International Journal of
Technology and
Systems
(IJTS)

# Post-Quantum Readiness for Federated Identity and API Platforms: Migrating TLS, JWT Signing, KMS, and Encrypted Persona Identifiers without Breaking Oauth/OIDC Flows



Journal of Technology and Systems

ISSN: 2788-6344 (Online)

Vol. 8, Issue No. 4, pp 61 – 74, 2026

www.carijournals.org

# Post-Quantum Readiness for Federated Identity and API Platforms: Migrating TLS, JWT Signing, KMS, and Encrypted Persona Identifiers without Breaking Oauth/OIDC Flows

**Sailesh Kotha, Rajesh Babu Kodali**

***Accepted: 19th Oct, 2026, Received in Revised Form: 9th Nov, 2026, Published: 19th Nov, 2026***

## Abstract

**Purpose:** This paper develops a layered framework that lets federated identity and API platforms built on OAuth 2.0 and OpenID Connect adopt post-quantum cryptography without breaking existing OAuth/OIDC flows.

**Methodology:** The paper synthesizes standards, vendor announcements and research available through June 30, 2026, including NIST's August 2024 standards (ML-KEM, ML-DSA and SLH-DSA) and hybrid specifications from the IETF, the OpenID Foundation and major cloud providers. It presents a reference architecture spanning TLS transport, JWT signing, key custody and encrypted persona identifiers, a per-layer migration lever taxonomy, three illustrative use cases and a qualitative comparison of staged and single cutover migration.

**Findings:** Data protected by classical key exchange is already exposed to harvest now, decrypt later attacks. OAuth/OIDC flows can break when the layers are not coordinated, for example when relying parties cannot parse unfamiliar JSON Web Key Set entries, when middleboxes reject the larger hybrid TLS ClientHello, or when stored identifiers are re-encrypted without a rotation plan. Because several required specifications were still Internet-Drafts in June 2026, a staged, hybrid, crypto-agile approach is preferable to a single cutover.

**Unique Contribution to Theory, Policy and Practice:** The paper treats the four layers as coordinated migration surfaces rather than independent projects; gives platform teams a lever taxonomy, a catalogue of breakage points and a persona identifier migration pattern; and supports policies that prioritize confidentiality-sensitive layers and hybrid, dual-stack designs while standards mature.

**Keywords:** *Post-Quantum Cryptography, OAuth 2.0, OpenID Connect, JSON Web Token, TLS, Key Management, Hybrid Key Exchange, ML-KEM, ML-DSA, Federated Identity, Crypto-Agility*

**JEL Codes:** *L86, O33, M15*

## 1. Introduction

Federated identity is distributed trust backed by public key cryptography. An OpenID Provider signs ID tokens with a private key that relying parties verify against keys published at a JSON Web Key Set (JWKS) endpoint; clients and authorization servers negotiate TLS sessions to protect authorization codes, access tokens and refresh tokens; a key management service (KMS) or hardware security module (HSM) holds the private keys behind both guarantees; and many consumer platforms also encrypt persona or account identifiers so that an identifier seen in a log, URL or downstream system does not reveal the underlying account. All four layers rely on RSA or elliptic curve cryptography, and all four would be compromised by a fault tolerant quantum computer running Shor's algorithm, which efficiently solves the factoring and discrete logarithm problems on which these schemes rest [5].

No such computer exists, and estimates of when one might vary widely. Migration has nevertheless begun, for two reasons. First, under the harvest now, decrypt later threat model, traffic recorded today can be decrypted once a cryptographically relevant quantum computer exists, so the effective deadline for confidential data arrives well before the computer itself [5]. Second, migrating federated identity is a multi-party coordination problem: an OpenID Provider cannot change its signing algorithm without breaking relying parties that have not updated, and large platforms depend on partners and long lived mobile clients they do not control. Government guidance points the same way; the U.S. National Security Agency's CNSA 2.0 suite sets post-quantum transition timelines for national security systems [6].

This paper makes two contributions specific to federated identity and API platforms. First, it presents a layered framework that treats TLS transport, JWT signing, KMS/HSM key custody and encrypted persona identifiers as four coordinated migration surfaces, because a change at one layer, such as moving a signing key to a post-quantum algorithm, affects others, such as which relying parties can still verify tokens. Second, it identifies the specific points at which OAuth 2.0 and OpenID Connect deployments can break during migration, drawing on compatibility problems already surfaced by transport layer rollouts at major browsers and content delivery networks.

