Quantum-Secure AI Integration Strategies

Preparing AI systems for the post-quantum cryptography era

38 min read | December 2025 | Security

KEY TAKEAWAYS

- Current public-key cryptography is vulnerable to Shor's algorithm on quantum computers
- NIST post-quantum standards (ML-KEM, ML-DSA) provide the path forward
- Hybrid approaches combining classical and post-quantum algorithms reduce transition risk
- AI systems face unique vulnerabilities including model confidentiality and inference data protection

Preparing Enterprise AI Systems for the Post-Quantum Era

Architetto Research | January 2025

Executive Summary

The development of cryptographically relevant quantum computers poses an existential threat to the security foundations upon which modern AI systems are built. While large-scale, fault-tolerant quantum computers capable of breaking current cryptographic standards do not yet exist, the timeline for their emergence is measured in years, not decades. For enterprise AI systems processing sensitive data with long-term confidentiality requirements, the transition to quantum-resistant cryptography must begin now.

This whitepaper examines the intersection of quantum computing threats and AI system security, presents the emerging post-quantum cryptographic standards, and provides a practical roadmap for transitioning enterprise AI infrastructure to quantum-resistant architectures. We address the unique challenges posed by AI systems, including the protection of model intellectual property, inference data, and the cryptographic protocols underlying distributed AI deployments.

1. The Quantum Threat Landscape

© 2026 Architetto. All rights reserved. | architetto.io

1.1 Understanding the Threat

Current public-key cryptography relies on mathematical problems that are computationally intractable for classical computers but can be efficiently solved by sufficiently powerful quantum computers.

RSA and Discrete Logarithm Systems

RSA, DSA, ECDSA, and ECDH—the algorithms underpinning TLS, code signing, secure communications, and most enterprise security infrastructure—are vulnerable to Shor's algorithm. A quantum computer with approximately 4,000 logical qubits could break 2048-bit RSA in hours.

Symmetric Cryptography

AES and other symmetric algorithms are less affected. Grover's algorithm provides only a quadratic speedup for brute-force attacks, meaning AES-256 remains secure (equivalent to 128-bit security against quantum attack), while AES-128 is reduced to 64-bit security and should be phased out.

Hash Functions

SHA-256 and SHA-3 retain adequate security against known quantum attacks. Longer hash outputs (384-bit or 512-bit) provide additional margin.

1.2 Timeline Considerations

The "harvest now, decrypt later" threat is already active. Adversaries with long-term interests—nation-states, sophisticated criminal organisations—are collecting encrypted data today with the expectation of decrypting it once quantum computers become available.

Critical Timeline Factors:

- **Data Shelf Life**: How long must the data remain confidential? Medical records, trade secrets, and national security information may require protection for decades.
- **Cryptographic Migration Time**: Large enterprises typically require 5-10 years to fully transition cryptographic infrastructure.
- **Quantum Computer Development**: Estimates for cryptographically relevant quantum computers range from 5-15 years, with significant uncertainty.

For organisations where data shelf life plus migration time exceeds the expected quantum timeline, the transition must begin immediately.

1.3 AI-Specific Vulnerabilities

AI systems present unique cryptographic attack surfaces:

Model Confidentiality

Trained models represent significant intellectual property and competitive advantage. Models transmitted or stored with vulnerable encryption could be exposed to future quantum decryption.

Training Data Protection

Training datasets often contain sensitive information. Even if personal data is removed, model inversion attacks may be enhanced by quantum computing capabilities.

© 2026 Architetto. All rights reserved. | architetto.io

Inference Data

API communications, prompt content, and response data are protected by TLS with RSA or ECDH key exchange—both quantum-vulnerable.

Distributed AI Protocols

Federated learning, distributed training, and multi-party computation protocols rely on cryptographic primitives that may be quantum-vulnerable.

Cryptographic Attestation

AI system integrity verification, including secure boot, model signing, and hardware attestation, depends on digital signatures that quantum computers could forge.

