

N2SF- Based Post- Quantum Cryptography Migration Framework for Data in National and Public Institutions

Do-Yun Park¹[0009-0002-2201-6161], Yu-Lim Hyoun¹[0009-0009-7530-6497],
and Hwa-Jeong Seo¹[0000-0003-0069-9061]*

Department of Convergence Security,
Hansung University, Seoul (02876), South Korea
{rhcp030418,yulin4hyoung}@gmail.com

Abstract. This paper proposes post-quantum cryptography (PQC) migration decision criteria for data in national and public institutions, based on the protection requirements of the National Network Security Framework (N2SF) and the cryptographic dependencies and operational states of data. The criteria distinguish payload retention, key rewrapping, re-encryption, communication migration, and signature renewal. Required actions are separated from decisions to allow, defer, or deny execution. Completion requires consistent results for the same object and version, recovery of registered copies, and current authority. A reference implementation examined these decisions under supported conditions. In 225 policy trials, completion/read rechecking and per-stage rechecking exhibited none of the unauthorized completions or reads observed with initial-decision reuse. Nine interruption trials confirmed resumption by reconciling artifacts with records. These results support the feasibility of the criteria in the local reference environment; institutional approval, production key management, and long-term preservation continuity remain unvalidated.

Keywords: Post-quantum cryptography · N2SF · Migration decision criteria · Execution conditions · Assessment evidence

1 Introduction

Cryptographic migration in response to advances in quantum computing requires not only replacing algorithms but also identifying the cryptographic dependencies of data and systems and establishing corresponding migration actions and verification procedures. Information that requires long-term confidentiality may be exposed to the harvest now, decrypt later (HNDL) threat, in which ciphertext collected today is decrypted in the future. Migration priorities should therefore be determined by considering the remaining protection period of the data, the possibility of ciphertext collection, and the time required for system migration.[16,7]

* Corresponding author: hwajeong84@gmail.com.

For data at rest, the data encryption key (DEK) used to encrypt the payload must be distinguished from the key that protects the DEK.[3] The stored encrypted representation of a DEK is referred to as a key envelope. Rewrapping replaces this envelope for a new recipient key while retaining the payload and DEK. Determining whether the payload can be retained requires examining not only algorithm security but also DEK compromise, nonce usage history, and payload integrity. A nonce is a value used during encryption. In GCM, a mode of operation that protects both confidentiality and integrity, nonce reuse must be prevented when different data are encrypted under the same key.[22,8]

Even after these conditions establish the technically necessary actions, execution and completion require separate decisions. Insufficient approval evidence requires deferring the selected action, and a backup that has not been migrated and still depends on the decryption key for the old envelope requires deferring retirement of that key. It is therefore necessary to distinguish required migration actions, current permission to execute them, and completion after execution.

National and public institutions examine information classifications and threats and select security controls according to the National Network Security Framework (N2SF).[18,19] Applying these controls to post-quantum cryptography (PQC) migration for individual data items requires specifying the necessary migration actions and the conditions under which execution and completion can be accepted. Decision criteria must therefore consider institutional protection requirements together with the cryptographic dependencies and operational states of data.

The main contribution of this paper is N2SF-based PQC migration decision criteria for data exchanged between institutions and subsequently stored and preserved internally. The criteria are specified and examined as follows.

- **Migration decision criteria.** Information classifications and security controls determined under N2SF, together with institution-provided protection periods and exposure information, are examined alongside cryptographic dependencies and operational states. These inputs are used to specify the conditions for retaining the payload and those requiring key rewrapping, re-encryption, communication migration, or signature renewal.
- **N2SF assessment evidence.** Technically necessary actions are distinguished from execution permission based on approval and support evidence. Processing results for the same data and version, recovery of registered copies, and current policy are linked to migration completion decisions. Re-assessment and recovery results following policy changes and interruptions are organized as N2SF assessment materials.
- **Feasibility evaluation.** Migration, deferral, denial, and recovery are examined under supported conditions in a reference implementation that connects transmission, storage, signatures, and copy recovery. The processing costs of rewrapping and re-encryption, behavioral differences associated with policy check timing, and recovery at designated interruption points are evaluated to establish the feasibility and applicability limits of the criteria.

2 Background

2.1 N2SF

According to the *National Network Security Framework (N2SF) Security Guidelines 1.0*, N2SF classifies business information and information systems as Classified (C), Sensitive (S), or Open (O), and calls for protection measures based on information flows and threats.[18,19] The users, resources, and information flows considered in information service modeling are used for analysis at the business and system levels. A data object in this paper is a file-level data item to which the results of that analysis are applied for migration and verification; this definition does not restrict the scope of N2SF analysis to files.

N2SF is applied in five stages: preparation (Prepare), classification (Categorize), threat identification (Identify), formulation of security measures (Select), and adequacy assessment and adjustment (Assess). Security requirements are derived from identified threats, and controls are selected according to the C/S/O classification and institutional environment.[18] Relevant controls include cryptographic key management (EK), use of cryptographic modules (EA), data transmission (DT), and data use (DU). Information flow (IF) and Cross Domain Solution (CDS; control domain CD) address transfer boundaries, while information system components (IN) addresses inventories and change management.[19]

Korea's 2025 comprehensive PQC migration plan presents a roadmap from 2025 to 2035, including asset discovery, risk assessment, migration procedures, and cryptographic module validation.[26] The U.S. National Cybersecurity Center of Excellence (NCCoE) likewise emphasizes cryptographic inventories and interoperability testing.[20,21] Algorithm standardization, library support, module validation, and institutional deployment approval are separate items to be verified; a national target year does not establish approval or completed migration of an individual system.

2.2 PQC

Data storage and transmission use distinct cryptographic functions, including payload encryption, key establishment and protection, and digital signatures. When a symmetrically encrypted payload depends on an encryption key protected by public-key cryptography, the payload and key protection layer are separate migration targets.[3]

The U.S. National Institute of Standards and Technology (NIST) states that AES-256 is expected to remain secure for a long time even when known quantum attacks are considered.[22] This provides a basis for examining whether the payload can be retained, although the actual decision depends on key management, nonce usage, and integrity conditions. GCM authenticated decryption returns failure when the authentication tag does not match.[8]

In 2024, NIST published ML-KEM (FIPS 203), ML-DSA (FIPS 204), and SLH-DSA (FIPS 205) as post-quantum cryptography standards. ML-KEM establishes a shared secret, whereas ML-DSA and SLH-DSA generate and verify

digital signatures.[24,23,25] Data protection using ML-KEM derives an encryption key from the shared secret and protects the payload using symmetric-key cryptography.[2]

2.3 Related Work

Cryptographic dependency analysis connects data, IT assets, and cryptographic assets to identify relationships among keys, certificates, and protocols. Hasan et al. use existing inventories and graphs to identify dependencies and replacement priorities.[12] Migration priority assessment considers data criticality, the remaining protection period, and potential ciphertext collection paths in relation to HNLD risk.[7] The review by Näther et al. and ETSI TR 103 619 organize migration activities and stages.[17,9] Software-Defined Cryptography (SDC) separates policy determination from enforcement and applies policy changes to cryptographic operations.[6]

PQConnect provides post-quantum protection at the network layer and formal verification of its handshake.[4] Vault’s rewrap function updates input ciphertext to a new key version without returning plaintext.[13] This function is used as an example of updating the key layer, whereas rewrapping in the proposed criteria is restricted to the envelope protecting the payload’s DEK.

