

Engineering the Post-Quantum Transition at Scale: Protocol Integration, Library Defaults, and the In-Between Era of Internet Security (2025)

Feng Hao

ORCID: 0000-0002-8664-5074

Department of Computer Science, University of Warwick, Coventry, United Kingdom

Danda B. Rawat

ORCID: 0000-0003-3638-3464

Department of Electrical Engineering and Computer Science, Howard University, Washington, DC, USA

Correspondence: Danda B. Rawat · contact details held in the journal record

Publication history. Assigned by the publisher to Cybersecurity Advances, Volume 2, Issue 1 (2025). This publisher-authorized digital edition was released on 29 October 2026 from the supplied final 2025 manuscript. Review-stage notices were removed and journal typography, pagination, metadata, and accessibility were normalized; substantive scholarly content was preserved.

© 2025 the author(s). Published by AlphaTech Scholarly Press under the Creative Commons Attribution 4.0 International License (CC BY 4.0). No DOI is displayed unless and until registration is independently verified.

Background: The August 2024 finalization of FIPS 203, 204, and 205 established the cryptographic core of the post-quantum transition; 2025 was the year that core was *engineered into the default path of the Internet*. Within twelve months, ML-KEM-based hybrid key exchange became a default, zero-configuration behavior across the major browser engines, a mainstream language runtime (Go 1.24's crypto/mlkem standard-library package), and — decisively for server operators — OpenSSL 3.5.0 (April 2025, LTS), whose default TLS keyshares offer X25519MLKEM768. By late October 2025, more than half of human-initiated traffic observed at a major CDN was post-quantum protected. **Objective:** This paper characterizes the 2025 state of the post-quantum transition as an engineering achievement, and identifies what the achievement does *not* yet cover. **Methods:** Following a design-analytical methodology, we (i) verify the 2025 default-enablement sequence against primary release documentation (OpenSSL, Go, browser engines, platform vendors, CDN telemetry), (ii) analyze the standards pipeline as it stood at year-end (the HQC selection of March 2025; the pending FN-DSA draft; the IR 8547 deprecation schedule), and (iii) construct a residual-risk register mapping six risk classes against six mitigation classes. **Results:** The deployment achievement is real but asymmetric: confidentiality (key establishment) has moved to post-quantum hybrids at scale, while authentication remains entirely classical — public post-quantum certificates are not expected to be widely issued before 2026 and broadly trusted later. We term this the *in-between era* and show that it closes the harvest-now-decrypt-later window for transit confidentiality while leaving an open window for identity forgery and server impersonation. The residual-risk register identifies implementation maturity (the KyberSlash-class timing-defect lesson), certificate and PKI latency, and the non-upgradeable long-tail client population as the dominant open risks. **Conclusions:** The engineering problem of 2025 was enabling post-quantum key exchange by default; the engineering problem of 2026–2027 is post-quantum authentication on the same schedule — and organizations should plan PKI re-issuance, validation pipelines, and legacy-exception registers against that window now.

Keywords: post-quantum cryptography; ML-KEM; TLS 1.3; OpenSSL 3.5; hybrid key exchange; HQC; FN-DSA; PKI migration; digital trust

Contribution statement. This paper makes four contributions: (1) a primary-source-verified account of the 2025 default-enablement sequence across the TLS stack (browsers, runtimes, TLS libraries, server platforms), correcting several secondhand dates circulating in secondary reporting; (2) a standards-pipeline analysis as of year-end 2025, including the HQC selection and its 2027 horizon and the pending FN-DSA draft; (3) the characterization of the *in-between era* — post-quantum confidentiality operating on a classical authentication base — and its threat-window implications; and (4) a residual-risk register mapping six risk classes to six mitigation classes with qualitative coverage assessments.

1. Introduction

In 2024, the cryptographic community received its marching orders: NIST finalized ML-KEM (FIPS 203), ML-DSA (FIPS 204), and SLH-DSA (FIPS 205) [1][2][3], and NIST IR 8547 fixed the transition's deadline structure — classical public-key cryptography deprecated after 2030, disallowed after 2035 [4]. The question that report left open was empirical: would the Internet's software

ecosystem actually *ship* the migration? 2025 answered emphatically for one half of the problem. Over the course of the year, post-quantum key exchange stopped being an opt-in experiment and became the **default, zero-configuration behavior** of the software stack that terminates most encrypted traffic on Earth.

The sequence is worth stating precisely, because secondary reporting has garbled it. In February 2025, Go 1.24 shipped a `crypto/mlkem` package into the standard library and enabled the X25519MLKEM768 hybrid key exchange by default whenever applications leave TLS configuration unset [8]. In April 2025, the OpenSSL Project released version 3.5.0 — a Long-Term-Support release supported until 2030 — incorporating all three NIST algorithms and changing the **default** TLS supported-groups list to include and prefer hybrid post-quantum KEM groups, with default keyshares offering X25519MLKEM768 and X25519 [7]. Because nginx, HAProxy, and a long tail of server software build against OpenSSL, this single release converted a documentation exercise into an industry-wide default. Browser engines had already moved: Chrome enabled post-quantum key agreement on desktop by default in March 2024, and Chrome on Android and Firefox on desktop followed in November 2024 [10]. By the last week of October 2025, Cloudflare reported that a **majority of human-initiated traffic** crossing its network used post-quantum key exchange [10] — a milestone that took the field roughly three years from first defaults to majority coverage.