2. Post-Quantum Cryptography Standards

2.1 NIST Standardisation

The U.S. National Institute of Standards and Technology (NIST) has conducted a multi-year process to evaluate and standardise post-quantum cryptographic algorithms. The primary standards, published in 2024, include:

ML-KEM (CRYSTALS-Kyber)

A lattice-based key encapsulation mechanism selected as the primary standard for key exchange and encryption:

- Based on the Module Learning with Errors (MLWE) problem
- Three security levels: ML-KEM-512, ML-KEM-768, ML-KEM-1024
- Relatively small key sizes compared to other post-quantum approaches
- Good performance characteristics for software implementation

ML-DSA (CRYSTALS-Dilithium)

A lattice-based digital signature algorithm selected as the primary standard for digital signatures:

- Based on the Module Learning with Errors problem
- Three security levels: ML-DSA-44, ML-DSA-65, ML-DSA-87
- Larger signature sizes than classical algorithms but acceptable for most use cases
- Efficient signing and verification

SLH-DSA (SPHINCS+)

A hash-based digital signature algorithm selected as a conservative alternative:

- Based only on the security of hash functions—no structured mathematical assumptions
- Larger signatures and slower performance than ML-DSA
- Appropriate for applications requiring maximum confidence in security assumptions

© 2026 Architetto. All rights reserved. | architetto.io

2.2 Additional Algorithms

FN-DSA (FALCON)

Expected to be standardised as an additional signature option:

- Based on the NTRU lattice problem
- Smaller signatures than ML-DSA
- More complex implementation with potential side-channel concerns

Classic McEliece

A code-based key encapsulation mechanism:

- Based on error-correcting codes—a well-studied mathematical foundation
- Very large public keys (hundreds of kilobytes to megabytes)
- Small ciphertexts and fast operations
- Appropriate for applications where public key size is not a constraint

2.3 Hybrid Approaches

During the transition period, hybrid cryptographic constructions combine classical and post-quantum algorithms:

Benefits of Hybrid Approaches:

- Security is maintained if either algorithm remains secure
- Protects against potential weaknesses in new post-quantum algorithms
- Enables gradual transition without binary security dependencies

Implementation Patterns:

- Concatenated key exchange: classical + post-quantum KEM, with derived keys combined
- Dual signatures: messages signed with both classical and post-quantum algorithms
- Composite certificates: X.509 certificates containing both algorithm types

3. AI System Security Architecture

3.1 Data Protection

Encryption at Rest

Model files, training data, vector databases, and operational data require encryption with quantum-resistant algorithms:

- Transition to AES-256 for symmetric encryption (AES-128 should be deprecated)

© 2026 Architetto. All rights reserved. | architetto.io

- Key wrapping and key management systems must transition to post-quantum algorithms
- Hardware security modules (HSMs) must support post-quantum algorithms for key operations

Encryption in Transit

All network communications must transition to post-quantum TLS:

- TLS 1.3 with ML-KEM key exchange
- Certificate chains with ML-DSA or SLH-DSA signatures
- Hybrid modes during transition (e.g., X25519 + ML-KEM-768)

Long-Term Archives

Data archives with multi-decade confidentiality requirements demand immediate attention:

- Re-encrypt existing archives with post-quantum algorithms
- Implement forward secrecy to limit exposure if keys are later compromised
- Consider "harvest-resistant" architectures that assume passive collection

3.2 Model Security

Model Signing

Code signing and model integrity verification must transition to post-quantum signatures:

- Model files signed with ML-DSA or SLH-DSA
- Build pipeline integrity protected with quantum-resistant signatures
- Verification enforced at deployment time

Model Encryption

Proprietary models distributed to edge devices or partner organisations require encryption:

- Symmetric encryption (AES-256) for model weights
- Key distribution using post-quantum KEM
- Consider hardware-bound keys for additional protection

Secure Inference

Protecting inference data in transit and potentially in use:

- Post-quantum TLS for API communications
- Confidential computing for inference (TEEs with post-quantum attestation)
- Encrypted inference protocols for sensitive applications

3.3 Distributed AI Protocols

Federated Learning

Federated learning protocols must transition their cryptographic foundations:

- Secure aggregation protocols using post-quantum primitives

© 2026 Architetto. All rights reserved. | architetto.io

- Differential privacy mechanisms remain quantum-resistant (based on noise addition)
- Communication encryption using post-quantum TLS

Multi-Party Computation

MPC protocols for collaborative AI face quantum vulnerabilities:

- Oblivious transfer protocols must transition to post-quantum assumptions
- Secret sharing schemes remain secure
- Garbled circuit implementations require updated key exchange

Blockchain and Distributed Ledgers

AI systems using blockchain for audit trails or governance face signature vulnerabilities:

- Transition to post-quantum signature schemes
- Address format changes may be required
- Existing transactions signed with classical algorithms remain vulnerable to retrospective forgery

3.4 Hardware Security

Secure Enclaves and TEEs

Trusted Execution Environments provide hardware-based isolation for sensitive AI workloads:

- Attestation protocols must transition to post-quantum signatures
- Key provisioning must use post-quantum key exchange
- Consider enclave-specific quantum-resistant implementations

Hardware Security Modules

HSMs protect cryptographic keys and perform sensitive operations:

- Vendor support for post-quantum algorithms is emerging but not universal
- Firmware updates may be required for post-quantum capability
- Evaluate HSM roadmaps as part of transition planning

AI Accelerators

Specialised AI hardware may have cryptographic functions that require updating:

- Secure boot and attestation mechanisms
- Encrypted communication with host systems
- Model encryption and decryption capabilities

4. Transition Roadmap

4.1 Assessment Phase (Months 1-6)

© 2026 Architetto. All rights reserved. | architetto.io

Cryptographic Inventory

Comprehensive identification of all cryptographic usage:

- Network protocols (TLS, SSH, VPN)
- Data encryption (disk, database, backup)
- Digital signatures (code signing, model signing, certificates)
- Key management systems
- Hardware cryptographic devices
- Third-party dependencies

Vulnerability Assessment

Classification of cryptographic usage by quantum vulnerability:

- Immediately vulnerable: RSA, DSA, ECDSA, ECDH, DH
- Reduced security: AES-128, SHA-1
- Secure with longer output: SHA-256 (acceptable), SHA-384/512 (preferred)
- Quantum-resistant: AES-256, SHA-3, new NIST standards

Data Classification

Identification of data with long-term confidentiality requirements:

- What data requires protection beyond 10 years?
- What is the estimated impact of future decryption?
- What are the regulatory requirements for data protection?

Dependency Mapping

Understanding external dependencies on cryptographic functions:

- Cloud provider cryptographic services
- Identity provider protocols
- Partner and customer integration points
- Compliance and audit requirements

4.2 Planning Phase (Months 7-12)

Architecture Design

Post-quantum security architecture incorporating:

- Algorithm selection based on use case requirements
- Hybrid approaches for transition period
- Performance impact mitigation strategies
- Backward compatibility requirements

Vendor Engagement

© 2026 Architetto. All rights reserved. | architetto.io

Coordination with technology vendors:

- Cloud provider post-quantum roadmaps
- HSM vendor upgrade paths
- Identity provider migration plans
- Application vendor compatibility

Standards and Policy

Internal standards development:

- Algorithm selection standards
- Key size and parameter requirements
- Transition timeline policies
- Compliance mapping

Proof of Concept

Targeted testing of post-quantum implementations:

- TLS 1.3 with post-quantum key exchange
- Post-quantum signature verification
- Performance benchmarking
- Compatibility testing

4.3 Implementation Phase (Years 2-4)

Infrastructure Layer

Network and infrastructure cryptographic transition:

- Load balancers and reverse proxies with post-quantum TLS
- VPN and SSH with post-quantum algorithms
- Certificate infrastructure with post-quantum signatures
- Key management system upgrades

Application Layer

Application-level cryptographic updates:

- API authentication and encryption
- Database encryption key rotation
- Model signing and verification
- Backup encryption updates

Data Migration

Re-encryption of data with long-term confidentiality requirements:

- Archive re-encryption projects

© 2026 Architetto. All rights reserved. | architetto.io

- Database encryption updates
- Model encryption upgrades
- Secure deletion of classically-encrypted copies where appropriate

Third-Party Integration

Coordination with external parties:

- Partner integration updates
- Customer-facing API transitions
- Compliance certification updates
- Audit trail documentation

4.4 Validation Phase (Year 5+)

Security Validation

Comprehensive verification of post-quantum implementation:

- Penetration testing with quantum-aware threat models
- Cryptographic implementation review
- Side-channel analysis where appropriate
- Third-party audit and certification

Operational Validation

Performance and reliability verification:

- Load testing with post-quantum protocols
- Failover and recovery testing
- Monitoring and alerting updates
- Runbook and procedure updates

Continuous Improvement

Ongoing adaptation as the field evolves:

- Monitor for new quantum computing developments
- Track cryptographic research for vulnerabilities
- Update standards as NIST and other bodies issue guidance
- Participate in industry coordination efforts

5. Performance Considerations

5.1 Computational Overhead

© 2026 Architetto. All rights reserved. | architetto.io

Post-quantum algorithms generally require more computational resources than their classical counterparts:

Key Exchange

- ML-KEM-768: approximately 2-3x slower than X25519 in software
- Hardware acceleration significantly improves performance
- Impact on TLS handshake latency is measurable but typically acceptable

