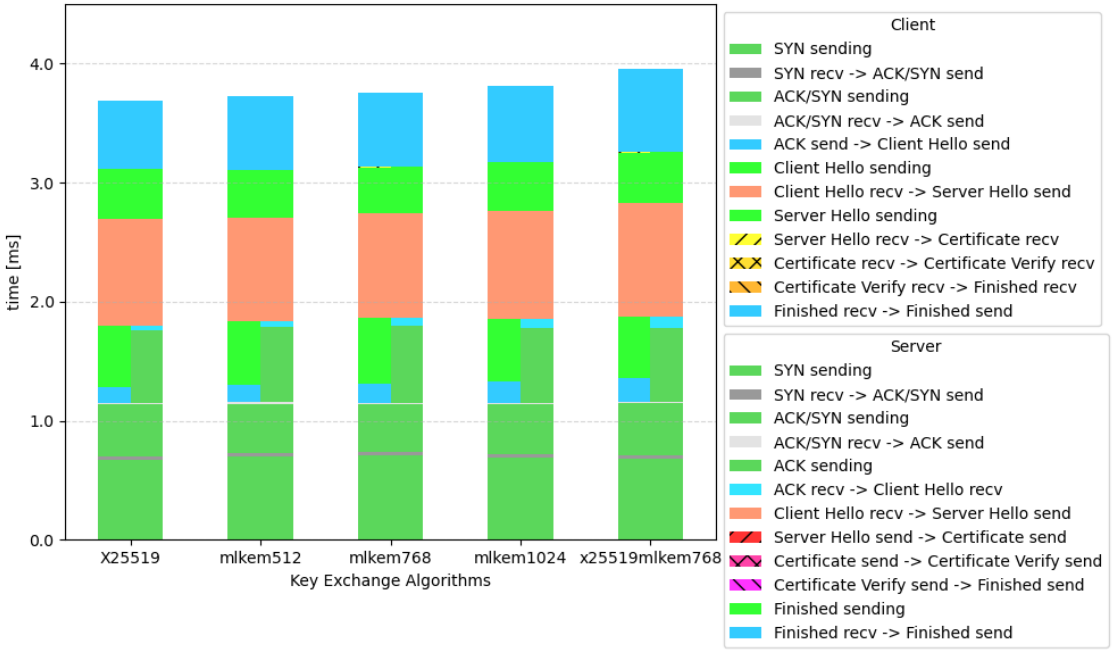


**図 4:サーバー及び認証局の署名アルゴリズムが 2048-bit RSA の場合**

### 3.1.4.2 計測結果に対する評価

鍵生成・カプセル化・デカプセル化にかかった時間の平均は数十マイクロ秒のオーダーである。これに対し、TLS ハンドシェイクの確立にかかる時間はミリ秒のオーダーであることから、通常の通信用途では暗号処理にかかる時間の差は誤差の範囲であるといえる。

その上で敢えて評価をするならば、ML-KEM の鍵生成及びカプセル化にかかる時間は平均及びロバストな変動係数の結果から X25519 よりも高速かつ安定していると言える。一方、デカプセル化については、図 3 の分布の形状及びロバストな変動係数の結果から ML-KEM 512 では上振れしやすい傾向があると考えられる。

また、いずれのアルゴリズムも CPU 使用率やメモリ使用量等の計算資源の観点では負荷は極めて低く、今回使用した端末以上のスペックであれば暗号処理の負荷は問題にならないと考える。

ML-KEM の公開鍵・秘密鍵・暗号文のサイズは X25519 の 20 倍以上となる。

TLS ハンドシェイクの用途では、公開鍵は Client Hello で、暗号文は Server Hello でそれぞれ通信相手に送るため、公開鍵や暗号文が大きいならばそれだけ送信に必要なパケット数が増えることになる。

TCP の最大セグメントサイズは 1460 バイト(または 1440 バイト)がデファクトスタンダードであることから、ML-KEM 1024 では超過し、Client Hello 及び Server Hello で必要な TCP パケットが増加する。但し、Client Hello 及び Server Hello を送信するためには TCP パケットが 2 つあれば十分であった。

最後に、ハンドシェイク全体にかかる時間は X25519 では 3.69 ミリ秒、ML-KEM 512 では 3.72 ミリ秒と極端な差は見られない。一方、ハイブリッドである X25519MLKEM768 では 3.96 ミリ秒となっており、X25519 と比較した場合には 0.27 ミリ秒の差である。

この差は主に、ACK の送信から Client Hello の送信までの 0.06 ミリ秒、Client Hello の受信から Server Hello の送信までの 0.06 ミリ秒、Finished の受信から Finished の送信までの 0.13 ミリ秒に由来する。

尚、通常の用途ではそこまで大きな差であるとは言えない。

## 3.2 システムを使った実証実験

### 3.2.1 環境設定

積み立てプラットフォーム(以下、TPF)と呼ばれるシステムを対象システムとした。大まかな通信経路は以下の通りである:

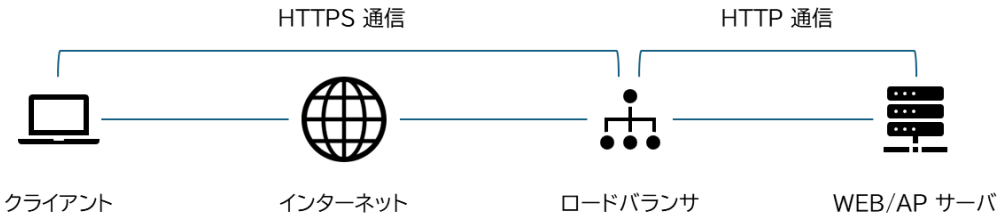


**図 5:システムの通信経路の概要図**

クライアントは「3.1 検証環境を使った基礎検証(鍵交換アルゴリズム)」に記載した EC2 インスタンスであり、以下のソフトウェアを使用している:

| | |
|---|---|
| OpenSSL | バージョンは 3.5.4。<br>TLS 通信のためのクライアントとして使用した。 |
| Nginx | バージョンは 1.28.0。<br>また、OpenSSL 3.5.4 を用いてビルドしている。<br>負荷テストにおいて鍵交換アルゴリズムを指定するために<br>JMeter のリバースプロキシとして使用している。 |
| Apache JMeter | バージョンは 5.6.3。<br>負荷テストで使用した。 |

実証実験にあたって F5 から提供されたロードバランサーは以下の通りである:

| | |
|---|---|
| BIG-IP | バージョンは 17.75.1.3。<br>X25519MLKEM768 での鍵交換に対応している。<br>尚、ML-DSA 等の耐量子計算機暗号のデジタル署名には未対応。 |

### 3.2.2 留意事項

執筆時点に於いて、Nginx は 1.28.1 が最新のバージョンとなっている。

また、BIG-IP 17.75.1.3 の耐量子計算機暗号への対応はソフトウェアレベルでのものである。尚、本バージョンでは X25519 もソフトウェアでの処理となっている。

### 3.2.3 検証項目及び検証概要

今回の実証実験では、性能に関して以下の項目について、X25519 を使用した場合と X25519MLKEM768 を使用した場合とでの差を検証した：

- ハンドシェイクの確立にかかる時間
- 初回接続時の通信遅延
- ロードバランサーの負荷
- 長時間通信（セッション維持）が安定して行えるか

また、既存アプリケーション（WEB サーバ）との統合の観点で負荷テストを実施した。

ハンドシェイクの確立にかかる時間については、クライアントからロードバランサーに向けてフルハンドシェイクを 1000 回行い、その通信データを基に分析を行っている。尚、サーバー及び認証局（DigiCert）の署名アルゴリズムとして 2048-bit RSA が用いられている。

セッション維持は、事前にセッションチケットを取得した上で、セッション再開によるハンドシェイクを 1000 回行い、その通信データを基に分析を行っている。

また、負荷テストは、過去に実施された負荷テストと同一のシナリオで負荷テストを実施し、暗号を変更したことに起因するエラー等の有無を調査する。

### 3.2.4 検証結果

#### 3.2.4.1 ハンドシェイクの確立について

まず、フルハンドシェイクによる TLS セッションの確立にかかった時間の平均は以下の通りであった：

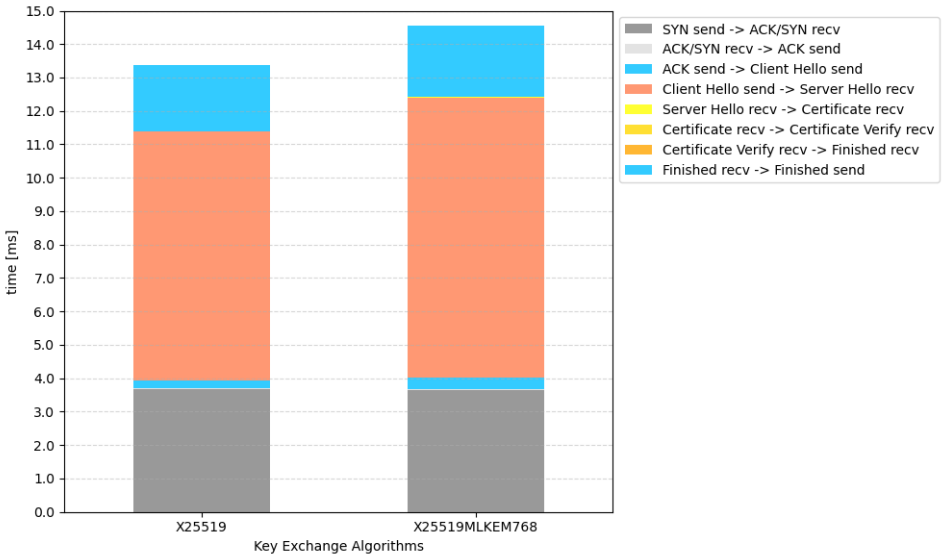


**図 6：フルハンドシェイクによる TLS セッションの確立にかかった時間**

具体的な数値及び標準偏差は以下の通りである：

| 鍵交換アルゴリズム | 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） |
| 合計 | 13.372453 | | 14.553073 | |

表 5：フルハンドシェイクによる TLS セッションの確立にかかった時間（単位：ミリ秒）

この結果について、以下の三点が指摘できる：

1. 基礎検証の結果と比べてハンドシェイクの確立までにかかった時間が 3.6 倍以上となっている
2. Client Hello の送信から Server Hello の受信までにかかる時間の変動係数は、X25519 の場合で 1.73、X25519MLKEM768 の場合で 1.65 と極めて大きい
3. X25519 では ACK の送信から Client Hello の送信までにかかる時間の変動係数が 0.87 と比較的大きい

基礎検証では EC2 インスタンス間の通信であるのに対し、TPF との通信ではインターネットを経由していることから、これらの事象には通信品質が影響している可能性が考えられる。

そこで、まずは 3 の変動の大きさについて考察する。

変動の大きさは外れ値によってもたらされていると考えられる。そこで、各ハンドシェイクにおいて Client Hello の送信から Server Hello の受信までにかかった時間を調査した。