Bindel et al. analyzed resistance to forgery and the separability of component signatures when multiple signature schemes are combined.[5] Evidence Record Syntax (ERS) links data to time evidence and supports its renewal, and Digital Signature Service (DSS) processes ERS preservation structures.[11,10] The confidentiality period and authenticity verification period are distinct: even after a document becomes public, it may remain necessary to establish the validity of a past signature, and merely adding a new signature does not establish that validity.

The proposed criteria use these foundations to distinguish necessary actions from execution conditions for each target and to record their grounds as N2SF assessment materials.

3 Migration Decision Criteria

N2SF-based PQC migration decisions determine necessary actions and examine urgency based on protection requirements, cryptographic dependencies, and operational states. They then determine whether to allow, defer, or deny execution. After execution, completion requires verification of processing results and registered copies. Changes in conditions or interruptions trigger reassessment and recovery. The grounds and results are recorded as materials for N2SF adequacy assessment and adjustment.

3.1 Inputs

The criteria concern data exchanged between institutions and stored internally, together with their encryption, authentication, signatures, and copies. Items to

be checked under N2SF cryptographic key management and data protection controls are specified as conditions for action selection, execution, and completion. Section 3.4 maps these proposed conditions to official controls.

A data object o denotes an individual managed data item, and version v denotes its content version. The pair (o, v) is the protection unit to which migration and verification results pertain. Each unit is associated with payload ciphertext C , the key envelope W protecting its DEK, and registered copies R . Object and version identifiers establish which unit the materials belong to, and hash comparisons establish whether they match the recorded ones. In Figure 1, S denotes signature and verification materials for a migration record containing the identifiers and hashes of the payload, envelope, and copy manifest.

Artifacts and encryption relationships for a protection unit

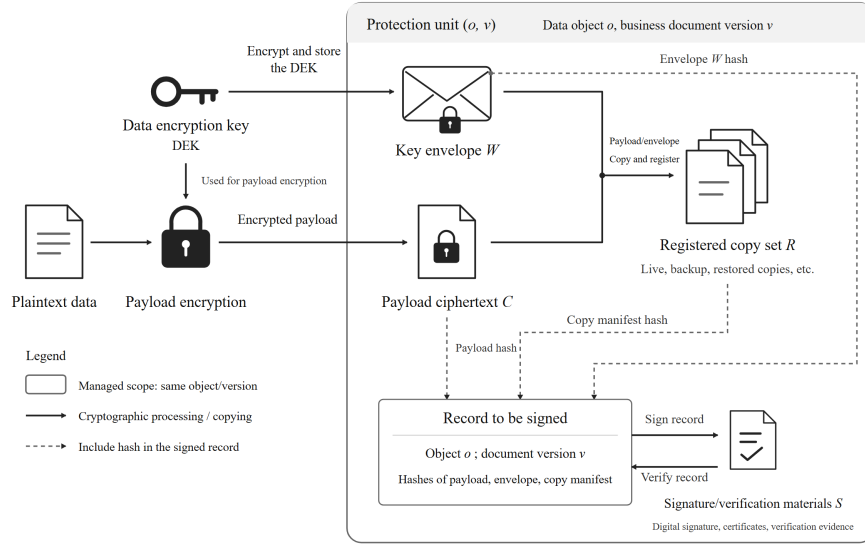


Fig. 1. Protection unit and associated materials. Signatures cover the migration record’s identifiers and hashes of the payload, envelope, and copies.

The threat model assumes HNDL attacks on data in transit and at rest and forgery of quantum-vulnerable public-key signatures. Ciphertext collection and confidentiality periods determine HNDL exposure; signature schemes and authenticity periods determine signature migration requirements. Risks at plaintext processing boundaries are examined under institutional access, information flow, and system protection controls. The operating system, cryptographic libraries, and policy repository are assumed to be trusted. Complete compromise

of administrative privileges, side-channel attacks, and actual quantum cryptanalysis are outside the evaluation scope.

Protection requirements. These specify the purpose and duration of data protection. The institution's C/S/O classification, confidentiality and authenticity requirement periods, retention period, and business impact are examined. The provided classification and its supporting evidence are used to check the security controls selected for that classification and the applicable module and usage requirements. Protection periods are used to examine communication and signature migration requirements and urgency. Migration requirements derived from classifications and periods are recorded with their policy basis. The remaining protection period of an individual object is distinguished from the protection period required for each data item in a business process that continues to generate and transmit data; the estimated migration duration is also recorded. Urgency assessment considers both current communication and storage paths through which ciphertext may be exposed and the possibility that an attacker has already obtained ciphertext.

Cryptographic dependencies. These represent dependencies among the cryptographic algorithms, keys, certificates, and verifiers used for data protection. Algorithm parameters and purposes are examined to identify cryptographic layers requiring replacement and dependency paths involving quantum-vulnerable public-key cryptography. Supported migration mechanisms, support deadlines, update paths, and the possibility of replacing trust roots used as the basis for certificate validation are also examined for products and services. The algorithms, parameters, and combinations to be applied after migration are recorded as the target profile.

Operational states. Operational states are used to determine necessary migration actions, current execution permission, and completion after execution. Key compromise, nonce usage history, and payload integrity provide the basis for payload retention or re-encryption. Peer system support, mechanisms actually used, and the validity of policy and approval serve as execution conditions; recovery results for registered copies using the new key support completion and old key retirement decisions. Operational states also include whether automatic fallback to existing cryptographic mechanisms is permitted, together with the owner, applicability period, and review date of each exception.

Each input records its source, responsible party, assessment time, and evidence validity period. Asset and copy inventories identify the survey scope and omissions. Measurements are distinguished from estimates, and product specifications from observed usage. Tag, signature, and envelope decryption checks provide runtime evidence; classification, key history, authority, and inventory completeness also require institutional evidence. Insufficient evidence leaves the relevant requirement unknown, with its reason and responsibility for confirmation recorded.

3.2 Actions

Actions by role are denoted by $A = (A_C, A_W, A_T, A_S, A_R)$ for the payload, key protection, communications, signatures, and copy/key lifecycle. Each component lists required actions and any prerequisites. A role with insufficient information is undetermined; one with no applicable target is not applicable. Confirmed actions are preserved even when other roles remain unknown. Table 1 summarizes action selection.

Table 1. Migration actions

Confirmed state	Required action
Payload retention conditions satisfied; quantum-vulnerable public-key envelope	Retain payload and DEK; rewrap the key envelope
DEK compromise, nonce reuse, or disallowed payload algorithm/parameters	Re-encrypt with a new DEK and nonce; obtain trusted plaintext before processing
Insufficient algorithm, nonce, or payload evidence	Retain confirmed actions; leave only actions with insufficient evidence undetermined
Payload authentication failure	Suspend use and recover a trusted original
Quantum-vulnerable public-key dependency in a segment requiring migration	Migrate the relevant key establishment or authentication
Classical signature requiring long-term verification	Renew signature and preservation evidence
Unmigrated registered copies depending on the old key	Migrate copies and verify recovery with the new key
End of support or inability to update or replace trust roots	Replace, isolate, or retire; preserve required data

The payload action (A_C) is determined from the algorithm and parameters, key and nonce usage states, and payload integrity. The payload and DEK may be retained if the payload uses a permitted symmetric-key algorithm and parameters and satisfies the DEK, nonce, and integrity conditions. Confirmed DEK compromise or nonce reuse under the same key invalidates the payload retention conditions and requires re-encryption with a new DEK and nonce. Re-encryption is also selected if the payload algorithm is confirmed to be disallowed by policy. If evidence about the algorithm or payload state is insufficient and no other confirmed condition determines the action, that action remains undetermined. If a defect such as DEK compromise establishes the need for re-encryption, the action and its reason are retained, and execution permission is assessed separately. Even when a necessary action is confirmed, unknown support in the system that will perform it requires further confirmation of execution conditions.

