AffixIO Technical Paper — WP-002
June 2026
affix-io.com

AFFIXIO WHITE PAPER · WP-002

Post-Quantum Attestation in Production: ML-DSA-65 for Long-Lived Record Integrity

Sign audit roots today so quantum cannot rewrite them tomorrow.

AffixIO | United Kingdom | affix-io.com | June 2026

ABSTRACT

Store-now-decrypt-later makes classical signatures a liability for records kept ten years. AffixIO signs Merkle roots with ML-DSA-65 inside FIPS 140-2 Level 3 HSMs. This paper is the production guide we wished existed when we migrated.

CONTENTS

SECTION 1

Introduction

Classical digital signatures (ECDSA, RSA, Ed25519) rest on mathematical problems that classical computers cannot solve in reasonable time but that quantum computers running Shor's algorithm can. For most of the past decade, this vulnerability felt theoretical: the quantum computers needed to exploit it did not exist, and credible projections placed their arrival far enough away that a planned migration felt adequate. That position is no longer defensible.

In March 2026, Google published a zero-knowledge proof demonstrating that a first-generation fault-tolerant quantum computer could break elliptic curve cryptography keys in under nine minutes using optimised circuit designs. Research published between mid-2025 and early 2026 has reduced the qubit estimate for breaking RSA-2048 to under 100,000, down from 20 million in 2019. The compression is driven entirely by engineering optimisations to known techniques, and there is no strong theoretical argument that it has reached its floor. The conversation in the security community has shifted from "if Q-Day arrives" to "when Q-Day arrives and how much warning we will have."

For short-lived encrypted communications, a planned migration is adequate. A TLS session encrypted yesterday does not need to remain confidential in fifteen years. Rotating keys to a quantum-safe scheme when the threat materialises is sufficient for that category of data.

Attestation records are a different category entirely. A signed governance certificate, an AI decision audit trail, a compliance attestation, or a signed proof of eligibility may need to withstand scrutiny in legal proceedings, regulatory investigations, or Freedom of Information requests years or decades from now. If the signature on such a record uses a classical algorithm, and a sufficiently capable quantum computer exists at the time of that scrutiny, the signature can be forged. A forged signature produces a record indistinguishable from a genuine one. No audit trail built on classical cryptography is safe from this outcome.

The correct response is to apply quantum-safe signatures to attestation records from the point of generation. Records generated before a PQC migration carry classical signatures that will eventually be forgeable. Records generated after carry post-quantum signatures that resist forgery for any foreseeable timeline. Closing the gap requires starting now, not at a future migration date.

NIST finalised ML-DSA-65 as FIPS 204 in August 2024, completing an eight-year public review process that included continuous cryptanalysis against lattice-based cryptography attacks. The algorithm runs on standard hardware, is deployable today via production-ready libraries, and its security rests on the hardness of the Module Learning With Errors (M-LWE) problem, for which no quantum algorithm provides a meaningful speedup. AffixIO has been running ML-DSA-65 in production for AI governance attestation since late 2025. This paper explains why, how, and what engineering teams need to know when evaluating quantum-safe attestation for their own cryptographic audit trail infrastructure.

SECTION 2

The Compressed Quantum Timeline

The urgency of post-quantum migration depends directly on how quickly the resource estimates for quantum attacks have changed. The key metric is the number of logical qubits required to break RSA-2048 or ECDSA P-256 using Shor's algorithm in a practical timeframe. That number has fallen by more than two orders of magnitude in under a decade.

YEAR	QUBIT ESTIMATE (RSA-2048)	BASIS
2019	~20 million	Gidney and Ekerå (Google) baseline
2022	~4 million	Refined surface code analysis
2025	under 1 million	QLDPC codes and improved error correction
Early 2026	under 100,000	Iceberg Quantum; Google, Stanford, and Ethereum collaborative analysis

Each revision represents a factor of ten to twenty reduction from the previous estimate. Three distinct engineering advances drive the compression. First, quantum error correction has improved: low-density parity-check (LDPC) codes and their quantum variants require fewer physical qubits per logical qubit than earlier surface code assumptions. Second, the quantum circuits that implement Shor's algorithm have been optimised, reducing gate operations per cryptographic operation. Third, architectural improvements to quantum hardware have increased the ratio of useful computation to error-correcting overhead.

None of these improvements required a theoretical breakthrough. They are engineering refinements of known techniques. This is the detail that makes the quantum threat timeline genuinely alarming for PQC readiness planners: the compression is likely to continue as engineering matures, not level off at a floor set by physics.

The first-generation fault-tolerant window

Current NISQ (noisy intermediate-scale quantum) computers have hundreds to thousands of physical qubits with error rates that prevent them from running Shor's algorithm at any useful scale. The estimates above describe fault-tolerant quantum computers, which require physical qubit counts much higher than logical qubit counts due to error-correction overhead. The question is when fault-tolerant machines at the relevant scale will exist.

The US National Security Agency's CNSA 2.0 guidance, the most authoritative public statement from a national security body, treats the threat as credible enough to mandate quantum-safe migration for new national security systems by 2027. Google's internal planning targets 2029 for migrating its own infrastructure to post-quantum cryptography. Cloudflare reports that the majority of human-generated internet traffic it handles was already protected by post-quantum key exchange by late 2025. Neither Google nor Cloudflare has incentives to overstate the threat. Both have engineering organisations with direct access to the most current quantum computing research.

For planning purposes, the threat should be treated as a realistic horizon of five to fifteen years, with meaningful probability at the lower end. For data that does not need to remain valid beyond five years, a deferred migration is a reasonable risk acceptance. For records that must remain valid for decades, the lower end of that range is already inside the retention window for data being generated today.

