IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 26 January 2026, accepted 26 February 2026, date of publication 2 March 2026, date of current version 5 March 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3669437

RESEARCH ARTICLE

From Policy to Practice: A Sector-Agnostic Operational Framework for Post-Quantum Cryptography Transition

BERAT BIRGIN AND BARIS CELIKTAS
Computer Engineering Department, Işık University, 34398 Istanbul, Türkiye
Corresponding author: Berat Birgin (24sibe5002@isik.edu.tr)

ABSTRACT The pace of quantum computing development necessitates not only the adoption of post-quantum cryptographic algorithms, but also the establishment of an executable and auditable institutional transition process. Although guidance documents published by the National Institute of Standards and Technology (NIST) and roadmaps proposed by the Post-Quantum Cryptography Coalition (PQCC) articulate strategic objectives, they largely remain procedural constructs lacking a concrete operational execution model. This paper presents an industry-neutral operational framework that translates policy-level post-quantum cryptography (PQC) guidance into deterministic, proof-producing process flows encompassing cryptographic asset discovery, classification, risk modeling, algorithm selection, deployment, monitoring, and governance enforcement. Central to the framework is a deterministic Quantum Risk Scoring (QRS) function, calibrated using the Analytical Hierarchy Process (AHP), which enables reproducible asset prioritization and policy-driven enforcement decisions. Framework executability is further strengthened through cryptography-aware continuous integration/continuous deployment (CI/CD) validation gates and downgrade protection mechanisms, ensuring the generation of verifiable and immutable audit artifacts. A scenario-based operational validation, implemented using open-source toolchains, demonstrates the framework's operability, auditability, and governance alignment without relying on empirical cryptographic performance benchmarks, confirming that PQC transition can be operationalized as a verifiable lifecycle process bridging policy guidance with enforceable technical actions. Rather than introducing new cryptographic primitives, this work formalizes PQC transition as an operational systems-engineering problem centered on governance-enforced execution and lifecycle verifiability.

INDEX TERMS Analytic hierarchy process (AHP), cryptographic transition framework, governance feedback loop, post-quantum cryptography (PQC), quantum risk scoring (QRS), scenario-based validation.

I. INTRODUCTION

The rapid progress of quantum computer science and the concomitant development of efficient algorithms have restated the traditional theoretical problem as a pressing security concern. Specifically, the existence and the known efficiency of Shor's algorithm undermine the mathematical foundations of the most commonly employed public-key cryptoschemes, such as the well-known and deployed algorithms of Rivest–Shamir–Adleman (RSA) and elliptic curve cryptography (ECC), and consequently compromise the security guarantees of the digital infrastructure. Moreover, the "harvest-now, decrypt-later" paradigm, which enables the collection of data today and decrypting it later with the availability of efficient quantum computers, makes the security of data and digital signatures, which is traditionally required to be long-lived, a pressing concern, and the need to adopt post-quantum cryptography (PQC) is unavoidable [1].

In reaction to this newly emerging risk, global standardization activities have increased in scale and intensity. The U.S.

The associate editor coordinating the review of this manuscript and approving it for publication was Mohuya Chakraborty.

National Institute of Standards and Technology (NIST) has finalized its first set of PQC standards, including Kyber for key encapsulation, as well as Dilithium [2] and SPHINCS+ for digital signatures [3], [4], after a multi-year public evaluation process documented in the NIST IR 8240-8545 sequence [5], [6], [7], [8], [9]. A fourth standard, the draft FIPS 206, defines the Falcon (FN-DSA) digital signature scheme and is in public comment after the Sixth NIST PQC Conference [10]. In parallel with these standardization activities, industry-led transition initiatives, such as the Post-Quantum Cryptography Coalition (PQCC) Roadmap, have defined high-level domains for PQC transition planning and implementation, including preparation, planning, execution, and monitoring [11]. In summary, these activities define a strong platform for standardization and strategic planning in the field of cryptography.

Despite the existence of standardized algorithms and migration guidance, it is apparent that most of the extant PQC migration recommendations provide very little support for translating policy objectives into executable processes, enforceable controls, and auditable operational evidence [12], [13]. Most organizations face challenges such as a lack of integrated visibility of cryptographic assets, sectoral constraints on operations, and a lack of understanding of how to prioritize and enforce migration decisions across a diverse environment. Indeed, PQC migration is mostly treated as a ''compliance'' activity rather than a reproducible process [9], [14], [15], [16], [17], [18]. Clearly, this calls for a framework that can effectively translate policy recommendations into measurable and reproducible institutional processes and practices.

This study introduces a sector-agnostic PQC Transition Framework that is intended to facilitate the realization of a quantum migration through a structured lifecycle approach, quantification and verification of the PQC transition lifecycle phases, as well as the governance of the same lifecycle phases in a deterministic manner. A Quantum Risk Scoring model (QRS), which is based on the Analytic Hierarchy Process (AHP), is the foundation of this framework, which allows the deterministic and reproducible assessment of the priority of cryptographic assets based on various risk factors, which include the longevity of the data, the vulnerability of the algorithm, the keys, as well as the dependence of the system, among others [18], [19].

The framework consolidates the migration lifecycle into seven interdependent phases encompassing cryptographic asset inventory, classification, risk modeling, algorithm selection, deployment, monitoring, and governance, each of which is designed to be compatible with open-source tooling and verifiable artifacts.

The primary contributions of the work are as follows:

- A structured, reproducible, and sector-agnostic operational framework that bridges the existing policy-level PQC recommendations with enforceable technical execution.
- A deterministic QRS model, which is also AHP-calibrated, for quantitative and auditable prioritization of cryptographic assets, validated by performing a sensitivity analysis.
- A runnable validation process that incorporates QRS output into cryptographic-aware continuous integration/continuous deployment (CI/CD) enforcement models to create verifiable audit trails.
- Scenario-based validation to show end-to-end executability, alignment with the governance model, and reproducibility, but without relying on any empirical performance benchmark related to cryptography.

The current work does not propose novel cryptographic primitives or design novel protocols. Instead, it focuses on operationalizing existing standards and guidelines on PQC standards and transitions. This work makes a technical contribution by providing examples of how the intent of the policy can be consistently translated to deterministic decisions and controls.

The rest of the paper is structured as follows. Section II discusses the related work, Section III presents the methodology, Section IV presents the main framework of the PQC transition, Section V discusses the implementation, Section VI presents the scenario-based validation, Section VII discusses the implications, and Section VIII concludes the paper.

II. RELATED WORK

The migration toward PQC has evolved from a specialized research topic into a global security priority, as the obsolescence of classical cryptography is now an operational concern [20]. There has been considerable study of this shift process by standards organizations, industry collaboration, and the academic community in terms of algorithmic research, hybrid approaches to deployment, risk management, and industry-level customization. Despite the magnitude of this research work, there has been a lack of operation-level migration strategies. While prior studies on cryptographic agility and automated compliance primarily focus on adaptive mechanisms within specific protocol or tooling contexts, this study addresses institution-level orchestration of PQC transition through governance-enforced lifecycle control. This section synthesizes the literature across foundational standards, sector-specific adaptations, and process-oriented implementation studies, and identifies the resulting research gap.

A. FOUNDATIONAL GUIDANCE FROM STANDARDS BODIES

NIST provides the most comprehensive foundation for PQC transition. Its publications, from the initial Report on PQC [1] through four standardization rounds [5], [6], [7], [9], culminated in the first finalized standards, including Kyber for key encapsulation and Dilithium and Stateless Hash-Based Digital Signature Standard (SLH-DSA) for digital signatures [2], [3], [4]. Complementary reports,

including Transition to PQC Standards [18] and Recommendations for Key-Encapsulation Mechanisms [21], introduced hybrid deployment and risk-based prioritization models. In parallel, the PQCC Migration Roadmap [11] organized the transition process into four categories: Preparation, Baseline Understanding, Planning and Execution, and Monitoring and Evaluation. Although both NIST and PQCC [11] offer strategic direction, they remain largely abstract, focusing on policy alignment rather than step-by-step operationalization.

B. SECTOR-SPECIFIC ADAPTATION FRAMEWORKS

Applied research regards the process of migrating to PQC in a domain-specific manner, for example, in critical infrastructure, medicine, aviation, or the energy sector [14], [15], [22], [23]. Clearly, there is innovation on a sectoral basis, with adapted cryptographic architectures, but also bounded by local conditions in terms of regulations and performance.

C. PROCESS-ORIENTED AND IMPLEMENTATION STUDIES

Beyond domain focus, several studies explore migration processes themselves. Hasan et al. [12] proposed dependency-based analysis for identifying cryptographic bottlenecks in enterprise systems. Näther et al. [13] underscored the absence of unified models linking software engineering and governance. The PMMP-PQC process [24] introduced role-based maturity stages, though with limited operational tooling. Implementation-level studies again proved the feasibility of the deployment of post-quantum cryptography solutions in hardware-limited settings [25], [26], without incorporating automation and quantitative risk prioritization with verifiable governance models in their frameworks.

As indicated in Table 1, existing methodologies primarily provide strategic guidance, sector-specific adaptations, or maturity-based assessments. However, none formalize an institution-level transition architecture that integrates deterministic risk quantification, governance-enforced execution mechanisms, and verifiable validation artifacts within a unified operational model. The proposed PQC Transition Framework addresses this gap by consolidating these dimensions into a reproducible and auditable systems-engineering process.

D. SYNTHESIS OF LITERATURE AND IDENTIFIED GAP

Across existing studies, two persistent patterns emerge. First, research progress in post-quantum cryptography consistently outpaces institutional adoption, creating a structural disconnect between cryptographic innovation and system-level readiness [20], [27]. Second, as emphasized in NIST IR 8545 (2025) [9], hybrid and staged deployments require context-specific orchestration across assets, risks, and governance domains, yet such orchestration is rarely operationalized in existing frameworks.

Beyond standards and academic roadmaps, industry-driven migration handbooks provide insight into how PQC transition is approached in practice. For example, the TNO PQC Migration Handbook documents phased migration programs adopted by public and private institutions, emphasizing asset discovery, risk assessment, pilot deployment, and governance alignment [28]. These efforts demonstrate that real-world PQC migration is already underway, but they remain largely descriptive and do not define executable or verifiable workflows.

Complementary to institutional perspectives, protocol-level transition studies further illustrate the feasibility of post-quantum migration in operational systems. Practical PQC adaptations of standardized protocols, such as 5G-AKA [29], show how hybrid and staged deployment strategies can be applied within large-scale communication infrastructures, while post-quantum extensions of lightweight protocols like EDHOC [30] indicate that even constrained environments are subject to cryptographic transition pressures. However, these works are inherently scoped to specific protocols and do not address organization-wide transition management.

