

Post-Quantum Cryptography Migration Recommendations

A globally applicable governance, technical and roadmap framework
informed by FINMA Guidance 05/2026

Hui Gong | UCL Institute of Finance & Technology

Audience: banks, insurers, securities firms, funds, asset managers, financial market infrastructures and fintech providers

Use: board decisions, risk assessment, technology selection, migration planning, pilot acceptance and audit evidence

Publication date: 1 October 2026 **Version:** 1.4

Executive Summary

Decision in one page

Large-scale quantum attacks are not yet a practical reality, but the migration window for financial institutions is already open. The case for action does not depend on predicting a single “Q-Day”: long-lived sensitive data continues to be created, complex technology changes take years, and third-party product cycles are not fully controlled by the institution. The Swiss Financial Market Supervisory Authority (**FINMA**) brought this issue into operational-risk and resilience governance in Guidance 05/2026; its control logic is relevant in other jurisdictions too.

Building on the governance priorities in FINMA Guidance 05/2026, this report recommends that a financial institution, within 12 months of starting, approve a strategy and roadmap, establish a high-risk **cryptographic bill of materials (CBOM)** and critical-vendor baseline, and start at least one **hybrid migration of traditional and post-quantum cryptography** pilot. It can then migrate critical processes in risk-based waves over a reference period of 36–60 months. These periods are this report’s reference schedule, not FINMA’s universal completion deadlines. The long-term objective is **cryptographic agility**: the ability to discover, replace, test and evidence that cryptographic controls remain effective.

Why act

Financial institutions rely on **Rivest–Shamir–Adleman (RSA) public-key cryptography** and **elliptic-curve cryptography (ECC)** for authentication, key establishment, digital signatures, software updates, inter-firm communication and long-lived records. A future **cryptographically relevant quantum computer (CRQC)** could undermine these mechanisms. Data that must remain confidential for many years is also exposed to **harvest-now, decrypt-later (HNDL)** risk: ciphertext copied today may become readable in the future.

Post-quantum cryptography (PQC) migration is therefore both a security programme and a business-continuity, customer-trust and third-party-governance programme. It can deliver immediate benefits in certificate management, key governance, technical-debt discovery, supplier transparency and emergency algorithm replacement.

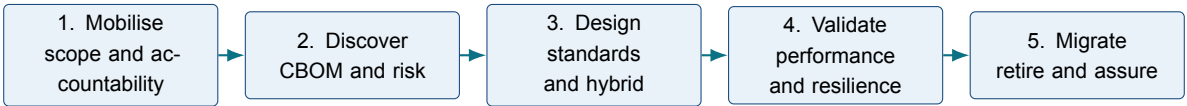
Why now

Current fact	Business and governance effect	Action available now
Long-lived data is already being created	A later upgrade cannot reverse prior capture of ciphertext	Identify data lifetime, exposure and priority flows
Migration takes years	Applications, certificates, hardware security modules, peers and vendors are interdependent	Build inventory, dependency map and waves
Formal standards exist	Work can move from research into controlled pilots	Test standard algorithms and hybrid protocols
New systems can add vulnerable dependencies	Each non-agile purchase increases future cost	Add PQC readiness to architecture and procurement
Supervisory attention is moving to governance	Boards must show that risk is identified, prioritised and overseen	Approve roadmap, risk appetite, resources and exceptions

Commercial and operating value of early action

Visibility: know which data, systems and suppliers matter first; **Controlled investment:** divide a multi-year programme into fundable and testable waves; **Supply-chain leverage:** use procurement and renewal windows to secure versions, evidence and exit rights; **Continuity:** reduce concentrated change risk through coexistence, controlled cutover and safe rollback; **Assurance:** explain risk status and progress to boards, clients, auditors and supervisors.

How to act



- 1. Mobilise.** The governing body approves risk appetite, dates and resources; security leads with business, technology, data, procurement, legal, risk and audit participation.
- 2. Discover.** Build the CBOM and connect algorithms, certificates and keys to data, processes, systems and vendors.
- 3. Design.** Define separate migration mechanisms for key establishment, signatures, data at rest and the software supply chain, including controlled hybrid operation.
- 4. Validate.** Test function, performance, interoperability, failure handling and continuity, retaining reviewable evidence.
- 5. Migrate.** Move production in waves. Retire legacy mechanisms only after dependencies, data and keys are treated, observation is complete and business owners sign off. Bring inventory updates, algorithm versions, vendor support and periodic retesting into normal operations to sustain future upgrades.

Reference time frame

FINMA recommends that institutions under its supervision establish a roadmap by mid-2027. Other jurisdictions should treat this as a useful governance benchmark, not an automatically applicable statutory deadline.

Stage	Reference timing	Outcome
Mobilise	Day 0–90	Governance, scope, initial vendor questionnaire, long-lived data and pilot candidates
Roadmap and base-line	Month 3–12	Approved roadmap, high-risk CBOM, risk order, target architecture, budget and procurement rules
Pilots and critical migration	Month 12–36	Hybrid Transport Layer Security (TLS) or signature pilots, critical-process migration, interoperability and capacity evidence
Scale and close-out	Month 36–60	Legacy treatment, weak-algorithm retirement, exception closure and continuous monitoring

Contents

Executive Summary

1	FINMA Guidance and Cross-Jurisdictional Application	1
1.1	Supervisory Implications of FINMA Guidance 05/2026	1
1.2	Cross-Jurisdictional Relevance of the Control Objectives	1
1.3	Intended Users and Responsibilities	2
1.4	Implementation Sequence	2
1.5	Time-Based Risk Assessment	2
1.6	Scope and Localisation	2
2	How Quantum Computing Changes Current Security Assumptions	3
2.1	What financial systems rely on today	3
2.2	Do quantum algorithms require physical hardware?	3
2.3	Shor's algorithm: a structural public-key threat	3
2.4	Grover's algorithm: square-root search acceleration	3
2.5	How Shor and Grover differ	4
2.6	From Algorithms to Business Exposure	4
2.7	HNDL and the migration window	5
2.8	Boundary of the threat model	5
3	PQC Standards and Their Mathematical Basis	6
3.1	Rationale for the Initial Standardised Algorithms	6
3.2	Lattices and Module-Lattice Structures	6
3.3	Learning with Errors and Computational Hardness	7
3.4	ML-KEM: Module-Lattice Key Encapsulation	7
3.4.1	Chosen-Plaintext and Chosen-Ciphertext Security	7
3.5	ML-DSA: module-lattice signatures	7
3.6	SLH-DSA: Stateless Hash-Based Signatures	7
3.7	Hybrid Traditional and Post-Quantum Constructions	8
3.8	Algorithm Selection and Implementation Constraints	8
4	End-to-End Migration Mechanism and Target Architecture	9
4.1	Migration Lifecycle and Completeness Criteria	9
4.2	Layered Target Architecture	9
4.3	TLS, APIs and inter-firm communication	10
4.4	Digital signatures, certificates and long-term evidence	10
4.5	Data at rest and key re-wrapping	10
4.6	Identity, software supply chain and devices	10
4.7	Third parties and financial market infrastructures	11
4.8	Coexistence, cutover and safe rollback	11
4.9	Cryptographic Policy Service, Observability and Completion	11
5	Cryptographic Inventory and Risk Prioritisation	12
5.1	Foundational Role of the Cryptographic Inventory	12
5.2	Minimum CBOM fields	12
5.3	Discovery requires multiple sources	13
5.4	Dependency graph	13
5.5	Risk scoring	13
5.6	HNDL scenario queue	14
5.7	Reviewable prioritisation	14
5.8	Keeping the inventory current	15
6	A Reference Roadmap for Financial Institutions	16

6.1	Roadmap Model and Conditions of Use	16
6.2	Mobilisation Phase: First 90 Days	16
6.3	Phase 1: governance, baseline and roadmap (months 3–12)	16
6.4	Phase 2: pilots and common capability (months 6–18)	16
6.5	Phase 3: high-risk migration (months 12–36)	16
6.6	Phase 4: scale, tail and retirement (months 24–60)	17
6.7	Reference timeline and investment	17
6.8	Cross-Jurisdictional Applicability and Localisation	17
6.8.1	International reference milestones and institutional scheduling	18
6.9	Annual and event-driven review	18
7	Pilot Validation, Resilience and Production Acceptance	19
7.1	Four questions for every pilot	19
7.2	Pilot 1: hybrid TLS/API	19
7.3	Pilot 2: file exchange or digital signature	19
7.3.1	Financial infrastructure reference case: Project Leap	19
7.4	Disturbance-testing framework	20
7.5	Safe rollback and fail closed	20
7.6	Implementation security and side channels	20
7.7	Capacity and release gates	20
7.8	Evidence package and observation period	20
8	Governance, Supply Chain and Continuous Assurance	21
8.1	Governance model	21
8.2	Board reporting	21
8.3	Vendor governance and contracts	21
8.4	Exception management	22
8.5	Metrics	22
8.6	Three lines and evidence architecture	22
8.7	Maturity and continuing update	23
8.8	Continuous cryptographic governance and operations	23
A	Glossary and Acronym Index	24
B	Mathematical Appendix	26
B.1	RSA correctness and Shor’s entry point	26
B.2	Grover amplitude amplification	26
B.3	Intuition for LWE correctness	26
B.4	KEM security: IND-CPA and IND-CCA	27
B.5	Signatures and rejection sampling	27
B.6	Hybrid key combination	27
C	Mobilisation and Vendor Questionnaire	28
C.1	Internal questions	28
C.2	Vendor questions	28
D	CBOM Field Template	29
E	Implementation and Acceptance Checklist	30
E.1	Before roadmap approval	30
E.2	Before pilot production entry	30
E.3	Before declaring completion	30
F	Standards and Regulatory Snapshot	31
	References	32

FINMA Guidance and Cross-Jurisdictional Application

1.1 Supervisory Implications of FINMA Guidance 05/2026

In July 2026, FINMA published [Guidance 05/2026 on cyber risks associated with quantum computing](#). It does not demand immediate full migration or predict a definite date for a capable quantum computer. Its supervisory meaning is more practical: institutions already have enough information to establish governance, an asset baseline and accountable planning.