> **Key planning fact:** A signed AI governance record generated today with ECDSA P-256 and retained for a fifteen-year regulatory compliance period will face a signing algorithm that may be quantum-compromised well before the retention period ends. The signature must be correct at the time of verification, not only at the time of generation.

SECTION 3

Store-Now-Decrypt-Later Attacks

The store-now-decrypt-later (SNDL) attack, also known as harvest-now-decrypt-later (HNDL), is the mechanism by which the quantum threat applies to data generated today. It proceeds in two phases. In the first phase, an adversary intercepts and stores encrypted communications or signed records without being able to process them cryptographically at the time of collection. In the second phase, once a quantum computer capable of running Shor's algorithm becomes available, the stored material is processed: ciphertext is decrypted, and signed records are either read or forged.

Multiple intelligence community assessments published between 2023 and 2026 confirm that nation-state adversaries are actively executing the first phase of SNDL attacks against high-value targets, systematically collecting encrypted communications in anticipation of future quantum computing capability. The UK National Cyber Security Centre, US CISA, and the Five Eyes alliance have all issued advisories acknowledging this activity. The practice is consistent with the long planning horizons and bulk data collection infrastructure that sophisticated state-level threat actors maintain.

Encryption and signatures fail differently under SNDL

SNDL applies to encrypted data and signed records in distinct ways that carry different implications for post-quantum migration priority.

For encrypted communications, the risk is confidentiality loss: the adversary will eventually read the plaintext. This is severe for long-term secrets but can be mitigated prospectively by adopting quantum-safe key encapsulation (NIST FIPS 203, ML-KEM) for new sessions, since older sessions are already compromised regardless of when migration happens.

For signed records, the failure mode is different. A digital signature provides two guarantees simultaneously: authentication (the signature was produced by the holder of the private key) and integrity (the signed content has not been altered since signing). Both guarantees fail if the underlying mathematical problem becomes tractable. Once a quantum computer exists, an adversary can produce a valid signature for any message under any

classical public key, without knowing the corresponding private key. They can generate entirely new records that appear to have been signed by the original signer at the original time. The signed audit trail becomes worthless as evidence because any record in it could have been fabricated.

The critical asymmetry: records signed with a classical algorithm today will be permanently vulnerable when quantum computing matures, even if the signing key has since been rotated. The signature is on the record, not on the key. Rotating keys does not retroactively make historical records quantum-safe.

What SNDL means for AI governance

AI governance records carry SNDL exposure in a specific commercial and regulatory context. An adversary who could forge AI governance attestations might dispute that a specific AI decision was made under the policy the organisation claims, fabricate evidence of governance compliance for systems that lacked it, or retroactively alter audit trails in ways that affect legal liability, regulatory standing, or contractual performance evidence. The commercial motivation exists wherever AI decisions carry legal or regulatory consequences, which is precisely the domain that the EU AI Act, NIST AI RMF, and ISO 42001 are designed to govern. Post-quantum attestation closes this exposure at the source.

SECTION 4

NIST FIPS 204: The ML-DSA Standard

NIST launched its Post-Quantum Cryptography Standardisation process in 2016, inviting candidate algorithms across three categories: digital signatures, key encapsulation mechanisms, and hash-based signatures. The process ran for eight years of open submission, public cryptanalysis, and multiple evaluation rounds, concluding in August 2024 with the finalisation of four standards. ML-DSA is the primary general-purpose digital signature standard in that set.

STANDARD	ALGORITHM	PRIMARY USE
FIPS 203	ML-KEM (lattice key encapsulation)	Key exchange and TLS
FIPS 204	ML-DSA (lattice digital signature)	Signatures and attestation
FIPS 205	SLH-DSA (stateless hash-based signature)	Conservative signatures
FIPS 206	FN-DSA (Falcon lattice signature)	Compact signatures

ML-DSA was previously known as CRYSTALS-Dilithium during the standardisation process. It is the general-purpose quantum-resistant signature algorithm recommended by NIST for most new deployments that do not have specific size constraints requiring Falcon (FN-DSA) or exceptional security conservatism requiring SLH-DSA. Its design draws on lattice-based cryptography, specifically the Module Learning With Errors (M-LWE) problem, which is structurally distinct from both the integer factorisation problem (targeted by Shor's algorithm against RSA) and the discrete logarithm problem (targeted against ECDSA).

Lattice-based cryptography foundations

Lattice-based cryptography is the family of post-quantum algorithms whose hardness assumptions rest on the computational difficulty of finding short vectors in high-dimensional lattices. The Learning With Errors (LWE) problem, introduced by Oded Regev in 2005, is the core hardness assumption

underpinning most lattice-based cryptographic constructions. In LWE, an adversary is given a set of noisy linear equations over a finite ring and asked to recover the secret coefficients. The noise term is small but sufficient to make the system intractable to solve efficiently, even for quantum computers.

ML-DSA uses the Module-LWE variant, where coefficients are structured as elements of a polynomial ring modulo a cyclotomic polynomial. This structure provides both efficient arithmetic and a clean security reduction to the hardness of M-LWE. No quantum algorithm is known to provide a significant speedup over classical lattice reduction algorithms for M-LWE. The best known quantum attacks reduce the effective security level by a modest factor that NIST accounts for in its parameter choices: ML-DSA-65 targets 128-bit post-quantum security after incorporating the best known quantum attacks against its lattice parameters.

Eight years of cryptanalysis

CRYSTALS-Dilithium was submitted in 2017 and subjected to continuous public cryptanalysis for eight years. During that period, no attack was found that materially reduced the security of the scheme below its claimed levels. Several attacks were found against related constructions and used to harden the Dilithium design. The resulting ML-DSA standard reflects accumulated cryptanalytic experience that no pre-standardisation post-quantum algorithm can match. For organisations making a long-term commitment to quantum-safe infrastructure, the depth of public review is a material advantage over any proprietary or draft-stage algorithm.