This paper's purpose is to celebrate that achievement *accurately*, and then to map what it did not achieve. The central structural fact of late 2025 is an asymmetry: **confidentiality has migrated; authentication has not**. TLS sessions are increasingly established with hybrid post-quantum key exchange, but the certificates and signatures that authenticate them remain RSA and ECDSA — and per ecosystem analysis, public post-quantum certificates are unlikely to be issued by CAs before 2026, with broad trust arriving later still [10]. We call this configuration the *in-between era*: the harvest-now-decrypt-later window on transit confidentiality is now substantially closed for the traffic majority, but the window for identity forgery and server impersonation — attacks that require breaking classical signatures *at the time of authentication*, which NIST IR 8547 explicitly notes cannot be attacked retroactively [4] — remains open until the certificate ecosystem follows. Meanwhile, the standards pipeline itself is still moving: NIST selected **HQC** as a fifth algorithm and backup to ML-KEM in March 2025, with a draft standard expected around 2026 and a final standard in 2027 [5][6]; the Falcon-derived FN-DSA signature draft (FIPS 206) was announced as forthcoming throughout 2025 [5].

1.1 Research questions

RQ1: What is the verified 2025 default-enablement sequence across the TLS stack, and what does it establish about deployment velocity? **RQ2:** What is the standards pipeline's state at year-end 2025, and what engineering commitments does it imply for 2026–2027? **RQ3:** What does the in-between era configuration imply for risk — which threat windows close, which remain open? **RQ4:** Which residual risks dominate, and which mitigation classes cover them?

1.2 Contributions

We contribute: (1) a primary-source-verified deployment timeline and layered readiness assessment (Sections 3–4, Figures 1–2, Table 1) (RQ1); (2) a year-end-2025 standards-pipeline analysis (Section 4.3, Table 2) (RQ2); (3) the in-between-era characterization with its threat-window analysis (Section 5, Figure 3) (RQ3); and (4) a residual-risk register with qualitative mitigation-coverage assessment (Section 6, Figure 4, Table 3) (RQ4). The register is an author assessment grounded in documented mechanisms, not an empirical measurement, and its limitations are stated explicitly.

2. Background and Related Work

2.1 The standards baseline and its deadlines

The cryptographic baseline was fixed in August 2024: ML-KEM for key establishment [1], ML-DSA as the primary signature standard [2], SLH-DSA as the conservative hash-based backup [3], with NIST IR 8413 documenting the third-round selection rationale [23]. The transition schedule — deprecation after 2030, disallowance after 2035 for classical public-key algorithms — arrived with the initial public draft of NIST IR 8547 in November 2024 [4]. Mosca's inequality remains the risk-side framing: organizations whose secrets must outlive the early 2030s needed the migration to be *underway* by now [14], and resource-estimate analyses of cryptanalytically relevant quantum computers continue to support horizon estimates on the order of a decade [15]. 2025 added two pipeline facts. First, NIST selected **HQC** — a code-based KEM built on different mathematics from ML-KEM's module lattices — as the designated backup for general encryption, documented in the final round-4 status report (NIST IR 8545), with a draft standard planned for roughly 2026 and finalization in 2027 [5][6]. Second, the FN-DSA (Falcon-derived) signature standard was announced as imminent but did not arrive in 2025, leaving the post-quantum *signature* pipeline — the piece the in-between era most needs — deliberately incomplete while implementation experience accumulates.

2.2 Deployment research

Engineering studies of post-quantum TLS predate deployment: Paquin et al. benchmarked post-quantum schemes in TLS before standardization [16], Sikeridis et al. measured post-quantum authentication costs [17], and Schwabe, Stebila, and Wiggers' KEMTLS redesign showed how TLS authentication could be restructured around KEMs to avoid signature bloat [18]. The deployment-observability literature is thinner and mostly industry-published: Cloudflare's annual "state of the post-quantum Internet" reports provide the only longitudinal traffic-level measurements, with the March 2024 edition reporting approximately 2% of terminated TLS 1.3 connections using post-quantum key agreement [9] and the October 2025 edition reporting majority coverage [10]. This paper extends that observational record with a structured engineering analysis: what shipped, in what order, with what defaults, and with what residual risks — a perspective the industry reports gesture at but do not systematize.

2.3 The implementation-defect lesson

The transition's first years also delivered a cautionary record on implementation maturity: the KyberSlash disclosures of December 2023 revealed secret-dependent division timing defects across multiple independent Kyber/ML-KEM implementations [13] — none exploitable in the TLS handshake configuration as deployed, but a clear signal that new algorithm implementations arrive with novel defect classes. Implementation risk is therefore treated here as a first-class residual risk rather than a footnote (Section 6).

3. Methodology

The study applies a four-component design-analytical methodology.