FINMA surveyed 60 Swiss banks, insurers, collective-asset managers and financial market infrastructures between November 2025 and January 2026. It reported that 72% had not planned or implemented quantum-safe measures; only 8% had a specific roadmap; institutions with roadmaps commonly expected four to five years for critical data and processes. FINMA recommends a roadmap by mid-2027, with separate target dates for critical processes and complete migration[1].

The mid-2027 date is a **governance and roadmap milestone** for FINMA-supervised institutions, not a deadline for completing every technical migration. A credible roadmap should identify accountability, priority data and processes, cryptographic exposure, vendor timing, resources, acceptance criteria and approved exceptions.

1.2 Cross-Jurisdictional Relevance of the Control Objectives

The notice applies within FINMA's perimeter, but the underlying risk is not uniquely Swiss. Any digital financial institution depends on public-key infrastructure, external interfaces, third-party software, cloud services, long-lived data and digital signatures. Five control outcomes are therefore broadly reusable.

Table 1.1: From FINMA themes to universal control outcomes

FINMA theme	Universal outcome	Evidence
Strategy and roadmap	Governing body approves priorities, dates, resources and exceptions	Decisions, risk appetite, roadmap, budget and RACI
Risk analysis and inventory	Cryptographic dependencies are linked to data, processes, systems and vendors	CBOM, dependency map and owner attestations
Critical data	Long-lived confidential data and signatures receive priority treatment	HNDL queue, re-encryption/re-signing and retention decisions
Cryptographic agility	Algorithms can be replaced under policy without repeated business redesign	Abstraction, policy, telemetry, testing and retirement controls
External providers	Outsourcing does not dilute accountability, evidence or exit capability	Contract terms, vendor roadmaps, test results and exit plans

The control outcomes can align with operational risk, technology risk, data security, outsourcing and continuity rules in many jurisdictions. What is universal is the desired control; what must be localised is the legal basis, approved algorithms and products, assurance route and target date.

The UK National Cyber Security Centre (**NCSC**) also addresses discovery, agility, vendor dependencies and continuity in its migration guidance[2]. This report connects initial migration with continuing governance; Chapter 6 distinguishes the scope and milestones of the reference timelines.

1.3 Intended Users and Responsibilities

This report is intended for governing bodies and risk committees, security and technology leaders, business and data owners, architecture and operations teams, and procurement, legal, compliance and audit functions. Chapters 1 and 6 support programme decisions; Chapters 2–5 explain technical exposure and design; Chapters 7–8 cover validation, evidence and oversight. The appendices provide terminology and working checklists.

1.4 Implementation Sequence

The ordering reflects dependencies: governance defines scope; scope enables discovery; discovery supports risk prioritisation; target design and vendor readiness precede pilots; evidence from pilots supports production waves and retirement.



Figure 1.1: End-to-end implementation spine

Activities can overlap, but production migration should not proceed without ownership, business impact, approved design and testable acceptance criteria.

1.5 Time-Based Risk Assessment

Let

$$T_D = \text{remaining confidentiality or signature-validity period}, \quad (1.1)$$

$$T_M = \text{time needed to discover, design, procure, test and migrate}, \quad (1.2)$$

$$T_Q = \text{time until quantum capability threatens current public-key cryptography}. \quad (1.3)$$

If

$$T_D + T_M > T_Q, \quad (1.4)$$

the required protection and migration periods exceed the potential safety window. This is commonly expressed as Mosca's inequality[18]. T_Q is uncertain, but T_D and T_M can be estimated and managed.

1.6 Scope and Localisation

The guide covers transport security, key establishment, digital signatures, authentication, code and firmware signing, certificates, data-key wrapping, distributed ledgers and external services. Quantum key distribution, quantum-computing business applications and general cyber controls are adjacent topics, not PQC migration itself.

Each institution should produce a local applicability statement covering law and supervision, permitted algorithms and products, data location, critical-infrastructure status, outsourcing approval, evidence retention, incident reporting and compatibility with the cryptographic regime in its jurisdiction.

How Quantum Computing Changes Current Security Assumptions

2.1 What financial systems rely on today

Cryptographic controls support confidentiality, integrity, authenticity and non-repudiation. Symmetric cryptography uses a shared secret, while public-key cryptography uses a public/private key pair. Most financial protocols combine them: public-key mechanisms establish a session key or validate a signature; efficient symmetric cryptography then protects bulk data. Quantum risk primarily affects the public-key layer used for key establishment, signatures and trust.

2.2 Do quantum algorithms require physical hardware?

Yes. A quantum algorithm is a mathematical procedure, but genuine quantum acceleration requires a quantum processor. Classical computers can simulate small circuits for learning and testing, but simulation cost grows rapidly.

Noisy physical qubits must generally be protected by quantum error correction to create reliable logical qubits. A credible assessment therefore considers logical qubits, logical error rates, circuit depth, correction overhead and run time, not only a vendor's physical-qubit count.

A **cryptographically relevant quantum computer (CRQC)** is a fault-tolerant quantum system able to attack real cryptographic parameters within practical time and cost. It is not a fixed product category or a simple physical-qubit threshold. No publicly demonstrated CRQC can currently break real-world RSA-2048 or mainstream ECC parameters.

2.3 Shor's algorithm: a structural public-key threat

Shor's algorithm, proposed in 1994, reduces integer factoring and discrete logarithms to efficient quantum period/order finding[13]. RSA relies on factoring being hard. In finite fields, **Diffie–Hellman (DH) key agreement** and the **Digital Signature Algorithm (DSA)** rely on discrete logarithms; on elliptic curves, **Elliptic Curve Diffie–Hellman (ECDH) key agreement** and the **Elliptic Curve Digital Signature Algorithm (ECDSA)** rely on elliptic-curve discrete logarithms. A sufficiently large, reliable quantum computer running Shor's algorithm would solve these problems in polynomial time.

For RSA modulus $N = pq$, find the order r of a random a :

$$a^r \equiv 1 \pmod{N}. \quad (2.1)$$

If r is even and $a^{r/2} \not\equiv -1 \pmod{N}$, then

$$\gcd(a^{r/2} - 1, N) \quad \text{and} \quad \gcd(a^{r/2} + 1, N) \quad (2.2)$$

typically reveal non-trivial factors. The quantum component finds r ; classical computation completes the factorisation. Increasing RSA or ECC key length alone does not remove this structural threat.

2.4 Grover's algorithm: square-root search acceleration

Grover's algorithm, proposed in 1996, does not exploit an algebraic structure like RSA's. It assumes a quantum query procedure that recognises a correct candidate. Starting from an equal superposition, an iteration alternates a **phase flip**

of marked answers with a **reflection about the average amplitude**; repeated iterations amplify the probability of observing a correct answer. For an unstructured space of N candidates with one marked answer, it requires about

$$O(\sqrt{N}) \quad (2.3)$$

ideal quantum queries rather than $O(N)$ classical queries[14]. With M marked answers, the iteration count is approximately $(\pi/4)\sqrt{N/M}$. Appendix B gives the two-dimensional amplitude-amplification argument. A query is not a free operation: each one needs a reversible implementation of candidate verification, including key scheduling, encryption or hashing, and error correction.

For a k -bit symmetric key, $N = 2^k$ and idealised query complexity changes from $O(2^k)$ to $O(2^{k/2})$: 128-bit and 256-bit keys correspond to approximately 2^{64} and 2^{128} ideal queries, respectively. This is not a structural break of AES, nor does the same formula describe every hash-collision or signature problem. Gate count, serial depth, correction and parallelisation can dominate practical cost. NIST cautions against treating the square-root result as proof that AES-128 is presently broken or requires immediate universal replacement. Institutions may select AES-256 for new or long-lived high-sensitivity systems according to risk and applicable standards, while reviewing hash and message-authentication parameters separately[15].