多くのハンドシェイクでは Client Hello の送信から Server Hello の受信までにかかる時間は 4 ミリ秒から 25 ミリ秒の範囲であった。これに対し、30 ミリ秒以上かかっているハンドシェイクでは全て 90 ミリ秒から 105 ミリ秒の範囲にあることが判明した。実際の通信を確認すると、これらの場合では TCP の再送が発生していた。再送の件数は X25519 の場合が 22 件、X25519MLKEM の場合が 26 件であった。以降では外れ値として除外する。

また、X25519 では ACK の送信から Client Hello の送信までにかかる時間について、1 件のみ 6.95 ミリ秒かかっていた。それ以外は 0.5 ミリ秒以下であったことから、これも外れ値として除外する。

TCP 再送が発生したもの及び外れ値を除いた場合の時間の平均は以下である：

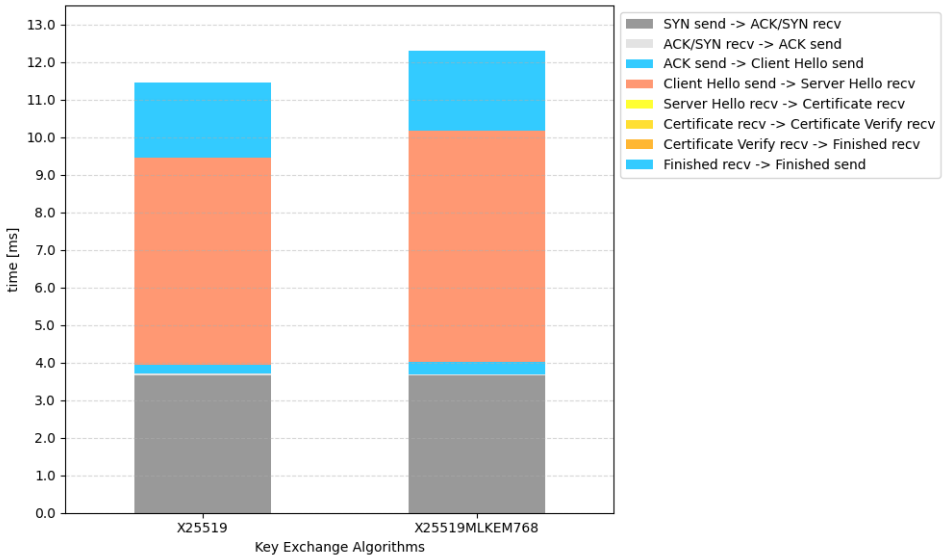


図 7:フルハンドシェイクによる TLS セッションの確立にかかった時間（外れ値を除く）

このときの Client Hello の送信から Server Hello の受信までと、確立にかかった時間の具体的な数値及び標準偏差は以下の通りである（単位：ミリ秒）：

| 鍵交換アルゴリズム | X25519 | X25519MLKEM768 |
|---|---|---|
| CLIENT HELLO SEND -> SERVER HELLO RECV | 5.515653 （±1.288781） | 6.135297 （±1.289622） |
| 合計 | 11.438157 | 12.301078 |

表 6:フルハンドシェイクによる TLS セッションの確立にかかった時間（外れ値を除く、一部抜粋）（単位：ミリ秒）

Client Hello の送信から Server Hello の受信までにかかる時間の標準偏差はかなり小さくなっている。この点について、まず、SYN パケットの送信から ACK/SYN パケットの受信までの変動係数の比較から、TPF との通信の場合の通信時間は基礎検証の場合と比べて

2 倍から 3 倍ばらつきやすいことが推察される。そして、TPF との通信の場合の Client Hello の送信から Server Hello の受信までにかかる時間の変動係数は基礎検証の場合の 2.3 倍または 2.6 倍となる。これらから、TCP 再送が生じたデータを除去したあとの平均及び標準偏差は妥当な値と考えられる。

次に、TCP 再送及び外れ値を除去した後の結果と基礎検証との結果を比較して通信速度の影響を検討する。TPF とのハンドシェイクの各パートの平均時間を基礎検証の場合と比較すると、概ね 3 倍から 3.5 倍の範囲にある。そして、SYN パケットの送信から ACK/SYN パケットを受信するまでの時間から、通信速度は概ね 2.7 倍程度になっていると考えられる。従って、ハンドシェイクの確立にかかる時間が 3.1 倍となるは一見尤もらしく思える。

しかし、Finished の受信から Finished の送信までの時間も 3.1 倍以上となっている。この過程で通信が行われた形跡は認められなかったことから、暗号処理でも基礎検証の場合より時間が掛かっていることになる。

この点について、基礎検証のときと比較して証明書のチェーンの深さが一段深いことやサーバー証明書・中間認証局証明書の記載内容が複雑となっていること、証明書のサイズの増加によって Certificate Verify や Finished で署名や認証の対象となるハンドシェイクコンテキストサイズも増加することが処理時間の増加に繋がっていると思われる。

最後に、鍵交換アルゴリズムを変えることでハンドシェイクの確立までにかかる時間の平均に 0.86 ミリ秒の差が生じている。基礎検証の場合の 3.2 倍となっている。

#### 3.2.4.2 セッションの維持について

セッション再開はいずれのアルゴリズムを使用しても問題無く行うことが出来た。セッション再開によってセッションを確立するのにかかった時間の平均は以下の通りである：

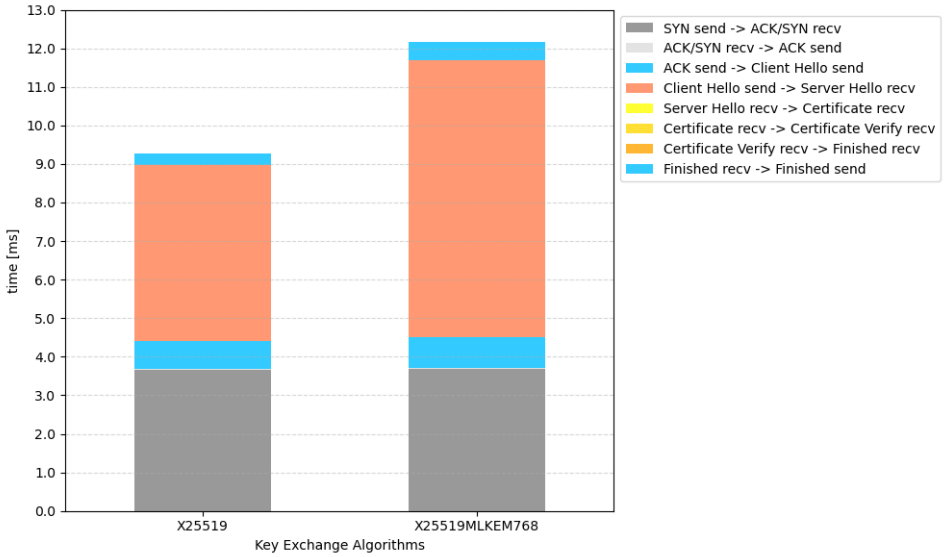


図 8:セッション再開によってセッションを確立するのにかかった時間

具体的な数値及び標準偏差は以下の通りである(単位:ミリ秒):

| 鍵交換アルゴリズム | 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) |
| 合計 | 9.261764 | | 12.154321 | |

表 7:セッション再開によってセッションを確立するのにかかった時間(単位:ミリ秒)

TCP 再送及び外れ値を除去する前のフルハンドシェイクのときの X25519 と X25519MLKEM768 では、Client Hello の送信から Server Hello の受信までの時間

の平均の差は 1.18 ミリ秒であった。これに対し、セッション再開では 2.89 ミリ秒と拡大している。加えて、Client Hello の受信から Server Hello の受信にかかる時間の変動係数も、X25519 では 1.61、X25519MLKEM768 では 2.06 と極めて大きい。

そこで、フルハンドシェイクのときと同様に、TCP 再送が発生しているデータを除去する。

鍵交換アルゴリズムが X25519 の場合に発生した TCP 再送の件数は 7 件、X25519MLKEM768 の場合に発生した TCP 再送の件数は 30 件であった。この差が平均の差に表れていると考えられる。

また、前述の通り、フルハンドシェイクでの TCP 再送の件数は、鍵交換アルゴリズムが X25519 のときに 22 件、X25519MLKEM768 のときに 26 件であった。通信量が減ると考えられるセッション再開で X25519MLKEM768 を採用した際に、却って TCP 再送が増えたことの妥当性について考察する。

クライアントが送信する Client Hello からサーバーが送信する Finished までの範囲で、クライアント及びサーバーが送信した TCP ペイロードのサイズ及び送信されたパケット数(推定)の合計は以下の通りであった:

| | フルハンドシェイク | | セッション再開 | |
|---|---|---|---|---|
| | X25519 | X25519MLKEM768 | X25519 | X25519MLKEM768 |
| **クライアント** | 455 | 1631 | 998 | 2174 |
| **サーバー** | 3661 | 4749 | 225 | 1313 |
| **合計パケット数** | 4 | 6 | 2 | 3 |
| **パケットロス率** | 0.55% | 0.43% | 0.35% | 1% |

**表 8:クライアント及びサーバーが送信した TCP ペイロードのサイズ(単位:バイト)及び送信されたパケット数(推定)の合計**

図 9 は、フルハンドシェイク及びセッション再開の実験中にいつ TCP 再送が発生したかを示している。尚、各実験中のハンドシェイクは 5 秒おきに行われた:

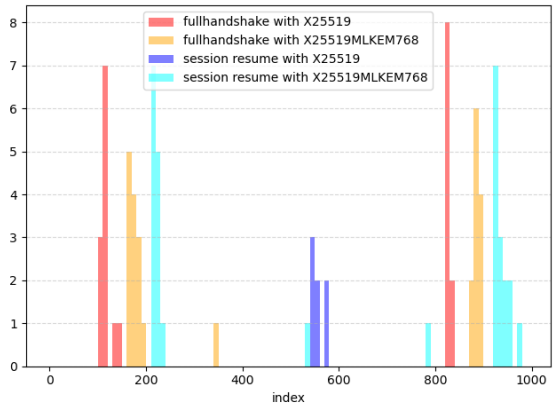


**図 9:フルハンドシェイク及びセッション再開の実験中の TCP 再送の発生タイミング**

図 9 から分かる通り、X25519 及び X25519MLKEM でのフルハンドシェイクと X25519MLKEM768 でのセッション再開では集中的な TCP 再送が二度発生している。また、山の間隔が同程度であることから周期的に発生している可能性がある。周期性を前提とすると、X25519 でのセッション再開でも実験の実施タイミングによっては TCP 再送の発生数が 2 倍程度になっていた可能性がある。更に、X25519MLKEM768 でのセッション再開では、二つ目の山で上振れしていることが示唆される。

これらのことから、セッション再開によるハンドシェイクかつ鍵交換アルゴリズムが X25519MLKEM768 の場合の 30 件の TCP 再送の件数は、他の実験結果と比較するとやや高いものの、インターネット経由の通信であることを考慮するとあり得る範囲と思われる。

フルハンドシェイクのときと同様に TCP 再送の影響を除去すると、平均時間は以下の通りとなった:

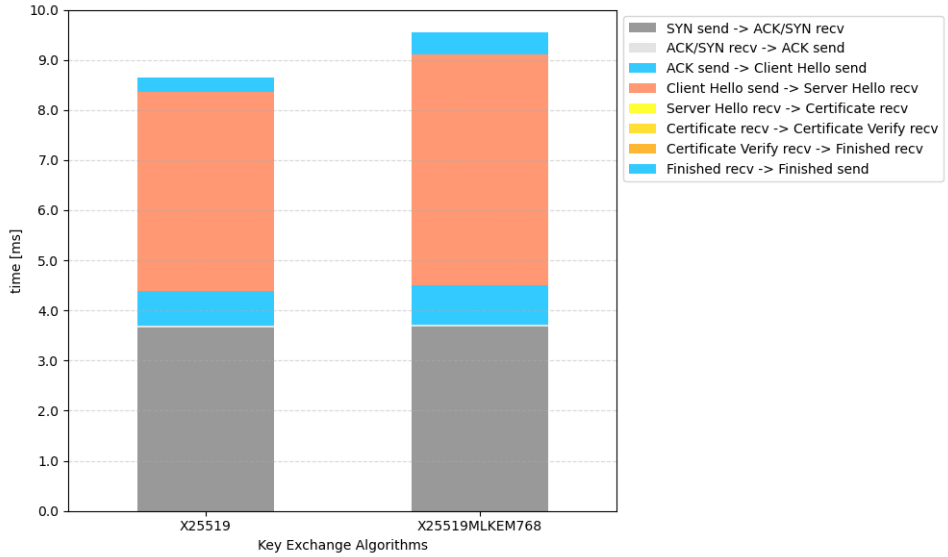


**図 10:セッション再開による TLS セッションの確立にかかった時間(外れ値を除く)**

このときの各パートにかかった時間(一部抜粋)は以下の通りである:

| 鍵交換アルゴリズム | 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) |
| 合計 | 8.646324 | 9.557866 |

表 9:セッション再開による TLS セッションの確立にかかった時間(外れ値を除く、一部抜粋)(単位:ミリ秒)

フルハンドシェイクと異なり、ACK パケットの送信から Client Hello の送信までにかかる時間が増加している。一方、Client Hello の送信から Server Hello の受信までの時間とサーバーから Finished を受け取ってからクライアントが Finished を返すまでの時間は短縮されている。

鍵交換自体は行われるため、セッション再開を利用してもハイブリッド化による時間の増加を抑えることはできない。しかし、鍵交換アルゴリズムとしてどちらを選んでも、ハンドシェイク全体としてはセッション再開による短縮の恩恵を同程度に受けられるといえる。

### 3.2.4.3 負荷テストの結果について

今回のシステムに対する負荷テストでは、ログイン及びウェブアプリケーションに対する一定の操作を行うシナリオで実施した。負荷の程度は、システムが想定する程度のものからその数倍に至るものまで実施した。

高負荷下ではアプリケーションサーバ側の CPU 使用率が 100%近くになる状況があったが、そのような状況にあってもロードバランサーの CPU 使用率が 8%を超えることはなかった。具体的なリソースのデータは以下の通りである。

まず、負荷実験日の実験時間中及び実験時間外における具体的な CPU 使用率は以下の通りであった:

| | X25519 | X25519MLKEM768 | 実験時間外 |
|---|---|---|---|
| 平均値 | 6.69 | 6.63 | 6.81 |
| 最小値 | 5.64 | 5.94 | 4.75 |
| 第一四分位 | 6.54 | 6.42 | 6.64 |
| 中央値 | 6.83 | 6.63 | 6.85 |
| 第三四分位 | 6.88 | 6.85 | 7.07 |

| | | | |
|---|---|---|---|
| **最大値** | 7.38 | 7.23 | 7.74 |
| **標準偏差** | 0.47 | 0.41 | 0.43 |

表 10:実験時間中及び実験時間外におけるロードバランサーの CPU 使用率

また、TMM メモリ使用率は以下の通りであった:

| | X25519 | X25519MLKEM768 | 実験時間外 |
|---|---|---|---|
| **平均値** | 6.89 | 6.89 | 6.89 |
| **最小値** | 6.89 | 6.89 | 6.89 |
| **第一四分位** | 6.89 | 6.89 | 6.89 |
| **中央値** | 6.89 | 6.89 | 6.89 |
| **第三四分位** | 6.89 | 6.89 | 6.89 |
| **最大値** | 6.90 | 6.90 | 6.90 |
| **標準偏差** | 0.004 | 0.004 | 0.002 |

表 11:実験時間中及び実験時間外におけるロードバランサーの TMM メモリ使用率

メモリや CPU の使用率という観点では、X25519 と X25519MLKEM768 との間だけでなく、実験時間外のハンドシェイク自体が行われていない状況と比較しても大きな差があるとは言えない結果であった。尚、スワップ領域は使われなかった。

これらの結果から、高負荷下においては、先にシステム側に限界が来るため、耐量子計算機暗号に切り替えたことによってロードバランサーがボトルネックになるとはいえないといえる。加えて、そのような状況下では X25519 と X25519MLKEM768 の性能差は顕在化しないと考えられる。

また、システムが想定する負荷の範囲内であれば負荷テストは問題無く実行出来たため、耐量子計算機暗号が原因でサービスが機能しなくなるといったことはないと判断した。

次に、通信量という点でロードバランサーのリソースデータを基に Total Bytes In/Out や Total Throughput を比較すると、X25519 から X25519MLKEM768 に変えることで 5%程度の増加となった。

但し、負荷テスト中はシナリオを実施するために必要な情報のみを JSON 形式で送受信しており、実務的には必要な画像などの他のリソースは取得していない。そのため、実務的な通信であれば増加分は更に低いと思われる。

## 3.3 検証環境を使った基礎検証(署名アルゴリズム)

### 3.3.1 検証環境

鍵交換アルゴリズムと同一である。

### 3.3.2 検証項目

まず、純粋な署名アルゴリズムの性能の測定として、鍵の生成、署名の生成・検証にかかる時間を調査した。

検証対象としたアルゴリズムは、2048-bit RSA、3072-bit RSA、P-256、ML-DSA 44 とした。尚、Falcon は標準化対象に選定されたものの 2026 年 1 月時点では FIPS の最終版が承認されていないことから、SLH-DSA は性能面で常用に堪えるものではないことから今回は対象外とした。

各アルゴリズムについて、鍵の生成、署名の生成、署名の検証それぞれについてかかる時間を 1000 回分計測し、オーバーヘッドの影響がある初回を除いた 999 回分のデータを基に算出した。

次に、実際の通信において、TLS ハンドシェイクの確立までにかかる時間を調査した。この通信は、二つの異なるアベイラビリティゾーンにある EC2 インスタンス間で、一方をクライアント、他方をサーバーとして行う。このとき、サーバーには基礎検証用に用意したサーバー証明書及び認証局証明書を提示させる。

検証対象とした鍵交換アルゴリズムは、X25519、ML-KEM 512、ML-KEM 768、ML-KEM 1024、X25519MLKEM768 である。また、サーバー及び認証局の署名アルゴリズムとして 2048-bit RSA、3072-bit RSA、P-256、ML-DSA 44 を使用している。尚、ここでいうサーバーの署名アルゴリズムとは、Certificate Verify でサーバーが署名に使用するアルゴリズムを指す。認証局の署名アルゴリズムとは、証明書に付されている署名アルゴリズムを指す。

ハンドシェイクの回数は 1000 回とし、その通信データを基に分析を行った。

後に提示する結果において、通信時間はクライアント及びサーバーで記録された時間の差分によって算出している。尚、クライアントとサーバーを動かしている EC2 インスタンスは同一の NTP サーバ(169.254.169.123)と同期させ、キャプチャファイルに記録された時間は NTP サーバーとのずれを基に補正している。

また、参考までに、現在標準化候補として考察されている MAYO、CROSS、UOV、SNOVA をサーバーの署名アルゴリズムとして採用した場合の結果も一部掲載する。

### 3.3.3 検証結果

#### 3.3.3.1 性能の計測結果

まず、鍵及び署名のサイズは以下の通りである。幅があるものについては、今回の基礎検証での実測値に依る：

| | 検証鍵 | 署名鍵 | 署名 |
|---|---|---|---|
| 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 |

表 12：検証鍵・署名鍵・署名のサイズ（単位はバイト）

次に、純粋なアルゴリズムの性能について、鍵の生成にかかった時間の平均、標準偏差、四分位及びロバストな変動係数は以下の通りであった。尚、時間の分布については別紙報告書を参照のこと：

| | 平均 | 標準偏差 | 第一四分位 | 中央値 | 第三四分位 | ロバストな変動係数 |
|---|---|---|---|---|---|---|
| 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 |

表 13：署名アルゴリズムの鍵の生成にかかった時間の平均、標準偏差、四分位及びロバストな変動係数（平均、標準偏差、各四分位の単位はミリ秒）

次に、署名の生成にかかった時間の平均、標準偏差、四分位及びロバストな変動係数は以下の通りであった。尚、時間の分布については別紙報告書を参照のこと：

| | 平均 | 標準偏差 | 第一四分位 | 中央値 | 第三四分位 | ロバストな変動係数 |
|---|---|---|---|---|---|---|
| 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 |

表 14：署名の生成にかかった時間の平均、標準偏差、四分位及びロバストな変動係数（平均、標準偏差、各四分位の単位はミリ秒）

最後に、検証にかかった時間の平均、標準偏差、四分位及びロバストな変動係数は以下の通りであった。尚、時間の分布については別紙報告書を参照のこと：

| | 平均 | 標準偏差 | 第一四分位 | 中央値 | 第三四分位 | ロバストな変動係数 |
|---|---|---|---|---|---|---|
| 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 |

表 15：署名の検証にかかった時間の平均、標準偏差、四分位及びロバストな変動係数（平均、標準偏差、各四分位の単位はミリ秒）

また、ハンドシェイクの結果については別紙報告書及び CSV を参照のこと。

#### 3.3.3.2 計測結果に対する評価

鍵生成にかかる時間について、RSA は平均もロバストな変動係数も他と比べるとかなり大きいことが分かる。即ち、生成に時間が掛かり、その時間も比較的幅があるといえる。但し、現時点では TLS ハンドシェイク毎に鍵を生成するといったことはないため、TLS 通信という観点ではそれほど問題にはならないと考えられる。

署名の生成や検証にかかる時間は、サーバー側の Certificate Verity の生成やクライアント側の Certificate/Certificate Verify の検証にかかる時間に影響を与える。

RSA での署名の生成にかかる時間は P-256 と比較して 10 倍以上、ML-DSA 44 と比較しても 5 倍以上となっている。尤も、1 ミリ秒未満であることから、よほど高速に通信を確立したいといった要望が無い限り、クライアントにとっては致命的といえるような遅さとはいえない。一方で、サーバーにとっては署名アルゴリズムを変えることでわずかではあるが計算資源を節約できることになる。頻繁にハンドシェイクを行っている場合には考察の余地はある。

また、分布の形状及びロバストな変動係数の結果から、ML-DSA 44 は署名の生成にかかる時間がやや上振れしやすいと考えられる。

署名の検証にかかる時間は P-256 が最も遅いが、数十マイクロ秒のオーダーであり、またロバストな変動係数もそこまで大きくないことから、通常の用途ではどのアルゴリズムを選択しても大差はないといえる。