SECTION 5

ML-DSA-65 Technical Specification

ML-DSA is available at three security levels: ML-DSA-44, ML-DSA-65, and ML-DSA-87. The numerical suffix reflects a parameter set that controls the security-size trade-off. AffixIO uses ML-DSA-65, which targets NIST Security Level 3, the level equivalent to AES-192. For long-lived attestation records that may need to remain valid for decades, Level 3 provides a security margin above the minimum that buffers against future cryptanalytic improvements without the size penalty of Level 5.

PARAMETER	ML-DSA-44	ML-DSA-65	ML-DSA-87
NIST Security Level	Level 2 (AES-128 equiv.)	Level 3 (AES-192 equiv.)	Level 5 (AES-256 equiv.)
Post-quantum security	~128 bits	~128 bits (conservative)	~128 bits (highest margin)
Public key size	1,312 bytes	1,952 bytes	2,592 bytes
Signature size	2,420 bytes	3,309 bytes	4,627 bytes
Private key size	2,560 bytes	4,032 bytes	4,896 bytes

Signing and verification

ML-DSA signing takes a private key and a message, then produces a signature using a Fiat-Shamir-with-aborts construction over module lattices. The process samples a random polynomial vector, computes a commitment via a hash function, and produces a response vector that proves knowledge of the private key without revealing it. ML-DSA is deterministic when used with a fixed randomness seed: the same message signed with the same private key always produces the same signature. This property is valuable for audit infrastructure because a claimed attestation can be reproduced and compared to the stored record without involving the original signer.

Verification takes the public key, the message, and the signature, then checks whether the response vector lies within the expected norm bound and whether it is consistent with the commitment. Verification requires no secret

material and no connection to the signing infrastructure. Any party holding the public key and the signature can verify offline, using any standards-compliant ML-DSA implementation. This is the foundation of the third-party verifiability described in Section 11.

Size comparison with classical schemes

SCHEME	PUBLIC KEY	SIGNATURE	QUANTUM-SAFE	STANDARD
ECDSA P-256	64 bytes	64 bytes	No	FIPS 186-4
Ed25519	32 bytes	64 bytes	No	RFC 8032
RSA-2048	256 bytes	256 bytes	No	PKCS#1
ML-DSA-65	1,952 bytes	3,309 bytes	Yes	NIST FIPS 204

The 3,309-byte ML-DSA-65 signature is approximately 52 times larger than an ECDSA P-256 signature. For high-frequency per-record signing at volume, this difference matters. For AI governance attestation where each signed record covers the Merkle root of an entire batch of governed decisions, a single signature amortises across all records in the batch, making the size overhead operationally manageable. This Merkle tree integration approach is described in detail in Section 9.

Implementation availability

Production-ready ML-DSA implementations are available across all major platforms. For Node.js and TypeScript, the `@noble/post-quantum` library provides a zero-dependency, audited implementation that passes NIST FIPS 204 test vectors. Python deployments can use the `pyca/cryptography` package, which added ML-DSA support in 2024, or the `pqcrypto` bindings. Java developers have Bouncy Castle 1.76 and later. a certified HSM supports ML-DSA signing operations directly. The algorithm is implementable without proprietary dependencies or licence fees, and the reference implementation is provided by NIST as part of the FIPS 204 publication package.

SECTION 6

Why Attestation Records Are Uniquely Exposed

Not all signed data carries the same quantum risk profile. The exposure is highest for data whose validity must be maintained far into the future and where a successful signature forgery would be indistinguishable from a genuine record. Attestation records meet both criteria more directly than almost any other category of signed data.

The long-validity problem

The quantum threat materialises as a function of two timescales intersecting: the expected arrival of a cryptographically relevant quantum computer, and the retention period of the records in question. A signed communication that needs to remain confidential for two years carries negligible SNDL risk even under aggressive quantum timeline assumptions. A signed compliance certificate that must remain valid and verifiable for twenty years carries substantial risk even under conservative assumptions.

Three data categories face the longest validity requirements: classified government records (twenty-five to seventy-five year retention in most jurisdictions), medical records (typically eight to thirty years after last treatment, with certain categories retained permanently), and legal and compliance records (indefinite retention in many contexts, particularly where ongoing litigation or regulatory obligations exist). All three categories generate large volumes of signed attestations. All three face retention periods that overlap meaningfully with the Q-Day timeline.

The forgery-versus-decryption distinction

For encrypted data, SNDL produces confidentiality loss: the adversary reads the plaintext. For signed records, SNDL produces something worse: forgery capability. An adversary who can forge signatures can produce new signed records dated to any point in the past that verify correctly against the original public key. This is not a theoretical weakness in the content of the record; it is the collapse of the evidentiary framework around it. A signed audit trail where any record can be forged is no audit trail at all.

This distinction is why attestation records require quantum-safe migration before encrypted communications in many AI governance contexts. The confidentiality failure from SNDL against a governance record is relatively contained: the adversary learns what decision was made. The integrity failure is catastrophic: the adversary can fabricate any decision they choose, with signatures that appear authentic.

AI governance records specifically

AI governance records inherit the long-validity and forgery-risk characteristics of other compliance records, and add a domain-specific exposure driven by the regulatory trajectory of AI deployment. The EU AI Act, which entered into force in August 2024, requires providers of high-risk AI systems to maintain technical documentation throughout the system lifecycle. High-risk AI systems in employment, credit, healthcare, law enforcement, and public administration may operate for decades. The documentation maintained under the Act includes records of AI decisions and the policies that governed them.

