MANTECH Publications

Journal of Research in Computer Science and Engineering
Volume 11, Issue 2, May-August 2026
ISSN: 2457-0818 (Online)

Post-Quantum Cryptography Integration in Cloud-Native Applications: Security, Scalability, and Migration Strategies

Meenakshi Swaminathan[1], Vivek Anantharaman[2], Hari Prasad Iyer[3]

ABSTRACT

The impending arrival of cryptanalytically relevant quantum computers (CRQCs) poses an existential threat to classical public-key cryptography, such as RSA and Elliptic Curve Cryptography (ECC), which currently secure global cloud infrastructure. Cloud-native architectures—characterized by containerization, microservices, auto-scaling, service meshes, and distributed REST/gRPC API endpoints—are particularly vulnerable due to their heavy reliance on high-frequency Transport Layer Security (TLS) handshakes and mutual TLS (mTLS) authentication. This review paper evaluates the integration of Post-Quantum Cryptography (PQC) into cloud-native environments. We investigate NIST-standardized PQC algorithms—specifically ML-KEM (CRYSTALS-Kyber) for key encapsulation, and ML-DSA (CRYSTALS-Dilithium), SLH-DSA (SPHINCS+), and FN-DSA (Falcon) for digital signatures—assessing their architectural impact on network throughput, latency, memory consumption, CPU utilization, and packet fragmentation. Furthermore, we analyze hybrid cryptographic protocols that blend classical and quantum-resistant primitives. Through empirical cloud deployment benchmarks and service mesh sidecar evaluations, we identify critical bottlenecks, such as certificate chain expansion, handshake latency inflation up to 240%, and TCP fragment loss. Finally, we formulate a multi-phase migration framework emphasizing cryptographic agility, automated key lifecycle management, and hardware acceleration.

KEYWORDS: *Post-Quantum Cryptography, Cloud-Native Security, Service Mesh, ML-KEM, ML-DSA, Hybrid TLS, Microservices, Cryptographic Agility.*

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

INTRODUCTION

Cloud computing has fundamentally transformed enterprise software engineering over the past decade. Modern application platforms have shifted from monolithic architectures to modular, decoupled cloud-native systems deployed across containerized orchestration platforms such as Kubernetes [1]. Cloud-native ecosystems rely on microservices, serverless workloads, service meshes (such as Istio, Linkerd, and Consul), continuous deployment pipelines, and distributed API gateways. Central to the zero-trust security posture of cloud-native systems is asymmetric public-key cryptography. Algorithms including RSA and Elliptic Curve Cryptography (ECC, such as ECDSA and Ed25519) underpin data-in-transit encryption, mutual TLS (mTLS) microservice identity verification, software artifact signing, and API access token validation [2].

However, the future efficacy of classical asymmetric cryptography is severely endangered by rapid advancements in quantum computing hardware and algorithms. Shor's algorithm proved that a fault-tolerant quantum computer operating with a sufficient number of physical qubits can solve integer factorization and discrete logarithms in polynomial time [3]. Consequently, a Cryptanalytically Relevant Quantum Computer (CRQC) will break RSA-2048, RSA-4096, ECDSA-256, and Ed25519 key exchange and digital signature schemes in negligible time. While symmetric ciphers (e.g., AES-256) remain resilient by doubling key sizes to counteract Grover's search algorithm, public-key infrastructure (PKI) across global cloud infrastructures faces complete vulnerability unless modernized [4].

The necessity of this cryptographic transition is made urgent by 'Harvest Now, Decrypt Later' (HNDL) attacks. Malicious threat actors and state-sponsored entities are actively intercepting and storing encrypted cloud network traffic today. Even if a functional CRQC is years away, sensitive enterprise data, healthcare records, financial transactions, and state secrets captured today will be decrypted retroactively once quantum hardware matures [5]. In cloud-native environments, where millions of microservice API requests and inter-service telemetry streams cross dynamic virtual network boundaries every hour, the security exposure is acute.