The key protection action (A_W) considers both the cryptographic mechanism of the envelope protecting the DEK and the required payload action. When the payload and DEK are retained, key envelope rewrapping is selected if the envelope depends on quantum-vulnerable public-key cryptography. When the payload is re-encrypted, the new DEK is protected in a key envelope conforming to the target profile.

The communication action (A_T) assesses key establishment and authentication separately for communication and key transfer paths. Migration requirements for each are established by examining the protection period, information classification, exposure paths, and applicable policy. Replacement targets are then identified from quantum-vulnerable public-key dependencies in segments requiring migration. Migration of a storage envelope does not eliminate the risk associated with past communication ciphertext; ciphertext that may already have been collected is therefore recorded as a residual risk.

The signature action (A_S) is determined by whether verification can be maintained throughout the authenticity requirement period. For business documents requiring long-term verification, actions specify preservation of the original document signature, certificates and revocation/time evidence required for verification, and subsequent renewal history. Readiness of the new verifier is assessed as an execution condition in Section 3.3.

The copy/key lifecycle action (A_R) is determined from the scope and migration state of registered copies that depend on the old key. Copies that have not been migrated are assigned the necessary payload and envelope migration and new-key recovery verification. Where old key retirement is required, verification of the entire registered scope using that key is recorded as a prerequisite. The need for copy migration is distinguished from permission to retire the key based on actual verification results.

Software and equipment operated by the institution are checked for support of the functions required for PQC migration. If support has ended, or required updates or trust root replacement are impossible, the feasibility of data migration is examined to select component replacement, isolation, or retirement.

Urgency. Let T_C denote the confidentiality requirement period, T_A the remaining authenticity requirement period, T_M the estimated migration duration, T_Q the time until assumed emergence of a cryptographically relevant quantum computer (CRQC), and T_E the time until end of support. For an ongoing business process, T_C is the period required for each newly generated data item; for a fixed dataset it is the remaining period for current data. The CRQC date is a scenario assumption, not a prediction.[16,7] Remaining periods use the same assessment reference date and unit. Estimates and their grounds are recorded; uncertain periods are entered as intervals.

Following Mosca's perspective, $T_C + T_M$ is compared with T_Q for ongoing generation and transmission.[16] If $T_C + T_M > T_Q$, the confidentiality requirement of data produced immediately before migration completion extends beyond the assumed emergence date. Equality means that the end of protection coincides with that date; a smaller sum leaves a margin equal to the difference. If

the comparison outcome varies within the entered intervals, the decision remains unknown.

For a fixed dataset, T_C and T_M are compared separately with T_Q . If $T_C > T_Q$, confidentiality extends beyond the assumed date; if $T_M > T_Q$, migration completes after it. Equality is recorded separately from exceeding the date. Authenticity urgency compares T_A with T_Q ; end-of-support urgency compares T_M with T_E .

These comparisons provide grounds for assessing migration urgency. Under the same protection objective and assumptions, targets with smaller time margins can be considered first. However, no weights are defined to convert different classifications, business impacts, and exposure paths into a single ranking. Institutions must determine the final order by considering their classification and retention policies together with paths currently available for ciphertext collection. If required periods or their supporting evidence are unknown, the priority is not finalized; responsibility for confirmation and items requiring reassessment are recorded.

3.3 Execution

After the necessary migration actions have been determined, a decision is made on whether to execute them. Current policy and approval, peer system support, and the state of the target data are checked before execution. Each request specifies the action and target object/version; if several actions are requested together, the prerequisites for each are checked. Required actions and their grounds remain in the record even when execution is deferred or denied.

Execution is *denied* if a prohibition or a defect violating a requested action's prerequisite is confirmed. It is *deferred* if no denial ground exists but evidence is insufficient to establish required prerequisites. It is *allowed* only when no denial ground exists and all applicable prerequisites are satisfied. Denial takes precedence over deferral, with both reasons retained. Permission to begin does not establish completion. Expired evidence leaving the current state unknown requires deferral; confirmed expiration or revocation of execution authority requires denial.

Before rewrapping, the existing envelope must permit DEK recovery and payload authentication. Copy accessibility and support and approval for the new mechanism are also checked. Authentication failure, DEK compromise, or nonce reuse denies a rewrapping request based on retention. A confirmed re-encryption requirement remains recorded, but damaged input must first be replaced by a trusted original. Communication migration requires peer support for the target profile and trust framework; signature renewal requires original signature evidence and a ready new verifier.

Automatic fallback to a classical mechanism prohibited by policy is denied. If approved temporary coexistence is permitted, the execution decision also records whether an exception applies. The exception's scope, owner, reason, validity period, and reassessment date are checked, along with compensating controls

applied to reduce the associated risk. Expired exceptions require reassessment; cryptographic defects cannot be waived through an exception.

Completion. After execution, completion is determined by checking verification results for each requested action and the policy and approval conditions currently in effect. Identifiers and hashes must establish that payload, envelope, signatures, and copies refer to the same object/version and stored result. All applicable authentication, decryption, and signature checks must pass, and each registered migrated copy must be recoverable with the new recipient key.

The old key is the recipient key decrypting pre-migration envelopes, distinct from the payload DEK. Its retirement requires four conditions: (1) confirmation of the complete registered dependency scope; (2) new-key recovery, payload authentication, and plaintext equality for every registered item; (3) stopped writes using the old key; and (4) current policy, approval, and trust permitting retirement. These conditions are checked again immediately before removal. Completion for one object does not authorize retirement while another registered dependency remains unverified. Changes to keys, copies, or policy require repeated verification. Removal of old envelopes is recorded separately and does not establish erasure of external copies or past attacker-held ciphertext.

Recovery. Changes in inputs or policy trigger reassessment of affected tasks. Following interruption, execution records are reconciled with actual file, key, and copy states. Intermediate artifacts may be discarded or regenerated. Once the old key has been removed, recovery must use the verified new key and copies. Insufficient recovery evidence defers use; confirmed loss of authority or prohibition denies it. Completion history is retained, while successful responses and later reads are assessed against authority at their respective times.

3.4 N2SF Assessment

Cryptographic key management (EK), cryptographic module use (EA), data transmission (DT), and data use (DU) provide the basis for migration targets and execution conditions.[18,19] Secure key destruction (EK-5) and protection of data at rest (DU-2) are considered jointly to require recoverability before old key retirement. Algorithm standardization or product support alone does not establish compliance with validated module use (EA-1); module and usage approval must be checked separately.

Transmission authority and suspension of information exchange (DT-1, DT-2) inform permission to transmit under current authority. Storage integrity (DU-M4) informs acceptance of stored results based on identification and verification. IF and CD identify controls at plaintext boundaries; IN identifies survey scope and change history. Table 2 maps the five N2SF stages to proposed decision outputs. This allocation of outputs and roles is a proposal for institutional application.