その上で敢えて評価をするならば、ML-DSA 44 は平均及びロバストな変動係数の結果から従来の暗号と比べて高速かつ安定していると言える。

検証鍵はサーバー証明書に含まれ、署名は Certificate Verify で送信するため、これらのサイズが大きい場合にはそれだけサーバーが送信する TCP パケットの数も増加することに

なる。

基礎検証においてサーバーが Server Hello から Finished までを送信するために使用した TCP パケットの数について、結果を一部抜粋すると以下の通りであった：

| 認証局 | 2048-BIT RSA | | | ML-DSA 44 | |
|---|---|---|---|---|---|
| サーバー | 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 |

表 16：Server Hello から Finished までを送信するために使用した TCP パケットの数（抜粋）

尚、この値は署名アルゴリズムだけではなく証明書の記載の内容や中間証明書の有無にも左右されることは注記しておく。基礎検証において使用した証明書のサイズは別紙報告書を参照のこと。

通信品質が理想的ではない場合は TCP 再送が発生し得る。TCP パケットの数が増えた場合、ハンドシェイク中に TCP 再送が発生する可能性は高くはなる。ハンドシェイク確立以降の通信が軽量であり、かつ可能な限りレイテンシを抑えたい場合には留意する必要があるだろう。加えて、輻輳ウィンドウの制限に留意する必要もある。

ハンドシェイクの確立にかかる時間について、以下の結論を得た：

1. サーバーと認証局で同じ署名アルゴリズムを使用することでハンドシェイクの確立が早くなる。
2. Server Hello で送信する暗号文のサイズと Certificate で送信する証明書のサイズの合計が増加することで却ってハンドシェイクの確立が早くなる場合がある。
3. サーバーの署名アルゴリズムとして 3072-bit RSA を使用する場合と ML-DSA 44 を使用する場合とでは、ML-DSA 44 を使用した方がハンドシェイクの確立が早い[3]。

尚、これらは EC2 インスタンス同士の通信という理想的な通信環境下での結論である。インターネットを経由する場合にはパケットロスの可能性もあることから、ML-DSA 44 と 3072-bit RSA との差はパケットロス率や輻輳制御によっては逆転しうる。

[3] この比較は、認証局の署名アルゴリズムとして P-256 を使用した場合及びサーバと同じ署名アルゴリズムを使用した場合のものである。

## 3.4 検証結果の考察

本節では、3.1～3.3 で得られた測定結果を踏まえ、PQC を TLS に適用した場合の影響を「性能」「通信量」「運用・互換性」の観点から整理し、実務上の示唆を 3 点提示する。

**（1）暗号処理時間そのものはボトルネックになりにくい**

基礎検証（4.1）では、X25519 および ML-KEM（512/768/1024）の鍵生成・カプセル化・デカプセル化はいずれも数十マイクロ秒程度であり、TLS ハンドシェイク全体（ミリ秒オーダー）に比べて十分に小さいことを確認した。従って、一般的な Web 用途においては「暗号計算の遅さ」よりも、ネットワーク遅延や再送の有無が体感性能に影響しやすい。

一方で、インターネット経由の実証（4.2）では、通信品質によるばらつき（外れ値）や TCP 再送の発生がハンドシェイク時間の分散を大きくし得ることも確認されている。よって、PQC 導入時の評価では平均値だけでなく、分散・外れ値（例：95/99 パーセンタイル）や再送率の観測が重要となる。

ただし、IoT デバイスや組み込み機器など計算資源が限られる環境では、暗号処理自体がボトルネックとなる可能性がある点には留意が必要である。

**（2）PQC 導入の主要な影響は「通信量の増加」に現れる**

ML-KEM は公開鍵・暗号文サイズが X25519（32 バイト）に比べて大きく、TLS ハンドシェイクでは ClientHello／ServerHello の送信データが増加する。基礎検証では、ML-KEM 1024 や X25519MLKEM768 は MSS を超過しやすく、パケット数増加の要因になり得ることを整理した。

もっとも、一般的な Web 利用では初期ウィンドウ拡大等により、パケットが増えても致命的な問題に直結しないケースが多いのではないかと考えられる。一方、回線が不安定な環境、帯域制限のある環境、もしくは低遅延・低ばらつきが要求されるシステムでは、パケット増が再送率を押し上げ、結果として“たまに遅い”現象が顕在化する可能性がある。

**（3）終端機器での PQC 移行が現実的か**

今回の実証環境では、ロードバランサ（F5 BIG-IP）が X25519MLKEM768 の鍵交換に対応しており、既存アプリケーション（Web/AP）側に大きな改修を加えずに検証できた。また、負荷試験においてロードバランサーCPU 使用率の顕著な増加は観測されず、想定負荷の範囲ではボトルネックになりにくいことが示唆されている。

一方で、実証検証時点での同機器は ML-DSA 等の PQC 署名に未対応であり、署名・証明書の PQC 化は鍵交換よりも遅れる計画となっている。従って、実務的には「鍵交換（KEM）を

先行し、署名・PKI は当面従来方式(RSA/ECSDA)を利用し、対応状況を見て段階的に進める」方針が現実的である。

## 3.5 コラム: 耐量子コンピュータ暗号(PQC)標準化の最新動向

DigiCert Inc. 業界スタンダード VP ティモシー・ホロビーク(Timothy Hollebeek)

デジサート・ジャパン合同会社シニアプロダクトマーケティングマネージャー 林 正人

### ― IETF とパブリックトラスト PKI ―

NIST は主に暗号標準の策定を担っている米国の団体ですが、全世界的に相互接続運用ができるインターネットの共通技術標準を定めるボランティア団体が Internet Engineering Task Force(IETF、インターネット技術タスクフォース)になり、現在 PQC 標準のインターネット実装を議論・策定しています。PQC に関しては **PQUIP**、**LAMPS**、**TLS** といった主に3つのワーキンググループにおいて、それぞれの議論が進んでいます。

また不特定多数がインターネット上で「相手が本物であること」を証明し、通信を暗号化、改竄を検知できる仕組みとして提供されるパブリック PKI の PQC 標準化に関しては、電子証明書を発行する認証局とパブリック TLS 証明書を表示するブラウザが中心となって運営される会員任意団体 CA/B フォーラムが担っておりこちらも複数のワーキンググループに分かれて議論・標準の策定を行なっています。

デジサートのような認証局が正式に PQC 対応した証明書を発行するためには、IETF での署名の RFC 標準化、CA/B フォーラムでの BaselineRequirement への追加が行われ、また PQC 標準を定めた NIST が暗号モジュールの安全性評価基準として定める FIPS140-3 対応 HSM 等デバイスが必要となります。しかしながら、デジサートではこれらの標準化を見越して検証用証明書をさまざまな環境、利用シーンでテストすることで備えています。

---

**IETF PQUIP ワーキンググループ**

**PQUIP（Post-Quantum Use In Protocols）**は、IETF における PQC 移行のための横断的な調整・ガイダンスを行い各プロトコル WG が進める PQC 共通課題・用語・移行上の知見を整理します。PQC 対応の進捗状況を横断的に可視化し、WG が存在しない領域の受け皿も担います。

---

### IETF LAMPS ワーキンググループ（X.509 / PKI）

**LAMPS（Limited Additional Mechanisms for PKIX and S/MIME）** ワーキンググループは、X.509 証明書に関連する標準を担当しています。PQC に関わる主な文書は以下のとおりです。（IETF 文書の策定中のステータスは、ドラフトが作成され6ヶ月の有効期間の間に議論が重ねられ新たな加筆修正が行われている状態を示しています。この議論が成熟して合意を得ると RFC 標準として公開されますが、6ヶ月議論も加筆も行われないドラフトは失効されます。）

- **RFC 9881**：X.509 向け ML-DSA アルゴリズム識別子（公開済み）
- **draft-ietf-lamps-x509-slhdsa**：X.509 向け SLH-DSA アルゴリズム識別子（策定中）
- **draft-ietf-lamps-pq-composite-sigs**：X.509 向け複合（コンポジット）ML-DSA 署名（策定中）
- **draft-ietf-lamps-pq-composite-kem**：X.509 向け複合（コンポジット）ML-KEM（策定中）

---

### IETF TLS ワーキンググループ

TLS ワーキンググループは TLS に関する技術標準を担当しています。この標準に関しては **X25519MLKEM768** を用いたハイブリッド鍵交換が、すでに Chrome や Cloudflare により本番環境で展開されています（**draft-ietf-tls-ecdhe-mlkem**）。これは「**今収集し、後で解読（Harvest Now, Decrypt Later）**」攻撃に対処するものです。

TLS ワーキンググループでは、TLS の PQC 標準を実用できるようにするため下記のような標準化が積極的に進められています。

- **draft-ietf-tls-mldsa**（Hollebeek、Schmieg、Westerbaan）（策定中）

TLS 1.3 における ML-DSA を用いた認証方式を規定し、**ML-DSA-44**、**ML-DSA-65**、**ML-DSA-87** の SignatureScheme 値を定義、署名検証方法を指定しています。

- **draft-reddy-tls-composite-mldsa**(策定中)

  TLS 1.3 向けの複合(コンポジット)ML-DSA 認証を定義しています。

---

**ハイブリッド vs ピュア PQC vs デュアルの議論**

現在 IETF で進行中の大きな議論の一つが、PQC 移行のアプローチです。

- **ピュア PQC**:PQC アルゴリズムのみを使用
- **ハイブリッド/コンポジット**:1 つの証明書に従来暗号と PQC を組み合わせ、両方を破らないと安全性が損なわれない方式(RFC 9794 で定義)
- **デュアル証明書**:従来暗号と PQC の証明書チェーンを並行して維持

**PQC 移行のためのシナリオの比較**

| | ピュア PQC | ハイブリッド | コンポジット | デュアル |
|---|---|---|---|---|
| 構造 | 1枚の証明書(PQC1枚のみ使用。従来暗号は排除。) | 1枚の証明書(従来暗号鍵を Subject Public Key Info に置き、PQC 暗号鍵を拡張領域に配置) | 1枚の証明書(従来暗号と PQC 暗号の鍵を Subject Public Key Info に配置) | 独立した 2 枚の証明書 |
| 主な目的 | 長期的な安全性と効率化 | 未知の脆弱性への耐性 | 強固な身元証明(両暗号要チェックのため厳格) | 最高の互換性 |
| 後方互換性 | なし(非対応機器は通信不可) | 中～高(拡張領域で対応) | 低い(システムは新しい複合アルゴリズムを理 | 最高(従来通り動作) |

| | | | | |
|---|---|---|---|---|
| | | | 解できる必要あり。理解できない古いシステムはエラー。） | |
| データ量 | PQC のみのため比較的小さい | 大きい（2 組分の鍵と署名を一部結合） | 最も大きい（2 組分の鍵と署名を全結合） | 選択した 1 組分のみ（効率的） |
| 実装の複雑さ | 低（シンプル） | 中（計算処理が増加） | 高（規格の再定義が必要） | 中（サーバー側の管理負荷） |
| 課題 | 旧システムの完全切り捨て | 通信・計算能力問題<br><br>一部システムの切り捨て | 通信・計算能力問題<br><br>新たな通信規格と機器へのパッチ<br><br>一部システムの切り捨て（ハイブリッドより多い） | 運用管理が倍増<br><br>弱い暗号を意図的に狙うダウングレード攻撃脆弱性 |
| コンセプト | 完全移行 | 多層防御 | 厳格なセキュリティ | 使い分け、併存 |