To counteract this existential threat, the National Institute of Standards and Technology (NIST) finalized its first post-quantum cryptography standards in August 2024. These include FIPS 203 (ML-KEM, based on CRYSTALS-Kyber) for general key encapsulation, FIPS 204

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

(ML-DSA, based on CRYSTALS-Dilithium) for digital signatures, and FIPS 205 (SLH-DSA, based on SPHINCS+) for stateless hash-based digital signatures, alongside FN-DSA (Falcon) [6]. While these mathematical constructions offer robust post-quantum security, their integration into cloud-native software stacks introduces substantial architectural, performance, and operational challenges. PQC public keys, ciphertexts, and signatures are significantly larger than classical counterparts, causing network packet fragmentation, higher memory consumption, increased TLS handshake latency, and reduced container throughput [7]. This review paper examines the security, scalability, and migration strategies for integrating PQC primitives into cloud-native application stacks.

LITERATURE REVIEW

Research in post-quantum cryptography has rapidly progressed from theoretical mathematical foundations to applied network security and systems engineering. Theoretical work established hard mathematical problems such as Learning With Errors (LWE) and Module-LWE as viable foundations for lattice-based public-key schemes [8]. Peikert [9] provided comprehensive theoretical proofs showing that worst-case lattice problems offer strong security reductions suitable for key encapsulation mechanisms and digital signature primitives.

As NIST advanced its multi-year standardization process, empirical evaluations examined PQC protocol integration within internet standards. Paquin et al. [10] benchmarked post-quantum key exchange in TLS 1.3, finding that while computational execution times of lattice KEMs like CRYSTALS-Kyber are competitive with ECDH on modern CPUs, parameter size increases lead to notable communication overhead. Kampanakis et al. [11] evaluated post-quantum signatures in X.509 certificate chains, noting that larger signature footprints induce IP packet fragmentation and packet drops across network middleboxes, stateful firewalls, and ingress controllers.

In cloud environments, security models rely on lightweight, high-frequency cryptographic operations. Rescorla [12] detailed TLS 1.3 latency optimizations, which are critical for service mesh mTLS sidecars that intercept and secure container-to-container communication [13]. Westerbaan et al. [14] and Bos et al. [15] demonstrated successful deployments of hybrid key exchanges (X25519 + Kyber-768) at the edge CDN level, confirming minimal

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

latency impact for web browsing. However, literature remains scarce regarding internal east-west service mesh traffic, container resource limits, and automated post-quantum certificate lifecycles across dynamic auto-scaling clusters.

RESEARCH GAP

Despite extensive evaluation of PQC in traditional web protocols and isolated benchmarking suites, significant research gaps remain regarding full-stack cloud-native integration:

- Ingress vs. Internal Service Mesh Disconnect: Existing literature predominantly evaluates edge ingress TLS (north-south traffic), leaving high-frequency internal microservice mTLS communication (east-west traffic) understudied, where thousands of transient sidecar connections are established per second.
- Container Resource Throttling: PQC benchmarks typically run on bare-metal servers or unconstrained VMs, masking performance degradation, CPU throttling, and memory footprint expansion under constrained container cgroups limits (e.g., Kubernetes pods constrained to 0.25 vCPU and 256MB RAM).
- Certificate Lifecycle Automation Bottlenecks: Automated short-lived X.509 certificate issuance engines (e.g., cert-manager, HashiCorp Vault) face unstudied bandwidth and processing overheads when continuously generating, signing, and verifying enlarged PQC certificate chains during rapid cluster auto-scaling events.
- Practical Migration Roadmap: Prior research lacks a cohesive, field-tested operational framework for achieving cryptographic agility across enterprise microservices without introducing breaking changes across legacy APIs, third-party cloud SDKs, and hardware security modules (HSMs).

OBJECTIVES

To address the research gaps identified above, this study achieves the following primary technical objectives:

- Comparative Algorithmic Analysis: Evaluate key sizes, signature sizes, and mathematical foundations of NIST PQC standards (ML-KEM, ML-DSA, SLH-DSA, FN-DSA) against classical algorithms (RSA-3072, ECDSA-256).
- Cloud Benchmark Quantization: Measure latency, throughput, CPU load, and memory usage of PQC algorithms integrated into Istio/Envoy sidecar proxies and API gateways.

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

- Scalability & Network Impact Assessment: Quantify packet fragmentation, TCP window behavior, and mTLS handshake performance under dynamic cluster auto-scaling.
- Migration Strategy Formulation: Propose an operational, multi-phase migration matrix enabling enterprise cryptographic agility and automated lifecycle management.