An organisation that produces AI governance records signed with ECDSA in 2026 and migrates to quantum-safe signatures in 2030 will have a four-year gap in its audit trail where records are classically signed and therefore eventually forgeable. A regulator or court examining a decision made in 2027 will be presented with a record that, in time, cannot be proven genuine and cannot be proven false. The only way to avoid this outcome is to start with quantum-safe signatures from the point of generation.

SECTION 7

The Attestation Gap in AI Governance

The majority of AI governance infrastructure currently in production either does not sign its audit records at all or signs them with classical algorithms. This is not negligence: it reflects the state of tooling and standards as recently as two years ago, before NIST FIPS 204 was finalised and before the quantum timeline compressed as sharply as it has. The gap is structural and widespread, and its significance is only now becoming clear to compliance and engineering teams who are beginning PQC readiness assessments.

Log-based audit trails and their cryptographic gaps

The dominant pattern for AI audit infrastructure is application logging: the AI system writes records to a database or structured log store, records are retained according to a policy, and records are retrieved during audits. Log-based approaches carry no cryptographic integrity guarantees unless additional tooling is layered on top. Any party with sufficient database access can alter log records after the fact. Compliance frameworks acknowledge this weakness but typically address it through access controls and procedural safeguards rather than cryptographic integrity proofs. These controls work when the threat is a rogue administrator. They offer no protection when the threat is a quantum computer applied to classically-signed log certification records.

Classical signatures where present

Where AI audit records are cryptographically signed, ECDSA and RSA are the common choices, the same algorithms used for TLS certificates and code signing. These are technically sound choices for data that does not need to withstand quantum computing attacks, but they are structurally insufficient for the retention periods that AI governance records require. An AI governance record signed with ECDSA P-256 in 2026 carries exactly the same eventual vulnerability as a record signed in 2020: it is classically secure today and will be quantum-vulnerable when Q-Day arrives, whenever that proves to be.

The compliance-creation gap

The compliance-creation gap is the period during which classically-signed governance records accumulate while a PQC migration is planned, procured, and deployed. A migration that begins in 2027 and completes in 2028 leaves a two-year gap. A migration delayed to 2030 leaves a four-year gap. Records in the gap carry signatures that will eventually become forgeable. The gap cannot be closed retroactively by post-quantum signing of new records: old records carry old signatures. The only way to eliminate the compliance-creation gap is to adopt quantum-safe signatures before the gap opens, not after it has accumulated.

We think AI governance infrastructure procured in 2026 should not have a compliance-creation gap at all. This requires post-quantum attestation from day one of deployment, not from a future migration date. Section 8 describes how AffixIO's production system achieves this.

SECTION 8

AffixIO's Production Implementation

AffixIO has deployed ML-DSA-65 attestation in production at a production deployment since late 2025. The system signs Merkle tree roots covering AI governance proofs generated by the platform's zero-knowledge proof pipeline. Every AI response governed by the platform produces a ZK proof that is anchored in a Merkle tree; the Merkle root after each proof insertion is signed with ML-DSA-65 using a key held in an a certified HSM cluster operating in the a designated region region.

The decision to use ML-DSA-65 in production from the outset, rather than deferring migration, was driven by three considerations. First, the record retention requirements of AI governance: governance records must remain valid and verifiable for the lifetime of the deployed system, which is measured in years to decades. Second, procurement signals from regulated-industry customers: banks and government agencies are already asking PQC readiness questions in vendor security questionnaires, and the answer "we deploy ML-DSA-65 in production today" is qualitatively different from "we have a migration roadmap." Third, the stability of the standard: FIPS 204 was finalised after eight years of public review, and implementing it now does not carry the uncertainty associated with draft or candidate algorithms.

The signing flow

The signing flow integrates into the ZK proof verification pipeline documented in AffixIO's WP-001 (Cryptographic AI Governance: A Technical Framework). When a ZK proof is verified, the following sequence executes synchronously before the proof response is returned:

1. The proof digest is computed as `a composite digest over circuit identity, outcome, and proof material`
2. The proof digest is inserted as a new leaf in the Merkle tree using SHA-256 sorted-pair hashing
3. The new Merkle root is computed over the complete leaf set
4. The attestation payload is assembled: `{ attestation_id, proof_digest, tree_root, tree_leaf_hash, signed_at }`

5. The SHA-256 hash of the JSON-serialised attestation payload is submitted to a certified HSM signing endpoint via KMS
6. The HSM executes the ML-DSA-65 signing operation internally and returns only the signature
7. The attestation record is included in the proof verification response

Total signing latency from submitting the signing request to receiving the signature is consistently under 50 milliseconds in the eu-west-2 region under normal load. This is well within the overall proof pipeline latency budget.

Attestation record structure

```
{
  "signed_at":             "2026-06-14T09:41:22.103Z",
  "payload_digest":        "a3f8c2...",    // SHA-256 of the attest
  "mldsa_signature_b64":  "BQID...",      // Base64-encoded ML-DSA-
  "algorithm":             "ML-DSA-65",  // NIST FIPS 204, Securit
  "standard":              "NIST FIPS 204"
}
```

Key management

The ML-DSA-65 key pair is generated inside the CloudHSM hardware using the HSM's certified random number generator. The private key never exists outside the HSM boundary in plaintext form. The public key is exported and published at a well-known URL under the AffixIO domain, making it available for third-party verification without any interaction with AffixIO's infrastructure. Key rotation follows an annual schedule. When a rotation occurs, both the old and new public keys are retained in the published key set, so that records signed under either key remain verifiable for their full retention period.