Table 2. N2SF assessment evidence

Stage	Inputs to examine	Proposed outputs
Prepare	Service, asset, flow, and copy inventories	Target scope, dependencies, and omissions
Categorize	C/S/O, protection and retention periods	Protection requirements linked to evidence
Identify	Vulnerable public-key, plaintext, and copy paths	Exposure, migration targets, residual risks
Select	Actions and execution decisions	Target profile, plan, execution/recovery conditions, exceptions
Assess	Verification results by object/version	Completion, deferral, denial, recovery evidence, reassessment items

Operators confirm inventories, dependencies, and support; business owners provide classification and retention grounds; information security personnel examine exposure; approvers and reviewers assess authority and evidence sufficiency. Records bind input and policy versions, selected actions and reasons, execution decisions, and verification targets, methods, and results. Requirement status (satisfied, not satisfied, unknown, not applicable), execution decisions (allow, defer, deny), and check outcomes (pass, fail, inconclusive, not executed) remain distinct. Reviewers can trace why actions were selected and whether execution and completion conditions were checked.

3.5 Implementation

The reference implementation checks the supported subset of the criteria and performs migration or defers or denies execution. Table 4 relates the proposed criteria to the evaluated paths and their scope. Policy and state management are implemented in Python; TLS communication, certificate analysis, and streaming cryptographic processing are implemented in C and OpenSSL. Inputs comprise the provided asset inventory, protection requirements, cryptographic dependencies, operational states, and policy. The implementation does not automatically discover all institutional assets or copies, or guarantee the accuracy of the provided materials.

An execution plan specifies tasks and prerequisite order for an object/version, separately from the record of confirmed actions. Even if the protection period is unknown, confirmed payload retention or rewrapping remains recorded, while plan issuance is deferred until all transmission, signature, and storage requirements are determined. Denial at the input stage is retained in the final decision. An allowed plan proceeds through transmission, storage, signatures, copy verification, and key retirement. Execution conditions are rechecked before each task and before finalizing results.

Storage and transmission. For storage migration, the DEK is recovered from the existing envelope and used to authenticate the payload before rewrapping. The experimental format separates the AES-256-GCM payload from a Cryptographic Message Syntax (CMS) envelope containing the DEK. When retention conditions are satisfied, the payload, DEK, and nonce are retained and the RSA-OAEP envelope is replaced with an ML-KEM-768 envelope. Re-encryption uses a new DEK and nonce. Both methods authenticate the payload before and after migration and compare plaintext hash and length against values registered by the test caller. The DEK exists in plaintext in process memory and standard input; non-exportable key processing in a physical hardware security module (HSM) is outside the verification scope.

The two TLS 1.3 segments connecting the sender, boundary proxy, and receiver check the server hostname, certificate chain, and revocation status. The integrated profile uses X25519MLKEM768 key establishment and ML-DSA-65 server certificates, with hostname, chain, and revocation checks in each segment. The proxy processes plaintext. A separate content inspection path treats explicit rejection and tampering as denial grounds and timeouts or missing responses as deferral grounds. Institutional business authorization and actual CDS and content disarm and reconstruction (CDR) effectiveness were not validated.

Signatures. After individual checks, the implementation verifies that all results pertain to the same object and document version. A document version distinguishes revisions to data content, whereas a storage generation distinguishes the stored state of the payload and envelope protecting that content. Even when the same plaintext is rewrapped or re-encrypted, committing a new stored result increments the storage generation. The migration record contains the object, document version, storage generation, hashes of the payload, envelope, and copy manifest, and versions of the applied policy and trust materials. Actual artifact identifiers and hashes are compared with this record, and both the classical ECDSA-P256 signature and post-quantum ML-DSA-65 signature over the record are verified. These checks alone cannot establish the plaintext provider’s business authority or the validity of the original business document signature.

To preserve verification materials for the migration record, a renewal manifest contains hashes of the signature record, certificates, and revocation materials, together with the hash of the previous time evidence. The integrated path obtains and verifies new time evidence for this manifest. The test time-stamping authority (TSA) signs using RSA/SHA256, and the integrated test includes initial issuance and one renewal. This preservation renewal occurs before completion; post-completion preservation verification over the completion manifest uses a separate path. The test does not establish long-term operational effectiveness or the maintenance of business document authenticity.

Completion and recovery. Before old key retirement, each registered copy is opened with the new recipient key to confirm payload authentication and plaintext equality. The integrated path uses test keys per request; the copy manager checks all registered manifests depending on the same key. New writes using

that key are blocked within the managed scope, and current policy, approval, and trust are rechecked immediately before removal. The removed item is a local RSA private key file; physical media erasure is outside the verification scope.

After the stored result is committed, copies, signatures, preservation materials, key retirement status, and current authority are still checked. A verified request/version is registered as the object’s latest completion record. A successful response requires the completed request/version and current response authority. Subsequent reads verify the corresponding artifacts and recheck authority at return time. An intermediate stored result alone does not permit completion or release.

The task journal records stage entry, commit, and interruption events. Following interruption, the journal is reconciled with actual payload, envelope, signature, copy, and key states so that verified artifacts can be reused and incomplete artifacts regenerated. Recovery after removal of the old key uses the new key and registered copies. Conditions are rechecked at resumption, completion, response, and read time. Policy changes and authority/certificate revocation are checked against locally reflected states; distributed failures and remote propagation are outside the evaluation scope. Manifests and receipts bind identifiers, hashes, outcomes, and policy/trust versions to the assessment evidence in Section 3.4.

4 Evaluation

4.1 Setup

The evaluation covered the conditions supported by the reference implementation among the decision criteria proposed in Section 3. First, it examined whether inputs led to distinct required actions and execution decisions to allow, defer, or deny. It then examined completion of permitted actions, reassessment after policy changes, and recovery after interruptions. Processing time was used as supplementary evidence of the execution cost of permitted actions. A test passed when observations matched predefined behavior. Expected deferrals and denials counted as passes, whereas inconclusive and unexecuted tests did not. In the comparison of recheck locations, the occurrence or absence of policy violations was itself part of the expected outcome; the proportion of tests matching expectations must therefore not be interpreted as a migration success rate.

Unit tests of the decision code ran mainly on Windows, while cryptographic execution ran on WSL2 Linux on the same PC. The integrated environment used an AMD Ryzen 7 7800X3D, Linux 6.18.33.2-microsoft-standard-WSL2, Python 3.12.3, and OpenSSL 3.6.4 built for linux-x86_64 with GCC and principal options `-O3 -m64 -pthread`. WSL reported 28,739,184 KiB MemTotal; this was not a physical RAM measurement. Communication used loopback, synthetic files, software keys, and test certification authorities (CAs) and a TSA. Decision regressions used Windows Python 3.12.14; F22’s cryptographic path checks used a separate WSL2 environment with Python 3.14.4 and OpenSSL 3.6.4. Ta-

ble 3 distinguishes the test sets; their source versions and execution dates differ. Optional reproduction details are retained in Appendix A.

Table 3. Evaluation materials

Set	Materials and scale	Purpose
S08	2026-09-08; 270 storage trials	Costs of permitted migration actions
X09	2026-09-09; 16 cryptographic and 20 TLS cases	Acceptance of normal and tampered inputs across implementations
I10	2026-09-10; 108 batches and nine recovery trials	Integrated completion, copy verification, and resumption
P15	2026-09-15; 225 policy trials	Effects of recheck location on completion, response, and reads
D21	2026-09-21; decision regression and reaggregation	Actions/reasons and retained measurement aggregates
F22	2026-09-22; decision follow-up, nine cryptographic runs, eight boundary tests	Partial decisions, current execution paths, and recovery boundaries

The core decision engine in the integrated tests (I10) and policy change tests (P15) used an experimental threshold of at least five years, together with information classification, exposure, and explicit policy, to select transmission, authentication, and signature actions. This threshold is not an official N2SF criterion. The protection lifetime assessment based on period intervals and evidence in Section 3 was evaluated through a separate input path in follow-up validation (F22); I10 and P15 did not integrate and execute that assessment in its entirety.