Sections 2 and 3 review the background and state the problem; Sections 4 to 6 present the framework, its levers and three use cases; Sections 7 to 9 evaluate, discuss and conclude.

## 2. Background and Related Work

### 2.1 The quantum threat model and harvest now, decrypt later

Quantum computing does not threaten all cryptography equally. Symmetric ciphers and hash functions, including the SHA-256 hash used in the OAuth PKCE code challenge, face only a quadratic speedup from Grover's algorithm, which longer keys or outputs offset. RSA and elliptic curve schemes such as ECDSA and ECDH, which underpin TLS and JOSE, are fully broken by Shor's algorithm, and no larger key size provides a stopgap. Post-quantum migration therefore

concentrates on the key exchange and signature mechanisms that TLS, JWT signing, X.509 certificates and key management services depend on.

Mosca's framing holds that migration should begin once the time data must remain secure plus the time migration will take exceeds the time remaining before a cryptographically relevant quantum computer appears [5]. For identity platforms it separates two priorities. Confidentiality mechanisms, principally TLS key exchange and encryption of persona identifiers or tokens at rest, are exposed to harvest now, decrypt later risk today. Integrity mechanisms, principally JWT and certificate signatures, are at risk only once signatures can be forged, since a signature captured today cannot be retroactively forged for an event that has already occurred. Hence the framework's first principle: encryption migrates first, while signature migration can be sequenced with more attention to compatibility.

### 2.2 NIST post-quantum cryptography standards

NIST finalized its first three post-quantum standards on August 13, 2024: FIPS 203 (ML-KEM), a module lattice key encapsulation mechanism that replaces classical key exchange such as elliptic curve Diffie-Hellman; FIPS 204 (ML-DSA), a module lattice signature scheme that replaces RSA and ECDSA signatures; and FIPS 205 (SLH-DSA), a stateless hash based signature scheme that offers a conservative hedge because its security rests on different assumptions [1], [2].

Two further algorithms were not yet final as of June 2026. FIPS 206, specifying the Falcon derived FN-DSA signature, reached Initial Public Draft status in 2025, with a final standard projected for late 2026 or early 2027 [4]; its compact keys and signatures make it attractive for certificate chains and HSMs. In March 2025 NIST also selected HQC, a code based key encapsulation mechanism, as a structurally independent backup to ML-KEM, with a final standard not expected before 2027 [3], [21]. Table 1 summarizes these algorithm families.

**Table 1. NIST post-quantum algorithm families referenced in this paper.**

| Algorithm (standard) | Category | Mathematical basis | Standardization status as of June 2026 | Primary role |
|---|---|---|---|---|
| ML-KEM (FIPS 203) | Key encapsulation | Structured (module) lattices | Final, published Aug. 2024 | Replaces classical key exchange (TLS, hybrid encryption) |
| ML-DSA (FIPS 204) | Digital signature | Structured (module) lattices | Final, published Aug. 2024 | General-purpose replacement for RSA/ECDSA signatures |
| SLH-DSA (FIPS 205) | Digital signature | Hash-based | Final, published Aug. 2024 | Conservative signature hedge, different assumptions than lattices |
| FN-DSA / Falcon (FIPS 206) | Digital signature | Structured (NTRU) lattices | Initial Public Draft (2025); final projected late 2026 or early 2027 | Compact signatures for certificates and constrained hardware |
| HQC (future FIPS) | Key encapsulation | Code-based | Selected Mar. 2025 as backup KEM; draft and final still pending | Structurally independent backup to ML-KEM |

### 2.3 Post-quantum TLS

Transport migration has progressed further in deployment than in standardization because of hybrid key exchange, which combines a classical algorithm such as X25519 with ML-KEM and derives the session key from both, so the connection remains secure if either component holds. Google enabled the hybrid X25519Kyber768 group by default in desktop Chrome 124 in May 2024, reporting about a four percent increase in median handshake latency and middlebox compatibility problems serious enough to warrant a temporary enterprise opt-out policy [9]. Cloudflare had already enabled hybrid key exchange on connections to origin servers in September 2023 [10].