Deployment verification (RQ1). The 2025 default-enablement sequence was reconstructed exclusively from primary release documentation: the OpenSSL 3.5.0 release notes [7], the Go 1.24 release notes and standard-library documentation [8], browser release and platform documentation as consolidated in Cloudflare's longitudinal timeline [10], and the Kubernetes project's July 2025 deployment analysis [11]. Every date in Table 1 traces to a release document or the vendor's own timeline; secondhand dates circulating in secondary reporting were discarded where they conflicted with primary sources. This discipline proved necessary: at least one widely repeated secondary claim (that ML-KEM hybrid support first appeared in a 2025 browser release) misdates the Firefox/Chrome defaults by months.

Pipeline analysis (RQ2). The standards pipeline was assessed as of year-end 2025 from NIST's own announcements and reports: the finalized FIPS 203/204/205 [1][2][3], the HQC selection announcement and its underlying round-4 report NIST IR 8545 [5][6], the pending FN-DSA draft status [5], and the IR 8547 deadline structure [4].

In-between-era analysis (RQ3). The configuration analysis (post-quantum key exchange over classical authentication) is derived from the deployment record plus ecosystem statements on certificate availability [10], and its threat-window implications follow directly from the retroactivity asymmetry documented in IR 8547 [4].

Residual-risk register (RQ4). Six risk classes were derived from the deployment record, the implementation-defect literature, and the pipeline analysis; six mitigation classes were derived from documented ecosystem mechanisms. Coverage levels (strong / partial / none) are qualitative author assessments, each tied to a specific mechanism argument (Table 2), with the subjectivity stated in Section 8.

4. The 2025 Deployment Record (RQ1)

4.1 The default-enablement sequence

Figure 1 presents the verified sequence. Its structure is a two-track story. The deployment track (green) is about who turns the feature on: Cloudflare enabled post-quantum key agreement by default server-side in October 2022 — two years before any client default — and then watched client defaults arrive in waves: Chrome's experiment ramp to 10% of desktop users (November 2023), Chrome desktop default (March 2024, when roughly 2% of Cloudflare-terminated TLS 1.3 connections were post-quantum [9]), Chrome on Android and Firefox on desktop (November 2024), and Apple's rollout with iOS/iPadOS/macOS 26 (October 2025). The library track (blue) is about who makes the feature the path of least resistance for everyone else: Go's standard library carried a post-quantum hybrid by default from August 2024 and formalized it as `crypto/mlkem` with the standard X25519MLKEM768 group in Go 1.24 (February 2025)

[8]; and OpenSSL 3.5.0 — the April 2025 LTS release supported until 2030 — made the decisive move, changing the default TLS supported-groups list to prefer hybrid post-quantum KEM groups and offering X25519MLKEM768 and X25519 as default keyshares [7].

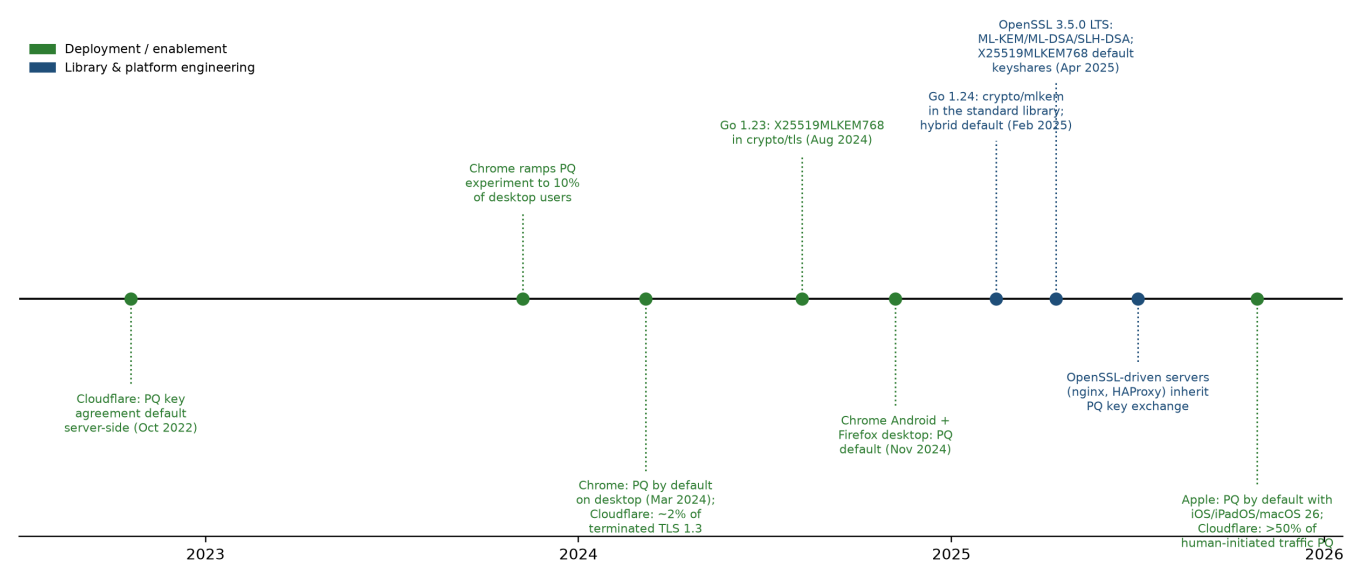


Figure 1. The post-quantum default-enablement sequence, 2022-2025: deployment milestones (green) and library/platform engineering (blue), verified against primary release documentation [7][8][9][10].