Table 4 connects the proposed criteria to these test sets. Integrated execution, separate input checks, and validation still required for institutional application are distinguished for each criterion.

4.2 Decision Results

Synthetic inputs from F22 were used for the period comparisons determining migration urgency. Periods were entered as integer numbers of days. For business processes with ongoing generation and transmission, the tests covered cases in which the sum of the confidentiality requirement period and migration duration was greater than, equal to, or less than the assumed time until CRQC emergence. For fixed datasets, the remaining protection period and migration duration were compared separately with the assumed CRQC period. Results for both types matched the predefined decisions. These results concern comparisons of the entered periods and do not establish the validity of institutional period estimates or final migration rankings. Additional input details are provided in Appendix A.

Table 4. Coverage of the proposed criteria

Criterion	Evaluated path	Scope and remaining validation
Protection periods and urgency	F22: separate period comparisons. I10/P15: experimental five-year rule.	Interval assessment not fully integrated into I10/P15; institutional period estimates unvalidated.
Actions and execution permission	D21/F22: actions, reasons, unknowns, prohibitions, and transfer to execution plans.	Support and coexistence assessed through supplied inputs and exception rules.
Payload, envelope, communications, and copies	I10: integrated migration and copy recovery. S08: storage costs. X09: selected cross-implementation cases.	Loopback communication, file objects, software keys, and registered copies.
Signatures and preservation	I10: migration-record signatures and pre-completion renewal. Post-completion manifest verification uses a separate path.	Original business-document authenticity and long-term preservation continuity unvalidated.
Current authority and recovery	P15: recheck locations. I10: process resumption. F22: key-removal boundaries.	Locally reflected state and specified process interruptions/errors; power loss and distributed failures unvalidated.

The separation of required actions from execution decisions was examined using normal, unknown, and defective inputs in D21 and F22. When the technical state was confirmed and only approval evidence was insufficient, payload retention and rewrapping were recorded while execution was deferred. When nonce history or the payload algorithm was unknown and no other defect was confirmed, the payload action remained undetermined. When DEK compromise, nonce reuse, and payload authentication failure were supplied together, the need for re-encryption and all reasons were retained while use of the damaged input was denied. Explicit prohibitions also resulted in execution denial; format and data type errors were returned as separate input errors.

Confirmed actions must remain distinct from execution permission when decision results are passed to an execution plan. F22 examined $3^3 = 27$ input combinations assigning true, false, or unknown to requirements for key establishment migration, post-quantum authentication, and signature renewal. When a mandatory requirement was unknown, the confirmed payload retention action remained in the record and plan issuance was deferred. Requests denied at the input stage remained denied in the final decision. The 27 combinations are internal checks within unit tests and are not added to the number of independent tests. Because confirmed actions and reasons preventing execution are retained separately, subsequent review can distinguish required migration from missing

approval or verification. F22 recorded 128 unit tests: 127 passed on Windows, and the remaining OpenSSL signature test passed on WSL2.

4.3 Completion and Recovery

Completion. I10 examined whether permitted actions completed after transmission, storage, signature, and copy verification. The tests used 36 conditions combining rewrapping, Python buffered re-encryption, and C streaming re-encryption with per-object sizes of 64 KiB, 1 MiB, and 8 MiB; new or reused connections; and one or two worker threads. KiB and MiB denote 2^{10} and 2^{20} bytes, respectively. Worker threads manage object processing and the execution of required subprocesses. A batch processing four objects together was run three times per condition; all 432 objects in 108 batches completed.

Migration results were checked separately for each action. For rewrapping, the checks confirmed retention of the payload, DEK, and nonce and equality of the recovered plaintext. For re-encryption, they confirmed use of a new DEK and nonce and plaintext equality. In the separate cross-validation X09, results for normal and erroneous cryptographic and TLS inputs under supported profiles matched predefined expectations. X09 checked eight ML-KEM-768 and eight ML-DSA-65 cases in both directions between OpenSSL 3.6.4 and wolfCrypt/wolfSSL 5.9.1. Twenty TLS cases comprised four normal connections and 16 expected failures involving hostname, trust, group, or certificate errors. Storage error conditions identified only in test code were not counted as observed X09 passes.

Policy changes. Following normal completion tests, P15 examined changes to policy or certificate trust/revocation status after execution had been permitted. Initial-decision reuse reuses the request’s initial policy and trust/revocation assessment in subsequent processing and reads. Completion/read rechecking refreshes that assessment at completion finalization, receipt issuance, return, and read; per-stage rechecking also refreshes it before intermediate tasks. Independently of these differences, all configurations retained preliminary and final authorization for old key removal, together with certificate and recovery checks. Payload, envelope, signature, and copy verification; actual TLS peer verification; direct authority checks before return to the sender; final version comparison; and receipt binding were also common to all configurations.

All three configurations used streaming re-encryption, connection reuse, one object, and one worker thread. Normal conditions were repeated six times per configuration at each object size of 64 KiB, 1 MiB, and 8 MiB. Nineteen change, tampering, and interruption conditions were repeated three times per configuration at 1 MiB. This yielded 75 trials per configuration and 225 trials in total. Policy violations were determined using the permission conditions at each completion, response, and read after the change had been reflected locally. Revocation after key removal was not applied retroactively to a previously valid removal.

In Table 5, initial-decision reuse produced three unauthorized completions and three unauthorized successful responses after changes. Its 36 violating reads represent reads of the live, backup, and restored copies of 12 objects. No such violations were observed with completion/read rechecking or per-stage rechecking. All configurations had zero violating key removals, and all 54 normal trials across the three configurations completed. Under the tested conditions, these differences demonstrate the need to recheck authority at completion and use, even after key retirement has been authorized.

Table 5. Policy violations

Configuration	Trials	Violating completions	Violating successful responses	Objects with violating reads	Violating reads
Initial-decision reuse	75	3	3	12	36
Completion/read rechecking	75	0	0	0	0
Per-stage rechecking	75	0	0	0	0

The two rechecking configurations differed in when processing stopped. In five conditions involving intermediate changes to policy, revocation materials, or trust state, completion/read rechecking entered five stages after the change and committed results from four before stopping at the common key retirement check. Per-stage rechecking stopped before entering the next stage. The same result was observed in all three repetitions of each condition; stage counts refer to a single trial. Differences in stage counts were not converted into time or computational savings. Additional policy results are given in Appendix B.

Recovery. When a process terminated during processing, the tests examined whether execution could resume from the remaining stored state and records. Separately from the normal performance tests, I10 recovery tests applied streaming re-encryption to a single 1 MiB object. Processes were forcibly terminated three times at each of three points: after committing storage migration but before recording it in the task journal; after old key removal but before recording it in the journal; and after the completion record. In all nine trials, exactly one completion event was recorded after resumption, and the plaintext recovered from the live, backup, and restored copies matched.