Formally, both the general hybrid design for TLS 1.3 (draft-ietf-tls-hybrid-design) and the ML-KEM groups such as X25519MLKEM768 (draft-ietf-tls-mlkem) remained Internet-Drafts as of June 2026 [7], [8]. Waiting for RFC publication would leave identity platforms years behind the ecosystem's largest operators.

### 2.4 Post-quantum JOSE, COSE, and JWT signing

JSON Web Signature identifies its algorithm through the alg header, using identifiers registered with IANA. The specification for ML-DSA in JOSE and COSE was published as RFC 9964 in May 2026, making it one of the few finalized building blocks for the token layer [15]. The FN-DSA counterpart, and the specification for PQ/traditional hybrid composite signatures that combine a post-quantum and a classical algorithm in a single JWS or COSE signature, both remained Internet-Drafts [16]. This difference in maturity informs the recommendation in Section 5.2 to prefer dual, separately verifiable signatures over composite signatures during the transition.

### 2.5 Post-quantum X.509 certificates and key management

X.509 certificates, which anchor TLS server authentication and often mutual TLS client authentication, face the same problem: a certificate authority cannot switch to a post-quantum signature without breaking relying parties that do not recognize it. The IETF LAMPS working group's composite signature and composite key encapsulation specifications carry classical and post-quantum components under a single algorithm identifier, but both remained Internet-Drafts as of June 2026 [17], [18].

Cloud key management support arrived unevenly. AWS added ML-DSA signing to AWS KMS in June 2025 and enabled hybrid ML-KEM TLS by default in several AWS Secrets Manager clients in April 2026 [11], [12]; Google Cloud offered ML-KEM and the X-Wing hybrid in Cloud KMS preview from October 2025 [13]. NIST's National Cybersecurity Center of Excellence has also published a preliminary draft migration practice guide, SP 1800-38 [19].

### 2.6 Post-quantum guidance for federated identity

The OpenID Foundation's May 2024 statement on post-quantum readiness identified the same exposure points used here, namely token signatures, TLS and certificate infrastructure, and HSM key generation, and asked its working groups to monitor emerging standards while maintaining practices such as key rotation [14]. As of June 2026, no OpenID Connect or FAPI specification had incorporated post-quantum signatures.

Adoption also lags awareness. A May 2025 survey of 1,042 senior and C-level cybersecurity managers in the United States, United Kingdom and Australia, conducted by Propeller Insights for DigiCert, found that only 5 percent of organizations had implemented quantum-safe encryption, although 69 percent recognized the quantum risk to current encryption and 69 percent expected encryption breaking quantum computers within five years [20]. Section 7.4 returns to this gap.

## 3. Problem Statement and Motivation

Three risks motivate the framework, each concerning whether a platform can adopt post-quantum cryptography without breaking production identity flows.

The first is relying party incompatibility at the token layer. Relying parties discover verification keys through the JWKS, and their JOSE libraries must recognize each key type and algorithm. An unfamiliar ML-DSA key may simply be ignored, which is the safe outcome, or a poorly implemented client may reject the entire JWKS response and stop verifying tokens it previously accepted. Because platforms do not control the update cadence of partners or long lived mobile apps, token layer rollout must follow the slowest realistic JWKS consumer rather than the platform's own release cycle.

The second is transport compatibility. In Chrome's rollout, TLS-inspecting appliances that assumed classical ClientHello sizes rejected or mishandled the larger hybrid ClientHello, since ML-KEM keys and ciphertexts are much larger than elliptic curve points [9]. Platforms that run their own API gateways, load balancers or mutual TLS infrastructure face the same risk, and enabling hybrid key exchange without first auditing these components invites an outage rather than graceful degradation.

The third, specific to platforms that encrypt persona identifiers, is that re-encrypting stored identifiers is not a drop-in change, because caches, logs and downstream systems may hold the ciphertext form that a naive re-encryption invalidates all at once. This risk is largely absent from the protocol-focused TLS and JWT migration literature and is addressed in Sections 5.4 and 6.3.