2.5 How Shor and Grover differ

Table 2.1: Engineering implications of Shor and Grover

Dimension	Shor	Grover
Problem	Factoring and discrete logarithms	Unstructured search
Main impact	RSA, DH, DSA, ECDH and ECDSA	Symmetric key search and hash preimages
Nature	Structural change in solvability	Square-root reduction in ideal queries
Response	Replace public-key key-establishment and signatures	Increase parameters where standards require
Hardware	Deep, fault-tolerant quantum computation	Quantum hardware is also required; practical resources remain substantial

The two algorithms are independent and both require physical quantum hardware. Shor exploits specific algebraic structure; Grover accelerates unstructured search.

2.6 From Algorithms to Business Exposure

Quantum algorithms become business attacks only when capable hardware, a vulnerable protocol, obtainable material and a usable business entry point coincide. A **confidentiality path** begins with captured traffic and handshake data protected by vulnerable public-key key exchange. A capable attacker could derive the shared secret, reconstruct the session key and decrypt stored ciphertext while it is still sensitive. An **authenticity path** begins with a public verification key: deriving the signing key could allow forged instructions, certificates or software updates to be submitted where a verifier still accepts the old algorithm. Fraud or supply-chain harm requires the attacker to reach that submission path and the forged item to be accepted; it is not automatic on the arrival of quantum hardware.

The institution must also distinguish **attack risk** from **migration risk**. Changes to algorithms, certificates and interfaces can themselves create incompatibility, latency or outages. Both the attack route and the safety of the proposed change need evidence from pilots, capacity testing and controlled recovery.

Table 2.2: Business mapping of public-key quantum exposure

Business object	Exposure and attack path	Potential effect and control
APIs and inter-firm links	Recorded vulnerable handshakes and ciphertext, or continued acceptance of old certificate signatures	Disclosure or peer impersonation; test hybrid exchange, chains and downgrade protection
Archives and backups	Obtainable public-key-wrapped data keys and historical copies	Exposure during the confidentiality period; locate copies, reduce capture and plan re-wrapping
Instructions and contracts	Verifier still accepts forgeable RSA/ECDSA signatures	False instructions may be accepted; migrate signing and retain trusted timestamps and validation evidence
Software and firmware	Attacker can both forge an old signature and enter the update-distribution chain	Malicious release accepted by legacy verifier; update signing and release controls
Customer/device identity	Legacy client or device certificate remains valid at an authentication endpoint	Identity impersonation; shorten old-certificate window and monitor negotiated mechanisms

Historical signatures do not automatically become forgeries when capable hardware emerges. Proving that a signature existed before key compromise depends on trustworthy timestamps, revocation status and archival evidence. Likewise, re-wrapping protects controlled data keys but cannot retrieve ciphertext already copied by a third party. Confidentiality and authenticity therefore require distinct prioritisation and acceptance tests.

2.7 HNDL and the migration window

Harvest now, decrypt later (HNDL) means copying encrypted traffic or data today and decrypting it when stronger capability becomes available. Prioritisation should consider confidentiality period, capture exposure, copy location, re-wrapping options, impact and migration time. Equation (1.4) enables scenario-based judgement without pretending that T_Q is precisely known.

2.8 Boundary of the threat model

Correctly implemented and deployed PQC can protect key establishment and digital signatures, and thus support future authentication and software-release trust chains against known quantum attacks. It allows new communications and long-lived sensitive data to be protected before a capable attacker exists.

Its scope is bounded. PQC cannot recover plaintext already disclosed, undo exposure of ciphertext that was captured and already decipherable, or fix weak authentication, stolen private keys, access-control errors, software defects, side channels, malicious insiders or insecure endpoints. New implementations can introduce interoperability, performance and security defects. Migration therefore belongs within broader key management, vulnerability control, incident response and continuity arrangements, with effectiveness demonstrated in actual systems and protocols.

PQC Standards and Their Mathematical Basis

3.1 Rationale for the Initial Standardised Algorithms

NIST began an open PQC standardisation process in 2016. Following multiple rounds of international review, cryptanalysis and implementation testing, it published three initial standards in August 2024:

- **ML-KEM (FIPS 203)** for shared-key establishment;
- **ML-DSA (FIPS 204)** for general-purpose digital signatures;
- **SLH-DSA (FIPS 205)** for stateless hash-based signatures[5, 6, 7].

These are not the only PQC algorithms, nor are they automatically mandatory in every jurisdiction. They are the first finalised general standards with public specifications and a broad implementation ecosystem. Production choices remain subject to local law, approved products, network rules and vendor support.

Table 3.1: Mapping current functions to PQC standards

Function	Threat	PQC response	Mathematical basis
TLS/VPN/API key establishment	Shor against ECDH, DH, RSA	ML-KEM; hybrid during transition	Module-LWE
Certificates, contracts and code signatures	Shor against RSA/ECDSA/DSA	ML-DSA	Module-LWE and Module-SIS
High-assurance/diverse signatures	Same	SLH-DSA	Hash preimage, second-preimage and collision security
Symmetric encryption and hashing	Grover	Parameter adjustment, not replacement by these three	Symmetric and hash assumptions

3.2 Lattices and Module-Lattice Structures

A lattice is a discrete set generated by integer combinations of independent basis vectors:

$$\mathcal{L}(\mathbf{B}) = \left\{ \sum_{i=1}^n z_i \mathbf{b}_i : z_i \in \mathbb{Z} \right\}. \quad (3.1)$$

Finding short or close vectors in high-dimensional lattices is believed to be hard. ML-KEM and ML-DSA use structured module lattices, providing compact and efficient operations. Institutions must use standardised parameters and reviewed implementations; changing noise, sampling or compression invalidates the security basis.

3.3 Learning with Errors and Computational Hardness

The Learning With Errors (LWE) pattern is

$$\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \pmod{q}, \quad (3.2)$$

where $\mathbf{A}$ is public, $\mathbf{s}$ is secret and $\mathbf{e}$ is a small sampled error. Without $\mathbf{e}$, linear algebra reveals $\mathbf{s}$; with suitable error, recovery relates to hard lattice problems[16, 17]. The noise is an intentional cryptographic component, not a system fault. It must hide the secret while remaining small enough for correct recovery.

3.4 ML-KEM: Module-Lattice Key Encapsulation

A **key-encapsulation mechanism (KEM)** creates a short shared secret that is then expanded into symmetric keys:

$$(pk, sk) \leftarrow \text{KeyGen}(), \quad (3.3)$$

$$(ct, K) \leftarrow \text{Encaps}(pk), \quad (3.4)$$

$$K' \leftarrow \text{Decaps}(sk, ct). \quad (3.5)$$

Correct execution gives $K' = K$. ML-KEM is derived from Kyber and uses Module-LWE, compression and a standard transformation intended to achieve chosen-ciphertext security. FIPS 203 defines ML-KEM-512, 768 and 1024; the number is a parameter-set name, not a key length.

3.4.1 Chosen-Plaintext and Chosen-Ciphertext Security

CPA (chosen-plaintext attack) security considers an attacker able to choose plaintexts and observe ciphertexts. **CCA (chosen-ciphertext attack)** security also allows selected ciphertexts to be submitted to a decryption interface, excluding the challenge itself. Network protocols face malformed input and active probing, so KEM use generally requires the stronger CCA objective and uniform invalid-ciphertext handling.

3.5 ML-DSA: module-lattice signatures

ML-DSA is derived from Dilithium. Its analysis draws on Module-LWE and Module-SIS. A simplified Module-SIS objective is to find a non-zero short vector $\mathbf{z}$ such that

$$\mathbf{A}\mathbf{z} \equiv \mathbf{0} \pmod{q}. \quad (3.6)$$

Rejection sampling prevents signature distributions from revealing private-key information. Implementations must protect randomness, sampling and timing. FIPS 204 defines ML-DSA-44, 65 and 87; selection balances signature size, throughput, device limits, certificate-chain expansion and long-term validation.

3.6 SLH-DSA: Stateless Hash-Based Signatures

SLH-DSA is derived from SPHINCS+ and does not rely on lattice problems. It combines one-time/few-time signatures with Merkle trees, anchoring many temporary public keys in a short root. Its advantages are mature hash assumptions, stateless operation and algorithmic diversity; its trade-offs are larger signatures and generally slower signing.

It can be considered for low-frequency, long-lived trust objects such as roots or firmware, or where a non-lattice alternative is required. It is not automatically superior to ML-DSA and should not be placed into high-volume messaging without capacity testing.