Taken together, the literature reveals a clear gap between descriptive guidance, protocol-specific feasibility studies, and the need for executable, institution-level transition models. This study addresses that gap by synthesizing foundational standards, sectoral insights, and implementation-oriented research into a structured, scenario-driven framework that integrates asset inventory, quantitative risk prioritization, algorithm selection, and governance enforcement into a cohesive operational process. In doing so, it advances PQC transition from high-level readiness concepts toward verifiable and reproducible institutional execution.

III. METHODOLOGY

This study employs a constructive research methodology to develop a prescriptive and reproducible framework that guides institutions through the operational transition to PQC. The goal is not to propose a new cryptographic primitive but to synthesize existing standards, empirical findings, and regulatory guidance into a coherent, actionable system. The methodological process comprises three interdependent phases: (1) knowledge synthesis and source analysis, (2) capability-oriented framework modeling, and (3) assessment of applicability and reproducibility.

A. PHASE 1: KNOWLEDGE SYNTHESIS AND SOURCE ANALYSIS

The first phase involved a systematic review of technical, academic, and regulatory literature forming the empirical basis for the framework. The corpus included:

- The full suite of NIST PQC publications, such as FIPS 203–205 (Kyber, Dilithium, SPHINCS+) [2], [3], [4], the draft FIPS 206 (Falcon) [10] and internal reports IR 8105–8547 [1], [5], [6], [7], [8], [9], [18] and SP 800-227 [21];
- The PQCC Migration Roadmap (2025) [11];

TABLE 1. Comparison of transition models and this work.

Model / Framework	Scope and Orientation	Decision Formalization	Tooling and Enforcement	Validation and Evidence Model
NIST IR 8547 (2024) [18]	Policy and governance-level transition guidance for federal agencies.	Narrative recommendations; no deterministic risk computation or threshold mapping.	Conceptual reference to automation; no prescriptive tool integration.	Self-assessment documentation and readiness tracking; no machine-verifiable artifacts.
PQCC Roadmap (2025) [11]	Strategic migration roadmap structured across four macro domains.	Structured categorization; lacks executable decision logic or algorithmic mapping.	Vendor-neutral advisory guidance; no enforcement mechanism defined.	Continuous readiness philosophy; no reproducible execution artifacts.
PMMP-PQC (2024) [24]	Organizational maturity and role-based process framework.	Maturity-stage progression model; does not define deterministic transition triggers.	Suggests integrated workflows; no explicit CI/CD or enforcement gates.	Maturity scoring and qualitative assessment; limited empirical verification.
This Work – PQC Transition Framework (2026)	Sector-agnostic operational systems-engineering model translating policy guidance into executable institutional workflows.	Deterministic QRS-based risk computation with threshold-mapped migration states (MIGRATE, HYBRID, MONITOR, DEFER).	Embedded CI/CD validation gates with downgrade protection and automated compliance checks.	Scenario-based operational validation producing audit logs, QRS outputs, and reproducible enforcement artifacts.

- More than twenty peer-reviewed studies addressing PQC implementation, risk modeling, and migration processes.

Each source was analyzed using a five-dimensional coding scheme encompassing scope, granularity, maturity, dependency depth, and implementability, establishing a structured empirical basis for framework construction.

B. PHASE 2: CAPABILITY-ORIENTED FRAMEWORK MODELING

The synthesized insights were formalized into a two-layer capability-oriented model comprising a core transition framework and a practical implementation guide, structured to support reproducibility and operational deployment. Layer 1 defines the logical structure of the transition process, while Layer 2 maps this structure to executable practices and toolchains.

A capability-centric design approach was adopted to structure the framework around deployable functional capabilities rather than fixed architectural states, supporting partial deployment, hybrid coexistence, and incremental governance scaling. Framework components were evaluated using a five-level maturity scale covering readiness, interoperability, scalability, auditability, and cost efficiency [9], [24].

C. PHASE 3: APPLICABILITY AND REPRODUCIBILITY ASSESSMENT

The final phase validated the framework's practicality across different institutional environments using a structured evaluation rubric built on five criteria:

- Cost Efficiency — prioritization of open-source tooling.
- Hardware Accessibility — support for both enterprise and resource-constrained platforms [26].
- Cryptographic Support — alignment with NIST-approved algorithms and hybrid readiness.
- Integration Capacity — compatibility with CI/CD, configuration management database (CMDB), and monitoring pipelines.
- Auditability and Transparency — traceability for verification and compliance assessment.

Validation followed a scenario-based reasoning approach, aligning assessment logic with sectoral constraints in energy [14] and healthcare systems [15]. A traceability matrix linked framework elements to supporting sources, tools, and evaluation criteria to support methodological transparency and reproducibility.

IV. CORE TRANSITION FRAMEWORK

The proposed PQC Transition Framework operationalizes NIST and PQCC [11] guidance into seven interdependent phases forming a reproducible lifecycle executable with open-source tools and measurable governance checkpoints. As illustrated in Figure 1, the framework aligns technical migration with evolving governance and risk dimensions, situating PQC adoption as a sustained institutional maturity process rather than a one-time cryptographic replacement.

Complementing this temporal overview, Figure 2 presents the internal operational lifecycle of the proposed framework. It organizes the seven phases into three macro layers: Assessment (P1–P3), Transition (P4–P5), and Monitoring (P6–P7), and illustrates the dependency and feedback relations among them. This visualization bridges the conceptual alignment shown in Figure 1 with the concrete, executable workflow later applied in the scenario-based validation.

Table 2 demonstrates that while NIST [18] and PQCC [11] define strategic migration domains, the proposed framework augments them with executable phases and verifiable open-source evidence artifacts.

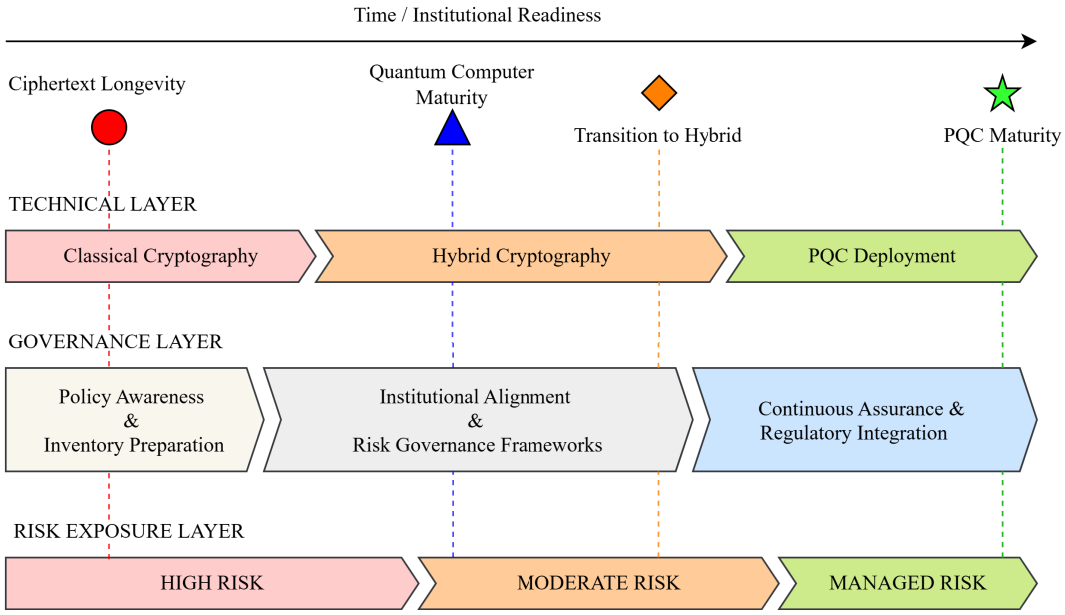

FIGURE 1. Temporal alignment of cryptographic maturity, governance evolution, and quantum risk across the PQC transition lifecycle.

TABLE 2. Alignment of NIST/PQCC guidance with the proposed PQC transition framework.

Macro Stage	NIST / PQCC Domains	Corresponding Framework Phases	Open-Source Evidence Artifacts / Tools
Assessment (Evaluation)	NIST IR 8547 Sec. 3.1 Preparation and Baseline Inventory PQCC Domain 1–2 (Preparation, Baseline Understanding)	Phase 1 – Asset Inventory and Cryptographic Discovery Phase 2 – Cryptographic Classification and Categorization Phase 3 – Risk Modeling and Quantum Threat Analysis	Snipe-IT export (CMDB snapshot), Cryptolyzer scan results, OWASP Dependency-Track report, Python QRS Engine output
Transition (Implementation)	NIST IR 8547 Sec. 3.2 Planning & Execution PQCC Domain 3 (Planning and Execution)	Phase 4 – Algorithm Selection and Hybridization Strategy Phase 5 – Deployment and Toolchain Integration	liboqs benchmark log, pqm4 latency report, GitLab CI "crypto-validation" YAML, container build checksum manifest
Monitoring & Governance (Continuous Assurance)	NIST IR 8547 Sec. 3.3 Monitoring & Evaluation PQCC Domain 4 (Monitoring and Evaluation)	Phase 6 – Monitoring, Validation and Operational Readiness Phase 7 – Governance, Policy and Institutional Alignment	Prometheus metric export, Grafana dashboard screenshot, OPA policy audit log, periodic review minutes

A. PHASE 1 - ASSET INVENTORY AND CRYPTOGRAPHIC DISCOVERY

One of the most persistent obstacles in PQC migration is the lack of comprehensive visibility into an organization's cryptographic footprint. Legacy components, unmanaged application programming interfaces (API) and undocumented encryption libraries create "shadow cryptography," which NIST IR 8545 identifies as a primary source of migration failure [9]. To address this challenge, the first phase combines manual code inspection, configuration auditing, and automated dependency scanning to construct a complete inventory of cryptographic assets, including keys, certificates, and algorithms embedded in firmware, applications, and communication interfaces. Open-source tools such as Snipe-IT, Cryptolyzer, and Open Web Application Security Project (OWASP) Dependency-Track collect standardized metadata, including algorithm type, key length, exposure level, and updatability, and integrate it into an enriched CMDB. The resulting verified inventory establishes a traceable baseline that directly supports subsequent classification, risk modeling, and migration planning.