## 4. Proposed Framework: Crypto-Agile Federated Identity Migration

### 4.1 Design principles

Four principles guide the framework. First, confidentiality before integrity: because harvest now, decrypt later risk accrues today while forgery risk does not, TLS hybrid key exchange and persona identifier migration take priority over JWT signature migration. Second, hybrid and dual-stack by default: every layer runs classical and post-quantum algorithms together, through hybrid key exchange, dual signatures or composite certificates where finalized, because several underlying specifications are still drafts and a premature cutover may need to be redone. Third, relying party pace rather than platform pace: new algorithms are advertised through OpenID Connect discovery metadata without removing existing ones, and no classical algorithm is deprecated until relying party capability is confirmed. Fourth, layer independence with coordinated sequencing: layers can migrate on separate timelines, but in a defined order so that no layer silently depends on another layer's assumptions.

### 4.2 Reference architecture

Figure 1 shows the reference architecture: four layers coordinated by a shared crypto-agility control plane. The transport layer covers TLS between clients, the authorization server, relying parties and internal gateways. The token layer covers JWS signing of ID and access tokens, keyed to JWKS entries and advertised in discovery metadata. The key custody layer, a cloud KMS or HSM, holds the private keys and is where new key types such as ML-DSA are provisioned. The

identifier encryption layer migrates through re-encryption and dual reads rather than protocol negotiation. The control plane tracks which algorithms are enabled, deprecated or retired at each layer and blocks disabling a classical algorithm until that layer's consumers are ready.

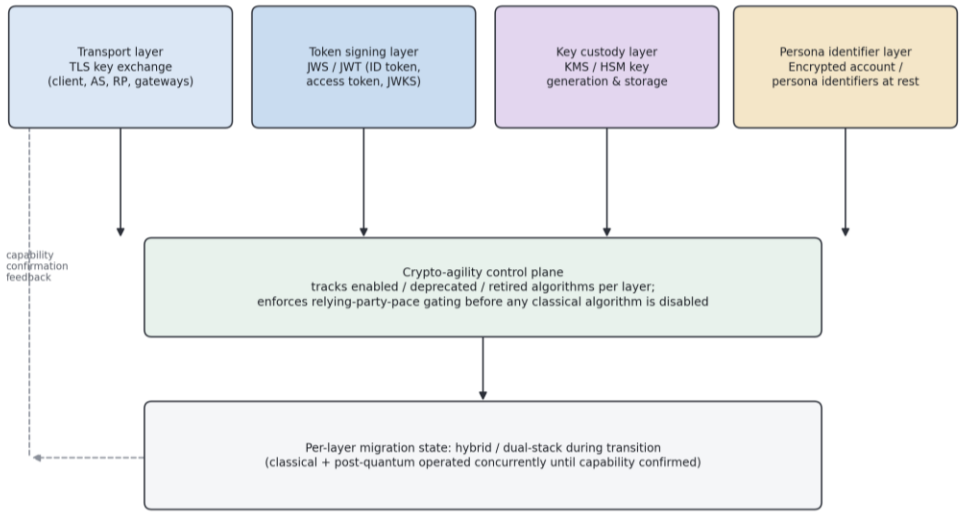



*Figure 1. Layered crypto-agile migration architecture for federated identity and API platforms, coordinated through a shared crypto-agility control plane.*

### 4.3 Standards maturity as a migration input

Table 2 consolidates the maturity of the protocol, certificate and guidance specifications the framework depends on (the algorithm standards themselves appear in Table 1), distinguishing finalized standards from Internet-Drafts that may still change and that the dual-stack principle is designed to absorb. Figure 2 places the main standardization and deployment milestones on a timeline.

Journal of Technology and Systems
ISSN: 2788-6344 (Online)
Vol. 8, Issue No. 4, pp 61 – 74, 2026
www.carijournals.org

**Table 2. Standardization status of specifications underlying this framework, as of June 2026.**

| Layer | Specification | Status as of June 2026 |
|---|---|---|
| Transport | Hybrid key exchange design for TLS 1.3 (draft-ietf-tls-hybrid-design) | Internet-Draft, not yet an RFC |
| Transport | ML-KEM key agreement for TLS 1.3 (draft-ietf-tls-mlkem) | Internet-Draft, in IESG evaluation |
| Token signing | ML-DSA for JOSE and COSE | Final (RFC 9964, published May 2026) |
| Token signing | FN-DSA for JOSE and COSE | Internet-Draft |
| Token signing | PQ/T hybrid composite signatures for JOSE and COSE | Internet-Draft |
| Certificates | Composite ML-DSA / ML-KEM for X.509 PKI | Internet-Draft |
| Migration guidance | NIST SP 1800-38 (Migration to Post-Quantum Cryptography) | Preliminary draft, open for public comment |