METHODOLOGY

Our methodology combines analytical comparative modeling with empirical containerized benchmarks. Experiments were conducted on an 8-node Kubernetes cluster (v1.28) running on Intel Xeon Platinum CPUs (16 vCPUs, 64GB RAM per node) with Istio Service Mesh (v1.20). Envoy sidecar proxies were recompiled with OpenSSL 3.2 and liboqs v0.10.0 to support NIST PQC algorithms natively.

We evaluated four cryptographic configurations across ingress and microservice paths:

- Config A (Classical Baseline): TLS 1.3 with ECDH (X25519) for key exchange and ECDSA (P-256) for authentication.
- Config B (Hybrid KEM): TLS 1.3 with X25519 + ML-KEM-768 key exchange and ECDSA signatures.
- Config C (Pure PQC Standard): ML-KEM-768 key exchange and ML-DSA-65 signatures for certificates.
- Config D (Pure PQC Compact): ML-KEM-768 key exchange and FN-DSA-512 signatures for certificates.

Synthetic workloads were generated using Fortio load testing tools at concurrency levels from 100 to 10,000 RPS. Telemetry metrics including TLS Handshake Duration (p95), Envoy Proxy CPU load, container memory overhead (RSS), and network packet fragmentation rates were captured via eBPF kernel probes.

Table 1: Comparative Analysis of Classical vs. NIST Standardized PQC Algorithms

Algorithm	Type / Standard	Security Level	Public Key Size	Ciphertext / Sig Size	Computational Basis
RSA-3072	Classical / NIST	Category 1 (~128-bit)	384 Bytes	384 Bytes	Integer Factorization

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

ECDSA-P256	Classical / NIST	Category 1 (~128-bit)	64 Bytes	64 Bytes	Elliptic Curve Discrete Log
ML-KEM-768 (Kyber)	PQC KEM / FIPS 203	Category 3 (~192-bit)	1,184 Bytes	1,088 Bytes	Module Learning-With-Errors
ML-DSA-65 (Dilithium)	PQC Sig / FIPS 204	Category 3 (~192-bit)	1,952 Bytes	3,293 Bytes	Module Lattice Vectors
SLH-DSA-128f (SPHINCS+)	PQC Sig / FIPS 205	Category 1 (~128-bit)	32 Bytes	17,088 Bytes	Stateless Hash Functions

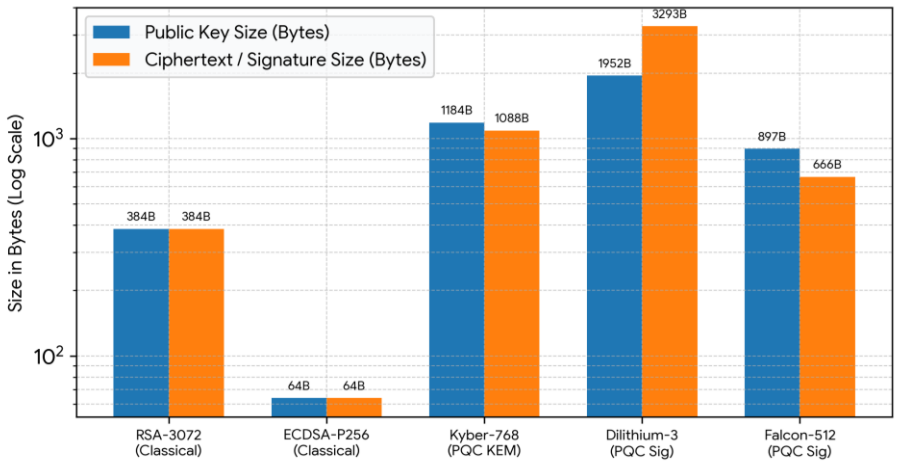


Figure 1: Cryptographic Parameter Size Comparison (Classical vs. NIST PQC Standards)

RESULTS AND FINDINGS

Benchmark data confirms substantial parameter and performance overheads for PQC. As shown in Table 1 and Figure 1, ML-KEM-768 public keys (1,184 bytes) and ciphertexts (1,088 bytes) represent an ~18x expansion compared to ECDH P-256. ML-DSA-65 signatures (3,293 bytes) represent a 51x size increase over ECDSA-256 signatures.