B. PHASE 2 - CRYPTOGRAPHIC CLASSIFICATION AND CATEGORIZATION

Following asset discovery, institutions must determine which cryptographic components are most critical, exposed, and feasible to migrate. Rather than treating all cryptographic instances uniformly, this phase applies structured classification criteria based on functional role, data sensitivity, and regulatory dependency, as emphasized by NIST IR 8545 and the PQCC roadmap [9], [11]. Externally exposed systems, long-lived data repositories, and safety-critical applications,

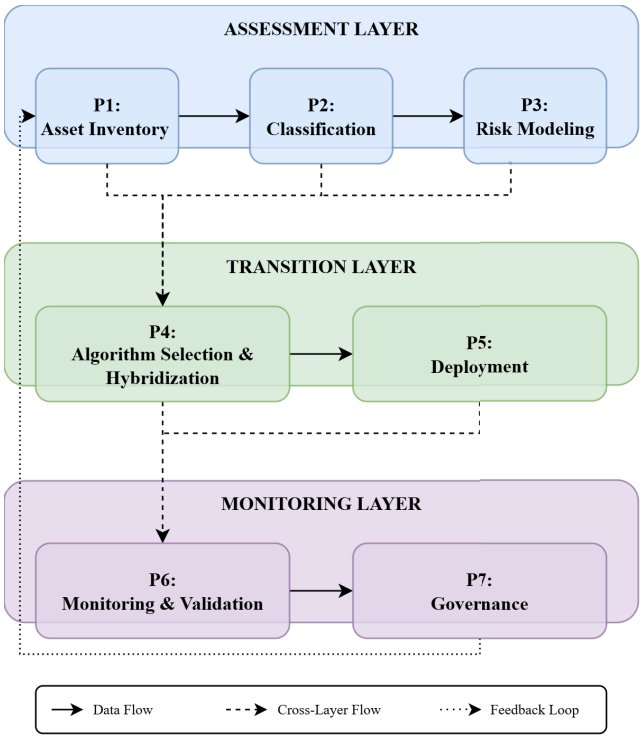

FIGURE 2. The seven interdependent phases are organized across three macro layers—Assessment, Transition, and Monitoring—to ensure continuous feedback between technical and governance processes. Dotted arrows indicate dependency and feedback relations between phases.

such as avionics or healthcare telemetry, are prioritized for early remediation, while low-impact or short-lived assets are deferred. Classification is operationalized through tagging and metadata annotation within the cryptographic inventory, using attributes including exposure level, key rotation policy, and interoperability constraints. The resulting classification produces a structured transition map that aligns technical prioritization with organizational and regulatory objectives.

C. PHASE 3 - RISK MODELING AND QUANTUM THREAT ANALYSIS

Following classification, institutions must identify which cryptographic assets pose the highest operational and confidentiality risks in a post-quantum context.

This stage involves replacing compliance list checks with the QRS model, which has its roots in NIST SP 800-227 [21] and IR 8105 [1]. The QRS model also evaluates assets based on factors of data longevity, vulnerability of algorithms, key exposure, and criticality of dependencies, while using temporal factors like harvest-now and decrypt-later. These scores are then obtained using Python analyses, and they correspond to migration priorities within the predefined classification system. Additional risk factors, based on decentralized and resource-constrained settings like the blockchain and IoT, with reference to energy efficiency, length of long-lived state, and trust, were also established by earlier research [31]. The resulting prioritized risk matrix provides measurable, auditable, and context-specific guidance for allocating PQC migration resources.

D. PHASE 4 - ALGORITHM SELECTION AND HYBRIDIZATION STRATEGY

With risks quantified, institutions must select post-quantum algorithms and deployment models that satisfy security, performance, and interoperability requirements. Guided by NIST's finalized standards, including Kyber for key encapsulation [2] and Dilithium and SLH-DSA for digital signatures [3], [4], and by hybridization guidance in NIST IR 8547 and the PQCC roadmap [11], [18], this phase evaluates candidate algorithms based on implementation maturity, protocol compatibility, and fallback behavior within legacy systems. Current literature overwhelmingly attests the viability, maturity of security, and performance feasibility of lattice cryptographic primitives in both the enterprise setting and resource-constrained environments, thereby underpinning their specification as standardized primitives [32], [33], [34]. Sustaining research efforts suggest that a hybrid cryptographic framework, combining classical, post-quantum, and quantum key distribution techniques, can support interoperability and transition resilience if correctly harmonized [35], [36]. Benchmarking exercises conducted with open-source toolkits such as liboqs and pqm4 confirm the performance parameters of throughput, latency, and resource consumption, with stress-testing also showing that classical methods cannot aggressively dominate quantum-safe negotiation, especially in current analyses [12], [17].

The outcome of this phase is a vetted algorithm portfolio and hybrid deployment policy that balances cryptographic agility with operational stability, forming the technical baseline for deployment in the subsequent phase.

E. PHASE 5 - DEPLOYMENT AND TOOLCHAIN INTEGRATION

This phase translates algorithm selection decisions into controlled production deployments through automation and policy-enforced release workflows. Cryptographic requirements derived from NIST IR 8547 and the PQCC roadmap [11], [18] are enforced directly within CI/CD pipelines, where validation gates verify algorithm compliance, certificate validity, and hybrid negotiation parameters prior to release. Open-source tooling, including GitHub Actions, GitLab CI, and containerized build pipelines, ensures consistent deployment across environments, while integration with liboqs enables end-to-end testing of both classical and post-quantum configurations. Each deployment produces immutable audit artifacts, establishing traceable and repeatable implementation workflows.

F. PHASE 6 - MONITORING, VALIDATION, AND OPERATIONAL READINESS

This phase sustains post-quantum cryptographic integrity by continuously observing and validating deployed

configurations during live operation. Operational metrics and telemetry are aligned with the monitoring and evaluation principles defined in NIST IR 8547 and the PQCC roadmap [11], [18], providing real-time visibility into algorithm usage, key lifetimes, and fallback behavior. Monitoring stacks based on Prometheus and Grafana support the detection of configuration drift and non-compliant states, while periodic audits, automated key-rotation tests, and incident simulations generate verifiable evidence of operational readiness [38]. The outcome is a continuously verifiable assurance state supporting sustained cryptographic integrity.

G. PHASE 7 - GOVERNANCE, POLICY, AND INSTITUTIONAL ALIGNMENT

The final phase institutionalizes post-quantum cryptographic transition by embedding it within formal governance, policy, and accountability structures. Guided by NIST IR 8547, the PQCC roadmap and the PMMP-PQC maturity model [11], [18], [24], this phase defines role-based decision rights, exception handling procedures, and audit responsibilities, as summarized in Table 3. Executive oversight establishes approved algorithm baselines and review cycles, while DevSecOps and compliance functions enforce these baselines through measurable policy gates. Vendor management and procurement processes are aligned to require PQC conformance disclosures, reducing the risk of reintroducing non-compliant dependencies. Past studies have shown that alignment of governance is significant in the public domain, especially where there might be dispersed control, thus hindering the process of universal quantum-safe technology adoption [39]. Results of audits, risk measures, as well as algorithm lifecycle measures, are fed into regular governance assessments, hence ensuring that there is ongoing cryptographic alignment.

V. PRACTICAL IMPLEMENTATION GUIDE

This section demonstrates how the seven-phase PQC Transition Framework introduced in Section IV can be operationalized using reproducible, open-source workflows and standard governance roles. Rather than serving as a technical playbook, the purpose is to show that the proposed framework is executable and verifiable under realistic institutional constraints. In alignment with NIST IR 8547 [18], the PQCC roadmap [11], and the PMMP-PQC maturity model [24], the implementation approach emphasizes accountable execution across all transition phases. The following subsections illustrate how role-based governance structures, open-source toolchains, and automation mechanisms translate the conceptual framework into a practical and evidence-driven implementation.

A. ROLE-BASED IMPLEMENTATION MODEL

The process of transitioning to PQC requires a jointly governed process at various levels within organizations, as opposed to purely technological updates. Following the recommendations laid out in the NIST IR 8547 [18] and the process described within the PMMP-PQC [24] maturity model, the proposed implementation architecture to migrate to PQC defines duties at three levels: strategic, operational and technical. At the strategic level, duties are primarily the domain of Chief Information Security Officers (CISOs), compliance officers, and risk managers, where the duties involve the definition of algorithmic baselines, hybridization plan approvals, and the handling of exceptions. At the next level, duties are usually the domain of system architects and DevSecOps teams, where the duties include the realization of the above-mentioned strategic instructions, involving activities like the discovery, categorization, deployment, and verification of the process through the use of certified toolchains. At the technical level, the duties involve the integration testing, monitoring, and the verification of the fallback process by the developers, QA analysts, and auditors. These levels operate within a combined governance cycle.

B. PROPOSED OPEN-SOURCE TOOLCHAIN

To support reproducible and auditable execution of the PQC Transition Framework, each phase is mapped to representative open-source tools that provide measurable outputs and traceable workflows. As per the implementation guidelines set in NIST IR 8547 and discussed in the PQCC roadmap [11], [18], the proposed approach considers transparency and vendor neutrality for implementation rather than relying on proprietary tools. The degree to which open-source tools are used for discovery, classification, risk modeling analysis, choice of algorithms for implementation, implementation enforcement, and governance of the seven stages of the proposed approach has been outlined in Table 4. As emphasized by Hasan et al. [12], reliance on open-source ecosystems enhances reproducibility and lowers adoption barriers, enabling institutions to implement PQC transition processes without specialized or proprietary infrastructure.

C. AUTOMATION AND CI/CD INTEGRATION

Cryptographic assurance is operationalized by embedding validation controls directly into the software delivery pipeline, ensuring consistent enforcement across development and production environments. In alignment with DevSecOps principles outlined in NIST IR 8547 [18], the framework integrates cryptography-aware validation gates into CI/CD pipelines to produce auditable and self-verifying deployment workflows. Three control points are defined: (a) the pre-build gate, which pinpoints deprecated and non-compliant algorithms in source code and dependencies; (b) the post-build gate, which tests post-quantum key exchange and signatures based on standardized test libraries; and (c) the deployment gate, which ensures compliance with valid certificates, key lifetime, and configuration before being released. All these control points generate immutable verification artifacts. This ensures traceable verification of a cryptographic decision throughout the entire lifecycle. As highlighted by the PQCC roadmap [11], embedding

TABLE 3. Overview of the PQC transition framework.

Phase	Objective	Key Activities	Outputs
1. Asset Inventory	Identify all cryptographic components, including shadow assets	Manual audit, automated scanning, CMDB enrichment	Verified cryptographic inventory
2. Classification	Assess criticality and migration feasibility	Functional tagging, exposure mapping	Structured transition map
3. Risk Modeling	Prioritize assets by quantum risk and data longevity	Quantum Risk Score computation	Ranked migration plan
4. Algorithm Selection	Choose and pilot NIST-approved and hybrid algorithms	Benchmarking, hybrid validation	Algorithm baseline & hybrid policy
5. Deployment	Integrate PQC into CI/CD and infrastructure	Incremental rollout, crypto-aware gates	Operational PQC implementation
6. Monitoring	Sustain readiness through continuous validation	Telemetry, fallback detection, audits	Measurable cryptographic assurance
7. Governance	Institutionalize cryptographic change management	Policy update, role assignment, exception control	Long-term quantum resilience framework

TABLE 4. Representative open-source toolchain for PQC migration.