Errors and authority changes around key removal were examined separately in eight F22 unit tests. These contained 29 detailed observations covering termination immediately after key deletion and immediately after directory synchronization to persist deletion state; deletion, synchronization, and recording errors; approval changes; and concurrent requests. Requests with valid current

authority recovered using the new key, whereas denied requests remained incomplete with reads denied. Unit test counts and internal observation counts are not added to I10 interruption/resumption repetitions. Recovery results from both test sets are limited to the specified process interruptions and errors; power loss and distributed failures were not validated.

4.4 Processing Costs

Storage tests S08 compared the costs of three migration methods applied to identical initial data satisfying the payload retention conditions. They used nine conditions combining a single object of 1, 16, or 64 MiB with idle, read, or mixed read/write load, plus a mixed-load condition with eight 1 MiB objects. Read load used four threads with four reads each; mixed load used two read threads with four reads each and one write thread with two writes. Requests were finite and used a warm OS cache. Each method was measured three times in each of three sessions per condition, giving nine measurements per method and condition and 270 trials overall. Migration time was measured from invocation of migration for the target objects until migration of all target objects finished. It included recovery of the existing envelope, payload authentication, migration, verification of new results, commit, and retries on conflict, but excluded final verification after the test.

Table 6 shows that rewrapping had the shortest median migration time at all three sizes under idle load. For the 64 MiB object, medians across nine measurements per method were 121.010 ms for rewrapping, 1,191.906 ms for buffered re-encryption, and 234.685 ms for streaming re-encryption. Rewrapping also reads the entire payload for pre- and post-migration authentication, so its cost increased with object size. However, the method with the shortest migration time did not have the lowest latency for every read and write request. Because the implementations also differed in processing language, pipes, and file access methods, the observed differences cannot be attributed solely to cryptographic algorithms or buffer sizes. This cost comparison supports the choice of execution method among actions permitted by the payload, key, and nonce conditions. Additional load results and descriptive distributions are given in Appendix B.

Table 6. Migration time under idle load (ms)

Object size	Rewrapping	Buffered re-encryption	Streaming re-encryption
1 MiB	21.293	39.539	26.632
16 MiB	48.897	333.087	89.467
64 MiB	121.010	1,191.906	234.685

Medians of nine trials per method and condition (three sessions with three repetitions each).

Integrated processing time in I10 was measured per batch as described above. The interval from transmission invocation to return of the results for four objects included planning, subprocess execution, TLS transmission, test key generation, storage, signatures, time evidence, copy processing, internal preservation renewal, key retirement, and completion. It excluded service initialization, subsequent reads, and post-completion preservation verification. Median processing times calculated from three batches per condition ranged from 15.897 to 36.027 seconds. This is a range of condition-specific medians, not per-object latency or the minimum and maximum of all measurements. Because objects within a batch shared the host, worker threads, and connections, the 432 objects were not treated as independent performance samples. The reported range describes integrated execution rather than the cost of an individual cryptographic operation.

Times for each policy configuration in P15 were measured from invocation of transmission for a single object to its return. These times included test TSA processing and host execution variability, so the cost of policy rechecking alone cannot be isolated. Additional timing observations are retained in Appendix B.

4.5 Limitations

In institutional application, actions, reasons, input evidence, execution decisions, and verification results for the same object/version can be examined together as materials for N2SF adequacy assessment and adjustment. Unknown items are assigned a responsible party and required evidence; confirmed defects are assigned recovery and migration tasks. The validity of this application approach and the accuracy of personnel decisions require further validation in actual institutions.

Institutional application requires confirmation of input accuracy and survey scope. Incorrectly reported key compromise states and omitted copies cannot all be detected by internal tag and hash checks. Information classifications, protection periods, exposure, and approval require institutional evidence and review by responsible personnel. Protection period comparisons also depend on the supplied CRQC assumptions and migration duration estimates; the method does not include a model that automatically determines a final ranking across different business processes.

Execution results were obtained from repeated trials in a small number of sessions on one host and cannot be generalized as samples from independent institutions. Applying file-object and registered-copy results to full-disk encryption (FDE), transparent data encryption (TDE), or database snapshots and transaction logs requires redefining key dependencies and recovery units.

Related literature was selected by cryptographic dependencies, migration procedures, policy enforcement, and protection layers. The review was not a systematic literature review with a fixed search scope. Neither performance comparisons with external implementations in an identical environment nor formal security comparisons were conducted.

Local key and envelope removal does not guarantee erasure of external copies or physical media, and risks from previously collected ciphertext remain. Coex-

istence and end-of-support responses were evaluated only for input rules and exception state handling. Production public key infrastructure (PKI), key management systems (KMS), HSM, CDS, cross-product negotiation and coexistence termination, equipment at end of support, and distributed failure recovery remain external validation targets. Long-term preservation continuity and legal effect, acceptance in institutional audits, and improvements to migration schedules and staffing also require follow-up validation.

5 Conclusion

This paper proposed decision criteria for determining PQC migration actions and execution conditions based on N2SF protection requirements and the cryptographic dependencies and operational states of data. Required actions for payloads, key protection, communications, signatures, and copies are distinguished, and execution deferral, denial, and recovery are determined according to unknown evidence and confirmed defects. Completion and old key retirement require verification results for the same object/version, recovery of registered copies using the new key, and current authority.

The reference implementation and evaluation provided evidence of action selection, execution control, and resumption after interruption under supported conditions. Recording decision inputs, selected actions, reasons, and verification results by object/version enables the grounds for migration requirements and execution results to be examined together during N2SF adequacy assessment and adjustment. Application in actual institutions requires validation of input accuracy, product and peer institution support, and evidence sufficiency.