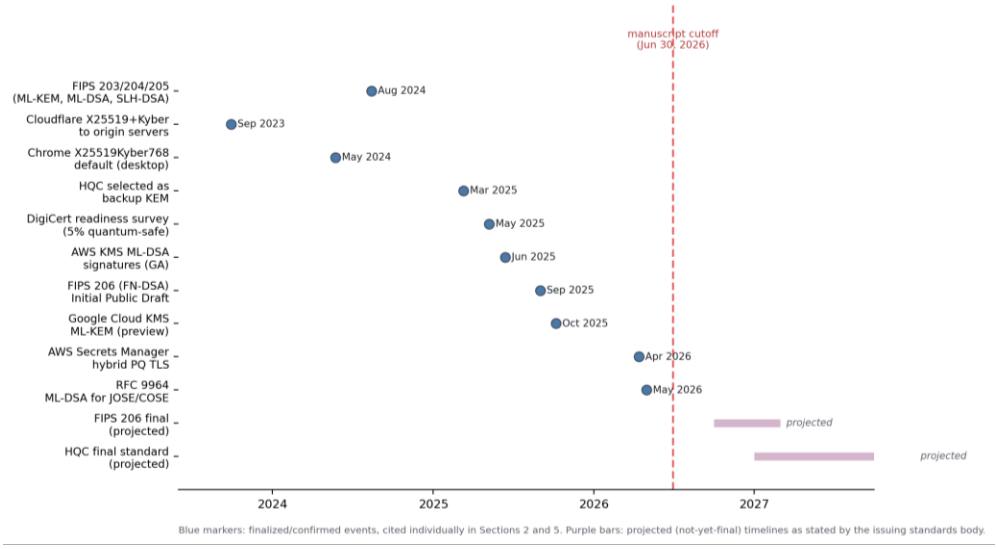



*Figure 2. Post-quantum standardization and deployment timeline referenced in this paper, spanning early hybrid TLS deployments through NIST FIPS finalization and projected completion of remaining standards.*

## 5. Migration Levers by Layer

### 5.1 Transport layer

The transport lever is hybrid key exchange, combining X25519 with ML-KEM, on every TLS connection the platform terminates or originates, including client-facing, service-to-service and mutual TLS. Given the middlebox failures seen in Chrome's rollout, the framework makes an audit of every component that parses or limits handshake message sizes a prerequisite rather than a risk to be discovered in production [9]. Because the governing specifications are still drafts, implementations should track the algorithm identifiers deployed by major browsers and content delivery networks, which serve as de facto reference implementations [7], [8].

### 5.2 Token signing layer

The token lever is dual signing: during the transition, tokens carry both a classical signature, such as ECDSA with P-256, and an ML-DSA signature using the RFC 9964 identifiers, so legacy relying parties verify the classical signature while updated ones verify the post-quantum one [15]. Dual signatures are preferred over composite signatures because the composite format is unfinished and because dual signatures degrade more gracefully: a relying party can ignore a signature it does not recognize, whereas it may be unable to extract the classical component of a composite. The lever requires that discovery metadata, such as id_token_signing_alg_values_supported, advertise the new algorithm before relying parties are expected to use it, and that the key custody layer can generate and hold ML-DSA keys.

### 5.3 Key management and hardware security module layer

The key custody lever is provisioning post-quantum key types alongside existing classical keys, which dual signing and hybrid key exchange continue to require. Because support arrived on different timelines, for example ML-DSA in AWS KMS in June 2025 and ML-KEM in Google Cloud KMS preview in October 2025, multi-cloud and on-premises HSM estates must be inventoried component by component [11], [13]. For certificate signing, FN-DSA's smaller signatures merit close tracking, since ML-DSA's larger signatures increase handshake overhead and certificate chain size [4].