Table 1 consolidates the matrix. The engineering significance of the OpenSSL event deserves emphasis: OpenSSL is not merely one more library — it is the TLS implementation underneath a large fraction of the world's non-browser server fleet, and its 3.5 line is an LTS release whose support window (until 2030) spans the IR 8547 deprecation horizon [4][7]. An administrator running a current OpenSSL-linked stack who has *done nothing* is now negotiating hybrid post-quantum key exchange by default. Defaults, not documentation, are how cryptographic transitions actually happen — the same lesson the browser-default wave of 2024 taught on the client side [9][10]. The default-on approach has reputable precedent: OpenSSH 9.0 shipped a hybrid post-quantum key exchange as its default in 2022 [22].

Platform / component	Version	Date	Post-quantum behavior	Source
Chrome (desktop)	124	Mar 2024	Hybrid PQ key exchange by default (X25519Kyber768, later X25519MLKEM768)	[9][10]
Chrome (Android) / Firefox (desktop)	131 / 133	Nov 2024	Hybrid PQ key exchange by default	[10]
Go runtime	1.23 -> 1.24	Aug 2024 -> Feb 2025	PQ key exchange default since 1.23; 1.24 switches to standardized X25519MLKEM768 via the crypto/mlkem standard-library package	[8][10]
OpenSSL	3.5.0 (LTS, supported to 2030)	Apr 2025	Default TLS supported groups prefer hybrid PQ KEMs; default keyshares X25519MLKEM768 + X25519; ML-KEM, ML-DSA, SLH-DSA supported	[7]
Firefox	135	Feb 2025	ML-KEM hybrid support consolidated	[11]
Apple platforms	iOS / iPadOS / macOS 26	Oct 2025	PQ key exchange rollout by default	[10]
Cloudflare (edge)	n/a	Oct 2022 -> late 2025	PQ key agreement default server-side; >50% of human-initiated traffic PQ-protected by late Oct 2025	[9][10]

Table 1. The 2025 default-enablement matrix for post-quantum key exchange, verified against primary sources [7][8][9][10][11].

4.2 Layered readiness

Figure 2 gives the author-assessed readiness picture across the stack as of late 2025. The top three layers are strong: browser engines (Chrome, Firefox) and the Go runtime ship post-quantum key exchange by default, and OpenSSL 3.5 puts it in the default path of the dominant server library. The gradient sharpens

downward. Server platforms inherit OpenSSL defaults only when they rebuild against 3.5+ — a packaging and ops problem, not a code problem, but a real one. Enterprise middleware is the weak middle: TLS-inspecting proxies, load balancers, and security appliances were the compatibility long tail in the browser-default wave — their ClientHello-size assumptions and session-handling regressions are what Chrome's multi-year low-rate ramp was spent discovering [9][10] — and they remain the layer where post-quantum key exchange most often silently falls back. The bottom layer, firmware-constrained and long-lived embedded clients, is on multi-year refresh cycles that will outlast the 2030 deprecation date regardless of what desktop software does.

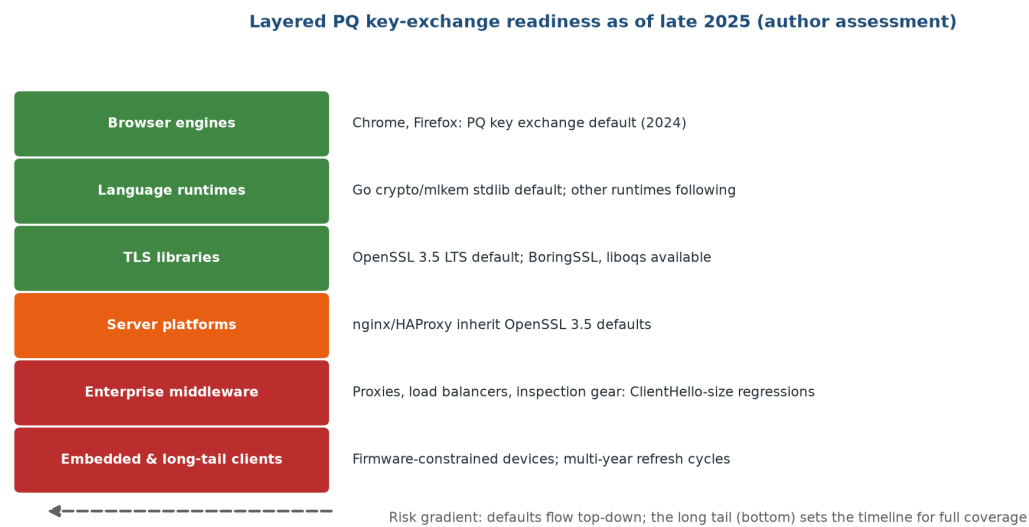


Figure 2. Layered post-quantum key-exchange readiness as of late 2025 (author assessment), with the top-down risk gradient.