References

1. Adams, C., Cain, P., Pinkas, D., Zuccherato, R.: Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP). RFC 3161, RFC Editor (8 2001). <https://doi.org/10.17487/RFC3161>, <https://www.rfc-editor.org/rfc/rfc3161.html>
2. Alagic, G., Barker, E., Chen, L., Moody, D., Robinson, A., Silberg, H., Waller, N.: Recommendations for Key-Encapsulation Mechanisms. SP 800-227, National Institute of Standards and Technology (9 2025). <https://doi.org/10.6028/NIST.SP.800-227>, <https://csrc.nist.gov/pubs/sp/800/227/final>
3. Barker, E.: Recommendation for Key Management: Part 1 - General. SP 800-57 Part 1 Rev. 5, National Institute of Standards and Technology (2020). <https://doi.org/10.6028/NIST.SP.800-57pt1r5>, <https://csrc.nist.gov/pubs/sp/800/57/pt1/r5/final>
4. Bernstein, D.J., Lange, T., Levin, J., Yang, B.Y.: PQConnect: Automated Post-Quantum End-to-End Tunnels. In: Network and Distributed System Security Symposium (2025). <https://doi.org/10.14722/ndss.2025.241879>, <https://www.ndss-symposium.org/wp-content/uploads/2025-1879-paper.pdf>
5. Bindel, N., Herath, U., McKague, M., Stebila, D.: Transitioning to a Quantum-Resistant Public Key Infrastructure. In: Lange, T., Takagi, T. (eds.) Post-Quantum Cryptography. Lecture Notes in Computer Science, vol. 10346, pp. 384–405. Springer (6 2017). https://doi.org/10.1007/978-3-319-59879-6_22, <https://www.douglas.stebila.ca/files/research/papers/PQCrypto-BHMS17.pdf>
6. Cho, J., Lee, C., Kim, E., Lee, J., Cho, B.: Software-Defined Cryptography: A Design Feature of Cryptographic Agility (2024), <https://arxiv.org/html/2404.01808v2>, arXiv:2404.01808v2. Preprint
7. CISA, NSA, NIST: Quantum-Readiness: Migration to Post-Quantum Cryptography. Tech. rep., CISA, NSA and NIST (8 2023), https://www.cisa.gov/sites/default/files/2023-08/Quantum%20Readiness_Final_CLEAR_508c%20%283%29.pdf, joint factsheet
8. Dworkin, M.: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. SP 800-38D, National Institute of Standards and Technology (11 2007). <https://doi.org/10.6028/NIST.SP.800-38D>, <https://csrc.nist.gov/pubs/sp/800/38/d/final>
9. ETSI: CYBER; Migration Strategies and Recommendations to Quantum Safe Schemes. ETSI TR 103 619, European Telecommunications Standards Institute (7 2020), https://www.etsi.org/deliver/etsi_tr/103600_103699/103619/01.01.01_60/tr_103619v010101p.pdf, v1.1.1
10. European Commission: Digital Signature Service Documentation, <https://ec.europa.eu/digital-building-blocks/DSS/webapp-demo/doc/dss-documentation.html>, evidence Records. Experiments used DSS 6.5; online documentation checked 21 September 2026
11. Gondrom, T., Brandner, R., Pordes, U.: Evidence Record Syntax (ERS). RFC 4998, RFC Editor (2007), <https://www.rfc-editor.org/rfc/rfc4998.html>
12. Hasan, K.F., Simpson, L., Baee, M.A.R., Islam, C., Rahman, Z., Armstrong, W., Gauravaram, P., McKague, M.: A Framework for Migrating to Post-Quantum Cryptography: Security Dependency Analysis and Case Studies. IEEE Access **12**, 23427–23450 (2024). <https://doi.org/10.1109/ACCESS.2024.3360412>, <https://arxiv.org/abs/2307.06520>
13. HashiCorp: Vault Transit Secrets Engine API: Rewrap Data, <https://developer.hashicorp.com/vault/api-docs/secret/transit#rewrap-data>, online documentation checked 21 September 2026

14. Housley, R.: Algorithm Identifiers for the HMAC-based Extract-and-Expand Key Derivation Function (HKDF). RFC 8619, RFC Editor (6 2019). <https://doi.org/10.17487/RFC8619>, <https://www.rfc-editor.org/rfc/rfc8619.html>
15. Kwiatkowski, K., Kampanakis, P., Westerbaan, B.E., Stebila, D.: Post-Quantum Traditional (PQ/T) Hybrid Key Agreement Mechanisms for TLS 1.3. RFC 10024, RFC Editor (2026), <https://www.rfc-editor.org/rfc/rfc10024.html>
16. Mosca, M.: Cybersecurity in an Era with Quantum Computers: Will We Be Ready? *IEEE Security & Privacy* **16**(5), 38–41 (2018). <https://doi.org/10.1109/MSP.2018.3761723>
17. Näther, C., Herzinger, D., Gazdag, S.L., Steghöfer, J.P., Daum, S., Loebenberger, D.: Migrating Software Systems Toward Post-Quantum Cryptography—A Systematic Literature Review. *IEEE Access* **12**, 132107–132126 (2024). <https://doi.org/10.1109/ACCESS.2024.3450306>, <https://publica.fraunhofer.de/entities/publication/c91c072a-c1d1-44f3-8f42-6cb00e83d7db>
18. National Cyber Security Center: National Network Security Framework (N2SF) Security Guidelines 1.0 (9 2025), in Korean; title translated
19. National Cyber Security Center: National Network Security Framework Security Guidelines: Explanatory Guide to Security Controls (9 2025), version 1.0, Appendix 1 (in Korean; title translated)
20. Newhouse, W., Souppaya, M., Barker, W., Brown, C., et al.: Migration to Post-Quantum Cryptography: Quantum Readiness: Cryptographic Discovery. SP 1800-38B, National Institute of Standards and Technology (12 2023), <https://www.nccoe.nist.gov/sites/default/files/2023-12/pqc-migration-nist-sp-1800-38b-preliminary-draft.pdf>, preliminary Draft. Volume B: Approach, Architecture, and Security Characteristics of Public Key Application Discovery Tools
21. Newhouse, W., Souppaya, M., Barker, W., Brown, C., et al.: Migration to Post-Quantum Cryptography: Quantum Readiness: Testing Draft Standards. SP 1800-38C, National Institute of Standards and Technology (12 2023), <https://www.nccoe.nist.gov/sites/default/files/2023-12/pqc-migration-nist-sp-1800-38c-preliminary-draft.pdf>, preliminary Draft. Volume C: Quantum-Resistant Cryptography Technology Interoperability and Performance Report
22. NIST: Post-Quantum Cryptography: FAQs, <https://csrc.nist.gov/projects/post-quantum-cryptography/faqs>, accessed 18 September 2026; AES and quantum attacks
23. NIST: Module-Lattice-Based Digital Signature Standard. FIPS 204, National Institute of Standards and Technology (2024). <https://doi.org/10.6028/NIST.FIPS.204>, <https://csrc.nist.gov/pubs/fips/204/final>
24. NIST: Module-Lattice-Based Key-Encapsulation Mechanism Standard. FIPS 203, National Institute of Standards and Technology (2024). <https://doi.org/10.6028/NIST.FIPS.203>, <https://csrc.nist.gov/pubs/fips/203/final>
25. NIST: Stateless Hash-Based Digital Signature Standard. FIPS 205, National Institute of Standards and Technology (2024). <https://doi.org/10.6028/NIST.FIPS.205>, <https://csrc.nist.gov/pubs/fips/205/final>
26. Relevant Ministries, Jointly: Comprehensive National Plan for Migration to Post-Quantum Cryptography (9 2025), in Korean; title translated
27. Rescorla, E.: The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, RFC Editor (2018), <https://www.rfc-editor.org/rfc/rfc8446.html>
28. Rescorla, E.: The Transport Layer Security (TLS) Protocol Version 1.3. RFC 9846, RFC Editor (7 2026). <https://doi.org/10.17487/RFC9846>, <https://www.rfc-editor.org/info/rfc9846/>, standard status checked separately from the experimental OpenSSL profile

Appendices

A Reproducibility

This appendix records optional reproduction details. The decision definitions, execution conditions, evaluation setup, principal results, and limitations are given in the main text.

A.1 Inputs and Profiles

The protection period tests in F22 used synthetic inputs. For ongoing generation/transmission, a 1,000-day confidentiality period and 100-day migration duration were compared with assumed CRQC periods of 1,099, 1,100, and 1,101 days. They respectively exceeded, equaled, or preceded the assumed date. For a fixed dataset, remaining protection of 1,000 days and migration of 100 days were compared separately with 1,050 days; both were within that period. These checks belong to 21 protection lifetime unit tests, distinct from the 27 internal plan-transfer combinations in Section 4.

The six readiness items G1–G6 respectively cover international algorithm standardization, standardization of protocol combinations, domestic cryptographic module validation and policy requirements, institutional deployment approval, product/version support, and confirmed interoperability with actual peer systems. They are research items, not official N2SF control numbers. The $4^6 = 4,096$ combinations of confirmed, not applicable, unknown, and prohibited, and 10 predefined cases, matched the expected deny/defer/allow rules. These combinations are internal checks and are not additional independent trials.