### 5.4 Encrypted persona identifier layer

Persona identifier migration requires a data strategy, because stored, cached or externally referenced ciphertexts cannot simply be reinterpreted under a new algorithm. The framework recommends a dual-read, forward-write pattern: from a defined cutover point, new identifiers are encrypted under a hybrid scheme combining classical public key encryption with an ML-KEM derived key, while the identifier resolution service continues to decrypt classical identifiers until they are reissued or re-encrypted by a background process. The pattern accepts a longer migration window in exchange for avoiding a single high-risk cutover across a live identifier space. Table 3 summarizes the levers for all four layers.

**Table 3. Summary of migration levers, primary risk addressed, and key dependency by layer.**

| Layer | Primary lever | Primary risk addressed | Key dependency |
|---|---|---|---|
| Transport (TLS) | Hybrid key exchange (X25519 + ML-KEM) | Harvest now, decrypt later on network traffic | Middlebox and load balancer ClientHello size support |
| Token signing (JWT/JOSE) | Dual classical and ML-DSA signatures | Relying party verification breakage | Discovery metadata updated ahead of relying party rollout |
| Key custody (KMS/HSM) | Provision PQC key types alongside classical keys | Signing/encryption capability gap | Provider-specific post-quantum key type availability |
| Persona identifiers (application data) | Dual-read, forward-write re-encryption | Breakage of downstream references to existing ciphertexts | Identifier resolution service supporting both schemes concurrently |

## 6. Use Cases

### 6.1 Migrating a consumer identity provider's JWKS to dual signing

A consumer identity platform whose OpenID Provider serves first-party mobile apps and external partner relying parties must introduce ML-DSA without breaking integrations it does not control. Following Section 5.2, it advertises ML-DSA alongside ECDSA in discovery metadata, begins issuing dual-signed ID tokens, and monitors introspection logs and partner reports to confirm that unrecognized algorithms are ignored rather than causing failures, before considering any classical deprecation. Because RFC 9964 was finalized only in May 2026, the platform must first confirm that its JOSE libraries implement the final specification rather than an earlier draft [15].

### 6.2 Hybrid TLS rollout across an API gateway fleet

A platform with an API gateway fleet serving external token endpoint traffic and internal service calls needs hybrid key exchange without repeating Chrome's middlebox problems. Following Section 5.1, it inventories load balancers, web application firewalls and inspection appliances for ClientHello size limits, then enables hybrid key exchange first on internal traffic, where it controls both endpoints and can roll back easily, and later on external endpoints. This internal-then-external order extends the relying party pace principle to the transport layer.

### 6.3 Re-encrypting persona identifiers under a hybrid scheme

A platform that encrypts persona identifiers to avoid exposing internal account identifiers in URLs or partner APIs must address harvest now, decrypt later exposure for identifiers logged or cached by third parties. Following Section 5.4, it issues new identifiers under a hybrid scheme from a cutover date, keeps its resolution service able to decrypt older identifiers, and re-encrypts high-value, long lived identifiers, such as those embedded in active partner integrations, ahead of natural expiry.

## 7. Evaluation and Comparative Analysis

### 7.1 Staged, crypto-agile migration versus single cutover

Table 4 compares staged, crypto-agile migration with a single cutover. The comparison is qualitative, because no controlled study of the two approaches in federated identity platforms was identified; it rests on the failure modes documented in Sections 2.3 and 3, notably the middlebox problems that surfaced even in Chrome's additive, hybrid rollout [9].

**Table 4. Qualitative comparison of staged, crypto-agile migration and single cutover migration.**

| Dimension | Single cutover | Staged, crypto-agile migration (proposed) |
|---|---|---|
| Relying party compatibility risk | High; any lagging relying party breaks at cutover | Lower; classical path remains available during transition |
| Exposure to unfinished specifications | High if cutover targets a still-drafting specification | Lower; dual-stack absorbs later specification changes |
| Operational complexity during migration | Lower (single algorithm to operate) | Higher (two algorithm stacks operated concurrently) |
| Time to full classical algorithm retirement | Potentially faster, if it succeeds | Slower by design |
| Resilience to a specific incident (e.g., middlebox failure) | Low; failure affects all traffic on that layer | Higher; classical path available as fallback |

### 7.2 Security analysis

Hybrid designs reduce but do not eliminate transition risk. They hedge against a future quantum attack on the classical component and an undiscovered weakness in the newer post-quantum component, without exceeding the strength of the stronger algorithm. For persona identifiers, the dual-read period prolongs exposure for identifiers not yet migrated, which is why Section 6.3 prioritizes long lived, externally referenced identifiers.