The @noble/post-quantum library

For software-based attestation in development environments and testing pipelines, AffixIO's implementation uses the `@noble/post-quantum` JavaScript library, a zero-dependency, audited, pure-JavaScript implementation of ML-DSA-65 maintained by Paulmillr. The library implements the full FIPS 204

specification and is used across the development and staging environments, with the production system substituting the HSM signing call at the key custody layer. This approach ensures that the same signing and verification code paths are exercised during development as in production, with the only difference being where the private key operation executes.

SECTION 9

Integration with Zero-Knowledge Proof Pipelines

Zero-knowledge proofs and post-quantum signatures address different problems in the same trust pipeline and compose naturally to produce a privacy-preserving, quantum-safe audit infrastructure. This section describes the composition pattern used in AffixIO's production system and the general principles that apply to any audit infrastructure combining ZK proofs with long-lived record integrity requirements.

Complementary problem sets

A ZK proof addresses a computation question: did this circuit evaluate to this result, given inputs satisfying these constraints? It provides a mathematical guarantee that the computation was performed correctly, without revealing the private inputs used. What it does not provide is a guarantee that the proof exists as a specific object that has not been tampered with after the fact, or that it exists at all without the prover's cooperation. That is the role of the attestation signature.

An ML-DSA-65 signature addresses a record integrity question: was this specific byte sequence produced by the holder of this key at this time, and has it been unmodified since? It provides tamper-evidence for any signed object, regardless of that object's content or structure. It does not substitute for the ZK proof; it provides a quantum-safe integrity envelope around the record that proves the ZK computation occurred.

The two technologies are not alternatives: they are complements. The ZK proof provides computation correctness and data privacy by proving a statement without revealing inputs. The ML-DSA-65 signature provides long-lived, quantum-safe record integrity by cryptographically binding the proof record to a specific time and key. Together they produce a privacy-preserving audit record that is computationally verifiable, tamper-evident, and resistant to quantum forgery.

Merkle tree as the binding layer

The Merkle tree sits between the ZK proof layer and the ML-DSA-65 signing layer. Its role is to allow one signature to attest to an arbitrarily large set of proofs without requiring a separate ML-DSA-65 signature per proof. Rather than signing each proof individually, which would produce one 3,309-byte signature per proof, the system signs the Merkle root after each proof insertion. The root encodes the complete ordered set of proofs anchored to date. Any individual proof's inclusion can be verified using a Merkle path logarithmic in the total number of proofs, typically under 400 bytes for a tree with tens of thousands of leaves.

This architecture means the storage overhead of ML-DSA-65 is bounded at approximately one signature per proof insertion in the current deployment. The signature covers all proofs in the cumulative set. A verifier checking the thousandth proof can do so using only the leaf hash, the Merkle path for that specific leaf, and the signed root, without processing or accessing any of the other 999 proofs.

Verifiable credentials and the attestation bundle

The proof-attestation structure produced by AffixIO is compatible with the W3C Verifiable Credentials Data Model. A Verifiable Credential wraps a set of claims, a proof section containing the cryptographic proof, and issuer metadata. AffixIO's attestation record maps naturally onto this structure: the claims describe the AI governance decision (circuit evaluated, outcome, timestamp), the proof section contains the ML-DSA-65 signature and the Merkle inclusion path, and the issuer is identified by the published public key. Organisations that already operate verifiable credentials infrastructure for privacy-preserving identity verification can integrate AI governance attestations into the same credential pipeline without requiring a separate record format.

The proof-attestation binding

The binding between a specific ZK proof and its ML-DSA-65 attestation is established through the leaf hash. The leaf hash is computed as the SHA-256 hash of a structured payload encoding the proof digest, circuit ID, proof ID,

event type, and timestamp. A verifier who holds the leaf hash, the Merkle sibling path, and the signed root can verify inclusion using three operations: reconstructing the Merkle path via SHA-256, comparing the result with the signed root, and verifying the ML-DSA-65 signature against the published public key. No AffixIO infrastructure needs to be reachable for this verification sequence to complete.

SECTION 10

HSM Key Custody and FIPS 140-2 Level 3

The security of a quantum-safe attestation system depends entirely on the security of the private signing key. An adversary with access to the private key can sign any record they choose, including fabricated records dated to any point in the past, defeating the entire purpose of the attestation infrastructure. Quantum-safe key custody is not a secondary concern after algorithm selection: it is the primary security requirement of the system.

Hardware Security Modules

A Hardware Security Module is computing hardware designed specifically to generate, store, and use cryptographic keys without exposing those keys to the surrounding computing environment. An HSM performs signing operations internally: the private key never leaves the hardware boundary, and the only external interface is to submit data and receive a signature. This is architecturally distinct from software key stores, where the private key exists in process memory on a general-purpose server and could be extracted by an attacker with operating-system-level access.

FIPS 140-2 Level 3 requirements

NIST's FIPS 140-2 standard defines four validation levels for cryptographic hardware. Level 3 adds physical tamper-evidence and tamper-response requirements to the logical controls required at Level 2. A Level 3 validated HSM must detect physical attempts to access its internal components, respond to detected tampering by automatically zeroising all key material, maintain a physical enclosure that requires destruction to bypass, and authenticate all operators before permitting cryptographic operations.

The consequence of these requirements is that physical possession of the HSM hardware does not yield the private key. An attempt to extract the key physically triggers its deletion. Key extraction requires either authenticated logical access via the defined interface, protected by access control policies, or a successful attack on the tamper-response circuitry itself, which is a sophisticated hardware attack beyond the capability of most threat actors.

a certified HSM in eu-west-2

AffixIO deploys an a certified HSM cluster in the a designated region region, operating in FIPS 140-2 Level 3 validated mode. The cluster maintains a minimum of two HSM instances in separate availability zones for high availability. AWS manages the physical infrastructure; AffixIO controls the key material and access policies exclusively. AWS has no access to keys generated within the customer partition of a CloudHSM cluster: this is a structural property of the CloudHSM architecture, not a policy commitment.