IETF LAMPS ワーキンググループでは、PQC と従来アルゴリズム（RSA、ECDSA、Ed25519、X25519 など）を組み合わせた**コンポジット仕様**が策定されています。

ただ、各国・地域で推奨方針は異なり、移行期間中はハイブリッド方式を勧めるケースもあれば、米国の **CNSA 2.0** ガイダンスのように、国家安全保障システムでは**ピュア PQC への直接移行**を推奨する例もあります。

また、コンポジット署名の標準化を進める過程で、当初の想定よりも上記表示した通りシステム側が従来存在しなかった証明書の“Subject Public Key Info”を理解できるようにする必要があり、これらの規格化やネットワークトラフィックのサイズの変更をインターネットスタックの全てに**実装する方法やその評価が大幅に複雑で多くの議論を要する**ことが明らかになり

ました。その一方で、**ピュア PQC アルゴリズムに対する検証と信頼は向上**してきています。これによりピュア PQC への直接移行の議論も多くなされています。しかしながら多くの機器によって構成されるインターネットの移行に関してはセキュリティを担保しながら切り捨てを最小限にする必要があるため PQC と RSA、ECC のコンポジットが移行議論の中心であることに変わりはありません。

一方、パブリック PKI 標準の急な変更への影響を受けない方法（例えば ATM 網、金融独自ネットワークなどの金融システム移行）として ANSI（米国国家規格協会）において金融業界向けの暗号・セキュリティ標準を定めている X9 では、PQC の実装に向けて準備を進めており、CA/B フォーラムの影響を排除した独自 PKI を確立し実装をすすめています。

---

**CA/B フォーラムにおける議論**

PQC の議論は CA/B フォーラムにも広がっていますが、Baseline Requirement への実装などにはワーキンググループごとに温度差があります。
S/MIME とコードサイニングワーキンググループでは、PQC に関するバロットが可決（例えば SMC-013）、もしくは提案段階にあり各 Baseline Requirement への追加が積極的にすすめられています。
しかし、パブリック TLS の Baseline Requirement の追加に関しては、TLS ハイブリッド鍵交換の実験がすすめられてきたのは周知の事実ですがその鍵交換だけに留まっており、署名認証を行う本当の意味での PQC TLS 通信の実現まではまだ出口が見えていないのが現状です。背景として、パブリック TLS 全体への影響度の大きさから慎重な意見があります。例えば、PQC アルゴリズムが想定より早く破られたり、実装上の欠陥が見つかった場合、一度実装として含めてしまうとパブリック TLS のアキレス腱になってしまいます。そして、当然 HSM、スマートカード、USB トークンの FIPS140-3 認証の遅れ、PQC による通信パフォーマンスへの課題など PQC の移行アプローチで紹介した課題と実装方式に関する議論などが山積しています。
現在進行中であるデジタル証明書の有効期限短縮化にも数年もの移行期間が必要になっている現状で、PQC へ向けた暗号アルゴリズム移行の仕組みの不在は業界全体のチャレンジとなっています。
そのような条件の中、ブラウザおよび OS のルートプログラム・ルートストアへの PQC 追加は世界中の信頼の基盤になるため技術互換性の確保など多くの前提条件が整う必要があり今後も保守的な更新になることが想定されます。

---

**その他の標準の PQC 検討状況**

また金融業界に直接影響を与えそうなトレンドしては EU において、政策・法規制・技術標準・産業支援を一体化した多層構造として準備が行われています。IETF や CA/B フォーラムの決定を参照しながらも強制力が明確なのが特徴となります。
まず Quantum Europe Strategy（量子欧州戦略：2025）で PQC へのビジョンが示されました。そして NIS2 指令（改正ネットワーク・情報システム指令）により金融機関を含む重要・必須事業者に対する法制化、2026年に定められる予定の EU Quantum Act によって、実行に移すための技術・製品・人材を域内で育成することを推進する予定です。（いずれも指令ですので EU 各国での法制化が必要です。）さらにそれらの実装を具体化する技術標準レイヤとして、ETSI と ENISA が具体的な実装要件を定めることで監督当局・事業者の実装判断を支援する構造になっています。

例えば法的な効力を判断する基準となるような長期署名フォーマット（AdES：:Advanced Electronic Signature）の PQC 対応やそれらの署名をするための証明書の発行を担う認証局（QTSP：Qualified Trust Service Provider）の要件更新、そして eIDAS2.0 で各国に義務付けられた EUDI Wallet の PQC 対応が ETSI で定められ、そのための推奨暗号方式や技術ガイダンスを ENISA が定めます。EUDI Wallet は、金融 KYC とも強く結び付いていますので動向が注目されるところです。

---

**最新情報について（参考リンク）**
可能な限り現在の議論を簡潔にご説明しましたが今後の進捗は以下のリンクで随時確認いただければ幸いです。IETF へは最新の情報の確認と同時に、メーリングリストへの登録で誰でも質問、要望を行うことができます。

- IETF PQUIP ワーキンググループ

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

- IETF LAMPS ワーキンググループ

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

- IETF TLS ワーキンググループ

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

- NIST PQC プロジェクト

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

- CA/B フォーラム

  https://cabforum.org/

- PKI Consortium PQC リソース

  https://pkic.org/

- ANSI X9 リソース

  https://x9.org

- Quantum Europe Strategy リソース

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

# 4. 暗号移行のアプローチ

本章では、PQC 移行を組織的に進めるための実務アプローチを提示する。

## 4.1 暗号移行の進め方

PQC 移行は、TLS や証明書、署名、鍵管理など広い範囲に影響するため、複数年にわたる全社的な取り組みとして推進体制（ガバナンス）を確立することが重要である。

また、PQC 移行は、IT 部門、セキュリティ部門だけで対応できるものではない。以下のように役割を含む横断チームを設置し、意思決定を迅速にすることが望ましい。

| 担当 | 主な役割 |
|---|---|
| 統括責任者 | チームの責任者、社内社外への説明、移行ロードマップの策定 |
| 推進担当 | 暗号の棚卸し、移行フレームワークの策定、全体進捗管理 |
| インフラ／ネットワーク担当 | TLS/VPN/負荷分散装置の暗号移行と検証 |
| アプリ開発担当 | アプリ内の暗号ライブラリ、署名、プロトコルの暗号移行と検証 |
| PKI／鍵管理担当 | 証明書ライフサイクル、鍵の保護方針の検討 |
| 調達担当 | サプライチェーンの PQC 対応状況と契約・SLA の確認 |
| 支援担当 | PQC に関する国内外の法規制の調査とチームへの周知 |

表 1 PQC 移行チームの役割（例）

横断チームの設置と並行して、この段階で新規/更改システムに適用する暗号の方針を定めておき、対応が二度手間になることを防ぐのが望ましい。

| 検討箇所 | 方針（案） |
|---|---|
| 設計の原則 | 暗号を切替可能にするクリプト・アジリティに関する設計を要求事項に含める |
| 通信 | 既存機器との互換性を確保するため、ハイブリッド（従来暗号＋ML-KEM 等）を採用する |
| 署名/証明書 | 長期利用するコードや文書で使用する署名について PQC の採用方針を定める |
| 例外管理 | 新規／更改段階で PQC 対応が難しい場合は、期限や対応方針について定め、管理する |

表 2 新規／更改システムに適用する暗号方針（案）

暗号は通信先／通信元の相互での対応することが前提となるため、外部委託先／クラウド・業界の共通基盤の対応状況で PQC 移行が行えないケースが考えられる。そのため、関係するベンダーに対し PQC の対応予定や移行方針を確認し、横断チームで管理していくことが望ましい。

PQC 移行は長期間に渡るため、暗号移行だけでなく、プロジェクト推進、暗号の棚卸し・リスク評価・検証などでも費用が発生するため、工程別に見積もり予算を確保することが望ましい。

| フェーズ | 主な見積観点 |
|---|---|
| プロジェクト推進 | 社内外ステークホルダーとの調整、全体進捗管理 |
| 暗号の棚卸し | 調査費用、ヒアリングによる工数<br>（利用する場合は調査ツールなども費用に含める） |
| リスク評価 | 検討工数、移行ロードマップの策定 |
| 検証 | 性能影響、互換性、運用手順確認 |
| 移行 | 機器更改、ソフト更新、暗号ライブラリの更新、テスト |

表 3 フェーズごとの見積観点

## 4.2 暗号アセットの棚卸し

暗号の棚卸しをする目的は、「どこで暗号が使われているか」を把握し、移行対象と優先度付けで必要な材料をそろえることである。

この段階で重要なのは、完全な網羅性を無理に最初から目指さないことである。まずは外部へ公開されている重要領域から着手し、未把握領域を明確化して段階的に埋めることが望ましい。

まずは量子コンピュータ出現による影響の大きさを考えて、まずは公開鍵暗号の利用箇所から優先的に調査し、その次にデータ保存などで利用される共通鍵暗号へ調査を広げる方法が考えられる。このとき、調査対象を場当たり的に洗い出すのではなく、システム内で公開鍵暗号が利用され得る箇所を整理し、どの領域から確認するか優先順位を付けることが有効である。

図 4 はシステムにおいて公開鍵暗号が利用される箇所を整理したものである。例えば、まずは外部インターネットと接続する経路上の機器・サービス（ブラウザ／アプリとCDN・WAF・FW・LB間の通信、公開サーバー、認証基盤等）を確認し、その後、VPN機器や他社データセンター接続、クラウド接続、自社データセンター内部の通信といった形に調査対象を広げる方法が考えられる。特に外部公開されている通信路、認証に関わる機能、複数の組織・複数の拠点をまたぐ接続に関しては、調査する際の確認先が複数になることが考えられるため、調査に時間がかかることを念頭に置く必要がある。