Journal of Technology and Systems
ISSN: 2788-6344 (Online)
Vol. 8, Issue No. 4, pp 61 – 74, 2026
www.carijournals.org

### 7.3 Standards volatility as an operational risk

Because several specifications in Table 2 remain drafts, the algorithm identifiers, wire formats and APIs referenced in Sections 5.1 to 5.3 may still change. The token layer illustrates this: implementations built on drafts of the ML-DSA JOSE specification may have needed adjustment when RFC 9964 was published in May 2026. The framework's independently timed layers are designed to confine such adjustments to the affected layer.

### 7.4 Adoption gap evidence

The DigiCert survey indicates that the gap this framework targets lies between awareness and coordinated execution rather than in recognition of risk: only 5 percent of organizations had implemented quantum-safe encryption by May 2025, while 69 percent recognized the risk (Figure 3) [20]. With fewer than 40 percent describing themselves as very prepared, the complexity of running two algorithm stacks concurrently appears to be a genuine barrier, which supports the framework's emphasis on sequencing and compatibility rather than algorithm selection alone.

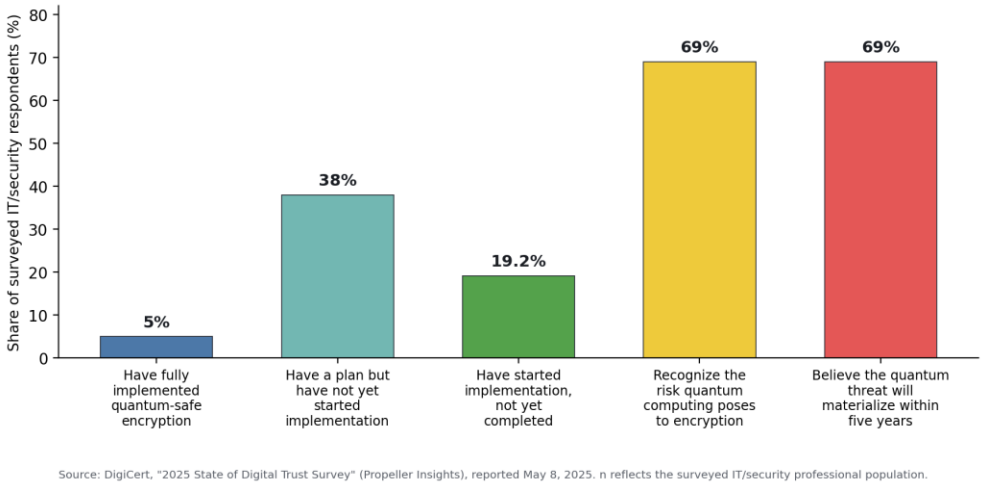



*Figure 3. Enterprise post-quantum cryptography preparedness reported by the DigiCert/Propeller Insights survey of 1,042 senior and C-level cybersecurity managers, published May 8, 2025.*

## 8. Discussion

### 8.1 Adoption path

A pragmatic sequence, consistent with Section 4.1, starts with the transport layer, then persona identifiers where they exist, followed by token signing and key custody once middleboxes have been validated and cloud providers offer post-quantum key types generally rather than in preview. Standards volatility is a reason to build crypto-agility, the ability to add, deprecate and monitor algorithms without re-architecture, rather than to wait; at a 5 percent implementation rate, delay is already the industry's dominant failure mode.

### 8.2 Limitations

The framework is an original synthesis of the standards and announcements reviewed in Section 2 and has not been validated in peer-reviewed research or measured in a production deployment. Several specifications it depends on may change before RFC publication, so readers should verify their current status. The use cases are illustrative scenarios based on common platform patterns, not case studies of a deployed migration. Finally, the paper adopts Mosca's urgency framing and offers no independent estimate of when a cryptographically relevant quantum computer will exist [5].