Signing requests are routed through a key management service (KMS). KMS resource policies specify which authorised service accounts are authorised to perform signing operations. The proof pipeline's service account is the only principal with signing permission. All signing requests and authorisation decisions are logged in AWS CloudTrail, which is itself tamper-evident. An independently auditable record of every signing operation exists separately from the application infrastructure.

Key backup and rotation

a certified HSM supports encrypted key backup to AWS S3, with the backup key wrapped under a key held in the HSM cluster. A backup can only be restored to the same cluster or to a cluster in the same CloudHSM account, preventing exfiltration via the backup path. Key rotation follows an annual schedule: a new ML-DSA-65 key pair is generated inside the HSM, the new public key is published alongside the old one, and the signing configuration is updated to use the new key for new attestations. Old keys are retained in the HSM for the full retention period of records signed under them, ensuring that records remain verifiable throughout their required retention period under the key used when they were signed.

SECTION 11

Third-Party Verification and Digital Sovereignty

A core design requirement for AffixIO's attestation system is that third-party verification must not require AffixIO's infrastructure to be operational. A governance record that can only be verified by querying a live AffixIO endpoint is dependent on AffixIO's continued operation and reachability. For records that may need to be verified in litigation or regulatory proceedings years hence, potentially after the company that generated them has been acquired, restructured, or dissolved, this dependency is an unacceptable single point of failure. Digital sovereignty in this context means that the verifiability of an attestation record is a property of the record itself, not of the vendor's infrastructure.

What a third-party verifier needs

To verify that a specific ZK proof is included in a quantum-safe signed Merkle tree, a third-party verifier needs four things:

1. **The leaf hash**: identifies the specific proof within the tree
2. **The Merkle sibling path**: the sequence of sibling hashes from the leaf to the root
3. **The signed Merkle root**: the root value and its ML-DSA-65 signature
4. **The ML-DSA-65 public key**: to verify the signature on the root

None of these items requires a live query to AffixIO infrastructure. The leaf hash and Merkle path are returned at proof verification time and should be retained as part of the governance record. The signed Merkle roots are published to a publicly readable log. The public key is published at a stable URL and should be downloaded and retained at the time of proof generation, just as a TLS certificate should be retained alongside the record it authenticates.

Self-contained verification bundles

For long-lived records, AffixIO recommends packaging proof records as self-contained verification bundles. A verification bundle is a single JSON document containing all four verification inputs. An organisation that retains verification bundles alongside its governance records can verify those records at any future point using any FIPS 204-compliant ML-DSA implementation, with no dependency on any external service.

```
{
  "version":        1,
  "leaf_hash":      "a3f8c2d1...",
  "merkle_path":    [
    { "sibling": "b9e4f2...", "side": "left" },
    { "sibling": "d4c7a1...", "side": "right" }
  ],
  "signed_root":    {
    "root":      "f1a2b3...",
    "signed_at": "2026-06-14T09:41:22.103Z",
    "signature": "BQID..."
  },
  "public_key_b64": "BQID...",
  "algorithm":      "ML-DSA-65",
  "standard":       "NIST FIPS 204"
}
```

Verification proceeds in three steps: (1) recompute the Merkle root from the leaf hash and the sibling path, (2) verify the ML-DSA-65 signature on the root using the public key, (3) confirm the recomputed root matches the signed root. All three steps use publicly documented algorithms requiring no proprietary code.

Open standards and implementation diversity

Because ML-DSA-65 is an open NIST standard and SHA-256 Merkle trees are specified in published RFC documents, a verifier can implement the full verification sequence in any programming language using publicly available specifications. They have no dependency on AffixIO's software, libraries, or proprietary format. This is the definition of digital sovereignty for an attestation record: verifiability is a property of open cryptographic standards, not of vendor availability.

SECTION 12

Regulatory Drivers

Post-quantum migration for attestation infrastructure is no longer solely a forward-looking engineering decision. Regulatory mandates are now in force for national security systems, and procurement expectations are spreading rapidly across regulated industries. The regulatory landscape as of mid-2026 reflects a coordinated international shift from voluntary preparation to mandatory compliance timelines.

NSA CNSA 2.0

The US National Security Agency published the Commercial National Security Algorithm Suite 2.0 (CNSA 2.0) in September 2022 and has updated its implementation guidance through 2025 and 2026. For digital signatures in National Security Systems, CNSA 2.0 sets three milestones: planning and inventory by 2025; new systems must use quantum-safe algorithms by 2027; all systems must complete migration by 2033. ML-DSA (referenced as CRYSTALS-Dilithium in earlier guidance) is the designated algorithm for digital signatures under CNSA 2.0. Contractors providing software or services to US government customers in national security contexts face these timelines in vendor questionnaires and contract requirements.

NIST IR 8547 and the deprecation schedule

NIST's Internal Report 8547, published in 2024, provides the federal transition roadmap. It recommends deprecating classical public-key algorithms after 2030 and disallowing their use after 2035 for most federal applications. For organisations with long-term evidence retention requirements, NIST explicitly recommends beginning PQC migration for signature schemes before 2030, because records generated in the 2026 to 2030 window will carry signatures under an algorithm that will be formally deprecated before those records' retention periods end.

UK NCSC and NIS2 alignment

The UK National Cyber Security Centre has published post-quantum guidance aligned with NIST's recommendations, with particular emphasis on critical national infrastructure and government supply chain participants. The NIS2 Directive, which entered into force across EU member states in October 2024, requires covered entities to maintain "state of the art" cryptographic security. The European Union Agency for Cybersecurity (ENISA) has confirmed that post-quantum cryptography is within scope of NIS2's state-of-the-art requirement for organisations with long-lived signed records. UK G-Cloud supplier frameworks are expected to include PQC readiness criteria as NIST deprecation timelines approach.