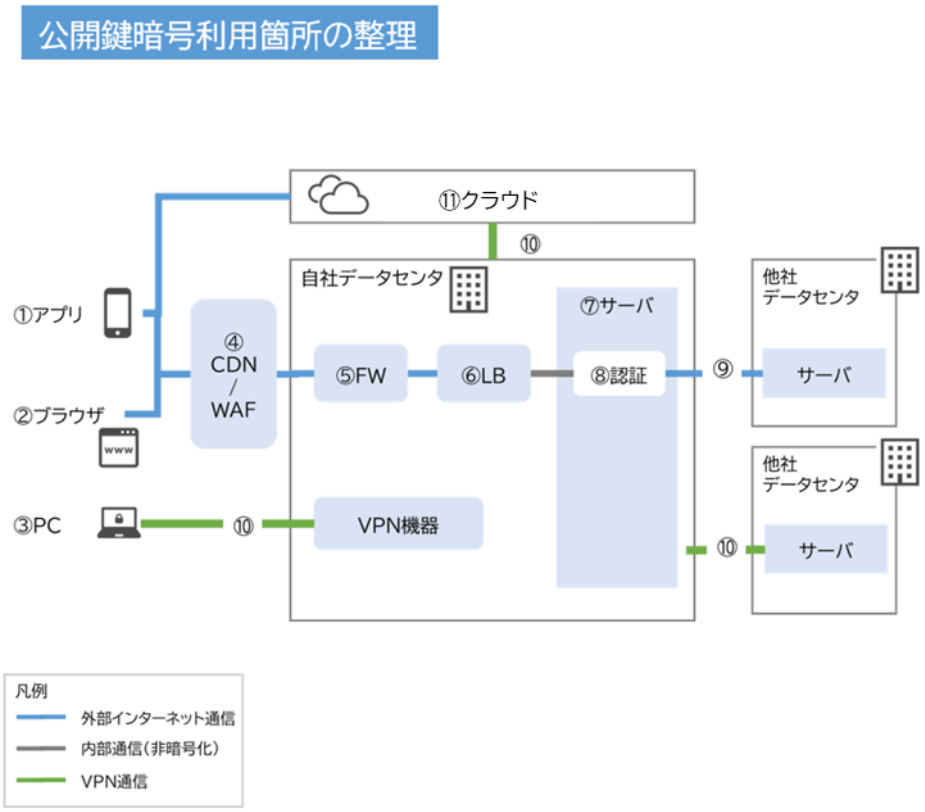


図 4 公開鍵利用箇所の整理

暗号の棚卸しにあたっては、英国 NCSC によると、トップダウンとボトムアップの両方のアプローチを取ると良いとされている。これは、業務フローや図 5-2-1 の様なサービス構成から全体像を把握するトップダウンの調査から始め、ソースコード、設定情報、証明書配置、通信ログ、パケットキャプチャなどから解析的に調査するボトムアップの調査を行うことでトップダウンでの考慮漏れを補完する進め方である。

| 調査手法 | 調査観点 | 調査内容 |
|---|---|---|
| トップダウン | 業務フロー・サービス | 設計資料などから業務フローやサービスを調べ、どの経路でデータが流れ、どこで暗号や署名が利用されるかを調べる。 |
| ボトムアップ | ソースコード・通信ログ | アプリケーションのソースコードを調査したり、通信のパケットをキャプチャし分析するなど、暗号の利用を解析する。<br>収集、 分析するための専用のツールが存在する。 |

表 5 暗号の調査手法

## 4.3 暗号インベントリの作成と管理

暗号インベントリは、調査した暗号の棚卸し結果を継続的に更新・参照できる台帳のことである。

暗号インベントリを作る際は、チーム全体で理解でき、かつ、次のリスク評価と優先順位付けで利用できる情報を含めた形で作成することが望ましい。逆に、暗号インベントリに不必要な情報を大量に含めてしまうと、調査が難航したり、リスク評価が困難になることが考えられる。

以下は、暗号インベントリとして最低限含まれるべきと考えられる項目である。

| 項目 | 観点 |
|---|---|
| システム／サービス関連情報 | 名称、主管部門、担当者、業務システムか業務外システムか？ |
| 暗号利用箇所 | Web サーバー上、機器上など |
| 暗号利用の目的 | 通信の秘匿、署名、認証、データ保存 |
| （通信の場合） | プロトコル(TLS、IPsec、SSH 等)、対向先情報 |
| アルゴリズム／パラメータ | アルゴリズム: 公開鍵暗号(RSA/ECDSA)、共通鍵暗号(AES)など<br>パラメータ: 2048bit など |
| 保護対象データの詳細 | （例: 証券口座へのログイン情報の秘匿、顧客との電子契約データ） |
| 最終更新日 | |

表 6 暗号インベントリの要素

暗号インベントリの放置や更新漏れを防ぐために、暗号インベントリの管理者・更新契機・点検周期を最初に決めるのが望ましい。下記は一例である。

| 項目 | 例示 |
|---|---|
| 管理者 | 推進担当者が管理・統制を行う |
| 更新契機 | PQC 移行、システム更改、ライブラリ更新など |
| 点検周期 | 最低でも年次で各システム主管部門で記載内容に齟齬がないか確認する |

表 7 暗号インベントリの管理ルール(例)

暗号インベントリについて、CMDB(構成管理データベース)と統合し管理できる製品についても存在しており、うまく活用できれば構成管理を含め効率的に実施することも可能と考えられる。しかし、暗号インベントリを管理する体制や運用が未整備だと製品の導入自体が目的化してしまう。そのため、まずはスプレッドシートなどの従来のツールで体制や運用を考えてから、不足する点や効率化できる点があると判断してから、導入を検討すべきと考える。

## 4.4 リスク評価と優先順位付け

すべてのシステム・アプリケーションを一度に PQC に移行することは現実的ではないため、優先順位付けが不可欠である。作成した暗号インベントリをもとに、どのシステム・アプリケーションのどの機能から優先的に PQC 移行すべきかを検討する。

PQC 移行の優先順位については、リスク評価でよく使われる「影響度（データが漏えいした場合の被害の大きさ）」と「蓋然性（攻撃者がデータを収集しやすいか）」の 2 軸に加えて、「秘密保持期間（秘匿が維持されるべき期間の長さ）」、「移行困難性（現時点では技術的・運用的に移行が可能な状態になっていない、制約により容易に移行できない）」の 2 つの追加観点で評価するのが良いと考える。

ただし、規制対応や業界インフラ（全銀システム、SWIFT 等）の移行スケジュールなど、外部から期限が定められるものについては、評価に優先して対応が必要となる。

## 4.5 暗号移行ロードマップの作成

前項で検討した優先順位を、年度計画・システム更改計画・ベンダー対応計画と整合させ、組織全体で合意・更新可能なロードマップとして具体化する必要がある。

2026 年 1 月、G7 サイバー・エキスパート・グループは、金融セクター向けの協調的なロードマップ推進に関するステートメント[xv]を公表した。このステートメントでは、重要システムと非重要システムで異なるタイムラインを設定する「デュアルトラック・アプローチ」が推奨されている。

### 4.5.1 デュアルトラック・アプローチと移行時期の目標

デュアルトラック・アプローチと、前項で検討したリスク評価の両面で移行時期の目標を立てることによって、リスクの高いシステムを優先的に保護し、全体として効率的な移行を実現する方法が考えられる。

**デュアルトラック・アプローチでの移行期限設定**

複数の国際機関による現行のガイダンスを踏まえ、G7 サイバー・エキスパート・グループではシステムの重要度に応じて、例として以下の移行時期を提示している。

| 対象システム | 移行完了時期 |
|---|---|
| 重要システム | 2030～2032 年 |
| 非重要システム | 2035 年 |

表 8 移行時期の例示（G7 サイバー・エキスパート・グループ）

重要システムかどうかの判断は、システム全体として見た際の業務継続への影響、金融システムへの影響、顧客影響、規制要求などを総合的に評価して行う。重要性の低いシステムは、重要システムの移行で得た経験を活かし、2032 年以降に移行することが考えられる。

なお、本書に記載する移行時期やタイムラインは、国内外の公表資料やガイダンスを整理した参考情報であり、当社グループとしての公式な確定計画や期限を示すものではない。

### 4.5.2 ロードマップの管理体制と継続的な取組

**ロードマップの管理体制**

ロードマップについては、次のような管理体制を用意することが望ましい。

| 管理体制 | 推奨事項 |
|---|---|
| 責任部署 | 5.1 で設置した横断チームの推進担当が統括的に管理する |
| 定期的な見直し | 半期または年次（経営報告・システム更改計画と連動） |
| 臨時的な見直し | 量子技術の大きな進展、標準化動向の大きな変化、規制・業界インフラの移行スケジュール変更などに柔軟に対応する |
| 関係部門とのレビュー | 四半期または半期ごとの横断チーム内とのレビュー、経営層への報告 |

表 9 ロードマップの管理体制

ロードマップは「作成して終わり」ではなく、外部動向を継続的に監視し、定期的に見直すことが重要である。特に、量子技術の進展や標準化の状況は現時点でも予測が困難であるため、柔軟に計画を調整できる体制を整えることが望ましい。

**継続的な取組（ロードマップを支える活動）**

移行期間においても、下記の取組を行い、ロードマップの見直しに備えることが望ましい。

| 継続的な取組 | 推奨事項 |
|---|---|
| ガバナンスとリスク管理 | 量子リスクを既存の組織ガバナンスに組み込む<br>(例)経営層への四半期報告、内部監査での確認項目化 |
| 外部依存性の管理 | 量子技術・標準化動向・ツール・脅威の動向をモニタリングする<br>(例) NIST/IETF 等の標準化動向の追跡、主要ベンダーの PQC 対応確認 |
| ステークホルダーとの対話 | 課題の特定・知見の共有・共通解決策の推進<br>(例) 業界団体(ISAC 等)での情報交換、委託先との移行計画調整 |

表 10 ロードマップの継続的な取組

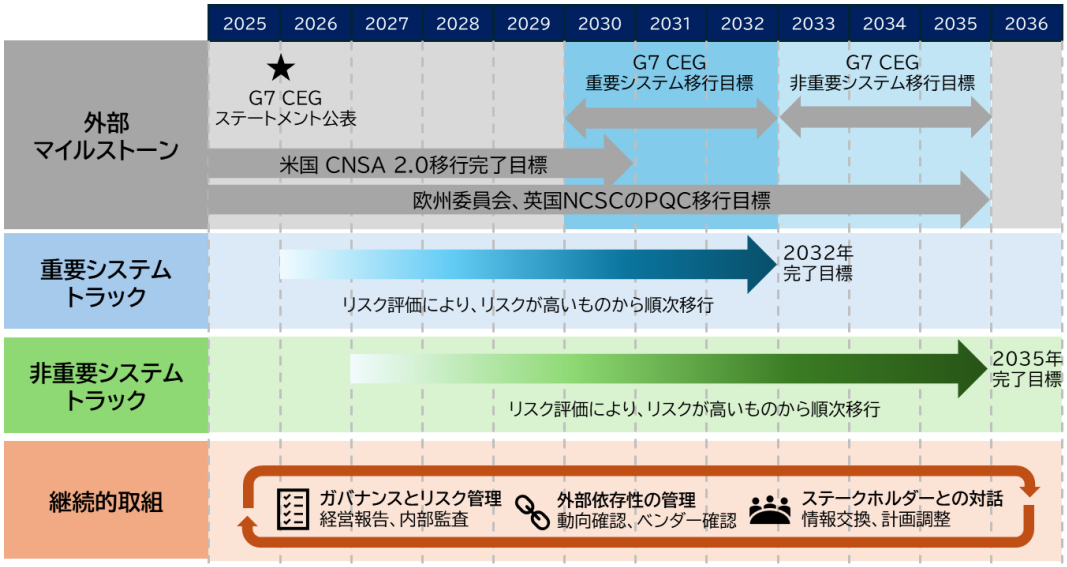


図 11 PQC 移行ロードマップ

## 4.6 具体的な暗号移行方法の検討

前項まで体制の確保から暗号移行のロードマップの作成までについて検討したが、この項では 2026 年 3 月時点の PQC を取り巻く状況を踏まえた暗号移行の基本的な考え方について述べる。