3.7 Hybrid Traditional and Post-Quantum Constructions

Hybrid cryptography combines a traditional and a PQC mechanism in one protocol. A hybrid key establishment may derive

$$K = \text{KDF}(K_{\text{traditional}} \parallel K_{\text{PQC}}, \text{context}). \quad (3.7)$$

RFC 10024 specifies several PQ/traditional TLS 1.3 key-agreement groups[9]. Hybrid value depends on a standard construction, context binding, downgrade resistance and consistent failure handling. Hybrid signatures must specify whether both signatures are mandatory, how certificates express the policy and when the legacy signature stops being accepted.

3.8 Algorithm Selection and Implementation Constraints

Selection follows four layers: local standards and supervision; protocol and market-infrastructure support; production-grade vendor implementation; and institution-specific performance, hardware and continuity testing. Draft protocols and preview implementations may be assessed in isolation, but must not bypass production approval. Standards, parameter sets, library versions, build options and test vectors must be recorded and checked against current errata.

End-to-End Migration Mechanism and Target Architecture

4.1 Migration Lifecycle and Completeness Criteria

Complete migration covers discovery, classification, design, coexistence, validation, cutover, observation, retirement and evidence. Replacing a library without treating certificate chains, key devices, counterparties, historical data and failure handling can create new availability or downgrade risk.



Figure 4.1: Complete migration lifecycle

A successful cutover is not completion. Completion requires dependency removal, key/certificate treatment, an adequate observation period, business sign-off and archived evidence.

4.2 Layered Target Architecture

Table 4.1: Target-architecture layers

Layer	Capability	Design requirement
Business and data	Classification, signature lifetime, transaction and regulatory processes	Protection period, outage tolerance and accountable owner
Application/protocol	TLS, API, file transfer, identity, signing and code release	No hard-coded algorithms; dual validation and version negotiation
Cryptographic service	KEM, signature, AEAD, certificate, derivation and policy	Approved catalogue, stable interface, context binding and fail closed
Key and hardware	KMS, HSM, random generation, backup and lifecycle	Size, firmware, import/export and high availability
Governance/evidence	CBOM, baseline, telemetry, exception, testing and audit	Traceable versions, explainable transitions and dated exceptions

Cryptographic agility means replacing algorithms, parameters, certificates or implementations without repeatedly rebuilding business logic, while retaining approval, testing, monitoring and safe recovery. It is policy-governed replaceability, not unrestricted switching.

4.3 TLS, APIs and inter-firm communication

First inventory clients, servers, gateways, load balancers, service meshes, counterparties and certificate chains. Test standard hybrid key establishment in isolation, measuring handshake size, CPU, memory, latency, retransmission, failure rates and middlebox compatibility. Roll out through policy cohorts or feature flags and observe the mechanism actually negotiated. Plan PQC certificate signatures separately as CA, trust-store, revocation and device support mature.

Protocols must resist downgrade. An endpoint must not silently fall back to a disallowed algorithm after malformed input or active interference. Any compatibility exception must be constrained by peer, period, data and monitoring; if confidentiality would fall below the policy floor, close the connection and invoke a continuity route.

For example, an asset manager exchanging daily valuation files with a custodian should examine the transport key exchange, the file's own encryption or signature, and certificate dependencies in gateways and batch processes. A pilot can reproduce end-of-day loads in isolated environments at both institutions and compare handshake success, file integrity, transfer time and failure recovery against the legacy baseline. Production rollout requires jointly approved compatibility windows and contingency procedures.

4.4 Digital signatures, certificates and long-term evidence

Separate short-lived online signatures, long-lived business signatures, and trust-root/code/firmware signatures. Migration options include dual signatures, parallel certificate chains, re-signing and renewed trusted timestamps. The policy must state whether both signatures are required, when legacy validation ends, how failure is recorded and how historical evidence remains verifiable.

For investment instructions or electronic contracts, the migration unit also includes the issuing authority, validation software, timestamp, revocation record and archive retrieval system. Pilots should test issuance, validation, revocation and later verification across documents of different ages and across old and new verifiers. Counterparty capability, legal effect and technical evidence must all be established before broad deployment.

4.5 Data at rest and key re-wrapping

Large datasets are commonly encrypted with a symmetric **data-encryption key (DEK)**, which is wrapped by a **key-encryption key (KEK)** or public-key mechanism. If bulk encryption remains adequate, an institution can unwrap the DEK in a controlled environment and re-wrap it under the new mechanism without rewriting all ciphertext. It must still locate backups, exports and third-party copies. Re-wrapping does not reverse prior capture of ciphertext.

For medical research data, customer records or long-lived contracts, establish the confidentiality period, all copies and key provenance before choosing between outer-key re-wrapping and full content re-encryption. The latter has greater compute, storage and business-coordination cost; the former requires assurance that the earlier ciphertext and backups were not already exposed. Both approaches require recovery exercises and evidence that data remains readable after key rotation.

4.6 Identity, software supply chain and devices

Identity providers, gateways, endpoints, build systems, artefact repositories, update services and device firmware must recognise new keys and signatures. HSMs, smart cards and secure elements must accommodate larger objects. Root trust must be updated before dependent software or devices, particularly for offline or rarely connected equipment.

Mobile applications and long-lived terminals require staged validation of client update rates, minimum support periods for old versions, offline verification and revocation propagation. The software-release chain should test dual signatures and version rollback on representative artefacts, preserving safe upgrades for old devices without allowing the compatibility period to legitimise unauthorised old releases.

4.7 Third parties and financial market infrastructures

Vendors should evidence algorithm and parameter support, product version and date, interoperability, HSM/KMS dependency, performance, errata handling, rollback, telemetry and exit. A promise to “support PQC” is not an acceptance criterion. Bilateral and multilateral links to payment networks, custodians, venues and clearing systems require joint migration windows.

4.8 Coexistence, cutover and safe rollback

Model each migration unit as a controlled state transition:

legacy baseline → hybrid pilot → hybrid production → PQC preferred → legacy retired. (4.1)

Each transition has technical gates, business approval, an observation period and recovery action. Rollback must be pre-approved, time-bound, observable and above the minimum security floor. If rollback would create unacceptable exposure, pause the channel or use an alternative business process rather than silently downgrade.

4.9 Cryptographic Policy Service, Observability and Completion

Approved algorithms, parameters, certificates and minimum versions should be centrally governed through a policy service or equivalent control plane. Telemetry records actual negotiation, parameter set, chain, version, failure, downgrade attempt, latency and responsible system, without logging secrets.

A system is complete only when inventory and ownership are confirmed; target standards and versions are approved; functional, security, performance, resilience and recovery tests pass; production telemetry proves intended use; legacy keys, certificates, paths and dependencies are retired or time-bounded; and business, technology and risk evidence is archived.

Cryptographic Inventory and Risk Prioritisation

5.1 Foundational Role of the Cryptographic Inventory

An institution cannot migrate an unknown dependency. A customer API may use ECDH at the gateway, another TLS profile internally, RSA certificates, an HSM-protected key and a vendor-specific recovery path. Configuration or certificate inventories alone miss embedded libraries, statically linked code, firmware and outsourced services.

A **cryptographic bill of materials (CBOM)** is therefore the factual basis of the roadmap. It records what algorithm is used, where, for what purpose, under whose ownership, with which dependency and migration date. Security architecture, asset management, engineering, data governance, procurement and business owners jointly maintain it; audit tests completeness and management uses an aggregated view.

5.2 Minimum CBOM fields

Table 5.1: Core CBOM fields

Group	Fields	Decision supported
Business	Process, service, owner, criticality and recovery objective	Who is affected and who approves
Data	Classification, secrecy/signature period, retention, location and exposure	HNDL priority and protection duration
Cryptographic object	Function, algorithm, parameter, protocol, certificate, library/version	What is used and whether it is vulnerable
Technical location	Application, host, container, gateway, HSM/KMS, environment and interface	Where and how to test and release
Supply chain	Vendor, product, contract, support, date and exit capability	Whether external timing supports the roadmap
Migration/evidence	Risk, target, wave, test, exception, date and discovery source	Status, confidence and remaining blockers

Use relational records rather than one unmanageable wide spreadsheet. Link business, data, systems, cryptographic objects and vendors by stable identifiers. Appendix D supplies a field template.

5.3 Discovery requires multiple sources

Table 5.2: Cryptographic discovery sources

Method	Strength	Limitation and complement
Certificate/key-platform export	Chains, algorithms, dates and key location	Misses embedded libraries; add code/runtime analysis
Code and dependency analysis	API calls, libraries, static links and hard-coded choices	False positives and dynamic configuration require review
Configuration/image scan	TLS, SSH, VPN, containers, gateways and middleware	Point-in-time view; map to environment and version
Network/protocol observation	Mechanisms actually negotiated with peers	Limited visibility; do not create new interception risk
HSM/KMS and cloud inventory	Managed keys, policy, rotation and use logs	Business purpose still needs linkage
Vendor evidence	SaaS, device and closed-product roadmap	Self-report needs test, version and contract evidence
Business/data interviews	Long-lived data, manual and shadow processes	Human knowledge needs attestation and refresh

Coverage must be reported by asset class. Scanning 90% of servers does not prove 90% coverage of processes, mobile apps, outsourcing and archives. Unknowns are themselves risks.