EU AI Act and long-lifecycle technical documentation

Article 11 of the EU AI Act requires providers of high-risk AI systems to maintain technical documentation throughout the system lifecycle. For AI systems in employment, credit, healthcare, law enforcement, and public administration, "the system lifecycle" may span many years or decades. Documentation maintained under Article 11 that includes signed attestation records carries an implied requirement that those signatures remain verifiable for the full documentation period. The Act does not prescribe cryptographic algorithms, but the practical consequence of long-lifecycle documentation combined with NIST's deprecation timeline is that quantum-safe signatures are the only approach that will remain compliant for the full lifecycle of AI systems deployed today.

The EU AI Act's August 2026 compliance deadline for certain obligations, combined with active CNSA 2.0 timelines and NIS2 enforcement, creates a convergence of pressure that makes 2026 the natural inflection point for post-quantum attestation adoption in regulated AI deployments.

Procurement signals

Across banking, insurance, healthcare, defence contracting, and central government, vendor security questionnaires have begun to include PQC readiness questions. The questions are not yet standardised but cover consistent ground: does the vendor have a PQC migration plan, are any

production systems currently using post-quantum algorithms, and what is the timeline to full migration? Being able to demonstrate production ML-DSA-65 deployment rather than a migration roadmap is a material differentiator in regulated-industry procurement conversations.

SECTION 13

Migration from Classical Signature Schemes

Organisations that currently sign AI governance records with ECDSA or RSA and want to migrate to ML-DSA-65 face a set of practical decisions worth addressing directly. Achieving full crypto-agility, the ability to swap cryptographic algorithms without changing the surrounding application architecture, requires planning at the key infrastructure, library, protocol, and historical record layers. Migration is an engineering project, not a cryptographic research project, but the engineering effort is non-trivial.

Key infrastructure

The most significant infrastructure change is at the key custody layer. Current deployments using software key stores or standard TLS certificate infrastructure do not support ML-DSA-65 keys. The available options are as follows.

HSM deployment is the recommended path for production attestation: CloudHSM, Thales Luna, or nShield HSMs with ML-DSA support provide FIPS 140-2 Level 3 key custody with quantum-safe algorithm support. AWS KMS supports ML-DSA signing via CloudHSM integration. Google Cloud KMS and Azure Key Vault have post-quantum support on their published roadmaps but are not yet in general availability as of mid-2026.

Software implementation using `@noble/post-quantum` or `liboqs` with keys held in encrypted software key stores is appropriate for development environments, lower-assurance deployments, and as a verification-only path for applications that do not perform signing. Software key stores do not provide the tamper-response properties of FIPS 140-2 Level 3 hardware and should not be used for production signing of high-assurance attestation records.

Crypto-agility in signing infrastructure

Crypto-agility is the property of a system that allows its cryptographic algorithms to be updated without redesigning the surrounding application. A crypto-agile signing infrastructure achieves this by abstracting the signing

operation behind an interface layer that does not encode a specific algorithm in the calling code. The interface accepts a payload and returns a signature with algorithm metadata; the underlying implementation can be switched from ECDSA to ML-DSA-65 by changing the implementation behind the interface without changing any code that calls it. AffixIO's signing interface follows this pattern: the attestation record's `algorithm` and `standard` fields carry the algorithm identifier, and verifiers are expected to dispatch on this field rather than assuming a fixed algorithm. This makes the system crypto-agile from day one.

Handling historical records

The most common migration planning question concerns historical records signed with classical algorithms. Three approaches are available, with different trade-offs.

The first is to acknowledge the boundary and document it clearly. Records generated before the migration carry classical signatures; records generated after carry ML-DSA-65 signatures. The migration date is the documented boundary. Historical records are managed under a separate retention and verification policy that acknowledges their classical-signature status. This is the pragmatic approach: the compliance gap is bounded and transparent.

The second is to counter-sign historical records with ML-DSA-65. Each historical record receives an additional ML-DSA-65 signature atesting that the record existed in its current state at the time of counter-signing. The counter-signature provides quantum-safe tamper-evidence from the counter-signing date forward. It does not retroactively secure the original creation timestamp, but it does provide a quantum-safe guarantee of record integrity from migration onwards.

The third approach, recommended for large historical record sets, is Merkle re-anchoring. All historical records are hashed, their hashes are inserted as leaves in a new Merkle tree, and the root is signed with ML-DSA-65. The signed root provides a quantum-safe attestation that the complete set of historical records existed in their current form at the time of re-anchoring, with a single ML-DSA-65 signature covering the entire history. Individual records can be verified via their Merkle paths against the re-anchoring

signature. The original records retain their classical signatures; the re-anchor provides a quantum-safe completeness guarantee from the re-anchoring date.

Hybrid signatures during transition

For systems that must maintain backward compatibility with verifiers that do not yet support ML-DSA-65, hybrid signatures allow a transition period during which each record carries both a classical signature and an ML-DSA-65 signature. A classical verifier uses the classical signature; an ML-DSA-65-capable verifier uses the post-quantum one. Both signatures are stored with the record. Once all verifiers in the ecosystem have been updated, the classical component can be dropped. The hybrid approach adds one classical signature worth of overhead per record during the transition and requires no changes to the record format beyond carrying both signatures.