Phase	Example Tools	Purpose / Integration
Phase 1 - Asset Inventory	Snipe-IT [41], Cryptolyzer [42]	Discovery of cryptographic assets, TLS and certificate analysis, metadata integration with CMDB systems.
Phase 2 - Classification	OWASP Dependency-Track [43]	Functional tagging, algorithm identification, exposure and dependency mapping.
Phase 3 - Risk Modeling	Python + Pandas (QRS Engine)	Quantitative risk scoring based on algorithm strength, key length, and data longevity.
Phase 4 - Algorithm Selection	liboqs [44], pqm4 [45]	Benchmarking and testing of NIST-approved and hybrid algorithms for performance and compatibility.
Phase 5 - Deployment	GitLab CI [46]	Automated cryptographic validation gates within CI/CD pipelines and configuration compliance checks.
Phase 6 - Monitoring	Prometheus [47], Grafana [48]	Continuous telemetry, fallback detection, key lifetime tracking, and audit logging.
Phase 7 - Governance	Open Policy Agent (OPA) [49]	Enforcement of algorithm baselines, access control, and cryptographic policy compliance.

governance logic into automated pipelines strengthens accountability while reducing reliance on manual compliance checks. In this context, embedding cryptographic validation into CI/CD pipelines aligns with the Secure Software Development Framework (SSDF), which emphasizes integrating security controls, verification activities, and traceable evidence throughout the software development lifecycle rather than relying on post-deployment checks [40].

A concrete downgrade-protection mechanism is incorporated into the CI/CD pipeline to prevent quiet regression to classical cryptography. Following deployment, a transient container establishes a Transport Layer Security (TLS) handshake with the target service and inspects the terms of the handshake for indicators of the usage of post-quantum cryptography and/or hybrid cryptography. If non-compliant negotiation occurs, the pipeline will be shut down, and a warning alert will be produced for governance analysis. This will provide a means to leverage downgrade-protection policy via the automated enforcement process. The ordered QRS output, including policy flags, is consumed by CI/CD validation gates to deterministically allow, restrict, or block cryptographic deployments, producing immutable audit artifacts for each enforcement decision. This enforcement logic is specified as a tool-agnostic reference design that focuses on decision flow and auditability rather than platform-specific pipeline syntax.

Algorithm 1 operationalizes governance intent by converting QRS policy flags into enforceable CI/CD decisions, while emitting immutable audit artifacts for every allow, restrict, or block outcome. CheckCompliance verifies cryptographic configuration against the declared baseline (e.g., approved or hybrid requirements) and flags downgrade-prone or non-compliant algorithm selections as violations. In this reference design, MIGRATE violations result in hard blocking, while

Algorithm 1 Crypto-Aware CI/CD Validation Gate (Reference Enforcement Design)

Require: Ordered QRS artifact `qrs_results.json` (asset_id, QRS score, policy flag)
Deployment crypto configuration manifest `crypto_config.json`
Policy thresholds and state definitions (MIGRATE, HYBRID, MONITOR, DEFER)
Ensure: Gate decision (ALLOW / RESTRICT / BLOCK) and audit artifact `audit_gate.json`

```
Parse qrs_results.json into QRSList
Parse crypto_config.json into CryptoCfg
Decision ← ALLOW
Initialize empty list Findings
for each target asset a referenced in CryptoCfg do
    (score, policy) ← QRSList[a.id]
    violations ← CheckCompliance(a, CryptoCfg)
    if violations ≠ ∅ then
        Append {a.id, score, policy, violations} to Findings
        if policy = MIGRATE then
            Decision ← BLOCK
        else if policy = HYBRID then
            Decision ← RESTRICT
        else if policy = MONITOR then
            Decision ← RESTRICT
        end if
    end if
end for
Emit audit artifact audit_gate.json containing (Decision, Findings, timestamp)
if Decision = BLOCK then
    Fail pipeline stage (non-zero exit code)
else if Decision = RESTRICT then
    Continue with restricted status and log the finding for governance review
else
    Continue pipeline
end if
= 0
```

HYBRID and MONITOR states permit controlled continuation with governance escalation, ensuring proportional enforcement aligned with policy intent.

1) COMPLIANCE CHECKS IMPLEMENTED BY CHECKCOMPLIANCE

To address practical enforceability, the reference CI/CD gate design makes the compliance logic explicit through the `CheckCompliance` function. This function evaluates a deployment against a declared cryptographic baseline and downgrade-resilience requirements, ensuring that policy intent is converted into deterministic, auditable findings rather than remaining a conceptual control.

In the reference design, `CheckCompliance(a, CryptoCfg)` flags the following violation categories:

- Classical-only public-key in MIGRATE paths: Any RSA/Elliptic Curve Digital Signature Algorithm(ECDSA)-only handshake or certificate chain detected for assets flagged as `MIGRATE`.
- Missing hybrid negotiation for HYBRID assets: Hybrid key establishment is required but the observed or declared configuration indicates fallback to classical-only negotiation.
- Protocol and cipher-suite noncompliance: Disallowed protocol versions (e.g., legacy TLS) or non-approved cipher suites relative to the declared baseline.
- Certificate and key policy violations: Invalid key sizes, expired certificates, missing rotation constraints, or key lifetime windows outside policy.
- Configuration drift: Mismatch between the declared `crypto_config.json` manifest and the observed handshake/configuration evidence.
- Downgrade-prone states: Any negotiation pattern that permits silent regression from post-quantum or hybrid modes to classical-only modes without explicit exception handling.

Each detected violation is recorded in `audit_gate.json` together with the asset identifier, QRS score, policy flag, and a timestamp. This produces an immutable enforcement artifact that can be reviewed by governance functions and re-verified in subsequent pipeline runs.

D. PILOT IMPLEMENTATION SCENARIO

The feasibility of the proposed framework for practical application has been assessed using a pilot implementation carried out in a mid-scale public sector firm that utilized a hybrid cloud infrastructure. The motivation behind the pilot is to assess the applicability of the seven-step migration process to be accomplished by utilizing open-source resources. The pilot process began with a cryptographic asset discovery process based on Snipe-IT and Cryptolyzer, where several unrecorded legacy libraries were found in the internal APIs and authentication services, which were expected based on the idea of shadow cryptography. The assets were classified based on the level of exposure, regulatory dependency, and lifetime of keys, based on the recommendations offered by the publication NIST IR 8545 [9].

A Python-based QRS implementation quantified algorithmic risk using parameters defined in NIST SP 800-227 [21], producing a prioritized migration roadmap that emphasized identity management components and secure communication channels. Algorithm evaluation using liboqs and pqm4 indicated that Kyber and Dilithium satisfied the organization's performance and interoperability constraints, while hybrid handshake testing preserved compatibility with existing classical systems. During deployment, PQC validation gates were integrated into a GitLab CI/CD pipeline to enforce certificate integrity and hybrid compliance. Post-deployment monitoring with Prometheus and Grafana collected telemetry on handshake outcomes and fallback behavior. Audit logs and monitoring outputs from the initial evaluation cycle indicated no regression to classical cryptographic defaults, illustrating the feasibility and reproducibility of the proposed transition approach in a resource-constrained institutional setting.

E. FEEDBACK AND GOVERNANCE LOOP

The final stage of implementation integrates automated assurance outputs into institutional governance through a closed policy–automation feedback loop. In alignment with NIST IR 8547 and the PQCC roadmap [11], [18], audit results, telemetry data, and cryptographic performance metrics are periodically reviewed by designated governance bodies to reassess algorithm baselines, key management policies, and exception handling procedures. Governance-defined policies directly inform algorithm selection and hybridization rules, which are enforced through CI/CD validation gates where QRS outputs serve as quantitative compliance thresholds. The results of each automated validation, including fallback incidents, key lifecycle metrics, and deployment outcomes, are aggregated through monitoring and policy engines such as Prometheus and Open Policy Agent (OPA) and reported back to governance dashboards. This bidirectional exchange ensures that policy decisions drive technical enforcement, while operational evidence continuously refines governance posture. By tightly coupling risk scoring, automation telemetry, and policy adaptation, the proposed framework achieves end-to-end traceability from governance directive to operational verification and back, addressing the governance–enforcement disconnect observed in prior transition models such as PQCC and PMMP-PQC [11], [24].

VI. SCENARIO-BASED VALIDATION

A. OVERVIEW AND OBJECTIVES

This section offers a scenario-based validation of the PQC transition framework. Instead of evaluating the cryptographic capabilities and the efficiency of algorithms, the scenario focuses on the ability to execute the process, make decisions, and audit the entire transition process. This is to ensure that the intention of the policy can be implemented as an operable task.

A representative public-sector scenario is considered, in which core institutional functions, including authentication, data exchange, and certificate management, are prepared for post-quantum migration. Within this context, selected framework phases are exercised to produce representative outputs such as a cryptographic asset inventory, a quantitative risk assessment, and algorithm operability indicators that support traceable evaluation of transition activities.

Performance criteria, such as latency and throughput, are also intentionally omitted in this assessment, as these criteria have been adequately covered in other standard publications by the NIST and pqm4 benchmarks. The purpose of this validation is to demonstrate the application of existing standards in the form of operational procedures within the proposed framework, which contribute to the achievement of governance maturity rather than simply being a technical solution for replacement purposes.

B. SCENARIO CONTEXT AND ENVIRONMENT

The scenario-based validation was conducted in a controlled open-source environment designed to emulate the operational context of a mid-size public-sector organization. The simulated ecosystem included authentication services, application gateways, database systems, and a public-key infrastructure (PKI) component, reflecting a typical hybrid environment observed in municipal or governmental institutions. The objective was to verify that each phase of the PQC Transition Framework could be instantiated, observed, and documented under constrained yet realistic operational conditions.

The test environment was set up on top of Ubuntu 22.04 LTS using freely available tools. Snipe-it was used as a simple CMDB for the enumeration of cryptographic resources and their corresponding tags. On the other hand, Cryptolyzer was employed to enable the analysis of TLS endpoint visibility. The functionality of the post-quantum algorithms was confirmed using the liboqs library. The QRS function was implemented using Python to estimate the weighted exposure score of registered resources.