5. The Standards Pipeline at Year-End 2025 (RQ2)

Table 2 summarizes the pipeline. Three facts define the engineering horizon.

First, the KEM portfolio is diversifying on schedule. HQC — code-based, mathematically unrelated to ML-KEM's module lattices — was selected in March 2025 as the designated backup for general encryption, with NIST's round-4 report (IR 8545) documenting the rationale and a draft standard planned for roughly 2026 and finalization in 2027 [5][6]. For protocol engineers this is a commitment device, not a to-do item for 2026: the reasonable 2025 posture is to ensure key-establishment interfaces are algorithm-parameterized so that a second KEM is a configuration event, not a code event.

Second, the signature pipeline is deliberately incomplete. The FN-DSA (Falcon-derived) standard was announced as forthcoming throughout 2025 but did not arrive — a consequence of genuine implementation difficulty (floating-point arithmetic is notoriously hard to implement in constant time), which ecosystem analysis has flagged as a deployment hazard in its own right [10]. Its absence is not an accident to be lamented but a sequencing decision: the certificate ecosystem's 2026-2027 window will arrive before the signature portfolio is fully diversified, which raises the value of the classical-signature maintenance discussed in Section 6.

Third, the deadline structure is unchanged and closer. IR 8547's deprecation (2030) and disallowance (2035) dates [4] now sit five and ten years out, with the migration's confidentiality half — the part with the retroactive-decryption exposure — substantially complete for the traffic majority. The authentication half is the open work item.

Algorithm / instrument	Status at year-end 2025	Date	Engineering note
ML-KEM (key establishment)	FIPS 203, final	Aug 2024	Primary KEM; deployed by default across the TLS stack
ML-DSA (signatures)	FIPS 204, final	Aug 2024	Standardized; certificate-ecosystem adoption pending
SLH-DSA (signatures)	FIPS 205, final	Aug 2024	Hash-based backup; conservative but large
FN-DSA (Falcon-derived signatures)	FIPS 206 draft forthcoming	announced through 2025	Implementation difficulty (floating-point constant-time concerns) cited as the bottleneck
HQC (key establishment)	Selected as fifth algorithm; draft ~2026; final 2027	Mar 2025	Code-based backup to ML-KEM on different mathematics; rationale in NIST IR 8545

Classical public-key cryptography	Deprecation / disallowance schedule	Nov 2024 (IR 8547 ipd)	Deprecated after 2030; disallowed after 2035 (≥ 112 -bit strength)
-----------------------------------	-------------------------------------	------------------------	--

Table 2. The post-quantum standards pipeline at year-end 2025 [1][2][3][4][5][6][10].

6. The In-Between Era (RQ3)

6.1 The asymmetric configuration

Figure 3 names the structural condition that the 2025 deployment record created. By late 2025, the majority of human-initiated traffic crossing major CDNs establishes its session keys with hybrid post-quantum key exchange [10] — the confidentiality half of the transition is, for practical purposes, on its way to done. But the authentication half has not moved: server certificates and their chains remain RSA and ECDSA, because the certificate ecosystem's prerequisites — CA hardware security modules with post-quantum support, audited and validated — were explicitly identified as the bottleneck, with public post-quantum certificates not expected before 2026 and broad trust later [10]. The result is an Internet running post-quantum confidentiality on a classical authentication base.

6.2 Threat windows: one closed, one open

NIST IR 8547 draws the operative distinction: key-establishment schemes protecting data with long confidentiality lifetimes are exposed to *retroactive* decryption — the harvest-now-decrypt-later attack — whereas authentication systems remain secure as long as their algorithms and keys are secure **at the time authentication is performed** [4]. The 2025 defaults therefore close (for the traffic majority) the threat that created the urgency in the first place: transit records captured today against future quantum decryption. What remains open is the contemporaneous window: an adversary who can forge or subvert classical signatures *today* — whether via cryptanalytic surprise, implementation failure, or CA compromise — faces a post-quantum key exchange that authenticates nothing by itself. Hybrid key exchange protects confidentiality against a future quantum attacker; it does not protect authentication against a classical one. The in-between era's risk profile is thus neither "secure" nor "vulnerable" but *specifically asymmetric*, and security engineering should treat it as such: the confidentiality threat model just shifted from harvest-now-decrypt-later to implementation-quality assurance, while the authentication threat model is unchanged from the pre-transition era — with a known end date (the 2026-2027 certificate window) instead of an open horizon.