5.4 Dependency graph

Connect

business process → data → application/interface → protocol/algorithm → key/certificate → platform/vendor.

Shared dependencies should be planned together, and upstream trust capability should precede downstream cutover. The graph enables impact analysis when a gateway, CA, HSM or vendor date changes.

5.5 Risk scoring

Scoring supports ordering but does not replace judgement. A reference score is

$$R_i = 100(0.24D_i + 0.18E_i + 0.18C_i + 0.14A_i + 0.14X_i + 0.12M_i), \quad (5.1)$$

where values in $[0, 1]$ represent data lifetime/sensitivity (D), external exposure (E), criticality (C), algorithm vulnerability (A), dependency concentration (X) and migration difficulty (M). Institutions calibrate weights and thresholds to risk appetite.

Table 5.3: Interpretable examples of risk-factor evidence

Factor	High-score condition	Evidence
<i>D</i> Data	Confidentiality exceeds ten years; sensitive customer, transaction or strategic records	Data catalogue, retention period, business sign-off
<i>E</i> Exposure	Internet-facing, inter-firm or cloud links; traffic can be captured for long periods	Data flows, network design, third-party locations
<i>C</i> Criticality	Outage affects core service, settlement, supervision or many customers	Business impact analysis, recovery objectives
<i>A</i> Algorithm	RSA or ECC supports long-lived confidentiality or signatures	CBOM, certificates, observed protocols
<i>X</i> Dependency	Multiple peers, concentrated vendor or long device refresh cycle	Dependency graph, contracts, product roadmap
<i>M</i> Difficulty	No source code, test environment or practical change window	Architecture assessment, test conditions, change history

High scores should trigger action: for example, $R_i \geq 75$ enters an early wave or a senior-approved time-bounded exception; $50 \leq R_i < 75$ requires vendor confirmation and a planned wave. Long-lived externally collectable data may be elevated regardless of the total score.

5.6 HNDL scenario queue

For confidentiality assets, calculate

$$G_i = T_{D,i} + T_{M,i} - T_Q. \quad (5.2)$$

$T_{D,i}$ is the **remaining** confidentiality period from the assessment date, $T_{M,i}$ is the time needed for verified migration, and T_Q is the assumed time from the same date to relevant quantum capability. They must use the same unit and starting point. For a record needing ten more years of secrecy and three years to migrate, an accelerated scenario of eight years gives $G_i = 10 + 3 - 8 = 5$ years. This does not predict disclosure in five years; it shows a mismatch between the protection period and the reference scenario.

Use accelerated, central and delayed T_Q scenarios, but do not rank solely by G_i . First identify flows that can be intercepted or copied and retained, then consider confidentiality period, customer harm, available mitigations and supplier delivery dates. High-exposure assets positive across scenarios enter the first protection or migration wave; assets positive only in the accelerated scenario may first need reduced collection, shorter unnecessary retention and a pilot. Negative-gap assets still require inventory and annual review. Re-wrapping cannot undo prior third-party capture of ciphertext. This time-gap test is primarily for **confidentiality**; long-lived signatures require a separate assessment of validity periods, trusted timestamps and verification chains.

5.7 Reviewable prioritisation

for each asset in CBOM:

```

validate owner, lifetime, algorithm, exposure and dependencies
if required fields are missing:
    open remediation; increase uncertainty
score = risk_model(asset)
if long-lived and externally exposed:
    score = max(score, HNDL_PRIORITY_FLOOR)
assign provisional wave

```

for each wave:

- resolve shared dependencies and vendor dates
- review with business, security, architecture and risk
- approve, or record a time-bounded exception

The first output is provisional. Human review resolves common vendors, trust chains, counterparties and peak periods. Every adjustment records rationale, approver and review date.

5.8 Keeping the inventory current

CBOM quality metrics include critical-business coverage, mandatory-field completeness, owner attestation, freshness, vendor-evidence coverage, unknown algorithms and overdue exceptions. New systems, certificates, library upgrades, acquisitions and vendor releases trigger updates. Refresh high-risk scope quarterly, critical vendors at least semi-annually and the full scope annually, with event-driven review after major standards or threat changes.

A Reference Roadmap for Financial Institutions

6.1 Roadmap Model and Conditions of Use

This chapter gives a 60-month reference model. It combines FINMA's governance direction, current standard maturity and common dependencies in large financial institutions. It is not a statutory timetable. Each institution should treat its formal start as Day 0 and recalibrate for data lifetime, supervision, technology scale, vendor windows and change freezes.

Set two dates: a **critical-process target** for high-risk data and core processes, and a **complete-migration target** for the supplier tail, legacy systems and approved exceptions.

6.2 Mobilisation Phase: First 90 Days

1. **Days 0–30:** appoint an executive sponsor and programme lead; approve charter, scope, risk assumptions and working group; require an architecture exception for new long-lived RSA/ECC dependence.
2. **Days 31–60:** define CBOM fields; identify initial long-lived data, critical processes and vendors; issue a vendor questionnaire; select two pilot candidates.
3. **Days 61–90:** produce an initial risk/dependency view and target-architecture principles; estimate 36–60-month resources; submit a draft roadmap and pilot decision.

The objective is not full discovery in 90 days. It is a scalable governance process, information model and decision cadence.

6.3 Phase 1: governance, baseline and roadmap (months 3–12)

Complete the high-risk CBOM, data-lifetime mapping and key dependencies; obtain written version and test plans from critical vendors; approve algorithm catalogue, hybrid principles, agility requirements and minimum security floor; set risk waves and target dates; and establish budget, skills, test environment, change windows and evidence plan.

6.4 Phase 2: pilots and common capability (months 6–18)

Run at least one key-establishment and one signature pilot. Build reusable policy interfaces, test vectors, telemetry, version records, experimental certificate chains and automated evidence. The pilot demonstrates an end-to-end service under peak load and failure conditions, not merely one successful cryptographic operation.

6.5 Phase 3: high-risk migration (months 12–36)

Move HNDL-sensitive, externally exposed and long-lived-signature processes first:

scope → design → pre-production test → progressive release → observe → retire.

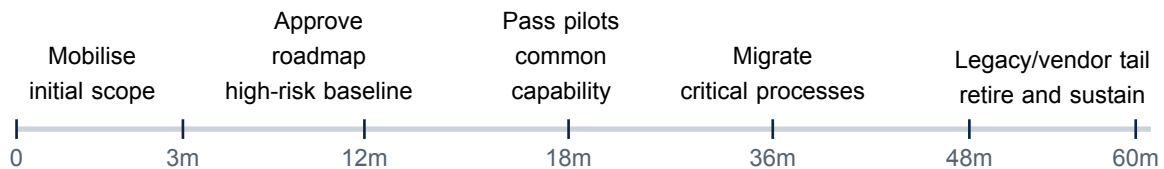
Organise waves around common protocols, trust chains, suppliers or data domains rather than organisational charts.

6.6 Phase 4: scale, tail and retirement (months 24–60)

Treat branches, mobile endpoints, firmware, acquired systems, offline archives, long contracts and unsupported legacy products. Options include replacement, isolation, a secure gateway, reduced retention/exposure or a time-bounded exception. Systems that cannot migrate and remain above appetite need a retirement date.

Transfer capabilities to normal operation: new systems use approved policy by default; change processes update the CBOM; vendors provide periodic evidence; management receives quarterly risk reporting; and standards errata or implementation vulnerabilities trigger rapid replacement.

6.7 Reference timeline and investment



Budget by capability and wave, not as a single “PQC software” purchase. Include discovery and data governance; PKI, HSM/KMS, gateway, endpoint and monitoring upgrades; vendor licences and services; engineering and business change; independent testing and audit; legacy replacement; and dual-running capacity. Link annual funding to approved baseline, passed pilots, production waves, retirement and exception closure.

6.8 Cross-Jurisdictional Applicability and Localisation

FINMA’s notice offers a useful governance structure, but its supervisory perimeter and dates apply directly only to the relevant Swiss-regulated institutions. Financial institutions elsewhere can adopt the control objectives while determining their own legal basis, sector-network dependencies and supplier constraints. The following is a localisation checklist, not a ranking of jurisdictions.

Table 6.1: Localisation checks for a cross-jurisdictional roadmap

Layer	Question to resolve	Evidence
International standards	Confirm algorithms, protocols, parameter sets and vendor versions	Standards map and interoperability results
Local regulation	Check cryptography, financial data, cyber, outsourcing and product-approval rules	Applicability register and approval route
Sector infrastructure	Confirm joint windows with payment, clearing, custody, trading and identity networks	Counterparty interface list and joint plan
Cross-border business	Determine location requirements for data, contracts, external services and audit evidence	Data-flow map and contract terms
Internal governance	Agree budget, dates and exceptions across risk, business, technology and audit	Approved roadmap and regular reporting

International standards assist technical selection and interoperability, but do not replace locally applicable standards, certification or supervisory processes. Nor should dates be copied mechanically across jurisdictions. The common foundation remains inventory, prioritisation, agility, supplier coordination, test evidence and an executable roadmap.