Six representative assets were manually entered into the CMDB to represent the key functions of the institution for services such as identity, APIs, database systems, email

Algorithm 2 QRS Evaluate Function

Require: CMDB asset list *cmdb_json* where each asset contains ordinal scores $(dl, av, ke, sd) \in [1, 10]$, AHP weight vector $W = \{w_{dl}, w_{av}, w_{ke}, w_{sd}\}$, Policy thresholds $T = \{T_{migrate}, T_{hybrid}, T_{monitor}\}$

Ensure: Ordered list of assets with QRS score and policy flag

Initialize empty list *Results*
for each asset a in *cmdb_json.assets* **do**
  $dl \leftarrow a.dl$
  $av \leftarrow a.av$
  $ke \leftarrow a.ke$
  $sd \leftarrow a.sd$
  Compute QRS score:

$$QRS(a) = dl \cdot w_{dl} + av \cdot w_{av} + ke \cdot w_{ke} + sd \cdot w_{sd}$$

  if $QRS(a) \geq T_{migrate}$ **then**
    $policy \leftarrow$ **MIGRATE**
  else if $QRS(a) \geq T_{hybrid}$ **then**
    $policy \leftarrow$ **HYBRID**
  else if $QRS(a) \geq T_{monitor}$ **then**
    $policy \leftarrow$ **MONITOR**
  else
    $policy \leftarrow$ **DEFER**
  end if
  Append $\{a.id,\ QRS(a),\ policy\}$ to *Results*
end for
Sort *Results* by *QRS* score in descending order
return *Results*

services, and certificate authorities. Categories of data included in each asset were the cryptographic algorithms used, their sizes, and the level of exposure, which are all system dependencies. The setup of all the components was done locally, which resulted in their operation without any proprietary dependence, thereby enabling the setup of the validation environment that was institutional in nature, focusing on the feasibility of operation rather than laboratory performance.

C. QUANTUM RISK SCORING (QRS) MODEL

To quantify the relative exposure of cryptographic assets to post-quantum threats, this study employs a QRS model. The model provides a reproducible and transparent mechanism for ranking assets based on their cryptographic vulnerability and institutional impact, in alignment with the risk-oriented assessment principles of NIST SP 800-227. [21].

The QRS component is designed as a deterministic decision function rather than a probabilistic assessment model. The deterministic evaluation and policy-state mapping process is formally defined in Algorithm 2. Given identical asset inventories and parameter configurations, QRS produces the same prioritization and policy outcomes, enabling reproducible enforcement decisions.

TABLE 5. Scenario instantiation of policy thresholds for QRS-to-State mapping.

Threshold	Value	Scenario intent
$T_{migrate}$	8.0	Force immediate action for externally exposed classical public-key dependencies (RSA/ECC) and long-lived confidentiality assets.
T_{hybrid}	7.0	Require hybrid controls where interoperability constraints exist but quantum exposure is still material.
$T_{monitor}$	6.0	Keep lower-risk assets under telemetry-based monitoring and periodic reassessment.

Thresholds are policy-defined decision boundaries selected to map continuous QRS values into actionable migration states.

1) POLICY THRESHOLD SELECTION RATIONALE

Thresholds $T_{migrate}$, T_{hybrid}, and $T_{monitor}$ are governance-defined decision boundaries rather than statistical cutoffs. In the scenario, thresholds are selected to (i) force immediate action on externally exposed classical public-key dependencies, (ii) mandate hybrid controls for medium-risk assets where interoperability is required, and (iii) keep lower-risk assets under telemetry-based monitoring. This mapping ensures that QRS values translate into actionable and auditable operational states. The scenario-specific threshold instantiation used for QRS-to-state mapping is presented in Table 5.

In this scenario, thresholds were chosen to classify externally exposed RSA/ECC paths as `MIGRATE`, while keeping medium-risk interoperable assets in `HYBRID` and lower-risk assets in `MONITOR`; institutions may adjust these values under governance without changing the deterministic QRS mechanism.

The QRS value for each asset is computed as a weighted linear combination of four factors:

$$\begin{aligned} QRS = &(w_1 \times \text{Data Longevity}) \\ &+ (w_2 \times \text{Algorithm Vulnerability}) \\ &+ (w_3 \times \text{Key Exposure}) \\ &+ (w_4 \times \text{System Dependence}) \end{aligned} \tag{1}$$

Data Longevity refers to the expected period of confidentiality of data processed by an asset, independent of the cryptographic algorithm employed. Algorithm Vulnerability considers both the basic cryptographic primitive and environment, such as key lifetime, sessions, and possible long-term effects of cryptanalysis. Key Exposure measures the likelihood of key compromise based upon storage, distribution, or usage behaviors, whereas System Dependence examines the operational necessities of the asset in organizational business practices. These are rated from 1 to 10 on a normalized scale.

TABLE 6. Ordinal Scoring Rubric for QRS factors (1–10), used to ensure repeatable scoring across assessors.

Factor	Low (1–3)	High (8–10)
Data Longevity (dl)	Ephemeral/session data; confidentiality needed $<$ 1 year	Long-lived confidentiality need (e.g., identity, medical, legal, regulated); $>$ 5–10 years
Algorithm Vulnerability (av)	PQC-ready (NIST PQC or approved hybrid) with no classical-only fallback	Classical-only public-key (RSA/ECC) in externally exposed paths; long-lived keys/certs
Key Exposure (ke)	Keys HSM-backed; strict rotation; limited distribution; strong separation	Keys widely distributed (files, containers, endpoints); weak rotation; high operator/vendor touchpoints
System Dependence (sd)	Non-critical service; easy rollback/replacement; limited blast radius	Mission-critical/PKI/identity core; high coupling; failure impacts multiple services

TABLE 7. AHP pairwise comparison matrix for QRS factors (dl, av, ke, sd).

	dl	av	ke	sd
dl	1	1	1	2
av	1	1	1	1
ke	1	1	1	1
sd	1/2	1	1	1

2) QRS FACTOR SCORING RUBRIC (1–10 SCALE)

To ensure repeatability across assessors and to avoid subjective scoring drift, the scenario uses an explicit ordinal scoring rubric for each QRS factor. Table 6 defines low and high anchor conditions for the 1–10 scale, enabling consistent scoring across replications.

For transparency of methods and ease of reproduction of the study's results, the weights w_1–w_4 were determined based on AHP as proposed by Saaty. For calculating pairwise comparisons of the relative weight of data longevity, algorithmic vulnerability to attacks, key exposure to attacks, and system dependence on individual parameters of importance, as set forth in NIST SP 800-227 [21] guidelines on risk assessment principles, a weight distribution of (0.30, 0.25, 0.25, 0.20) was determined based on the AHP corresponding to the normalized right-hand side column of a comparison matrix. This aligns with the weight distribution pattern already set forth in Table 7. Sensitivity analysis based on a variation of $\pm$ 20% showed that the institutional ranking using the QRS model was relatively unaffected [19].

3) AHP PAIRWISE MATRIX AND CONSISTENCY

For transparency and to address methodological reproducibility, the AHP weights were derived from an explicit pairwise comparison matrix using Saaty's scale. Table 7 presents a representative matrix consistent with the selected priority pattern. The resulting priority vector is approximately $W \approx (0.298,\ 0.246,\ 0.246,\ 0.210)$ for (dl, av, ke, sd), and the Consistency Ratio is $CR \approx 0.022 < 0.10$, indicating acceptable consistency.

Table 8 illustrates a representative QRS calculation by aggregating weighted risk factors for selected institutional cryptographic assets. The table demonstrates how dimensions such as data longevity, algorithm vulnerability, key exposure, and system dependence are combined using predefined weights to produce deterministic QRS values. These values are not enforcement decisions by themselves; rather, they serve as structured inputs to the subsequent policy mapping and CI/CD validation stages described in Section V.

The QRS outputs generated in this scenario-based validation are subsequently consumed by the CI/CD validation gates defined in Section V to enforce policy-driven allow, restrict, or block decisions while emitting auditable enforcement artifacts.

D. OPERATIONAL VALIDATION PROCESS

A validation exercise, based on scenario evaluation, was conducted in accordance with a sequential approach compatible with the seven stages of the PQC Transition Framework. All stages of this process undertook open-source components in order to provide reproducibility, traceability, and auditability. Procedural consistency, evidentiary development, and alignment, as opposed to cryptographic metrics, received particular attention in this validation exercise.

1) STEP 1 - ENVIRONMENT INITIALIZATION

To provide a reproducible execution environment, Ubuntu 22.04 LTS was chosen, incorporating Docker, Python 3.11, GNU Compiler Collection (GCC) support, and OpenSSL 3.0 with post-quantum cryptography dependencies. This setup offered a predictable and controllable platform to analyze the feasibility of the algorithms and perform the QRS calculation in a monitored and traceable environment.

2) STEP 2 - ASSET INVENTORY AND CLASSIFICATION

A total of six exemplary assets were manually registered within the Snipe-IT inventory, specifying the essential system components such as authentication and application gateways (including the external and IoT gateways), mail servers, database servers, and certificate authorities. A set of records included additional information regarding the used algorithms and the level of exposure. They formed the baseline dataset.

3) STEP 3 - QUANTUM RISK SCORING (QRS)

The implementation of QRS, based on a Python script, used the weighted scoring process identified within QRS Model for all assets that were registered on the system. Exposure factors were assessed through a deterministic

TABLE 8. Sample QRS calculation for institutional assets.

Asset ID	Asset Name	Current Cryptography	Data Longevity ($w_1 = 0.30$)	Algorithm Vulnerability ($w_2 = 0.25$)	Key Exposure ($w_3 = 0.25$)	System Dependence ($w_4 = 0.20$)	QRS Score
A1	Auth-Server	RSA-2048, TLS 1.2	8	9	7	8	8
A2	PKI-Node	RSA-2048, X.509 CA	9	9	7	9	8.5
A3	Database	AES-256 (RSA key wrapping)	9	6	5	8	7.1
A4	Mail Server	RSA-2048, S/MIME	8	9	6	7	7.6
A5	External API Gateway	RSA-2048, TLS 1.2	6	8	8	7	7.2
A6	IoT Gateway	ECDSA-P256, DTLS	4	7	8	5	6

process, ensuring reproducible and consistent results on a normalized scale, as a repeated execution would produce an identical outcome.

4) STEP 4 - ALGORITHM OPERABILITY VERIFICATION

Functional viability for post-quantum algorithms was assessed through use of a liboqs library by compiling and running PQC algorithms known as Kyber (ML-KEM-1024) and Dilithium (ML-DSA-65), and a successful run ensured functionality and post-quantum algorithmic interoperability within an institutional framework. Measurements of post-quantum algorithmic performance were not considered because said information is well-documented within respective NIST directives and pqm4 benchmarks.

5) STEP 5 - CRYPTOGRAPHIC CONFIGURATION VISIBILITY

The visibility of the TLS configuration was tested using the tool Cryptolyzer to assess whether the post-quantum or the hybrid negotiation of the cryptographic schemes was enabled at the simulated institutional endpoints. In the default settings established in this validation process, no parameters related to post-quantum or hybrid key exchange were established, hence establishing a representative pre-transition state. Although there are experimental post-quantum environments with the hybrid model installed within some public infrastructures and browser settings, this does not fall within the usual settings of the institutions. This confirms the usefulness of the model within the context of the pre-activation state of the post-quantum environment. This uniformity of visibility continues with the downgrade-guard approach using the CI/CD process, which will be discussed in Section V.