6.3 What to engineer now

Three concrete engineering commitments follow. **Protect the hybrid.** A deployment that negotiates X25519MLKEM768 by default can often be forced down to classical-only key exchange by an active adversary who tampers with group negotiation; downgrade protection is part of TLS 1.3's design [19], but middlebox regressions and misconfiguration can reintroduce implicit downgrade paths — and systematic field data on downgrade attempts against deployed post-quantum-hybrid estates is not yet public, so the risk is treated here as a theoretical class to engineer against rather than an observed one. The hybrid's security property — secure if *either* component holds — is only worth having if the post-quantum component actually engages; it is a property European national guidance has explicitly required or recommended for the transition's high-assurance deployments [20][21]. **Plan PKI re-issuance against the 2026-2027 window.** Organizations should inventory certificate ecosystems now (HSM capability, CA support, validation pipelines) so that post-quantum certificate adoption is a scheduled project rather than a scramble. **Keep classical signatures well-operated.** Through the in-between era, RSA and ECDSA remain the authenticator of record; their operational hygiene — key strength, rotation, revocation — is not deprecated work but the active defense of the era.

The in-between era: PQ confidentiality on a classical authentication base

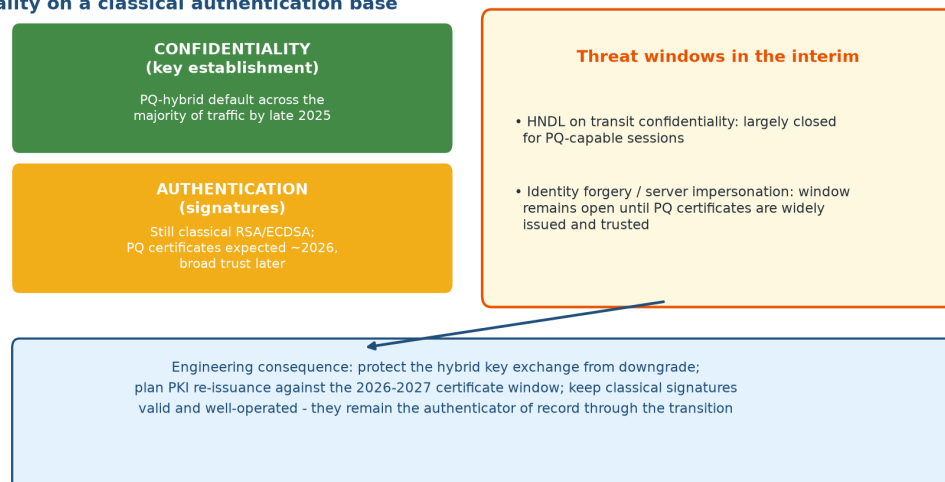


Figure 3. The in-between era: post-quantum confidentiality (key establishment) operating on a classical authentication base, with the interim threat windows and engineering consequences.

7. The Residual-Risk Register (RQ4)

Figure 4 maps six residual risk classes against six mitigation classes; Table 3 records the register. Four findings.

R1 — Implementation defects and side channels remain the live risk. The KyberSlash disclosures of December 2023 [13] showed timing defects across independent implementations before the algorithms were ever widely deployed; the 2025 defaults have massively expanded the deployed surface. The strongest mitigation class is memory-safe, independently audited implementations (the register's single "strong" cell), with telemetry and hybrid composition as partial cover. This risk never closes; it is the permanent cost of new cryptography.

R2 — Hybrid combiner semantics are subtle. Combining ML-KEM with X25519 must preserve the property that the session key remains secure if either component holds — the property IR 8547 describes and protocol designers have formalized [4][12]. The register scores most mitigation classes partial here because the risk is concentrated in edge cases (group-negotiation behavior, key-share fallback logic) that tests systematically under-cover.

R3 and R4 — the two latency risks. Certificate/PKI latency (R3) and the non-upgradeable client long tail (R4) are calendar risks, not code risks: standards pressure and CA/Browser-forum coordination score strong against R3, and adoption telemetry against R4, but neither is fully mitigable by any single actor — they are the transition's schedule, made visible.

R5 and R6 — the renewal loop. FN-DSA's pending draft and HQC's 2027 finalization mean the algorithm portfolio will change at least twice more this decade [5][6]; crypto-agility (algorithm-parameterized interfaces) and validation pipelines (FIPS/CMVP lead times) are the mitigation classes that keep those changes cheap. Organizations that treated 2024-2025's defaults as a one-time event will rediscover this in 2026-2027.

Residual-risk register vs. mitigation classes for the 2025 PQ transition (author assessment)

	Hybrid composition & downgrade defense	Telemetry & adoption monitoring	Standards & CA/Browser forum pressure	Memory-safe, audited implementations	Procurement & validation pipelines	Documented legacy exception register
R1 Implementation defects & side channels	~	~	-	+	~	~
R2 Hybrid combiner semantics	~	~	-	~	-	~
R3 Certificate / PKI latency	-	+	+	-	~	~
R4 Long-tail non-upgradeable clients	~	+	+	~	~	-
R5 Algorithm-renewal agility (FN-DSA, HQC)	+	~	+	-	+	+
R6 Validation lag (FIPS/CMVP)	~	~	+	-	+	+

Figure 4. Residual-risk register versus mitigation classes for the 2025 post-quantum transition (author assessment; register detail in Table 3).