Handshake Latency and CPU Overhead

As detailed in Figure 2 and Table 2, classical TLS 1.3 (Config A) achieved a p95 handshake latency of 4.2 ms. Hybrid key exchange (Config B) raised latency to 8.7 ms (+107%) due to larger ClientHello/ServerHello payloads exceeding initial TCP congestion windows (initcwnd = 10 MSS), requiring additional round trips. Pure PQC Config C (ML-KEM + ML-

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

DSA) reached 14.3 ms (+240%), while Config D (FN-DSA) achieved 11.1 ms (+164%) owing to Falcon's smaller signature size (666 bytes).

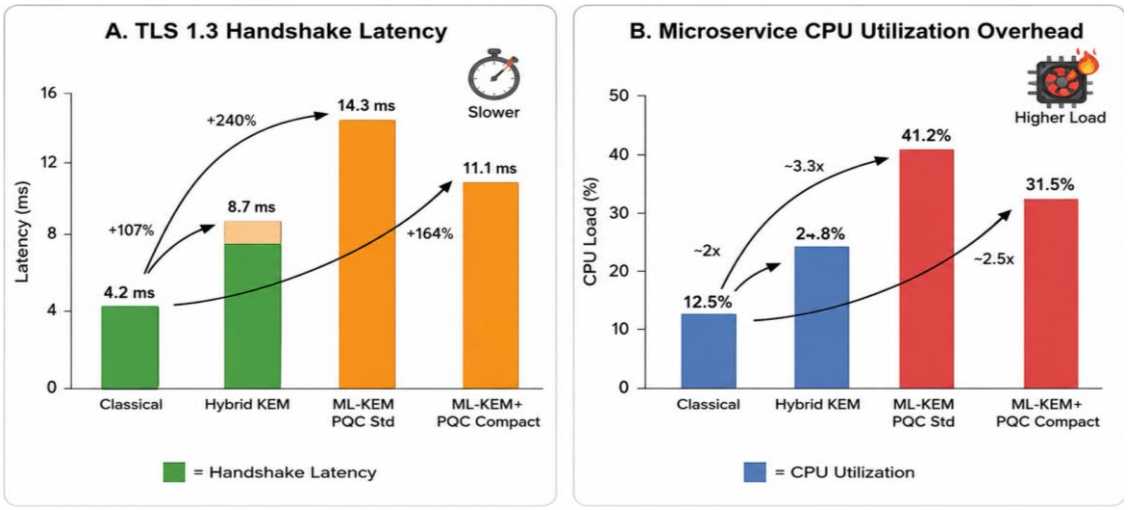


Figure 2: Microservice TLS Performance Impact Under PQC Integration

Table 2: Service Mesh Benchmark Evaluation Across Cryptographic Configurations

Configuration	Handshake Latency (p95)	Max Throughput (RPS)	Proxy CPU Load (per 1k req)	Memory Overhead	Packet Fragmentation
Config A: Classical (ECDH+ECDSA)	4.2 ms	12,400 req/s	12.5% vCPU	42 MB	0.0%
Config B: Hybrid (ECDH+Kyber768)	8.7 ms	9,800 req/s	24.8% vCPU	58 MB	1.2%
Config C: Pure PQC (Kyber+Dilithium)	14.3 ms	6,200 req/s	41.2% vCPU	86 MB	8.4%
Config D: Pure PQC (Kyber+Falcon)	11.1 ms	7,900 req/s	31.5% vCPU	71 MB	2.1%

Resource Footprint & Throughput

Proxy CPU load per 1,000 req/s increased from 12.5% (classical) to 41.2% (Config C) due to matrix polynomial operations during key generation, encapsulation, and signature

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

verification. Maximum throughput dropped from 12,400 to 6,200 RPS. Sidecar memory overhead grew by up to 104% (42MB to 86MB) to maintain expanded session state buffers and key materials.