6.8.1 International reference milestones and institutional scheduling

NCSC's UK reference milestones are **2028** for discovery, assessment and initial planning; **2031** for the highest-priority migrations and refined plans; and **2035** for completing PQC migration[2]. FINMA's mid-2027 milestone concerns roadmap preparation. The milestones address different deliverables within their respective jurisdictions.

Institutions can map their inventory baseline, pilots, high-risk migration and legacy retirement to relevant reference milestones, then schedule against local requirements, data-protection periods and vendor releases. This report's 60-month model starts at institutional mobilisation and does not imply a common global completion year. Prioritise controls where long-lived data has a narrowing protection window; where mature protocols or products are unavailable, document dependencies, interim controls and review dates.

6.9 Annual and event-driven review

Reassess the roadmap at least annually and after standards revisions, major algorithm or implementation defects, regulatory change, critical-vendor delay, acquisition or cloud migration, data-classification change or a credible quantum-capability step. Record how changed assumptions affect waves, budget and dates.

Pilot Validation, Resilience and Production Acceptance

7.1 Four questions for every pilot

A financial-sector pilot uses real protocols, representative product versions and load in an isolated or pre-production environment. It asks: is the algorithm/parameter/construction approved; does the service remain functional and interoperable; does it respond safely to network, key, certificate, device or vendor disturbance; and can configuration, actual negotiation, exceptions and evidence be operated continuously?

7.2 Pilot 1: hybrid TLS/API

Select a bounded API or inter-firm connection. Compare an approved hybrid key-agreement configuration with a traditional baseline under identical hardware, network, dataset and load. Cover clients, servers, gateways and libraries; full and resumed handshakes; concurrency and large messages; CPU, memory, bandwidth, error and retransmission; middleboxes and observability; approved, prohibited, unknown and downgrade combinations. Report P95/P99 as well as means.

7.3 Pilot 2: file exchange or digital signature

Use a contract, investment instruction, regulatory report, software artefact or batch file. Test signature, validation, timestamp, archive and audit flow. If dual signing is used, define the policy and end date. Measure throughput, object size, storage, batch window, HSM/KMS compatibility, invalid-signature rejection, key rotation, revocation, historical validation and disaster recovery.

7.3.1 Financial infrastructure reference case: Project Leap

The Bank for International Settlements (**BIS**) published Project Leap Phase 2 in December 2025. In an isolated Eurosystem T2 test environment, it tested post-quantum signatures for liquidity-transfer messages, including function, performance and interoperability, and identified processing differences and component changes[3].

The experiment used an earlier Dilithium version rather than final ML-DSA, substituted software key files for physical hardware security modules, and did not cover the full production flow. It supports controlled feasibility testing; deployment still requires current standards, protected key custody and complete-chain validation.

An institution can begin with one repeatable message or file flow, assigning sender, recipient, network-provider and verifier responsibilities. Include **standard version, key protection, message capacity, invalid-input rejection and peak load** in the pilot. Preserve reusable interfaces, test cases and delivery documents, then validate each subsequent wave's differences.

7.4 Disturbance-testing framework

Table 7.1: Resilience and disturbance tests

Disturbance	Injection	Expected safe behaviour
Network/message	Delay, loss, reordering, fragmentation, replay, truncation, oversized handshake	Bounded retry, no exhaustion; controlled failure and alert
Negotiation	Remove PQC group, fake non-support, duplicate extension, parameter mismatch	Block unauthorised downgrade; only approved compatibility fallback
Certificate/signature	Expired, revoked, unknown or invalid chain/signature	Reject consistently, preserve evidence and never bypass validation
Key/randomness	Missing key, HSM outage, interrupted rotation, random-source fault	Do not reuse unsafe keys; fail over safely or stop and alert
Capacity	Peak concurrency, CPU/memory pressure, queue/log exhaustion	Rate-limit and isolate; never select a weaker algorithm under pressure
Deployment/vendor	Partial upgrade, version rollback, cloud-region failure, peer drift	Detect drift, isolate/recover by wave, preserve state and traceability

Each test states precondition, injection, expected result, maximum recovery time, stop condition, evidence location and owner.

7.5 Safe rollback and fail closed

Define three outcomes in advance: return to the previous approved secure state when it remains within appetite; switch to an alternative secure channel or manual continuity process; or fail closed when any fallback would breach the minimum security floor. A rollback package must restore configuration, certificate, key reference, application version, database change and dependency state consistently, without duplicate processing or audit gaps.

7.6 Implementation security and side channels

Test constant-time behaviour, cache/branch leakage, uniform failure handling, randomness and sampling, memory clearing, malformed input, software provenance and HSM/accelerator boundaries. High-risk implementations require independent review or laboratory assessment. Algorithm conformance, protocol conformance, implementation security and business acceptance are separate forms of evidence.

7.7 Capacity and release gates

For scenario j , a planning model is

$$\Delta C_j = N_j(\Delta c_{\text{cpu}} + \Delta c_{\text{mem}}) + B_j \Delta s_{\text{wire}} + S_j \Delta s_{\text{store}}. \quad (7.1)$$

Calibrate it using peak testing and telemetry, including HSM queues, chain expansion, MTU/fragmentation, endpoint energy and recovery capacity.

Production entry requires approved standards and versions; passed function, interoperability, capacity, disturbance and recovery tests; no open high-severity defect; telemetry and alerting; an exercised rollback/fail-closed route; and business, security, technology, risk and change approval.

7.8 Evidence package and observation period

Retain the wave's approved versions, inventory, test evidence and release decisions. Monitor negotiated mechanisms, failures, downgrades, tail latency, resources, HSM queues, certificate anomalies, drift and customer impact. Completion requires an accepted observation period, legacy treatment and business sign-off; Chapter 8 defines the evidence package.

Governance, Supply Chain and Continuous Assurance

8.1 Governance model

PQC migration should sit within existing technology-risk and operational-resilience governance. The board or risk committee oversees appetite, dates, major investment and high-risk exceptions. An executive steering committee resolves cross-functional trade-offs. A programme office maintains roadmap, dependencies, budget and evidence.

Table 8.1: Core accountability

Activity	Business	Security	Tech	Risk	Note
Data lifetime/business impact	A/R	C	C	C	Business owns protection period and impact
CBOM/dependency map	C	A	R	C	Engineering discovers; security owns control criteria
Architecture/algorithm catalogue	C	A/R	R	C	Legal, procurement and local crypto expertise advise
Pilot/production acceptance	A	R	R	C	Business signs availability; security signs the floor
Risk exception	R	C	C	A	Material/long exceptions escalate
Independent assurance	C	C	C	R	Internal audit remains independent

A means accountable, R responsible and C consulted. Each material decision needs one clear accountable owner.

8.2 Board reporting

Quarterly reporting should show high-risk business and long-lived-data coverage; weighted exposure and trend; wave status, blockers and date variance; critical-vendor readiness and exit options; pilot/production performance and defects; exception count, impact and expiry; and whether funding, skills and environments are sufficient.

8.3 Vendor governance and contracts

Questionnaires cover algorithms, versions, PQC/hybrid support, dates, certificate/HSM dependencies, performance, upgrade, compatibility, vulnerability response, telemetry and retirement. Self-report is supported by release evidence, testing, independent assessment or contract.

New and renewed contracts should require an updated cryptographic inventory and roadmap; production versions and interoperability windows; timely notification and remediation after standards or implementation change; access to configuration, logs and evidence; upgrade, dual-run, data-export and exit terms; and equivalent obligations for subcontractors.

8.4 Exception management

An exception records the asset, data, reason, score, compensating control, business impact, vendor dependency, owner, approver, expiry and exit plan. Escalate where long-lived data or core business is involved, the minimum policy is breached, the term exceeds one year, the vendor has no credible roadmap or several systems share the exception. Isolation, retention reduction, gatewaying, monitoring or manual review are mitigations, not completed migration.

8.5 Metrics

Table 8.2: Suggested management indicators

Category	Examples	Purpose
Inventory quality	Critical-process coverage, field completeness, owner attestation, freshness	Confidence in the decision base
Risk reduction	HNDL flows, vulnerable interfaces, risk-weighted exposure	Substantive reduction in exposure
Delivery	On-time waves, pilot pass, retirement and exception closure	Execution quality
Supply chain	Vendor roadmap, interoperability and contract coverage	External dependency control
Operations	PQC/hybrid use, failure/downgrade, tail latency, incidents	Real production state
Capability	Agile-interface adoption, automated tests and trained roles	Long-term sustainability

Every coverage measure needs a clear denominator and disclosed uncertainty.