6) STEP 6 - EVIDENCE CONSOLIDATION AND AUDIT READINESS

All generated artifacts, including configuration files, inventory exports, QRS outputs, and validation logs, were archived within a structured `/evidence/` directory. Artifacts were versioned and labeled by framework phase to preserve full traceability. This consolidation produced an audit-ready evidence bundle demonstrating that each phase of the framework can generate verifiable and repeatable outputs.

Collectively, these steps demonstrate that the PQC Transition Framework can be instantiated end-to-end within an open-source environment, producing consistent and auditable evidence without reliance on proprietary tooling or high-cost infrastructure.

7) OPERATIONAL ENFORCEMENT TESTS (ALLOW/RESTRICT/BLOCK)

To demonstrate practical evaluation beyond narrative description, we executed policy enforcement tests in which QRS-derived policy states were mapped to CI/CD gate outcomes. For each test case, the CI/CD gate consumed `qrs_results.json` and `crypto_config.json`, produced `audit_gate.json`, and emitted an allow, restrict, or block decision. This directly validates that the framework performs enforceable controls and produces verifiable artifacts. Representative enforcement outcomes are summarized in Table 9.

These enforcement tests provide concrete evidence that QRS is not merely a descriptive score but an operational input that deterministically drives deployment controls and generates auditable enforcement records.

E. OBSERVED OUTCOMES

The scenario validation enables the following verifiable outputs that relate to the PQC Transition Framework operation phases, and collectively, the outputs have indicated that readiness for the post-quantum state can be accomplished through procedures that are cost-effective and auditable, even if the institution lacks resources. Every output forms an evidence artifact that identifies the operation phase of the conceptual framework.

The cryptographic asset audit yielded algorithmic dependencies, hidden or legacy components, thereby validating the notion that discovery and categorization processes can be accomplished via open-source CMDB tools. The QRS component provided deterministic values each time it ran, guaranteeing quantifiable levels of exposure to the quantum threat. Deterministic processes fulfill the verifiability requirements expressed within the recommendations outlined in NIST IR 8547 [18], thus ensuring the outcome of risk assessment can be verified.

TABLE 9. Representative enforcement tests demonstrating CI/CD decisions driven by QRS policy states.

TC	Input Condition	Detected Violation	Gate Decision
TC1	Asset A2 flagged as `MIGRATE`; deployment manifest declares RSA-only TLS handshake	Classical-only public-key in a MIGRATE path	**BLOCK**
TC2	Asset A3 flagged as `HYBRID`; manifest allows fallback to classical-only negotiation	Missing hybrid requirement / downgrade-prone configuration	**RESTRICT**
TC3	Asset A6 flagged as `MONITOR`; manifest matches approved baseline with no downgrade indicators	No baseline violation detected	**ALLOW**

TABLE 10. Evidence artifacts produced by the scenario validation.

Phase	Artifact	Description	Verification
P1	`cmdb_snapshot.json`	Exported cryptographic asset inventory with tags and metadata	Schema validation + checksum
P3	`qrs_results.json`	Ordered QRS scores with policy flags	Determinism check (same input → same output)
P5	`audit_gate.json`	CI/CD decision record (ALLOW/RESTRICT/BLOCK) with findings	Pipeline log + hash
P6	`cryptolyzer_report.txt`	Observed TLS configuration visibility and fallback indicators	Re-run against endpoint

Operability tests of the algorithm confirm that the Kyber and Dilithium primes function properly in the context of an institutional setting, thus certifying interoperability between NIST-approved algorithms and open-source toolchains. Performance metrics are not considered in this study. These are sufficiently covered in NIST reports and PQM4 benchmark datasets.

Therefore, the validation process maintained its focus on governance maturity, auditability, and procedural correctness as opposed to cryptographic performance.

Analysis of the visibility of the TLS configuration showed that post-quantum or hybrid negotiation was not being conducted through the default configurations in this particular case, thereby establishing a pre-transition readiness baseline. Although experimental hybrid implementations do currently exist in certain conditions, this observation is a reflection of the applicability of the framework, which relates to institutions that have not implemented PQC within a production setting.

In sum, the inclusion of all outputs into a systematic evidence repository establishes an audit trail that comprehends identification of assets, risk analysis, validation of algorithms, as well as synchronization with the governance process. Each entry in the logs, data, as well as config profiles, becomes an object that can be traced, implying the fitness of the framework's processes from start to finish via tangible evidence. On the whole, the set of aggregate observations tends to prove that the PQC Transition Framework performs the quantum transition more as an institutional transformation than a simple technology swap.

1) EVIDENCE BUNDLE AND REPLICATION STEPS

To support replication and to address validation concerns, all scenario outputs are organized under a phase-labeled evidence taxonomy. Each artifact is verifiable via re-execution and integrity checks (e.g., checksums and version tags), enabling independent reproduction of the enforcement workflow. The complete set of generated artifacts and their verification properties are summarized in Table 10.

Replication follows a minimal sequence: (i) instantiate the toolchain (Snipe-IT, Cryptolyzer, Python QRS engine), (ii) populate the CMDB with the same asset schema, (iii) generate `qrs_results.json`, (iv) run the CI/CD gate using the declared `crypto_config.json`, and (v) verify that `audit_gate.json` is emitted with consistent findings for identical inputs.

F. FUTURE EXPANSION

This particular research points out that the current validation focuses on reproducibility and verification from a governance point of view, although the framework has the potential for extension towards continuous assurance and adaptive monitoring. This kind of development focuses on the importance of real-time observability, autonomous compliance, as well as feedback-driven governance as part of the PQC transition lifecycle.

The future implementation may include the use of Prometheus and Grafana to monitor cryptographic activities, including the usage of the algorithms, rotation of keys, and fallback events. The use of these tools in the CI/CD systems ensures that the continuity of compliance checking occurs, and the whole system becomes self-verifying rather than being reproducibility-based.

There is also an extension that takes into consideration the creation of a Governance Maturity Indicator (GMI) that will be calculated based on quantum-resilient system distributions and incident response metrics. The GMI will therefore create a new dimension of evaluation based on quantum maturity.

In addition to these, cross-sector validation with a focus on areas including finance, healthcare, and energy can be utilized

TABLE 11. Scenario-based maturity-oriented comparison of PQC transition frameworks.

Evaluation Metric	PQCC Roadmap (2025)	PMMP-PQC (2024)	Proposed Framework (2025)
Procedural Depth (clarity of actionable steps)	2 / 5	3 / 5	5 / 5
Automation Integration (CI/CD, telemetry, rollback control)	1 / 5	2 / 5	5 / 5
Audit Traceability (verifiable evidence artifacts)	2 / 5	3 / 5	5 / 5
Governance Coupling (policy-to-execution feedback)	2 / 5	3 / 5	5 / 5
Reproducibility and Transparency (open-source verifiability)	3 / 5	3 / 5	5 / 5

TABLE 12. Qualitative rubric describing the documented procedural, Automation, Audit, and Governance characteristics used to derive the maturity scores summarized.

Evaluation Metric	PQCC Roadmap (2025)	PMMP-PQC (2024)	Proposed Framework (2025)
Procedural Depth	High-level domains, no stepwise execution	Role-based maturity stages	Explicit phase-level operational steps
Automation Integration	Conceptual references only	Partial workflow suggestions	CI/CD-enforced cryptographic validation
Audit Traceability	Documentation-based	Maturity scoring artifacts	Verifiable logs and evidence bundles
Governance Coupling	Advisory alignment	Role-policy linkage	Closed policy–automation feedback loop
Reproducibility and Transparency	Roadmap guidance	Process guidance	Open-source, executable workflows

to test its regulatory flexibility and scalability. In these areas, utilizing the same evidence taxonomy and feedback loops from governance cycles in sector development, the PQC Transition Framework can evolve from a scenario proof-of-concept to a verifiable and sector-independent operational process of quantum-resilient transformation.

G. GENERALIZABILITY AND REPLICATION PLAN

The scenario-based validation in this study was intentionally confined to a public-sector context to preserve reproducibility and governance focus. Nevertheless, the proposed framework is designed to be sector-agnostic and directly replicable across domains such as finance, healthcare, and energy. Each domain can instantiate the same seven-phase process using the identical evidence structure established in this

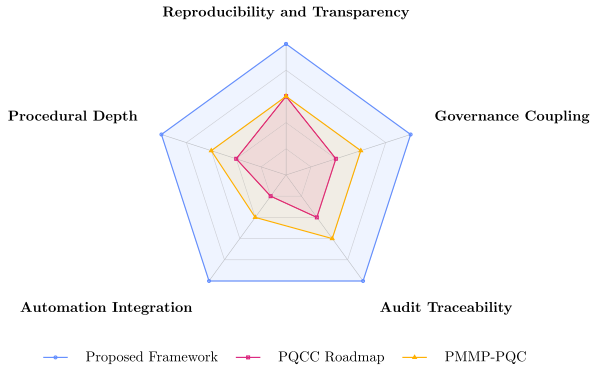

FIGURE 3. Radar chart illustrating normalized operational maturity scores across evaluation dimensions, derived from documented framework characteristics and scenario-based validation evidence. Values reflect relative procedural and governance coverage rather than cryptographic performance.

work, including QRS outputs, CI/CD verification logs, and OPA policy audits, without modification to the underlying methodology.

Replication in other sectors follows an identical validation rubric encompassing cost efficiency, audit traceability, automation integrity, and governance coupling. By preserving the same open-source toolchain and evidence taxonomy, external researchers and institutions can reproduce the full transition workflow on their own infrastructures and compare outcomes using the maturity indicators defined in Table 11. Extending validation beyond the public sector further enables assessment against distinct compliance baselines, such as PCI-DSS in finance or NERC-CIP in energy, demonstrating that the framework's evidence-driven methodology is not bound to a single organizational archetype but supports repeatable and auditable quantum-resilient transformation across heterogeneous regulatory environments.

VII. DISCUSSION

The proposed PQC Transition Framework was developed to explicitly bridge the gap between conceptual policy guidance and enforceable operational execution of post-quantum cryptography. Rather than positioning PQC transition as a planning or compliance exercise, the framework defines deterministic decision paths, CI/CD-enforced controls, and auditable evidence generation mechanisms. Through scenario-based validation, this study demonstrates that reproducible and governance-aligned PQC transition can be achieved using transparent, open-source–compatible operational constructs. Each phase of the framework, from asset discovery to algorithm verification, can be executed sequentially while producing verifiable artifacts, confirming both technical feasibility and governance alignment.

A. FORMAL VERIFICATION AND ASSURANCE CONSIDERATIONS

The current framework ensures operational assurance through evidence-based validation and audit artifacts, and

TABLE 13. Framework novelty and contribution.