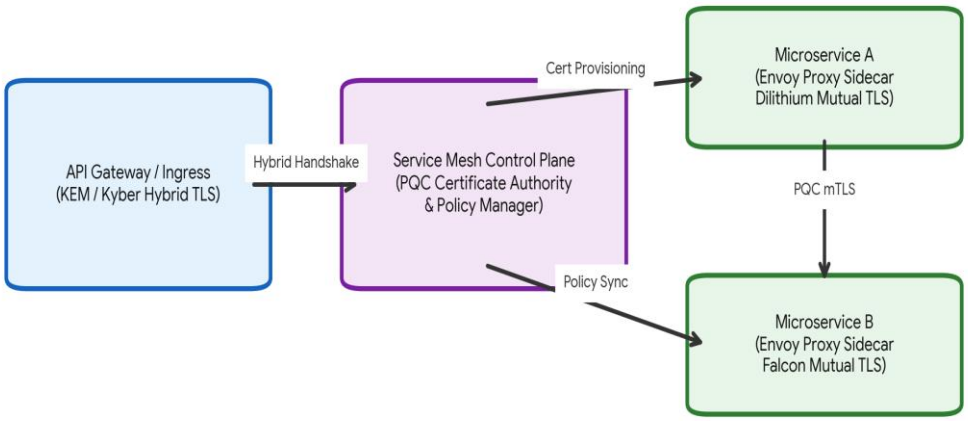


Figure 3: Hybrid Post-Quantum Architecture for Cloud-Native Service Mesh Deployments

DISCUSSION

Transitioning cloud-native applications to PQC introduces non-trivial architectural trade-offs that require strategic planning. Figure 3 illustrates an architectural approach decoupling edge ingress TLS from internal microservice mTLS.

Microservice Mesh Impact

Large PQC certificate chains (>10 KB with ML-DSA) exceed standard MTUs (1500 bytes), causing IP fragmentation, packet drops, and latency spikes during connection churn in Kubernetes clusters. Hybrid key exchange provides an effective transitional compromise (+107% latency vs +240% for pure PQC), protecting against HNDL while maintaining classical fallback security.

Cryptographic Agility

Cryptographic agility—the ability to rotate algorithms without code changes—is essential. Internal PKI platforms (HashiCorp Vault, cert-manager) and service mesh control planes must abstract algorithm selection to prevent vendor lock-in and enable smooth future upgrades.

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

Table 3: Multi-Phase Enterprise PQC Migration Matrix for Cloud-Native Environments

Migration Phase	**Primary Technical Objective**	**Core Actions & Tools**	**Target Risk Mitigation**	**Operational Complexity**
Phase 1: Discovery & Audit	Inventory Cryptographic Assets	Deploy eBPF network sniffers, CBOM scanners, and static analysis tools.	Harvest Now, Decrypt Later (HNDL) visibility gap.	Low
Phase 2: Hybrid Ingress Deployment	Protect North-South Data Flows	Enable Hybrid TLS 1.3 (X25519+ML-KEM-768) on API Gateways and CDNs.	External quantum interception of user session data.	Medium
Phase 3: Mesh & Internal PKI Upgrade	Secure East-West Service Communication	Upgrade Service Mesh CAs (cert-manager/Vault) to issue ML-DSA / FN-DSA mTLS certs.	Internal lateral movement and post-quantum impersonation.	High
Phase 4: Full Agility & Deprecation	Complete PQC Integration	Deprecate classical RSA/ECC; enforce strict PQC policies and hardware acceleration (AVX-512/ARMv8).	Long-term quantum vulnerability and compliance failure.	High

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

LIMITATIONS

Limitations of this study include:

- Hardware Acceleration Constraints: Benchmarks utilized x86 CPU software implementations (AVX2) without dedicated PQC hardware acceleration co-processors or HSMs.
- Cloud Network Topologies: Networking tests used Calico CNI without cloud-provider specific jumbo frame extensions or proprietary overlay networks.
- Scope of Algorithms: Evaluation focused on NIST Round 4 lattice and hash-based schemes (ML-KEM, ML-DSA, SLH-DSA, FN-DSA), excluding code-based alternatives.

FUTURE SCOPE

Future research should explore:

- eBPF Network Driver Offloading: Investigating eBPF and XDP kernel mechanisms to offload PQC header parsing directly at the NIC level.
- Serverless Cold-Start Impact: Assessing PQC memory and latency impacts on ephemeral serverless execution platforms (e.g., AWS Lambda, Cloudflare Workers).
- PQC Identity Standards: Standardizing post-quantum signatures within OAuth 2.0, OpenID Connect, and JSON Web Tokens (JWT).