8.6 Three lines and evidence architecture

The first line implements and operates controls; the second line sets policy, challenges assessments and oversees exceptions; internal audit independently tests governance, inventory completeness, test quality and reporting. These are distinct responsibilities within the same migration wave, not three signatures on one form. The first line supplies the inventory, code/configuration versions, business tests and post-release negotiation logs, and accepts operational continuity. The second line challenges omissions, parameter selection, vendor claims and rollback against risk appetite, escalating exceptions below the policy floor. Internal audit does not design or approve the technology change; it evaluates whether the first two lines' decisions and evidence are credible.

Independent assurance should use **two-way sampling**: trace a critical process backwards through data, interface, algorithm, key, supplier and test evidence to find missing dependencies; then sample CBOM entries forward into production configuration, actual negotiation and ownership. Recalculate selected risk scores, replay material failure and approved rollback tests, and confirm legacy mechanisms are retired or governed by dated exceptions. Increase sample depth around high-risk flows and major cutovers, then check telemetry again during the observation period. Inventory/production mismatches, unauthorised downgrades and missing evidence require remediation, retesting and escalation to the appropriate risk committee. Project status reports alone do not establish control effectiveness.

Index evidence under decision, asset, design, test, release, operation, exception and vendor. Each item has an ID, object, version, date, owner, approval, location, retention and access class. Content can remain in specialist systems, but a unified index must make it reviewable.

Each wave's package retains scope, architecture, inventory snapshot, standards and software versions, configuration, test-data summaries, cases and results, defects, performance baselines, disturbance and rollback records, vendor evidence, approvals and production telemetry.

8.7 Maturity and continuing update

Use five maturity levels: awareness; visibility; verifiability; migration capability; and sustained agility. Rate individual business domains, not just the institution as a whole. Continue standards watch, vendor review, inventory refresh, vulnerability response and annual exercises after the programme. External communication must distinguish roadmap, pilot, critical-process migration and full completion.

8.8 Continuous cryptographic governance and operations

Cryptographic assets change with application releases, certificate rotation, cloud changes and vendor upgrades. At operational handover, assign owners, funding, tools and escalation routes so that the inventory, policies and tests established during migration remain useful.

In its May 2026 company commentary, e23 organises discovery, agility and automation within its proprietary Triple-A framework[4]. Drawing on its attention to continuing management, this section proposes operational arrangements within financial institutions' change and risk processes. Applicable standards and approved policies govern algorithm and product selection.

1. **Inventory refresh and change closure.** Link scans, releases and certificate/key records to the CBOM. A new interface, library upgrade, certificate renewal or outsourced-service change triggers a difference check. New traditional dependencies, unknown versions or missing owners create dated remediation tickets. Preserve discovery source, confirmation date and version.
2. **Policy approval and algorithm lifecycle.** Record algorithms, parameters and implementation versions as approved, pilot-only, temporarily compatible or prohibited, with activation, replacement and retirement conditions. Automation executes approved policy. Material algorithm withdrawal, trust-root updates and production cutovers receive change approval; emergency actions follow pre-authorised plans and retain execution and review evidence.
3. **Actual-use monitoring and regression testing.** Check the mechanisms used for transport key establishment, authentication and digital signatures separately. After library, gateway, peer or certificate-chain changes, rerun affected interoperability, failure and performance cases. Unauthorised downgrade, drift or rising errors trigger controlled blocking, alerts and recovery under the runbook.
4. **Operating cadence and service handover.** Consider monthly reviews of assets, incidents and certificate expiry, quarterly reports on risk, suppliers and exceptions, and annual roadmap and recovery exercises, calibrated to business risk. Contracts for external services define update scope, response times, retest evidence, data custody and exit. The institution retains policy approval and risk accountability.

These arrangements carry discovery, controlled replacement, retesting and evidence updates through the system lifecycle. Assess both closure of old dependencies and the time and cost of later changes. Operating records, remediation timeliness and reviewable tests demonstrate sustainable maintenance capability.

Glossary and Acronym Index

Term	Plain definition	Page
FINMA	Swiss Financial Market Supervisory Authority; Guidance 05/2026 provides a governance reference for this report.	1
NCSC	National Cyber Security Centre, UK; its PQC guidance provides UK reference milestones and organisational migration methods.	1
BIS	Bank for International Settlements; Project Leap is a post-quantum validation case in financial payments.	19
PQC	Cryptography running on classical computers and designed to resist known classical and quantum attacks.	6
CRQC	Fault-tolerant quantum computer able to attack real cryptographic parameters.	3
HNDL	Capture ciphertext now and decrypt it after future capability becomes available.	5
RSA	Public-key cryptography based on the assumed hardness of integer factoring.	3
ECC	Elliptic-curve key agreement/signature cryptography based on discrete logarithms.	3
DH	Diffie–Hellman key agreement based on finite-field discrete logarithms.	3
DSA	Digital Signature Algorithm based on finite-field discrete logarithms.	3
ECDH	Elliptic Curve Diffie–Hellman key agreement based on elliptic-curve discrete logarithms.	3
ECDSA	Elliptic Curve Digital Signature Algorithm.	3
Shor	Quantum algorithm for factoring and discrete logarithms.	3
Grover	Quantum square-root acceleration for unstructured search.	3
ML-KEM	FIPS 203 module-lattice key-encapsulation standard.	7
ML-DSA	FIPS 204 module-lattice digital-signature standard.	7
SLH-DSA	FIPS 205 stateless hash-based signature standard.	7
KEM	Public-key mechanism that establishes a short shared secret.	7
CPA	Chosen-plaintext attack model.	7
CCA	Chosen-ciphertext attack model.	7
LWE / Module-LWE	Learning-with-errors problems used in ML-KEM/ML-DSA analysis.	7
Module-SIS	Short-integer-solution problem used in ML-DSA analysis.	7
CBOM	Inventory linking cryptographic mechanisms to location, purpose, owner and dependencies.	12
Cryptographic agility	Policy-controlled ability to replace algorithms, parameters, certificates and implementations.	9

Term	Plain definition	Page
Hybrid cryptography	Combining traditional and PQC mechanisms in one protocol.	8
TLS	Transport Layer Security; protocol protecting network connections.	10
PKI	Infrastructure for certificate issuance, trust, revocation and validation.	10
HSM	Hardware security module protecting key generation, storage and use.	9
KMS	Key-management system governing key creation, authorisation, rotation, backup and destruction.	10
DEK	Symmetric data-encryption key.	10
KEK	Key-encryption key used to wrap or protect a DEK.	10
RTO / RPO	Recovery-time objective / recovery-point objective.	12
QKD	Quantum key distribution; distinct from software-deployable PQC.	2

Mathematical Appendix

B.1 RSA correctness and Shor's entry point

Choose primes p, q , let $N = pq$ and $\varphi(N) = (p-1)(q-1)$. Select e with $\gcd(e, \varphi(N)) = 1$ and d such that

$$ed \equiv 1 \pmod{\varphi(N)}. \quad (\text{B.1})$$

For a suitable message m , encryption gives $c = m^e \bmod N$, and

$$c^d = m^{ed} = m^{1+k\varphi(N)} \equiv m \pmod{N}. \quad (\text{B.2})$$

Shor does not guess d . It finds the order r of random a modulo N . If r is even,

$$(a^{r/2} - 1)(a^{r/2} + 1) \equiv 0 \pmod{N},$$

and $\gcd(a^{r/2} \pm 1, N)$ usually reveals a factor under non-trivial conditions. Efficient quantum order finding is the structural break.

B.2 Grover amplitude amplification

For one target state $|w\rangle$ among N states,

$$|s\rangle = \frac{1}{\sqrt{N}}|w\rangle + \sqrt{\frac{N-1}{N}}|w^\perp\rangle. \quad (\text{B.3})$$

Let $\sin \theta = 1/\sqrt{N}$. Each Grover iteration rotates the state by about 2θ toward the target. After t iterations, target amplitude is approximately $\sin((2t+1)\theta)$. It is maximised near

$$t \approx \frac{\pi}{4}\sqrt{N}. \quad (\text{B.4})$$

This is a query-complexity result; real gates, depth, correction and parallelisation add major cost.

B.3 Intuition for LWE correctness

Given $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \pmod{q}$, a simplified encryption forms

$$\mathbf{u} = \mathbf{A}^\top \mathbf{r} + \mathbf{e}_1, \quad (\text{B.5})$$

$$v = \mathbf{b}^\top \mathbf{r} + e_2 + \lfloor q/2 \rfloor m. \quad (\text{B.6})$$

The receiver computes

$$v - \mathbf{s}^\top \mathbf{u} = \mathbf{e}^\top \mathbf{r} + e_2 - \mathbf{s}^\top \mathbf{e}_1 + \lfloor q/2 \rfloor m \pmod{q}. \quad (\text{B.7})$$

If total error is well below $q/4$, the result is near 0 or $q/2$ and recovers m . Real ML-KEM adds module-lattice structure, compression and a standard transformation; this simplified equation is not an implementation specification.