Risk class	Description	Primary mitigation class	Horizon
R1 Implementation defects & side channels	Timing and correctness defects across independent ML-KEM implementations (KyberSlash class, Dec 2023)	Memory-safe, independently audited implementations; telemetry	Ongoing
R2 Hybrid combiner semantics	Correctness of ML-KEM + X25519 composition, group negotiation, key-share fallback	Hybrid composition design; conformance testing	2025-2027
R3 Certificate / PKI latency	Public PQ certificates not expected before 2026; HSM/CA audits the bottleneck	Standards and CA/Browser-forum coordination; re-issuance planning	2026-2027
R4 Long-tail non-upgradeable clients	Embedded and firmware-constrained clients on multi-year refresh cycles	Adoption telemetry; legacy exception register	2030+
R5 Algorithm-renewal agility	FN-DSA and HQC will add portfolio changes in 2026-2027	Algorithm-parameterized interfaces; procurement clauses	2026-2027
R6 Validation lag	FIPS/CMVP validation lead times behind algorithm releases	Validation pipelines treated as critical-path schedule items	Ongoing

Table 3. The residual-risk register for the 2025 post-quantum transition (author assessment).

8. Discussion

8.1 What 2025 proved

The deepest lesson of the 2025 record is about *defaults as infrastructure*. The cryptographic community spent 2016-2024 producing algorithms; 2024-2025 showed that the transition actually happens when the default configuration of shipping software changes — Chrome's desktop default, then Go's standard library, then OpenSSL's LTS release. Each default converted a million deployment decisions into zero deployment decisions. For security engineering generally, this reframes the unit of deployment work: the artifact that matters is not the migration guide but the default.

8.2 What the in-between era demands

The in-between era is not a gap in the transition to be tolerated; it is a well-defined operating condition with its own engineering duties — downgrade-resistant hybrid deployment, classical-signature hygiene, and PKI preparation against a known 2026-2027 window. Framing it precisely matters because the two failure modes are opposite: complacency ("post-quantum is done, look at the traffic charts") and alarmism ("nothing is post-quantum until certificates move"). The accurate statement is the asymmetric one, and this paper's contribution is to state it with the deployment evidence attached.

8.3 Trust framing

Digital trust in this period rests on an unusual configuration: the confidentiality guarantee has been upgraded ahead of the authenticity guarantee, reversing the historical order in which authentication led. Trust frameworks — audit programs, procurement questionnaires, transparency dashboards — should track the two halves separately, because "is this connection post-quantum?" is now two questions with two different answers, and conflating them produces both false assurance and false alarm.

9. Limitations and Threats to Validity

Construct validity. The layered-readiness assessment (Figure 2) and the risk-coverage matrix (Figure 4) are qualitative author assessments grounded in documented mechanisms; a different assessor could shift individual cells, and no inter-rater reliability is claimed. **Internal validity.** The deployment timeline relies on vendor release documentation, which can itself be imprecise; where secondary sources conflicted with primary ones, the primary source was used, but vendor-document errors cannot be excluded. **External validity.** Traffic-coverage statistics reflect one major CDN's measurement [10] and may not generalize to all traffic mixes; the long-tail client population is heterogeneous by definition. **Reliability.** Default behavior can change between point releases (Go's own defaults have evolved across 1.23-1.26 [8]); the snapshot is valid as of the dates cited. **Temporal scope.** The analysis is frozen at year-end 2025; the HQC draft, the FN-DSA draft, the first post-quantum certificates, and IR 8547's finalization are all expected in 2026-2027 and are treated as pipeline commitments, not results.

10. Conclusion

2025 was the year the post-quantum transition stopped being a plan and became a default. In twelve months, the software that terminates most of the Internet's encrypted traffic moved to hybrid ML-KEM key exchange without asking administrators to do anything — the single most encouraging development in applied cryptography in a generation. But the achievement is half a transition: sessions are sealed against future quantum computers while the identities authenticating them remain classically signed, public post-quantum certificates are still ahead of us, and the algorithm portfolio will keep growing (FN-DSA, HQC) for years. The engineering agenda that follows from the evidence is specific: defend the hybrid against downgrade, prepare the PKI for the 2026-2027 certificate window, treat implementation quality as the standing front it has become, and build the agility that turns the next algorithm renewal into a configuration change. The transition's first half is being won by defaults; the second half will be won by the organizations that start planning it now.

Declarations

Funding. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

CRedit contribution statement. Feng Hao: conceptualization, methodology, investigation, deployment-record verification, writing – original draft. Danda B. Rawat: validation, deployment-record analysis, writing – review & editing. (Corresponding author: Danda B. Rawat.)

Competing interests. The authors declare no competing interests.

Data and code availability. No new datasets were generated. All deployment dates trace to the cited primary release documentation [7][8][9][10][11]; the risk register (Table 3) is a documented author assessment.

Ethics and dual-use. This paper contains no exploit material, live credentials, or operational attack guidance; discussions of implementation defects (KyberSlash) are limited to publicly disclosed, patched findings in a coordinated-disclosure context.

AI assistance declaration. AI-based tools were used in a limited supporting capacity (literature search, figure preparation, and language editing); all research content, analysis, and conclusions were produced, reviewed, and verified by the authors, who take full responsibility for the work.