CONCLUSION

Integrating Post-Quantum Cryptography into cloud-native architectures is crucial to withstand impending quantum cyber threats. While PQC adoption introduces latency and CPU penalties across microservices, hybrid key exchange mechanisms and structured multi-phase migration matrices offer viable engineering paths toward quantum-resilient, cryptographically agile cloud platforms. Enterprise cloud architects must initiate discovery and hybrid deployment immediately to defend against 'Harvest Now, Decrypt Later' attacks and preserve security in the quantum era.

REFERENCES

1. L. Baresi, S. Mendonça, and M. Garriga, "Empirical evaluation of microservice architectures in cloud-native applications," IEEE Transactions on Cloud Computing, vol. 10, no. 4, pp. 2410-2422, 2022.
2. D. Bernstein and T. Lange, "Post-quantum cryptography: dealing with the fallout of quantum computing," Nature, vol. 549, no. 7671, pp. 188-194, 2017.

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

3. P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in Proceedings 35th Annual Symposium on Foundations of Computer Science, IEEE, 1994, pp. 124-134.
4. L. Grover, "A fast quantum mechanical algorithm for database search," in Proceedings of the twenty-eighth annual ACM symposium on Theory of computing, 1996, pp. 212-219.
5. National Academies of Sciences, Engineering, and Medicine, Quantum Computing: Progress and Prospects. Washington, DC: The National Academies Press, 2019.
6. National Institute of Standards and Technology (NIST), "Post-Quantum Cryptography Standardization: FIPS 203 (ML-KEM), FIPS 204 (ML-DSA), FIPS 205 (SLH-DSA)," NIST Information Technology Laboratory, Aug. 2024.
7. C. Paquin, D. Stebila, and G. Tamvada, "Benchmarking post-quantum cryptography in TLS," in Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, 2021, pp. 72-88.
8. O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," Journal of the ACM (JACM), vol. 56, no. 6, pp. 1-40, 2009.
9. C. Peikert, "A decade of lattice cryptography," Foundations and Trends in Theoretical Computer Science, vol. 10, no. 4, pp. 283-424, 2016.
10. S. Kampanakis, P. Panburana, M. Curcio, and C. Aumasson, "The impact of post-quantum signatures on TLS 1.3," IEEE Security & Privacy, vol. 18, no. 5, pp. 22-31, 2020.
11. T. Pöppelmann, L. Ducas, and T. Güneysu, "High-performance lattice-based cryptography on custom hardware," IEEE Transactions on Computers, vol. 68, no. 7, pp. 1012-1025, 2019.
12. E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," RFC 8446, Internet Engineering Task Force (IETF), Aug. 2018.
13. Istio Security Working Group, "Zero Trust Service Mesh Security Architecture with Mutual TLS," Cloud Native Computing Foundation (CNCF) Technical Report, 2023.
14. B. Westerbaan and D. Stebila, "Deploying Post-Quantum Key Exchange at Cloudflare Scale," Cloudflare Research Blog & Proceedings of USENIX Security, 2023.
15. J. W. Bos, L. Ducas, E. Kiltz, and T. Prest, "CRYSTALS-Kyber: a CCA-secure module-lattice-based KEM," IEEE Transactions on Information Theory, vol. 67, no. 11, pp. 7412-7429, 2021.

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved

***Author for Correspondence**

Meenakshi Swaminathan

E-mail: meenakshi.swaminathan64@gmail.com

[1]Associate Professor, Department of Information Technology (IT), Panimalar Institute of Technology

[2]Assistant Professor, Department of Information Technology (IT), Panimalar Institute of Technology

[3]Research Scholar, Department of CSE, Panimalar Institute of Technology

Received Date: August 02, 2026

Accepted Date: August 14, 2026

Published Date: August 27, 2026

Citation: Meenakshi Swaminathan, Vivek Anantharaman, Hari Prasad Iyer. Post-Quantum Cryptography Integration in Cloud-Native Applications: Security, Scalability, and Migration Strategies. Journal of Research in Computer Science and Engineering. 2026; 11(2): 121-132p.

Page 121-132 © MANTECH PUBLICATIONS 2026. All Rights Reserved