CRYPTO4A

WHITEPAPER

Quantum-Safe by Design:

An Architectural Imperative for Hardware Security Modules in the Post-Quantum Era

CRYPTO4A

Quantum-Safe by Design: An Architectural Imperative for Hardware Security Modules in the Post-Quantum Era

Abstract

As quantum computing approaches practical viability, the cryptographic foundations of modern digital infrastructure face an existential threat. Hardware Security Modules (HSMs), the bedrock of digital trust, were designed around traditional cryptographic algorithms such as RSA and ECC, both of which could be rendered obsolete by quantum-powered attacks[1]. This white paper examines what it truly means for an HSM to be *Quantum-Safe by design*. Through a forward-looking thought experiment and a detailed exploration of nine architectural imperatives, it demonstrates why future-proofing the HSM cannot be achieved through software patches or simple algorithm swaps. Instead, it requires a fundamental re-engineering of the trust architecture itself. As the operational backbone of digital sovereignty, the HSM must be resilient in a post-quantum world, a challenge that carries both strategic and national importance.

Foreword: A Thought Experiment

Imagine stepping into a time machine and travelling to the year 2036 to find that the long-anticipated era of quantum computing has indeed arrived. Clinical trials are underway for transformative new drugs for diseases previously thought untreatable. Artificial Intelligence can find patterns in large datasets with a precision rivalling the computers from Star Trek, and quantum-powered cryptanalysts are cracking 50 years worth of captured Internet traffic.

You've brought with you one of today's[2] Hardware Security Modules (HSMs), the same trusted devices that safeguard cryptographic keys, authenticate systems, and uphold the integrity of our digital world. When you arrive and power it on, a sobering question emerges: *Is the HSM still permitted to operate, or has its cryptographic foundation been rendered obsolete?*

By 2036, traditional public-key cryptography (e.g. RSA, ECC, and all related primitives) is now disallowed as per NIST's guidance[3]. The once unshakeable guardians of digital trust can no longer operate using only traditional public-key cryptography. That HSM you brought, once the

---

[1] NSA, *"Post-Quantum Cryptography: CISA, NIST, and NSA Recommend How to Prepare Now,"* August 21, 2023, NSA.gov.

[2] Publication date: January 2026

[3] NIST IR 8547, "*Transition to Post-Quantum Cryptography Standards*"

bedrock of digital assurance, may now be too vulnerable to trust. Unless it was Quantum-Safe by design, it has no place in a post-quantum world.

What Makes an HSM Quantum-Safe by Design?

Traditionally, a Quantum-Ready device is one that utilizes internal cryptographic mechanisms[4] that are resilient to a cryptographically relevant quantum computer (CRQC). However, an HSM is a unique case as it also provides cryptographic services to external parties. We define a Quantum-Safe HSM as one that is both Quantum-Ready, and which provides post quantum cryptographic (PQC) services externally. It considers not only the quantum resilience of the cryptographic services it delivers but also the quantum resilience of the device's internal capabilities, ensuring both can withstand a quantum-powered adversary. Becoming Quantum-Safe is not a matter of software patches or isolated algorithm updates; it requires architectural foresight.

A truly Quantum-Safe HSM must be engineered from the ground up with post-quantum principles embedded into its roots of trust, internal architecture, and cryptographic lifecycle, while simultaneously enabling post-quantum cryptographic services for external systems.

The following nine critical questions determine whether an HSM is genuinely prepared to withstand the quantum future by assessing if it is truly quantum-safe by design.

1. Are the Roots of Trust (RoT) Quantum-Ready?

A Root of Trust (RoT) is the foundational, inherently trusted component within a system that serves as the secure anchor for all other security functions. HSMs are typically used to generate and protect RoTs (e.g., PKI root key pairs, FW signing keys, etc.), but HSMs also require a RoT to anchor their internal cryptographic functions. If that foundation is built on traditional cryptography, it inevitably inherits those algorithms' vulnerabilities, undermining the long term security of the device when CRQCs exist.

A Quantum-Safe HSM must be manufactured with post-quantum algorithms embedded directly into its trust chain, ensuring its foundational integrity remains intact, even in the presence of quantum-powered adversaries.