The AES-256-GCM payload uses a 32-byte DEK, 12-byte nonce, and 16-byte tag; it has no additional authenticated data. Object binding is checked using the signed external metadata. CMS DER contains the DEK, with RSA-OAEP or ML-KEM-768 recipient protection and internal AES-256-GCM encryption. The retained ML-KEM envelope derives its wrapping key using HKDF-SHA256 and the identifier specified by RFC 8619.[14] Parallel ECDSA-P256 and ML-DSA-65 signatures cover identical metadata bytes and must both verify; this is an acceptance rule, not a new composition proof.

The integrated TLS profile uses server authentication and is distinct from separate mutual-authentication tests. Protocol version references are recorded separately from the experimental OpenSSL profile.[27,28,15] Test RFC3161 time evidence uses RSA/SHA256 and checks the imprint, nonce, policy, and signer.[1] It is neither post-quantum time-stamping nor evidence from an accredited TSA. Separate post-completion ERS verification uses DSS 6.5 over completion-manifest bytes; detached ML-DSA verification is separate. These paths do not establish long-term operational continuity or legal validity.

A.2 Retained Evidence

The paths below identify locations relative to the archived research materials; they are not public download addresses or persistent identifiers. Performance

remeasurement requires the retained runner, binaries, and a new test directory. Test versions and sample units must remain separate.

- S08: `lab/results/d3-storage/d3-20260908T075841Z/`. Compare `raw/`, `summary.json`, and `source/profile/binary` hashes in `run.json`.
- X09: `report/interop_v7/main-20260909/`. Normal and tampered ML-KEM, ML-DSA, and TLS inputs have separate expected outcomes.
- I10: `report/inheritance_20260910/run-c/`. `source_manifest.json` identifies the source used in `performance/batches/`. Measurement collection was interrupted after 21 completed batches; these results were retained and collection resumed for the remaining 87. These 108 performance batches are separate from the nine deliberate process-termination/recovery trials.
- P15: `report/retirement_comparison_20260915/main-a/`. Post-change stage counts use the parent directory’s `summary-corrected.json`, aggregated by monotonic time because event numbers reset on restart.
- D21: `report/manuscript_revision_20260921/`. Source hashes, commands/logs, and reaggregated measurements are retained; reaggregation is not new performance measurement.
- F22: `report/manuscript_followup_20260922/`. Decision, planning, cryptographic, and recovery records distinguish successful runs from failed environment setup.

The storage archive contained Python caches generated after archival, so the strict file-inventory comparison failed. No listed file was missing or had a mismatched hash, and event invariants and recomputed summaries for all 270 trials agreed. This was not relabeled as verification of the entire archive. Reaggregation uses `report/manuscript_revision_20260921/audit.py` and does not remeasure cryptographic performance.

B Additional Results

B.1 Storage

Table 7 supplements the idle results in Table 6. Read load used four threads with four reads each; mixed load used two read threads with four reads each and one write thread with two writes. Requests were finite with a warm OS cache; sustained-load throughput was not measured.

For the 64 MiB idle condition, first-to-third-quartile ranges were 113.157–127.270 ms for rewapping, 1,025.140–1,483.183 ms for buffered re-encryption, and 228.113–245.700 ms for streaming re-encryption. Quartiles use linear interpolation over nine sorted observations. Within-session repetitions were not independent institutional samples; population confidence intervals were not estimated.

For 18 write requests under the 16 MiB mixed condition, median request latency was 98.545, 85.622, and 68.466 ms for rewapping, buffered re-encryption, and streaming re-encryption. Migration conflicts numbered 0, 9, and 18, whereas write conflicts numbered 9, 0, and 0. Request latency and retries are separate from migration time; aggregates do not establish causes of individual delays.

Table 7. Migration time under concurrent load (ms)

Size and load	Rewrapping	Buffered	Streaming
		re-encryption	re-encryption
1 MiB; read	40.001	75.827	48.243
16 MiB; read	65.957	310.122	110.953
64 MiB; read	154.678	1,214.825	324.822
1 MiB; mixed	58.904	102.718	95.051
16 MiB; mixed	59.255	630.006	246.660
64 MiB; mixed	150.042	2,488.868	783.767
1 MiB×8; mixed	194.787	377.999	273.097

Medians of nine trials per method and condition. The final row measures migration of all eight objects.

B.2 Policy Changes

Table 8 reports finalized completions/successful responses/reads with matching plaintext. n denotes trials per configuration; each trial reads three copies. Reuse, completion, and stage denote the three configurations in Section 4.3. CRL means certificate revocation list, OCSP means Online Certificate Status Protocol, and TTL means evidence time to live.

Under initial-decision reuse, the 3/3/9 outcome for revocation before completion, nine reads in each 3/0/9 outcome for revocation before return, and nine reads in the 3/3/9 outcome for revocation after return are violations after the change. Earlier completions and responses remain valid historical events. Confirmed revocation caused denial; missing current evidence caused deferral. Tampering was blocked by artifact checks, and interruption cases completed exactly once after resumption.

Table 8: Policy outcomes by condition

Condition	Event location	n	Reuse	Completion	Stage
Normal 64 KiB	No change	6	6/6/18	6/6/18	6/6/18
Normal 1 MiB	No change	6	6/6/18	6/6/18	6/6/18
Normal 8 MiB	No change	6	6/6/18	6/6/18	6/6/18
Policy revocation	After storage migration	3	0/0/0	0/0/0	0/0/0
CRL revocation	After storage migration	3	0/0/0	0/0/0	0/0/0
OCSP revocation	After storage migration	3	0/0/0	0/0/0	0/0/0
Trust material expiry	After storage migration	3	0/0/0	0/0/0	0/0/0
Distribution failure/TTL expiry	After storage migration	3	0/0/0	0/0/0	0/0/0
Policy revocation	After completion check, before finalization	3	3/3/9	0/0/0	0/0/0
Policy revocation	After receipt, immediately before return	3	3/0/9	3/0/0	3/0/0
Trust revocation	After receipt, immediately before return	3	3/0/9	3/0/0	3/0/0

Condition	Event location	n Reuse	Completion	Stage
Policy revocation	After successful return, before read	3 3/3/9	3/3/0	3/3/0
Payload tampering	After storage migration	3 0/0/0	0/0/0	0/0/0
Envelope tampering	After storage migration	3 0/0/0	0/0/0	0/0/0
Signature tampering	After signing stage	3 0/0/0	0/0/0	0/0/0
Backup tampering	After backup stage	3 0/0/0	0/0/0	0/0/0
Interruption after storage commit	Before task journal entry	3 3/3/9	3/3/9	3/3/9
Interruption after key removal	Before task journal entry	3 3/3/9	3/3/9	3/3/9
Interruption after completion	After completion record	3 3/3/9	3/3/9	3/3/9
Policy revocation	After preliminary removal check, before final authorization	3 0/0/0	0/0/0	0/0/0
CRL revocation	After preliminary removal check, before final authorization	3 0/0/0	0/0/0	0/0/0
Verification TTL expiry	After preliminary removal check, before final authorization	3 0/0/0	0/0/0	0/0/0

Normal P15 median times for completion/read versus per-stage rechecking were 5.591 versus 5.626 s at 64 KiB, 5.447 versus 5.712 s at 1 MiB, and 10.037 versus 10.046 s at 8 MiB, with six pairs per size. The corresponding median paired differences were 0.295, 0.392, and -0.053 s. A median paired difference need not equal the difference between configuration medians. Test TSA processing and host variability prevent interpreting these values as isolated policy-check costs.