References

- [1] National Institute of Standards and Technology, "Module-Lattice-Based Key-Encapsulation Mechanism Standard," FIPS 203, U.S. Dept. of Commerce, Aug. 2024. doi: 10.6028/NIST.FIPS.203.
- [2] National Institute of Standards and Technology, "Module-Lattice-Based Digital Signature Standard," FIPS 204, U.S. Dept. of Commerce, Aug. 2024. doi: 10.6028/NIST.FIPS.204.
- [3] National Institute of Standards and Technology, "Stateless Hash-Based Digital Signature Standard," FIPS 205, U.S. Dept. of Commerce, Aug. 2024. doi: 10.6028/NIST.FIPS.205.
- [4] D. Cooper, D. Apon, Q. Dang, M. Davidson, W. Dworkin, and C. Miller, "Transition to Post-Quantum Cryptography Standards," NIST IR 8547 (Initial Public Draft), Nov. 2024. doi: 10.6028/NIST.IR.8547.ipd.

- [5] National Institute of Standards and Technology, "NIST Selects HQC as Fifth Algorithm for Post-Quantum Encryption," Mar. 11, 2025. [Online]. Available: <https://www.nist.gov/news-events/news/2025/03/nist-selects-hqc-fifth-algorithm-post-quantum-encryption>
- [6] National Institute of Standards and Technology, "Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process," NIST IR 8545, 2025. doi: 10.6028/NIST.IR.8545.
- [7] OpenSSL Project, "OpenSSL 3.5.0 release notes," Apr. 8, 2025. [Online]. Available: <https://www.openssl.org/news/openssl-3.5-notes.html>
- [8] The Go Project, "Go 1.24 Release Notes," Feb. 2025. [Online]. Available: <https://go.dev/doc/go1.24>
- [9] B. Westerbaan, "The state of the post-quantum Internet," Cloudflare Blog, Mar. 2024. [Online]. Available: <https://blog.cloudflare.com/pq-2024/>
- [10] B. Westerbaan, "State of the post-quantum Internet in 2025," Cloudflare Blog, Oct. 28, 2025. [Online]. Available: <https://blog.cloudflare.com/pq-2025/>
- [11] F. Kammel, "Post-Quantum Cryptography in Kubernetes," Kubernetes Blog, Jul. 18, 2025. [Online]. Available: <https://kubernetes.io/blog/2025/07/18/pqc-in-k8s/>
- [12] K. Kwiatkowski, P. Kampanakis, B. E. Westerbaan, and D. Stebila, "Post-quantum hybrid ECDHE-MLKEM Key Agreement for TLSv1.3" (draft-kwiatkowski-tls-ecdhe-mlkem), IETF Internet-Draft, 2024. [Online]. Available: <https://datatracker.ietf.org/doc/draft-kwiatkowski-tls-ecdhe-mlkem/>
- [13] D. J. Bernstein et al., "KyberSlash: Exploiting secret-dependent division timings in Kyber implementations," Dec. 2023. [Online]. Available: <https://kyberslash.cr.yp.to/>
- [14] M. Mosca, "Cybersecurity in an era with quantum computers: will we be ready?" IEEE Security & Privacy, vol. 16, no. 5, pp. 38-41, 2018. doi: 10.1109/MSEC.2018.2888779.
- [15] C. Gidney and M. Ekerå, "How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits," Quantum, vol. 5, p. 433, 2021. doi: 10.22331/q-2021-04-15-433.
- [16] C. Paquin, D. Stebila, and G. Tamvada, "Benchmarking post-quantum cryptography in TLS," in Proc. Int. Conf. on Post-Quantum Cryptography (PQCrypto 2020), LNCS 12100, Springer, 2020. doi: 10.1007/978-3-030-64386-8_4.
- [17] D. Sikeridis, P. Kampanakis, and M. Devetsikiotis, "Post-quantum authentication in TLS 1.3: a performance study," in Proc. 2020 Network and Distributed System Security Symposium (NDSS), 2020. doi: 10.14722/ndss.2020.24203.
- [18] P. Schwabe, D. Stebila, and T. Wiggers, "Post-quantum TLS without handshake signatures," in Proc. 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS '20), pp. 1461-1480, 2020. doi: 10.1145/3372297.3423350. [Online]. Available: <https://eprint.iacr.org/2020/534>
- [19] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," IETF RFC 8446, Aug. 2018. doi: 10.17487/RFC8446.
- [20] Agence nationale de la sécurité des systèmes d'information (ANSSI), "Views on the post-quantum cryptography transition," Jan. 2024. [Online]. Available: <https://www.ssi.gouv.fr>
- [21] Bundesamt für Sicherheit in der Informationstechnik (BSI), "Quantum-Safe Cryptography: Fundamentals, Current Developments and Recommendations," 2024. [Online]. Available: <https://www.bsi.bund.de>
- [22] OpenSSH, "OpenSSH 9.0 release notes," Apr. 2022. [Online]. Available: <https://www.openssh.com/txt/release-9.0>
- [23] G. Alagic et al., "Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process," NIST IR 8413, Jul. 2022. doi: 10.6028/NIST.IR.8413.