Identified Gaps in Prior Work	Contributions of This Study
Conceptual frameworks lack operational detail - NIST IR 8547 and PQCC Roadmap [11] provide high-level guidance but omit step-by-step implementation paths.	Introduces a seven-phase operational framework that translates policy recommendations into executable technical and governance procedures.
Limited integration of open-source tools - Existing models seldom link standards to practical toolchains.	Maps each phase to open-source, reproducible tooling (e.g., Snipe-IT, liboqs, GitLab CI, Prometheus), ensuring transparency and vendor neutrality.
Absence of quantitative risk metrics - Most studies rely on qualitative readiness checklists.	Defines the QRS model, providing measurable prioritization of cryptographic assets and reproducible exposure analysis.
Governance treated as static compliance - Prior work rarely connects technical migration with organizational maturity.	Embeds governance and policy alignment as continuous lifecycle stages, converting compliance into adaptive institutional capability.
Validation limited to performance testing - Earlier demonstrations emphasized benchmark speed over operational reproducibility.	Implements a scenario-based validation focusing on auditability, cost-efficiency, and traceable evidence generation instead of latency metrics.
Lack of cross-sector adaptability - Existing models are confined to specific domains (e.g., healthcare, avionics).	Designs a sector-agnostic framework applicable across public, financial, and critical-infrastructure contexts.

then formal verification can provide a useful extension that could potentially strengthen the guarantees of assurance. In the later stages of conducting research, Tamarin Prover or ProVerif analysis could be used to carry out the verification of confidentiality, authenticity, and downgrade resilience properties of PQC-enabled workflows defined through the framework. These tools enable symbolic reasoning over cryptographic protocols, allowing CI/CD automation logic, governance policies, and QRS-driven decisions to be evaluated against formally defined security goals. Integrating such verification would elevate the framework from empirically validated assurance to provable correctness, reinforcing its trust boundaries and compliance credibility. This direction aligns with the assurance principles outlined in NIST SP 800-227 [21] and complements the existing operational validation by adding mathematically verifiable security guarantees.

B. CONCEPTUAL BOUNDARIES

The framework views the process of post-quantum migration as a lifecycle that does not occur once but instead constitutes an ongoing process. The structure demonstrates a quantitative relationship that exists between cryptographic modernization and governance maturity, such that the process of operational readiness keeps pace with algorithmic resilience. In this case, the process focuses on the attribute of transparency, an attribute that can be viewed from a reproducibility-focused paradigm that aligns with the recommended practices within the PQCC guidelines [11]and the NIST IR 8547 report guidelines on published works such as [18]. While the framework is expressed at a lifecycle level, its phases are defined through executable procedures and enforcement mechanisms rather than abstract conceptual stages.

C. OPERATIONAL LIMITATIONS

The validation process was conducted in the controlled execution environment, which excluded live network telemetry and hybrid TLS negotiation. Although the controlled environment enables the deterministic assessment of governance alignment and enforcement logic, it does not entirely capture the inherent variability associated with production-scale infrastructures.

The scalability of the proposed enforcement model relies on the completeness and accuracy of institutional asset inventories. In the case of distributed environments, incomplete metadata or configuration synchronization can lead to inconsistencies between the policy state derived from the QRS and the actual state of the cryptographic infrastructure. Such inconsistencies can lead to false positive or false negative results during CI/CD validation gate enforcement. Although the deterministic nature of the QRS model ensures reproducible results within the same input conditions, the reliability of the enforcement process remains dependent on the quality of input data and the quality of institutional data governance.

In addition, the model relies on the existence of mature governance infrastructure that can translate risk-based outputs into enforceable deployment controls. Institutions with less mature CI/CD pipelines, configuration management, or compliance monitoring capabilities may require additional integration overhead. Inertia, coordination constraints, and transitional policy conflicts can also affect deployment efficiency in real-world operational environments.

D. FRAMEWORK NOVELTY AND CONTRIBUTION

The novelty of the proposed framework does not lie in cryptographic algorithm design or performance optimization, but in the systematic operationalization of PQC transition. The comparative maturity analysis therefore evaluates procedural completeness, automation capability, auditability, and governance coupling, rather than cryptographic strength or benchmark efficiency.

As presented in Table 11, the comparative maturity scores reflect a structured assessment of procedural coverage,

automation capability, audit traceability, and governance coupling rather than empirical performance advantages. The maximum scores assigned to the proposed framework indicate completeness of lifecycle coverage within the defined evaluation dimensions, not superiority in cryptographic strength or implementation efficiency. All values represent relative maturity levels normalized on a 0–5 scale and derived from documented framework characteristics and reproducibility evidence in Section VI, rather than experimental benchmarks or quantitative performance measurements.

Table 12 summarizes the qualitative evaluation rubric underlying these scores, while Figure 3 provides a graphical representation of the same maturity dimensions.

Table 13 synthesizes the conceptual and operational contributions of this study by mapping persistent gaps in prior PQC transition models to the specific mechanisms introduced by the proposed framework.

E. FUTURE RESEARCH DIRECTIONS

Future research will extend the framework toward hybrid TLS environments, continuous cryptographic monitoring, and adaptive governance metrics that support long-term institutional assurance. Integrating QRS outputs and CI/CD validation evidence into a GMI will enable dynamic quantification of organizational readiness and resilience over time. Cross-sector collaboration with finance, energy, and healthcare institutions will further provide empirical data for evaluating scalability, regulatory adaptability, and domain-specific constraints, extending the framework's relevance for multi-domain PQC transition planning.

VIII. CONCLUSION

The findings of this study demonstrate that PQC transition can be addressed as an operationally executable and auditable institutional process rather than a purely conceptual planning exercise. By focusing on governance alignment, deterministic decision-making, and reproducible enforcement mechanisms, the proposed framework moves PQC transition beyond static compliance models.

A key outcome of this work is the integration of a deterministic QRS model, calibrated using the AHP, with cryptography-aware CI/CD validation gates. This integration enables consistent prioritization of cryptographic assets and the transformation of policy intent into enforceable deployment decisions, while producing verifiable and immutable audit artifacts.

Through scenario-based validation implemented with open-source tooling, the framework was shown to support end-to-end executability, auditability, and governance traceability without relying on empirical cryptographic performance benchmarks. These results indicate that quantum readiness can be advanced through measurable and repeatable operational mechanisms, framing PQC transition as a continuous lifecycle capability rather than a one-time technical replacement.

REFERENCES

[1] L. Chen, S. Jordan, Y. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, *Report on Post-Quantum Cryptography*, Standard NIST IR 8105, 2016, doi: 10.6028/NIST.IR.8105.

[2] *Module-Lattice-Based Key-Encapsulation Mechanism Standard*, Standard FIPS 203, 2024, doi: 10.6028/NIST.FIPS.203.

[3] *Module-Lattice-Based Digital Signature Standard*, Standard FIPS 204, 2024, doi: 10.6028/NIST.FIPS.204.

[4] *Stateless Hash-Based Digital Signature Standard*, Standard FIPS 205, National Institute of Standards and Technology, 2024, doi: 10.6028/NIST.FIPS.205.

[5] D. Moody, G. Alagic, J. M. Alperin-Sheriff, D. C. Apon, D. A. Cooper, Q. H. Dang, and R. A. Perlner, *Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process*, Standard NIST IR 8240, 2019, doi: 10.6028/NIST.IR.8240.

[6] G. Alagic, J. Alperin-Sheriff, D. Apon, D. Cooper, Q. Dang, J. Kelsey, Y.-K. Liu, C. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, and D. Smith-Tone, *Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process*, Standard NIST IR 8309, 2020, doi: 10.6028/NIST.IR.8309.

[7] G. Alagic, D. Apon, D. Cooper, Q. Dang, T. Dang, J. Kelsey, J. Lichtinger, Y.-K. Liu, C. Miller, D. Moody R. Peralta, R. Perlner, A. Robinson, and D. Smith-Tone, "*Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process*, Standard NIST IR 8413-upd1, 2022, doi: 10.6028/NIST.IR.8413.

[8] G. Alagic, M. Bros, P. Ciadoux, D. Cooper, Q. Dang, T. Dang, J. Kelsey, J. Lichtinger, Y.-K. Liu, C. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, H. Silberg, D. Smith-Tone, and N. Waller, *Status Report on the First Round of the Additional Digital Signature Schemes for the NIST Post-Quantum Cryptography Standardization Process*, Standard NIST IR 8528, 2024, doi: 10.6028/NIST.IR.8528.

[9] G. Alagic, M. Bros, P. Ciadoux, D. Cooper, Q. Dang, T. Dang, J. Kelsey, J. Lichtinger, Y.-K. Liu, C. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, H. Silberg, D. Smith-Tone, and N. Waller, *Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process*, Standard NIST IR 8545, 2025, doi: 10.6028/NIST.IR.8545.

[10] R. Perlner. (2025). *FIPS 206: FN-DSA (Falcon)*. Accessed: Oct. 26, 2025. [Online]. Available: https://csrc.nist.gov/presentations/2025/fips-206-fn-dsa-falcon

[11] Post-Quantum Cryptography Coalition (PQCC). (May 2025). *Post-Quantum Cryptography Migration Roadmap*. [Online]. Available: https://pqcc.org/wp-content/uploads/2025/05/PQC-Migration-Roadmap-PQCC-2.pdf

[12] K. F. Hasan, L. Simpson, M. A. R. Baee, C. Islam, Z. Rahman, W. Armstrong, P. Gauravaram, and M. McKague, "A framework for migrating to post-quantum cryptography: Security dependency analysis and case studies," *IEEE Access*, vol. 12, pp. 23427–23450, 2024, doi: 10.1109/ACCESS.2024.3360412.

[13] C. Näther, D. Herzinger, S.-L. Gazdag, J.-P. Steghöfer, S. Daum, and D. Loebenberger, "Migrating software systems toward post-quantum cryptography–A systematic literature review," *IEEE Access*, vol. 12, pp. 132107–132126, 2024, doi: 10.1109/ACCESS.2024.3450306.

[14] M. I. García-Cid, M.-A. Kourtis, D. Domingo, N. Tcholtchev, E. K. Markakis, M. Niemiec, J. Faba, L. Ortiz, V. Martín, D. López, G. Xilouris, M. Gagliardi, J. González, M. García, G. Comande, and N. Stoianov, "PQ-REACT: Post quantum cryptography framework for energy aware contexts," in *Proc. 19th Int. Conf. Availability, Rel. Secur.*, Jul. 2024, pp. 1–7, doi: 10.1145/3664476.3670868.