### 4.6.1 移行の基本方針:鍵交換から先行

インターネットの実装動向として、TLS 通信では鍵交換(鍵カプセル化メカニズム:KEM)の PQC 対応が先行している。これは、鍵交換の移行が比較的容易で、かつ「Harvest Now, Decrypt Later(HNDL:今収集し、後で解読)」リスクへの対策として即効性が高いためである。

一方、デジタル署名の PQC 対応は、発行・検証インフラ(認証局、証明書チェーン、ブラウザ

/OS の信頼基盤、監査・運用手順）が必要であり、時間がかかる見込みである。さらに、PQC 鍵や証明書に関する表現形式（鍵フォーマット等）は標準化の調整が続いている。したがって、**第一段階は鍵交換、第二段階は署名**という段階的アプローチが現実的である。

### 4.6.2 クリプト・アジリティを意識した移行パターン

2 章で述べた通り PQC は、「量子コンピュータでも現時点で効率的なアルゴリズムが発見されていない」という計算量的安全性を安全性の根拠に持つ。そのため、採用した PQC に万が一脆弱性が発見された場合でも、別の PQC に柔軟に移行が可能なクリプト・アジリティを備えたアーキテクチャを検討すべきである。

そのため、移行方式の選定では「どの方式を採用するか」だけでなく、「暗号変更の影響範囲をどこまで局所化できるか（＝変更時の波及を抑えられるか）」も合わせて評価する必要がある。具体的には、暗号方式の変更点を集中管理できる終端（エッジ）や共通暗号基盤（ライブラリ／ランタイム）に集約し、アプリケーション個別改修を最小化する設計が望ましい。

以下に、代表的な移行パターンについて、変更影響の少なさと集中管理の観点を含めて、クリプト・アジリティが高いものから順に整理する。

| パターン | 概要 | 適用場面 | 評価 |
|---|---|---|---|
| A. エッジ終端での吸収 | CDN／LB／WAF 等の TLS 終端で PQC を有効化する | Web/API の終端、共通経路 | クリプト・アジリティが高い<br>アプリケーションの改修不要で、導入が早く、適用範囲を統制しやすい |
| B. 共通ライブラリ／暗号基盤での吸収 | 組織共通の暗号基盤で方式を一元管理する | マイクロサービス、API サーバー群 | クリプト・アジリティが高い<br>エッジ終端が難しい環境でも実装の統一、統制が効きやすい |
| C. 例外領域でのアプリケーション改修 | アプリケーションに PQC を組み込む | 独自通信、特殊要件、終端で吸収できない領域 | クリプト・アジリティは低い<br>終端に非依存であるため、特殊な要件への対応を行える |

表 12 移行パターン別のクリプト・アジリティ評価

### 4.6.3 ハイブリッド方式の採用

ハイブリッド方式とは、従来暗号（例：ECDHE）と PQC（例：ML-KEM）を同時に用い、どち

らか一方が破られても秘匿性を維持する移行期の実装方式である。

TLS 1.3 においては ECDHE と ML-KEM を組み合わせるハイブリッド方式（X25519MLKEM768）の標準化・実装が進んでおり、ハイブリッド方式を採用した場合はクライアント側の対応状況に応じて、使用する暗号を使い分ける。

一方で、PQC のみを利用するピュア PQC は 2026 年 3 月時点では採用例は多くない。そのため、移行期に関してはハイブリッド方式を採用し、将来的にピュア PQC を利用するかどうか判断する、といった形が現実的である。

### 4.6.4 サプライチェーンと更改戦略

PQC 移行は、自組織だけでなくサプライチェーン全体に影響する。以下の観点で第三者の対応状況を確認し、ロードマップに組み込む必要がある。

確認対象と観点の例を以下に示す。

| 確認対象 | 確認観点 |
|---|---|
| クラウドサービス | PQC 対応ロードマップ、TLS 終端サービスの対応時期 |
| ソフトウェアベンダー | ミドルウェア（Web サーバー、DB 等）の PQC 対応予定、ライセンス条件 |
| ネットワーク機器ベンダー | LB、FW、VPN 装置等のファームウェア更新予定 |
| 認証局（CA） | パブリック認証局の PQC 証明書発行予定、プライベート CA の改修計画 |
| 業界インフラ | 全銀システム、SWIFT 等の移行スケジュール |

表 13 サプライチェーンでの確認対象と確認観点

また、移行のロードマップとシステム更改との整合性も重要である。

| 戦略 | 判断基準／対応 |
|---|---|
| 更改時に対応 | 更改予定がロードマップの目標より前に到来する場合 |
| 更改を待たずに対応 | ロードマップの目標が更改より前、あるいは不確定の場合 |
| 例外管理 | 技術的観点やサプライチェーンの観点・経済的観点で更改での移行困難な場合は、ロードマップ上にリスクを明示する |

表 14 移行戦略と判断基準

## 4.7 移行後の性能・安定性の影響評価

PQC 導入後に予期せぬ性能劣化や安定性問題が発生しないよう、事前に考慮すべき観点と

確認すべき性能指標について整理する。

### 4.7.1 事前に考慮すべき観点

PQC に関して、従来の暗号と比較して、以下の特性がある。

| 特性 | 理由と影響として考えられる点 |
|---|---|
| 鍵・証明書・署名のサイズが大きい | 例として、X25519 の公開鍵のサイズが 32 バイトに対して、ML-KEM の公開鍵のサイズは約 1～2 キロバイトとなっている。<br>TLS ハンドシェイクなどで通信量が増大する。 |
| 暗号方式の違い | 暗号方式が変更になることによって、従来暗号では使用できていたハードウェアアクセラレーションが使用できない可能性がある。 |
| 運用上の制約 | TLS での PQC は TLS 1.3 が前提となっている。<br>TLS 1.2 以前を利用している場合は、1.3 への移行が必要となる。 |

表 15 PQC と従来暗号の特性比較

3 章に記載した通り、大和証券グループの実証実験では、従来暗号である X25519 と PQC である ML-KEM で TLS ハンドシェイクの確立にかかる時間を測定したところ、鍵生成・カプセル化・デカプセルにかかる時間は PQC の方が短くなったものの、通信量自体が増えることによって、全体としては僅かながら時間が増加することを確認している。

Web サイトなどの通常の利用用途では影響は少なく、ユーザー側で体感できるものでないと言えるが、高頻度取引システムや帯域制限がある環境などの特殊な業務・環境においては移行方針を検討する前に考慮しておくことが望ましい。

### 4.7.2 評価すべき性能指標

PQC 移行前後で、以下の指標に少なからず影響を与え得る。そのため、特に重要システムにおいては指標を測定し、移行前後で比較することが望ましい。

| 指標 | 測定項目の例 |
|---|---|
| 遅延 | TLS ハンドシェイク時間（初回／セッション再開）<br>API 応答時間（平均、95 パーセンタイル） |
| スループット | TPS（Transactions Per Second）<br>同時接続数やデータ転送量 |
| リソース | CPU 使用率、メモリ使用量<br>HSM の処理能力 |
| 安定性 | TLS ハンドシェイク失敗率<br>再送・タイムアウト発生率<br>可用性 |

表 16 PQC で評価すべき性能指標

### 4.7.3 性能改善の方針

PQC 移行前の検討段階で性能劣化が懸念される場合や PQC 移行後の性能測定で性能劣化が顕在化した場合、以下の対策を検討する必要がある。

| 対策 | 対策の例 |
|---|---|
| ハードウェアの増強 | LB、Web サーバーなどのスケールアウト／アップ<br>ハードウェアアクセラレーション対応（ファームウェア更新、機器リプレイス）<br>HSM の追加導入 |
| セキュリティパラメータの調整 | 軽量なセキュリティパラメータ（例:ML-KEM768→512）への変更<br>※安全性とのトレードオフのため、慎重な判断が必要 |
| ネットワーク最適化 | ネットワーク機器の MTU 調整などのチューニング<br>サーバーからエッジ終端による暗号のオフロード（TLS アクセラレータの導入） |

表 17 性能劣化が顕在化した際の対策例

# 5. おわりに

本書では、耐量子計算機暗号（PQC）の基礎知識、国内外の制度・ガイドライン動向、当社グループのオンラインサービス開発環境で実施した概念実証、および暗号移行の実務アプローチを整理した。

米国 NIST が 2024 年 8 月に最初の PQC 標準（FIPS 203/204/205）を公表し、日本でも関係府省庁連絡会議が設置されるなど、移行準備の段階に入っている。

当社グループの実証実験では、PQC での鍵・証明書のサイズ増による通信量増加による通信遅延の影響はわずかに見られるものの、大きな影響はないことが確認できた。一方で、本実証実験は開発環境における PoC であり、それぞれのシステムの特性によって評価が変わり得る点に留意が必要である。

PQC 移行は暗号アルゴリズムの差し替えに留まらず、通信・認証基盤・鍵管理・調達・運用まで影響する中長期の取り組みとなる。現時点で多くのガイダンスに共通する実務要点として、暗号アセットの棚卸しとインベントリ整備、リスク評価に基づく優先順位付け、そして段階的移行（システムの重要度、鍵交換から署名・PKI へ）が推奨される。

PQC を巡る状況は変化が大きい分野であるため、量子コンピュータや標準化、実装の最新動向を継続的に確認する必要がある。本書が、金融分野をはじめ多くの組織における PQC 対応の検討・説明・実装の出発点となり、業界全体の円滑な移行に資することを期待する。