---

[4] While this whitepaper focuses on the vulnerabilities associated with public-key algorithms, other cryptographic mechanism such as symmetric algorithms and hash functions also have concerns but for the most part they are readily addressable by ensuring key and digest lengths of 256 bits or longer (e.g., AES-256 and SHA2-256/384/512 or SHA3).

2. Are All Firmware Updates Signed with Post-Quantum Cryptographic (PQC) Algorithms?

Firmware updates are the HSM's lifeline. If any of those updates are signed using traditional cryptographic methods, they become vulnerable to future quantum compromise. A quantum-powered attacker could forge signatures, inject malicious code, and corrupt the very foundation of digital trust.

A Quantum-Safe HSM must therefore rely on PQC-based signing algorithms to protect firmware integrity and implicitly authenticate that FW images are from legitimate sources throughout the device's entire lifecycle.

3. Is the Secure Boot Process Quantum-Ready?

A Quantum-Ready boot process must use PQC-based validation so that, even decades from now, the earliest system checks remain trustworthy and resilient in the presence of quantum-powered attacks.

Secure boot ensures a Quantum-Safe HSM starts in a known and trusted state. But if this process relies on traditional integrity mechanisms, that assurance collapses under a quantum-powered adversary.

4. Are Attestation Keys Quantum-Ready?

Attestation provides cryptographic proof of identity, origin, and system integrity. But if attestation keys rely on traditional cryptography, those proofs can be forged once quantum-powered attacks become practical.

A Quantum-Safe HSM providing attestation services must embed PQC-based attestation keys, provisioned and certified at manufacturing, to ensure enduring authenticity and trustworthiness.

5. Are Backup and Archival Processes Quantum-Ready?

Backups preserve critical key material and configuration data. If these archives are protected using components reliant on traditional public-key cryptography (e.g., RSA- or ECC-based key agreement/establishment), they become vulnerable the moment quantum-powered attacks are feasible.

A Quantum-Safe HSM must employ PQC-based key wrapping and encryption for all backups and archival data, ensuring long-term recoverability without the risk of compromise.

6. Are Inter-HSM Communications Quantum-Ready?

Clusters of HSMs continually exchange sensitive cryptographic material. If inter-HSM communication relies on traditional key agreement or authentication mechanisms (e.g., RSA- or ECC-based key agreement/establishment), a quantum-powered attacker could compromise internal trust boundaries.

A Quantum-Safe HSM must enforce PQC-based encryption and authentication for all inter-HSM communications, preserving confidentiality and integrity across distributed deployments, even in the presence of quantum-powered adversaries.

7. Are Access Authentication Mechanisms Quantum-Ready?

Access Authentication governs access to every secure operation, whether for users, administrators, or interconnected systems. If those credentials rely on traditional cryptography, they will eventually be forgeable by a quantum-powered attacker.

A Quantum-Safe HSM must support only PQC-based access authentication mechanisms to ensure that only legitimate users and systems retain control.

8. Are Logging/Audit Mechanisms Quantum-Ready?

Secure logging and auditing are essential for security, accountability, compliance, and incident response. They provide a verifiable, time-stamped record of system activity used to detect breaches, validate policy enforcement, and reconstruct events to determine the timeline and scope of an incident. But if the integrity of these logs relies on traditional cryptography, they become susceptible to modification or forgery by a quantum-powered attacker.

A Quantum-Safe HSM must employ PQC-based integrity mechanisms within its logging and auditing systems to ensure the integrity of this invaluable information.

9. Does the HSM Provide Certified Post-Quantum Cryptographic Capable Services?

An HSM ultimately exists to serve external systems, providing encryption, signing, and key-management services. The post-quantum transition demands that HSMs support both current traditional public-key algorithms, the newly minted standardized PQC algorithms and future PQC algorithms that are already being worked on, enabling a smooth evolution of cryptographic services and protocols throughout the transition period and into the future.

A Quantum-Safe HSM must therefore be *crypto-agile*[5]: capable of supporting traditional, first generation PQC as well as future PQC algorithms seamlessly. This crypto agility can be provided in a variety of ways such as quantum-safe FW updates and field-reprogrammable hardware engines.