## 9. Conclusion and Future Work

Federated identity platforms depend on classical public key cryptography at four layers often owned by different teams: TLS transport, JWT signing, KMS/HSM key custody and encrypted persona identifiers. This paper has argued that these layers should be migrated as coordinated but independently sequenced surfaces, with confidentiality protecting layers first and hybrid, dual-stack designs throughout, because key specifications remained drafts as of June 2026. The main obstacle is not awareness but coordinated execution that does not break relying parties, partners and downstream systems.

Future work should update Section 5 as the drafts in Table 2 are finalized, replace the qualitative comparison in Section 7.1 with measured outcomes from real staged migrations, and formalize the persona identifier pattern of Section 5.4 as a reusable reference implementation.

## References

[1] National Institute of Standards and Technology, "NIST Releases First 3 Finalized Post-Quantum Encryption Standards," Aug. 13, 2024.

[2] Federal Register, "Announcing Issuance of Federal Information Processing Standards (FIPS) FIPS 203, FIPS 204, and FIPS 205," Aug. 14, 2024.

[3] National Institute of Standards and Technology, "NIST Selects HQC as Fifth Algorithm for Post-Quantum Encryption," Mar. 11, 2025.

[4] National Institute of Standards and Technology, Computer Security Division, "FIPS 206: FN-DSA (Falcon)," NIST CSRC presentation and status update, 2025.

[5] M. Mosca, "Cybersecurity in an Era with Quantum Computers: Will We Be Ready?," IEEE Security & Privacy, vol. 16, no. 5, pp. 38-41, 2018.

[6] National Security Agency, "Commercial National Security Algorithm Suite 2.0," 2022, updated 2024.

[7] D. Stebila, S. Fluhrer, and S. Gueron, "Hybrid key exchange in TLS 1.3," Internet Engineering Task Force, Internet-Draft draft-ietf-tls-hybrid-design, work in progress, 2026.

[8] Internet Engineering Task Force, TLS Working Group, "ML-KEM Post-Quantum Key Agreement for TLS 1.3," Internet-Draft draft-ietf-tls-mlkem, work in progress, 2026.

[9] Google Security Blog, "Advancing Our Amazing Bet on Asymmetric Cryptography," May 23, 2024.

[10] Cloudflare Blog, "Cloudflare now uses post-quantum cryptography to talk to your origin server," Sept. 29, 2023.

[11] Amazon Web Services, "AWS KMS adds support for post-quantum ML-DSA digital signatures," June 13, 2025.

[12] Amazon Web Services, "AWS Secrets Manager clients now support hybrid post-quantum TLS to protect secrets from quantum risks," Apr. 14, 2026.

[13] Google Cloud Blog, "Announcing quantum-safe Key Encapsulation Mechanisms in Cloud KMS," Oct. 7, 2025.

[14] OpenID Foundation, "Post-Quantum Identity Standards," May 7, 2024.

[15] Internet Engineering Task Force, COSE Working Group, "ML-DSA for JOSE and COSE," RFC 9964, May 2026.

[16] Internet Engineering Task Force, JOSE Working Group, "PQ/T Hybrid Composite Signatures for JOSE and COSE," Internet-Draft draft-ietf-jose-pq-composite-sigs, work in progress, 2026.

[17] Internet Engineering Task Force, LAMPS Working Group, "Composite ML-DSA for use in X.509 Public Key Infrastructure," Internet-Draft draft-ietf-lamps-pq-composite-sigs, work in progress, 2026.

[18] Internet Engineering Task Force, LAMPS Working Group, "Composite ML-KEM for use in X.509 Public Key Infrastructure," Internet-Draft draft-ietf-lamps-pq-composite-kem, work in progress, 2026.

[19] National Institute of Standards and Technology, National Cybersecurity Center of Excellence, "Migration to Post-Quantum Cryptography," NIST Special Publication 1800-38, preliminary draft, Vols. A-C, 2023.

[20] DigiCert (survey conducted by Propeller Insights), "Quantum Readiness Gap: DigiCert Study Finds Just 5% of Enterprises Have Quantum-Safe Encryption in Place," May 8, 2025.

[21] National Institute of Standards and Technology, "Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process," NIST Internal Report 8545, Mar. 2025.

2026 by the Authors. This Article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/)