# 付録 1. 用語集

| 用語 | 説明 |
|---|---|
| PQC | Post Quantum Cryptography（耐量子計算機暗号）。量子コンピュータによる攻撃に対しても安全性を維持できると期待される暗号方式の総称。従来型コンピュータ上で実装・運用できる点が特徴。 |
| 量子暗号 | 量子力学的性質を利用した暗号技術の総称。QKD などが含まれる。PQC とは異なり、専用の機器を必要とする。 |
| QKD | Quantum Key Distribution（量子鍵配送）。光子などの量子力学的性質を利用して暗号鍵を配送する技術。理論上、盗聴があれば必ず検知でき、情報理論的安全性を持つ鍵交換を実現する。 |
| HNDL | Harvest Now, Decrypt Later（今収集し、後で解読）。攻撃者が現在暗号化されている通信データや保存データを収集・保管しておき、将来の量子コンピュータの実用化後に解読するという攻撃シナリオ。 |
| CRQC | Cryptographically Relevant Quantum Computer（暗号解読に適した量子コンピュータ）。現在広く利用されている公開鍵暗号を実用的な時間で解読できる性能を持つ量子コンピュータのこと。 |
| 情報理論的安全性 | 攻撃者の計算能力に依存せず、数学的に証明可能な安全性。QKD はこの安全性を持つとされている。 |
| 計算量的安全性 | 暗号を解読するために必要な計算量が、現実的な時間・コストでは実行不可能なほど膨大であることを根拠とする安全性。PQC を含む従来の暗号方式はこの安全性に基づいている。 |
| 公開鍵暗号 | 暗号化と復号に異なる鍵ペア（公開鍵と秘密鍵）を使用する暗号方式。鍵共有・鍵交換やデジタル署名に広く使用される。従来の暗号では RSA 暗号や楕円曲線暗号（ECDSA 等）が代表であるが、Shor のアルゴリズムにより、十分な性能を持つ量子コンピュータによって解読される可能性があり、PQC への移行が議論されている。 |
| 共通鍵暗号 | 送信者と受信者が同じ鍵を共有して暗号化・復号を行う方式。AES が代表例。Grover のアルゴリズムにより量子コンピュータによって鍵探索が二次的に高速化されるが、鍵長を倍増（例：AES-128→AES-256）することで対応可能であり、アルゴリズム自体の全面置換は不要とされている。 |
| KEM | Key Encapsulation Mechanism（鍵カプセル化機構）。受信者の公開鍵を用いて共有鍵を安全にカプセル化（暗号化）し、受信者が秘密鍵で取り出す（デカプセル化）ことで共有鍵を確立する方式。TLS ハンドシェイク等での鍵交換に用いられる。PQC では ML-KEM が標準化されている。 |

| 用語 | 説明 |
| --- | --- |
| 鍵交換 | 通信の開始時に、盗聴されても安全な方法で通信相手と共通鍵を共有するプロセス（鍵共有）。TLS/SSL のハンドシェイクにおいて実施され、以降の通信データを共通鍵暗号で暗号化するために用いられる。HNDL 攻撃の主な標的であり、PQC 移行の第一優先とされている。 |
| TLS | Transport Layer Security。インターネット通信において、データの暗号化・認証・改ざん検知を提供するプロトコル。HTTPS などで広く利用される。PQC の鍵交換対応は TLS 1.3 を前提としている。 |
| TLS ハンドシェイク | TLS 通信を開始する際に、クライアントとサーバーが暗号方式の合意・認証・鍵交換を行う一連のプロセス。PQC への移行により、鍵・証明書のサイズ増大に伴うパケット数増加や処理時間の変化が生じうる。 |
| TLS セッション再開 | 以前に確立した TLS セッションのセッションチケットを利用し、証明書の検証等を省略してセッションを再確立する仕組み。フルハンドシェイクと比べて接続確立時間を短縮できる。 |
| PKI | Public Key Infrastructure（公開鍵基盤）。デジタル証明書の発行・管理・失効を行う仕組みの総称。認証局（CA）が中心的な役割を担い、TLS 通信やコード署名等の信頼基盤となる。PQC への移行においては、署名・証明書の PQC 対応が課題となっている。 |
| CA | Certification Authority（認証局）。デジタル証明書を発行・管理する機関。不特定多数に対してサービスを提供するパブリック CA と、組織内部で利用するプライベート CA がある。PQC 対応証明書の発行には、IETF での RFC 標準化や CA/B フォーラムでの Baseline Requirement 更新が必要。 |
| 証明書チェーン | エンドエンティティ証明書（サーバー証明書等）から中間認証局を経てルート認証局に至る信頼の連鎖。TLS 通信において、サーバーの正当性を確認するために検証される。証明書チェーンの深さや証明書サイズはハンドシェイクの処理時間に影響する。 |
| HSM | Hardware Security Module（ハードウェアセキュリティモジュール）。暗号鍵の生成・保管・管理を安全に行うための専用ハードウェア。 |
| NIST | National Institute of Standards and Technology（米国国立標準技術研究所）。PQC アルゴリズムの標準化を主導してきた米国政府機関。2016 年に公募を開始し、2024 年 8 月に FIPS 203/204/205 として最初の PQC 標準を公表した。NIST の標準暗号は事実上の国際標準として世界中で採用されている。 |
| FIPS | Federal Information Processing Standards（米国政府の情報処理標準規格）。NIST が発行する標準規格群。 |

| 用語 | 説明 |
| --- | --- |
| ML-KEM | NIST FIPS 203 で標準化された鍵カプセル化方式。CRYSTALS-Kyber をベースとし、格子問題に基づく安全性を持つ。 |
| ML-DSA | NIST FIPS 204 で標準化されたデジタル署名方式。CRYSTALS-Dilithium をベースとし、格子問題に基づく安全性を持つ。 |
| SLH-DSA | NIST FIPS 205 で標準化されたデジタル署名方式。SPHINCS+をベースとし、ハッシュ関数の一方向性に基づく安全性を持つ。格子問題に依存しないため ML-DSA のバックアップアルゴリズムとして考えられている。 |
| FN-DSA | NIST で FIPS 206 として標準化が進んでいるデジタル署名方式。Falcon をベースとし、格子問題に基づく安全性を持つ。 |
| HQC | Hamming Quasi-Cyclic。符号問題(誤り訂正符号の復号問題)に基づく鍵交換方式。2025 年 3 月に NIST の標準化候補に選定され、ML-KEM とは異なる数学的根拠を持つことから、バックアップアルゴリズムとして考えられている。 |
| 暗号インベントリ | 組織内で使用されている暗号アルゴリズム・プロトコル・証明書・鍵管理状況などの暗号利用状況を一元的に管理する台帳。 |
| クリプト・アジリティ | 暗号アルゴリズムが危殆化した場合や新たな脆弱性が発見された場合に、システム全体への影響を最小化しながら別の暗号へ柔軟に移行できるための設計上の考え方。PQC の安全性は「現時点で効率的な解読アルゴリズムが知られていない」ことに基づくため、将来の変化に備えた設計原則として重要。 |
| デュアルトラック・アプローチ | 重要システムと非重要システムで異なる移行タイムラインを設定する PQC 移行アプローチ。G7 サイバー・エキスパート・グループが推奨しており、リスクの高いシステムを優先しつつ、全体として効率的な移行を実現する。 |
| ハイブリッド方式 | 従来暗号(例:X25519)と PQC(例:ML-KEM)を組み合わせて同時に使用する移行期の実装方式。どちらか一方が破られても秘匿性を維持できる多層防御の考え方に基づく。TLS 1.3 では X25519MLKEM768 が標準化・実装されており、主要ブラウザがすでに対応している。 |
| ピュア PQC | PQC アルゴリズムのみを使用する方式。従来暗号との組み合わせを持たないため、通信データ量は相対的に小さく実装もシンプルだが、旧システムとの互換性が失われる。2026 年 3 月時点での採用例は多くないが、PQC が一般化した場合に採用に関する議論が行われる可能性がある。 |
| コンポジット証明書 | 1 枚の証明書内に従来暗号と PQC の両方の鍵・署名を組み合わせた証明書。両方の暗号を破らない限り安全性が損なわれない方式。IETF の LAMPS ワーキンググループで標準化が進んでいるが、システム側が新しい証明書フォーマットを理解できるようにする必要があるなど実装の複雑さが課題。 |

| 用語 | 説明 |
| --- | --- |
| デュアル証明書 | 従来暗号と PQC の独立した 2 枚の証明書を並行して維持し、相手側の対応状況に応じて使い分ける方式。コンポジット証明書で生じる互換性の問題は生じないが、サーバー側の証明書管理負荷が倍増する点や、ダウングレード攻撃への脆弱性が懸念される。 |
| X25519 | Curve25519 を使用した楕円曲線ディフィー・ヘルマン鍵交換アルゴリズム。現在 TLS 通信で広く使用されている。公開鍵サイズは 32 バイトと非常に小さく高速だが、量子コンピュータによる解読リスクがあるため PQC への移行対象となる。 |
| X25519MLKEM768 | X25519 と ML-KEM 768 を組み合わせたハイブリッド鍵交換方式。TLS 1.3 での標準化・実装が進んでいる。HNDL 攻撃への対処として現時点で最も実用的な PQC 対応方式と言える。 |
| IETF | Internet Engineering Task Force（インターネット技術特別調査委員会）。全世界的なインターネット技術標準を策定する団体。PQC のインターネット実装について、PQUIP・LAMPS・TLS の 3 つの主要ワーキンググループを中心に議論・標準化を進めている。 |
| PQUIP | Post-Quantum Use In Protocols。IETF における PQC 移行のための横断的な調整・ガイダンスを行うワーキンググループ。 |
| LAMPS | Limited Additional Mechanisms for PKIX and S/MIME。IETF のワーキンググループで、X.509 証明書に関連する標準を担当している。 |
| 認証局／ブラウザフォーラム（CA/B フォーラム） | 電子証明書を発行する認証局とパブリック TLS 証明書を表示するブラウザが中心となって運営される団体。パブリック PKI における PQC 標準化を担い、Baseline Requirement への PQC 追加を議論・策定している。 |
| CNSA 2.0 | Commercial National Security Algorithm Suite 2.0。NSA（米国国家安全保障局）が公表した国家安全保障システム向けの暗号スイート。量子耐性アルゴリズムへの移行方針とタイムラインを示し、2030 年前後までの移行完了を想定している。 |
| CRYPTREC | 電子政府等で利用される暗号技術の安全性評価・監視や、適切な実装・運用に関する調査検討を行う日本の枠組み。 |
| ENISA | European Network and Information Security Agency（欧州ネットワーク・情報セキュリティ機関）。PQC 導入に関する報告書を公表し、2020 年代中盤からの準備・実装開始と 2030 年代前半までの重要インフラの移行完了を見据えた考え方を示している。 |
| NCSC | National Cyber Security Centre（英国国家サイバーセキュリティセンター）。英国内の重要国家インフラ事業者に対し早期の移行計画策定を推奨している。 |

| 用語 | 説明 |
| --- | --- |
| G7 サイバー・エキスパート・グループ | G7（主要 7 カ国）のサイバーセキュリティ専門家グループ。2026 年に金融セクターにおける PQC への移行について分野横断的・協調的なタイムラインの考え方を示す声明を公表した。 |
| 東京 QKD ネットワーク | NICT（情報通信研究機構）が構築・運用する QKD のテストベッドネットワーク。日本での QKD 実証実績を持ち、広域 QKD ネットワークの構築に向けた検討の基盤となっている。 |

# 付録 2. 参考文献・関連リンク

各リンクは 2026 年 3 月 26 日時点でアクセスできることを確認している。

---

[i] 内閣官房： 政府機関等における耐量子計算機暗号（PQC）への移行について（中間とりまとめ）, 2025-11-20, https://www.cas.go.jp/jp/seisaku/pqc/pdf/report_202511.pdf

[ii] 金融庁： 「預金取扱金融機関の耐量子計算機暗号への対応に関する検討会報告書」の公表について, 2025-11-26, https://www.fsa.go.jp/news/r6/singi/20241126.html

[iii] CRYPTREC： CRYPTREC 暗号技術ガイドライン（耐量子計算機暗号）2024 年度版, 2025-03, https://www.cryptrec.go.jp/report/cryptrec-gl-2007-2024.pdf

[iv] CRYPTREC： CRYPTREC 耐量子計算機暗号の研究動向調査報告書, 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] 金融庁： G7 サイバー・エキスパート・グループによる金融セクターにおける耐量子計算機暗号への移行に向けた協調的なロードマップの推進に関するステートメントの公表について, 2026-1-15, https://www.fsa.go.jp/inter/etc/20260115/pqc.html