Why the Details Matter

Items 1 through 4 above are so foundational that they *must* be implemented during the manufacturing phase of an HSM's lifecycle, the only point at which the device is under complete positive control. This is the stage where the HSM can be provisioned with hybrid traditional and PQC-based Roots of Trust (RoT) and unique keying material. Attempting to retrofit these elements later through alternative methods introduces significant, and often unacceptable, risk that requires careful evaluation by the HSM operator.

Heightening the challenge, none of these capabilities are currently required for FIPS 140-3 Level 3 validation. In practice, HSMs that rely exclusively on traditional cryptography can and do achieve FIPS 140 certification and will continue to do so until PQC-related requirements are formally incorporated into the CMVP and the FIPS 140 assessment process. Until that happens, customers bear the responsibility of determining whether a device is genuinely Quantum-Safe by asking vendors the critical questions outlined above. In the interest of transparency, we have included Crypto4A's responses in the Appendix.

Conclusion: The Bedrock of Digital Trust

Public Key Infrastructure (PKI) is often described as the pillar of modern digital trust, but every pillar requires a solid and secure foundation. That foundation is the HSM. Much like the Operational Technology (OT) that underpins critical infrastructure, the HSM's security and integrity are essential to safeguarding the world's digital systems in a quantum-powered future.

As the quantum era emerges, HSMs that are *not* Quantum-Safe by design will weaken the very systems they were intended to protect. Engineering them for crypto agility, PQC readiness, and architectural resilience is no longer optional, it is an absolute necessity!

When the time machine lands in 2036, you shouldn't be asking yourself, "Is my HSM still secure?", but instead, you should be saying, "I'm glad we used an HSM that was Quantum-Safe by design".

---

[5] Additional details regarding crypto agility can be found in NIST's publication, "*Considerations for Achieving Cryptographic Agility: Strategies and Practices*", https://doi.org/10.6028/NIST.CSWP.39

Appendix - Crypto4A's Responses

Throughout our responses you'll note that we leverage a hybridized approach in which both traditional (e.g., ECC-based) and PQC (e.g., HSS-based) cryptographic mechanisms are used to secure various aspects of the design (e.g., FW upgrading, attestation, etc.). This hybridization was required in order to comply with the FIPS certification requirements present during the design process. In addition, at times we were forced to develop proprietary solutions (e.g., Classic McEliece based inter-HSM transfers) as there weren't any certified/standardized PQC-based options available at the time we were designing our solution. Going forward we advocate the adoption of certified/standardized quantum-safe approaches as they become available. As such, we're leveraging our platform's crypto agility and quantum-ready FW update capabilities to adopt these approaches and continue to evolve our capabilities to ensure we deliver a resilient solution that adapts to whatever unforeseen threats arise in the future.

How does Crypto4A's Quantum-Assured Security Module (QASM™[6]) HSM fare on the Quantum-Safe by Design questionnaire? Here are our answers to the 9 questions we posed above:

1. Quantum-Ready RoTs: During the manufacturing phase, each QASM is provisioned with two immutable roots of trust (RoT): one based on ECDSA P-384 (traditional NIST-approved elliptic curve cryptography) and one based on HSS (Hierarchical Signature System, a NIST-approved hash-based PQC), to dual-sign the QASM firmware (FW) image. The corresponding private keys for these RoT are themselves stored off-line in our own QASM manufacturing HSMs.

2. PQC-based FW Updates: The QASM FW updates are signed using both ECDSA P-384 and HSS signatures. The RoTs installed during step#1 above, are used to verify all QASM FW images throughout the entire lifetime of the device such that at no point is there a risk of using a FW image that was NOT protected by PQC. The FW update process proceeds only once both signatures are validated against the installed RoTs. This creates a chain of trust that can be used throughout the HSM's lifecycle.