B.4 KEM security: IND-CPA and IND-CCA

An indistinguishability experiment asks an attacker to distinguish a real secret/ciphertext from a random challenge. Advantage is

$$\text{Adv} = \left| \Pr[b' = b] - \frac{1}{2} \right|. \quad (\text{B.8})$$

IND-CPA permits chosen plaintexts; IND-CCA additionally permits decryption/decapsulation queries other than the challenge. Network KEM use requires strong active-attack handling. Distinguishable errors or timing can violate proof assumptions.

B.5 Signatures and rejection sampling

Correctness requires

$$\Pr[\text{Verify}(pk, m, \text{Sign}(sk, m)) = 1] \approx 1.$$

Security commonly targets existential unforgeability under adaptive chosen-message attack. Lattice signatures use rejection sampling so output distribution does not expose secret-dependent values. Random generation, sampling bounds and timing must follow the standard.

B.6 Hybrid key combination

If traditional and PQ mechanisms produce K_1, K_2 , a standard protocol may derive

$$K = \text{HKDF-Extract}(\text{salt}, K_1 \parallel K_2) \quad (\text{B.9})$$

and bind protocol, role, algorithm identifiers and transcript in later derivation. Security depends on the construction, input independence, failure handling and authentication; it does not follow merely from concatenating two algorithm names.

Mobilisation and Vendor Questionnaire

C.1 Internal questions

1. Which data needs confidentiality for more than 5, 10 or more years? Which signatures remain legally or operationally valid for long periods?
2. Which external links, archives, backups or cloud services allow ciphertext to be collected?
3. Where are RSA, DH, ECDH, DSA and ECDSA used, and what evidence supports the inventory?
4. Which keys are managed by HSM/KMS and which are embedded in applications or devices?
5. Which critical processes share a CA, gateway, vendor or hardware dependency?
6. Can test environments reproduce peers, peak load, recovery and failure?
7. Who approves the algorithm catalogue, exception, production cutover and retirement?

C.2 Vendor questions

1. Which cryptographic algorithms, protocols and libraries are used in each supported version?
2. Which PQC or hybrid standards are supported, in which release and on what date?
3. Which clients, HSMs, KMSs and gateways have completed interoperability?
4. What are the effects of key, ciphertext, signature and certificate size?
5. How are downgrade, telemetry, standards errata, side channels and emergency withdrawal handled?
6. Does upgrade require hardware, downtime, data migration or licence change?
7. Are test environments, vectors, evidence, support and exit/data export available?

CBOM Field Template

Group	Suggested fields	Completion guidance
Identity	Record ID, discovery date/-source, environment	State production, recovery or test and automatic/manual source
Business	Process, service, owner, criticality	Link to service catalogue and impact analysis
Data	Classification, confidentiality/signature period, retention, location	Separate confidentiality, integrity, authenticity and legal period
System	Application, component, host/-container, interface, peer	Record version, location and asset identifier
Cryptography	Function, algorithm, parameter, protocol, library/version	Separate key establishment, signature, symmetric, hash and random
Key/certificate	ID, chain, HSM/KMS, rotation and validity	Never store private keys or shared secrets in the CBOM
Dependency	Upstream/downstream, gateway, PKI, hardware, vendor	Identify common dependencies and sequencing
Risk	HNDL, exposure, impact, vulnerability, difficulty and score	Preserve inputs, model version and adjustment rationale
Plan	Target, wave, owner, date and blocker	Link project, budget and vendor milestone
Validation/status	Test, defect, sign-off, evidence, state and exception	Every change records date, approver and evidence

Implementation and Acceptance Checklist

E.1 Before roadmap approval

- ☐ Executive sponsor, programme lead and cross-functional accountability assigned.
- ☐ Local legal, supervisory, cryptographic and data applicability documented.
- ☐ Critical business, long-lived data and critical vendors in scope.
- ☐ CBOM fields, discovery, quality and ownership approved.
- ☐ Separate critical-process and complete-migration dates set.
- ☐ Multi-year budget, skills, environments and vendor windows assessed.

E.2 Before pilot production entry

- ☐ Algorithm, parameter, standard, software and firmware versions approved.
- ☐ Function, interoperability, peak capacity, recovery and long-term validation tested.
- ☐ Disturbance, downgrade protection and rollback/fail-closed route exercised.
- ☐ No open high-severity defect; other defects have owners and dates.
- ☐ Telemetry, alerting and operational runbooks integrated.
- ☐ Business, security, technology, risk and change approval complete.

E.3 Before declaring completion

- ☐ Observation period passed and actual negotiation matches policy.
- ☐ Upstream, downstream, recovery, backup and third-party dependencies confirmed.
- ☐ Legacy keys, certificates, paths and configuration retired or time-bounded.
- ☐ CBOM, dependency map, risk and vendor status updated.
- ☐ Evidence is complete, searchable and retained.
- ☐ Business owner confirms risk reduction and service performance.
- ☐ Operational owner, inventory triggers, algorithm lifecycle and retesting agreed.
- ☐ Maintenance contracts or internal service arrangements define response, evidence custody and exit.

Standards and Regulatory Snapshot

Snapshot date: 1 October 2026. Confirm current versions and errata before implementation.

Table F.1: Core sources and use

Source	Status/topic	Use
FINMA Guidance 05/2026	July 2026 supervisory guidance	Strategy, roadmap, inventory, critical data, agility and vendors
NCSC PQC migration guidance	March 2025; UK reference timeline	Discovery, priorities, continuity and schedule calibration
BIS Project Leap Phase 2	December 2025; payment-system experiment	Signature migration, interoperability, performance and pilot limits
NIST FIPS 203/204/205	Final ML-KEM, ML-DSA, SLH-DSA standards	Algorithms, parameters and conformance baseline
NIST SP 800-227	KEM recommendations	KEM interface, use and security considerations
IETF RFC 10024	PQ/traditional hybrid TLS 1.3 key agreement	Hybrid groups and interoperability
NIST NCCoE SP 1800-38 drafts	Cryptographic discovery and migration practice	Inventory, tools, testing and migration
CISA/NSA/NIST readiness guidance	Organisational preparation	Roadmap, inventory and supply chain
Chinese cryptography/financial data rules	Law, commercial cryptography and data-security regime	China-specific applicability, product and assurance constraints

Bibliography

- [1] FINMA. [Guidance 05/2026: Cyber risks associated with quantum computing](#). 2026.
- [2] NCSC. [Timelines for migration to post-quantum cryptography](#). 20 March 2025. UK implementation reference.
- [3] BIS Innovation Hub. [Project Leap: Quantum-proofing payment systems, Phase 2](#). 11 December 2025. Financial-sector test report.
- [4] e23. [The Great Cryptographic Reset: The Rise of Quantum Safe Financial Infrastructure](#). 19 May 2026. Company commentary on continuing governance.
- [5] NIST. [FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard](#). 2024.
- [6] NIST. [FIPS 204: Module-Lattice-Based Digital Signature Standard](#). 2024.
- [7] NIST. [FIPS 205: Stateless Hash-Based Digital Signature Standard](#). 2024.
- [8] NIST. [SP 800-227: Recommendations for Key-Encapsulation Mechanisms](#). 2025.
- [9] IETF. [RFC 10024: Post-Quantum Traditional Hybrid Key Agreement Mechanisms for TLS 1.3](#). 2026.
- [10] NIST NCCoE. [Migration to Post-Quantum Cryptography and SP 1800-38 Drafts](#).
- [11] CISA, NSA and NIST. [Quantum-Readiness: Migration to Post-Quantum Cryptography](#). 2023.
- [12] ETSI. [TR 103 619: Migration Strategies and Recommendations to Quantum Safe Schemes](#). 2020.
- [13] Peter W. Shor. [Algorithms for Quantum Computation: Discrete Logarithms and Factoring](#). 1994.
- [14] Lov K. Grover. [A Fast Quantum Mechanical Algorithm for Database Search](#). 1996.
- [15] NIST. [Post-Quantum Cryptography: Frequently Asked Questions](#), discussion of Grover’s algorithm and AES key lengths.
- [16] Oded Regev. [On Lattices, Learning with Errors, Random Linear Codes, and Cryptography](#). 2009.
- [17] Chris Peikert. [A Decade of Lattice Cryptography](#). 2016.
- [18] Michele Mosca and Marco Piani. [Quantum Threat Timeline Report](#). 2018.
- [19] National Cryptography Administration. [Cryptography Law of the People’s Republic of China](#).
- [20] State Council of China. [Regulations on the Administration of Commercial Cryptography](#). 2023.
- [21] National Financial Regulatory Administration. [Measures for Data Security Management of Banking and Insurance Institutions](#). 2024.

From roadmap to a verifiable quantum-safe migration capability

See the assets. Explain the risk.
Control the transition. Evidence the outcome.

Hui Gong | UCL Institute of Finance & Technology

Version 1.4 | 1 October 2026 | Public release