[15] Z. G. Al-Mekhlaf, M. A. Saare, J. M. H. Altmemi, M. A. Al-Shareeda, B. A. Mohammed, G. Alshammari, R. Alrashdi, Y. A. Alkhabra, and I. Alreshidi, "A quantum-resilient lattice-based security framework for Internet of Medical Things in healthcare systems," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 37, no. 6, p. 126, Aug. 2025, doi: 10.1007/s44443-025-00140-0.

[16] L. Jančiūtė, "Cybersecurity in the financial sector and the quantum-safe cryptography transition: In search of a precautionary approach in the EU digital operational resilience act framework," *Int. Cybersecurity Law Rev.*, vol. 6, no. 2, pp. 145–154, Jun. 2025, doi: 10.1365/s43439-025-00135-7.

[17] T. G. Tan, P. Szalachowski, and J. Zhou, "Challenges of post-quantum digital signing in real-world applications: A survey," *Int. J. Inf. Secur.*, vol. 21, no. 4, pp. 937–952, Aug. 2022, doi: 10.1007/s10207-022-00587-6.

[18] D. Moody, R. Perlner, A. Regenscheid, A. Robinson, and D. Cooper, *Transition to Post-Quantum Cryptography Standards*, Standard NIST IR 8547-ipd, 2024, doi: 10.6028/NIST.IR.8547.ipd.

[19] R. W. Saaty, "The analytic hierarchy process—What it is and how it is used," *Math. Model.*, vol. 9, nos. 3–5, pp. 161–176, 1987, doi: 10.1016/0270-0255(87)90473-8.

[20] J. Dey and R. Dutta, "Progress in multivariate cryptography: Systematic review, challenges, and research directions," *ACM Comput. Surv.*, vol. 55, no. 12, pp. 1–34, Dec. 2023, doi: 10.1145/3571071.

[21] G. Alagic, H. Barker, L. Chen, D. Moody, A. Robinson, H. Silberg, and N. Waller, *Recommendations for Key-Encapsulation Mechanisms*, Standard NIST SP 800-227-ipd, 2025, doi: 10.6028/NIST.SP.800-227.ipd.

[22] J. O. del Moral, A. deMarti iOlius, G. Vidal, P. M. Crespo, and J. E. Martinez, "Cybersecurity in critical infrastructures: A post-quantum cryptography perspective," *IEEE Internet Things J.*, vol. 11, no. 18, pp. 30217–30244, Sep. 2024, doi: 10.1109/JIOT.2024.3410702.

[23] K. Varner, W. Zaeske, S. Friedrich, A. Kaiser, and A. Bowman, "Agile, post-quantum secure cryptography in avionics," *CEAS Aeronaut. J.*, May 2025, doi: 10.1007/s13272-025-00806-5.

[24] N. Von Nethen, A. Wiesmaier, N. Alnahawi, and J. Henrich, "PMMP-PQC migration management process," in *Proc. Eur. Interdiscipl. Cybersecurity Conf.*, Jun. 2024, pp. 144–154, doi: 10.1145/3655693.3655719.

[25] J. Hekkala, M. Muurman, K. Halunen, and V. Vallivaara, "Implementing post-quantum cryptography for developers," *Social Netw. Comput. Sci.*, vol. 4, no. 4, p. 365, Apr. 2023, doi: 10.1007/s42979-023-01724-1.

[26] M. A. Al-Shareeda, A. A. H. Ghadban, A. A. H. Glass, E. M. A. Hadi, and M. A. Almaiah, "Efficient implementation of post-quantum digital signatures on raspberry pi," *Discover Appl. Sci.*, vol. 7, no. 6, p. 597, Jun. 2025, doi: 10.1007/s42452-025-07201-z.

[27] S. Bhasin, F. De Santis, and F. Regazzoni, "Special issue on post-quantum cryptography for embedded systems," *ACM Trans. Embedded Comput. Syst.*, vol. 23, no. 2, pp. 1–3, Mar. 2024, doi: 10.1145/3641852.

[28] Netherlands Organisation for Applied Scientific Research (TNO). (2024). *Post-Quantum Cryptography Migration Handbook*. Accessed: Dec. 18, 2025. [Online]. Available: https://publications.tno.nl/publication/34643386/fXcPVHsX/TNO-2024-pqc-en.pdf

[29] A. Braeken, A. K. Yadav, and J. Munilla, "A practical transition to post-quantum security in 5G-AKA," *IEEE Trans. Inf. Forensics Security*, vol. 20, pp. 13071–13084, 2025, doi: 10.1109/TIFS.2025.3632234.

[30] L. P. Fraile, C. Koulamas, and A. P. Fournaris, "Reinventing EDHOC for the post-quantum era," *IEEE Access*, vol. 13, pp. 196622–196640, 2025, doi: 10.1109/ACCESS.2025.3633843.

[31] Y. Wang and E. Shahril Ismail, "A review on the advances, applications, and future prospects of post-quantum cryptography in blockchain and IoT," *IEEE Access*, vol. 13, pp. 112962–112977, 2025, doi: 10.1109/ACCESS.2025.3584473.

[32] H. Nguyen, S. Huda, Y. Nogami, and T. T. Nguyen, "Security in post-quantum era: A comprehensive survey on lattice-based algorithms," *IEEE Access*, vol. 13, pp. 89003–89024, 2025, doi: 10.1109/ACCESS.2025.3571307.

[33] H.-Y. Kwon, I. Bajuna, and M.-K. Lee, "Compact hybrid signature for secure transition to post-quantum era," *IEEE Access*, vol. 12, pp. 39417–39429, 2024, doi: 10.1109/ACCESS.2024.3374645.

[34] H. Nejatollahi, N. Dutt, S. Ray, F. Regazzoni, I. Banerjee, and R. Cammarota, "Post-quantum lattice-based cryptography implementations: A survey," *ACM Comput. Surv.*, vol. 51, no. 6, pp. 1–41, Nov. 2019, doi: 10.1145/3292548.

[35] C. R. García, A. C. Aguilera, C. Stan, J. J. V. Olmos, S. Rommel, and I. T. Monroy, "Enhanced network security protocols for the quantum era: Combining classical and post-quantum cryptography, and quantum key distribution," *IEEE J. Sel. Areas Commun.*, vol. 43, no. 8, pp. 2765–2781, Aug. 2025, doi: 10.1109/JSAC.2025.3568011.

[36] N. Aquina, B. Cimoli, S. Das, K. Hövelmanns, F. J. Weber, C. Okonkwo, S. Rommel, B. Škorić, I. Tafur Monroy, and S. Verschoor, "A critical analysis of deployed use cases for quantum key distribution and comparison with post-quantum cryptography," *EPJ Quantum Technol.*, vol. 12, no. 1, p. 51, Dec. 2025, doi: 10.1140/epjqt/s40507-025-00350-5.

[37] K. K. Singamaneni, "A novel lightweight hybrid cryptographic framework for secure smart card operations," *EURASIP J. Inf. Secur.*, vol. 2025, no. 1, p. 19, May 2025, doi: 10.1186/s13635-025-00204-8.

[38] L. Meng, Y. Fu, F. Zheng, M. Wang, Z. Ma, J. Dong, and J. Lin, "HTM-PQC: Hardening cryptography keys under the trend of post-quantum cryptography migration on industrial internet," *IEEE Trans. Ind. Informat.*, vol. 21, no. 4, pp. 3504–3514, Apr. 2025, doi: 10.1109/TII.2025.3528582.

[39] I. Kong, M. Janssen, and N. Bharosa, "Challenges in the transition towards a quantum-safe government," in *Proc. 23rd Annu. Int. Conf. Digit. Government Res.*, Jun. 2022, pp. 282–292, doi: 10.1145/3543434.3543644.

[40] *Secure Software Development Framework (SSDF) Version 1.1 (NIST SP 800-218)*, Standard NIST SP 800-218), 2022, doi: 10.6028/NIST.SP.800-218.

[41] *Snipe-IT—Open Source Asset Management System*. Accessed: Oct. 30, 2025. [Online]. Available: https://github.com/grokability/snipe-it

[42] *Cryptolyzer—TLS and Cryptographic Protocol Analyzer*. Accessed: Oct. 30, 2025. [Online]. Available: https://github.com/c0r0n3r/cryptolyzer

[43] OWASP Foundation. *Dependency-Track—Software Composition Analysis Platform*. Accessed: Oct. 30, 2025. [Online]. Available: https://github.com/DependencyTrack/dependency-track

[44] Open Quantum Safe Project. *LIBOQS—C Library for Quantum-Safe Cryptographic Algorithms*. Accessed: Oct. 30, 2025. [Online]. Available: https://github.com/open-quantum-safe/liboqs

[45] *PQM4—Benchmarking and Implementation Framework for Post-Quantum Cryptography on ARM Cortex-M4*. Accessed: Oct. 30, 2025. [Online]. Available: https://github.com/mupq/pqm4

[46] *GitLab CI/CD—Continuous Integration and Deployment Service*. Accessed: Oct. 30, 2025. [Online]. Available: https://gitlab.com/gitlab-org/gitlab

[47] *Prometheus—Monitoring and Alerting Toolkit*. Accessed: Oct. 30, 2025. [Online]. Available: https://github.com/prometheus/prometheus

[48] *Grafana – Open Source Analytics and Monitoring Platform*. Accessed: Oct. 30, 2025. [Online]. Available: https://github.com/grafana/grafana

[49] *OPA—Open Policy Agent: Policy-as-Code Engine for Cloud-Native Environments*. Accessed: Oct. 30, 2025. [Online]. Available: https://github.com/open-policy-agent/opa

BERAT BIRGIN received the B.S. degree in software engineering from Kırklareli University, Türkiye, in 2024. He is currently pursuing the M.S. degree in cybersecurity with FMV Işık University, Istanbul. His current research interests include post-quantum cryptography transition frameworks, cryptographic risk modeling, and secure automation in cloud environments.

BARIS CELIKTAS received the B.S. degree in systems engineering from the National Defense University, in 2008, the M.S. degree in applied informatics from Istanbul Technical University, in 2018, and the Ph.D. degree in cybersecurity engineering and cryptography from the Institute of Informatics, Istanbul Technical University, in 2022. He is currently an Assistant Professor with the Computer Science Engineering Department and the Director of the Cybersecurity Graduate Program, Işık University. In addition, he is also a Cybersecurity Consultant and an Architect, specializing in enterprise cybersecurity and cryptography solutions, cloud security, risk management, and governance. He holds several industry-recognized certifications, including CISSP, CCSP, CISM, CISA, CRISC, AAIA, SSCP, CCNP, Security+, CySA+, C|EH, and ISO/IEC 27001, 22301, 20000, 27701, and 42001 Lead Auditor/Lead Implementer credentials, and GDPR DPO and NIST cybersecurity consulting credentials. His research interests include cybersecurity, network security, cloud computing, cryptography, malware analysis, risk management, and security applications.