3. Quantum-Ready Secure Boot: All FW images are currently secured at boot with AES-256-GCM encryption (which is Quantum-Safe), and the loading of those FW images is protected by PQC signatures (see item #2 above). Furthermore, all of that is running inside our always-on 24/7 tamper-monitored operating environment to preclude substitution or manipulation of the FW image once it has been loaded and authenticated (using Quantum-Safe signatures).

4. Quantum-Ready Attestation: Crypto4A provides attestation services based on both traditional cryptography (ECDSA P-384) and PQC (HSS), which can be leveraged to provide traditional-only, PQC-only, or a hybrid attestation framework. This ensures interoperability with existing attestation ecosystems that may not currently be PQC-capable, while providing a migration path to evolve to a PQC-based capability in the long term. Additional attestation keys can be generated to support other attestation authority structures, and Crypto4A's attestation verification tools can enforce the dual validation approach being proposed in new IETF draft standards such as the PKIX Evidence for Remote Attestation of Hardware Security Modules (see: draft-ietf-rats-pkix-key-attestation-02).

5. Quantum-Ready Backups and Archiving: When cryptographic material must be backed up or archived (e.g. for disaster recovery or migration), AES-256 Key Wrap with Padding (SP800-38F) is used and the associated AES wrapping keys are protected by Shamir's Secret Sharing (a.k.a. M-of-N techniques) which allows a quorum of users to govern their reconstitution and usage. The sharded wrapping keys are encapsulated using either traditional or PQC encryption primitives to ensure long-term resilience. The wrapping/unwrapping mechanism and their associated keys are held exclusively within the secure boundaries of a participating QASM in the cluster.

6. Quantum-Ready Inter-HSM Communications: When two QASMs communicate or exchange user keys, the exchange is protected by Quantum-Safe symmetric key cryptography (e.g., AES-256) using a combination of ECDH P-384 and Classic McEliece to establish the session key, and a

[6] QASM™ is the cryptographic module at the core of the QxHSM™, QxVault™ and QxEDGE™

combination of ECDSA P-384 and HSS (which is Quantum-Safe) to authenticate the process. Further assurance can be provided using our Quantum-Safe attestation capability to attest to the participating parties and keys being used, and established, to ensure they are present on QASM devices.

NOTE: the choice of Classic McEliece was necessitated by the need to choose a very conservative quantum-resilient cryptographic encapsulation mechanism before NIST had published their selection of FIPS 203 (ML-KEM), and we had the luxury of not worrying about key size or requiring interoperability with other vendors for this use case. Given Classic McEliece's resilience to cryptanalytic attacks over the last 45+ years, it seemed a good choice, as validated by its selection for PQC key encapsulation by other agencies such as the BSI.

7. Quantum-Ready Authentication Mechanisms: The QASM supports both traditional cryptography (i.e., ECDSA and RSA) and PQC (i.e., ML-DSA, SLH-DSA, LMS/HSS, and XMSS/XMSS-MT) authentication for administrative and API access to the QASM. Users can authenticate via host-generated credentials or hardware tokens. Integration with traditional cryptographic USB tokens is already available, and collaborations are underway with manufacturers producing ML-DSA-capable hardware tokens for PQC-backed authentication use cases.

8. Quantum-Ready Logging/Audit Mechanisms: The QASM supports a secure logging and audit facility based on a blockchain whose state is validated via PQC-based attestation methods. In addition, Crypto4A provides tools to allow an external party/auditor to cryptographically verify the integrity of the audit log using methods based on both traditional (ECDSA P-384) and PQC (HSS).

9. PQC-Capable Services: The QASM natively supports the full complement of all NIST PQC algorithms (i.e. LMS, XMSS, FIPS-203, FIPS-204 and FIPS 205[7]) as well as HSS, XMSS-MT and Classic McEliece. The QASM also supports the complete set of traditional algorithms normally found on general purpose HSMs (e.g., RSA, ECC, AES, SHA-1/2/3, etc.). When new PQC algorithms are standardized, such as FN-DSA (FIPS 206), Crypto4A's Quantum-Safe FW update process can be used to add it to the QASM's capabilities either via FW-based or FPGA-based functionality, all of which is part of our standard Features, Maintenance and Support (F,M&S) contract.

[7] CAVP certificate #A5631