Digital Signatures

- ML-DSA: signing is comparable to ECDSA; verification is faster
- SLH-DSA: significantly slower (appropriate for infrequent, high-security operations)
- Batch verification can improve throughput for high-volume applications

Symmetric Operations

- AES-256 vs AES-128: approximately 40% slower (still very fast with hardware acceleration)
- Modern CPUs with AES-NI make this overhead negligible

5.2 Bandwidth and Storage

Post-quantum algorithms typically have larger key and signature sizes:

| Algorithm | Public Key | Signature/Ciphertext |

|-----------|------------|----------------------|

| ECDSA-P256 | 64 bytes | 64 bytes |

| ML-DSA-65 | 1,952 bytes | 3,293 bytes |

| ML-KEM-768 | 1,184 bytes | 1,088 bytes |

| SLH-DSA-128s | 32 bytes | 7,856 bytes |

Implications:

- Certificate chains become larger
- TLS handshakes exchange more data
- Signed data requires more storage
- Bandwidth-constrained environments (IoT, mobile) may require optimisation

5.3 Mitigation Strategies

Hardware Acceleration

- CPU instructions for lattice operations are emerging
- HSMs and cryptographic accelerators with post-quantum support
- GPU acceleration for high-throughput scenarios

© 2026 Architetto. All rights reserved. | architetto.io

Protocol Optimisation

- TLS session resumption reduces full handshake frequency
- Certificate compression reduces bandwidth impact
- Strategic algorithm selection based on use case requirements

Caching and Reuse

- Session keys can be reused within security bounds
- Public key caching reduces repeated exchanges
- Batch operations amortise overhead

6. Compliance and Standards

6.1 Regulatory Expectations

Government Requirements

Government agencies in many jurisdictions are mandating transition to post-quantum cryptography:

- U.S.: CNSA 2.0 requires post-quantum algorithms for National Security Systems by 2033
- EU: ENISA guidance recommends hybrid approaches for sensitive data
- Financial regulators are beginning to address quantum risk in guidance

Industry Standards

- PCI DSS: future versions expected to address quantum threats
- HIPAA: current guidance focuses on "reasonable" security measures
- SOC 2: control objectives may be interpreted to require quantum readiness

6.2 Certification Paths

FIPS 140-3

Post-quantum algorithms are being added to FIPS 140-3 validation:

- CAVP (Cryptographic Algorithm Validation Program) testing available
- CMVP (Cryptographic Module Validation Program) for modules implementing post-quantum algorithms
- Transition period allows hybrid implementations

Common Criteria

Protection profiles may be updated to include post-quantum requirements:

- Currently, post-quantum algorithms can be included as additional assurance
- Future protection profiles may mandate post-quantum for high-assurance levels

© 2026 Architetto. All rights reserved. | architetto.io

6.3 Documentation Requirements

Cryptographic Inventories

Regulators increasingly expect documented inventories of cryptographic usage:

- Algorithm identification
- Key sizes and parameters
- Data protected and retention requirements
- Transition plans and timelines

Risk Assessments

Quantum risk should be integrated into broader risk management:

- Data classification considering quantum vulnerability
- Timeline analysis (data shelf life vs. quantum computer development)
- Business impact assessment for delayed transition

7. Conclusion

The transition to quantum-resistant cryptography is not a future concern—it is a present imperative for organisations processing data with long-term confidentiality requirements. AI systems, with their reliance on cryptographic protocols for secure communication, model protection, and system integrity, are particularly exposed to quantum threats.

The path forward requires:

- **Immediate Action**: Begin cryptographic inventory and assessment now
- **Strategic Planning**: Develop transition architectures that balance security, performance, and compatibility
- **Phased Implementation**: Execute transition in stages, prioritising highest-risk systems
- **Continuous Adaptation**: Monitor the evolving threat landscape and adjust plans accordingly

Organisations that proactively address quantum risk will be better positioned to maintain security and compliance as quantum computing capabilities advance. Those that delay may find themselves racing against an adversary timeline they cannot control.

References

1. NIST Post-Quantum Cryptography Standardisation

2. CISA: Post-Quantum Cryptography Initiative

3. NSA/CISA: Quantum-Readiness Guidance

© 2026 Architetto. All rights reserved. | architetto.io

4. ETSI: Quantum-Safe Cryptography

5. Cloud Security Alliance: Quantum-Safe Security Working Group

This whitepaper is provided for informational purposes. Specific implementation decisions should be made in consultation with qualified technical and legal advisors.

© 2026 Architetto. All rights reserved. | architetto.io