SECTION 14

Performance and Crypto-Agility Considerations

ML-DSA-65 is computationally efficient on modern server hardware. The operations relevant to an attestation pipeline, specifically key generation, signing, and verification, execute in sub-millisecond time on current x86-64 and ARM64 hardware. The operational considerations that matter in practice are dominated by signature size, HSM round-trip latency, and the crypto-agility requirements for the surrounding infrastructure.

Algorithm performance

OPERATION	ML-DSA-65 (APPROX.)	ECDSA P-256 (APPROX.)
Key generation	0.3 ms	0.1 ms
Signing	0.8 ms	0.2 ms
Verification	0.5 ms	0.5 ms

Approximate figures for a modern x86-64 server running reference C implementations. HSM round-trip latency (network plus hardware processing) adds 5 to 50 milliseconds depending on deployment topology, and dominates the total signing time in practice.

For AI governance systems where signing occurs once per AI response, typically at most a few times per second per active user, the algorithm-level performance difference between ML-DSA-65 and ECDSA is irrelevant. The HSM network round-trip is the binding constraint, and it is identical regardless of which signing algorithm the HSM executes. At AffixIO, total signing latency including HSM round-trip is consistently under 50 milliseconds in the eu-west-2 region, well within the overall proof pipeline budget.

Storage considerations at scale

A 3,309-byte ML-DSA-65 signature is 52 times larger than a 64-byte ECDSA P-256 signature. For systems signing individual records at millions per day, this difference has meaningful storage cost implications. The Merkle tree architecture addresses this directly: AffixIO signs one root per proof insertion, and each signed root covers the cumulative set of all anchored proofs. The per-record storage overhead attributable to the quantum-safe signature is approximately 3,309 bytes per signing session, not per proof. At current AI governance proof volumes, the total signed root storage is measured in megabytes per year, not gigabytes.

For systems that require per-record individual signatures rather than a Merkle architecture, the storage trade-off must be evaluated against retention requirements. A system retaining 10 million individually ML-DSA-65-signed records per year will require approximately 33 gigabytes of signature storage annually. This is material but not prohibitive for most enterprise storage budgets, and should be weighed against the governance risk of classically signed records with long retention requirements.

Designing for crypto-agility

Crypto-agility in a production system means that the algorithm identifier, key identifier, and signature are always stored together with the signed record, and that the verification path dispatches on the algorithm identifier rather than assuming a fixed scheme. This design prevents the brittleness of systems where a future algorithm change requires a migration of all verification logic alongside the signing infrastructure. AffixIO's attestation record always carries `"algorithm": "ML-DSA-65"` and `"standard": "NIST FIPS 204"` explicitly, so that future algorithm changes, whether to ML-DSA-87 or to a post-lattice scheme, require only an update to the signing implementation rather than a schema change. Engineering teams designing new attestation infrastructure should build algorithm negotiation into the record format from the start, even if they begin with a single algorithm.

SECTION 15

Known Limitations

The following limitations apply to ML-DSA-65 as deployed in AffixIO's production system and to the current state of post-quantum attestation more broadly. They are described here to support accurate evaluation by engineering teams and compliance officers.

Signatures are substantially larger than classical alternatives

The 3,309-byte ML-DSA-65 signature is roughly 52 times larger than ECDSA P-256. For protocols that embed signatures in space-constrained formats, such as compact JWTs, short certificate extensions, or embedded firmware headers, this size is prohibitive without protocol changes. The Merkle tree architecture amortises the cost for bulk record attestation but does not help for protocols that require individual per-record signatures in constrained formats. Organisations with such constraints should evaluate FN-DSA (NIST FIPS 206), which provides more compact signatures at the cost of a stateful signing scheme that requires more careful key management.

HSM vendor support is not yet universal

As of mid-2026, ML-DSA-65 HSM support is available from a certified HSM and selected Thales and nShield hardware configurations. It is not yet universally available across all HSM vendors or cloud KMS offerings. Organisations with existing HSM estates from vendors without ML-DSA support may face a hardware refresh or a transition period during which ML-DSA signing is handled by a separate HSM running alongside existing hardware. This is a temporary limitation that will resolve as vendor support broadens, but it affects migration timelines for organisations with established HSM infrastructure.

The underlying ZK proof construction is not quantum-safe

AffixIO's ZK proofs use the production proving system, based on the BN254 elliptic curve. BN254 does not provide post-quantum security: Shor's algorithm applied to the discrete logarithm problem on BN254 would, in

principle, enable a quantum attacker to forge SNARK proofs. The ML-DSA-65 signature provides a quantum-safe tamper-evidence layer around the proof record, but the proof itself uses a classically secure construction. For AI governance use cases where the ZK proof's function is to prove computation correctness without revealing private inputs, this is an acceptable trade-off: the ML-DSA-65 attestation layer ensures that the record of the proof cannot be retroactively altered or fabricated, even if the proof format is eventually found to be quantum-vulnerable. Post-quantum ZK proof constructions based on hash functions (zkSTARKs) are available and would close this gap in a future version of the system.

M-LWE security depends on parameter correctness

ML-DSA-65's security claim rests on the hardness of M-LWE with the specific parameters chosen by NIST. The security analysis has been validated over eight years of public cryptanalysis, and NIST's parameter choices include a margin above the minimum required security level. However, the security claim is not unconditional: if a future algorithm reduces M-LWE more efficiently than currently known techniques, the effective security level could be lower than claimed. This is a residual risk inherent in any specific cryptographic algorithm and is not unique to ML-DSA-65: it applies equally to AES, SHA-256, and every other deployed cryptographic primitive.

Single-region HSM deployment in current production

AffixIO's production signing key is held in a single CloudHSM cluster in eu-west-2. A prolonged AWS regional outage affecting eu-west-2 would suspend proof signing for the duration of the outage. Records generated during an outage period can be signed retroactively once the cluster is restored, but the signing timestamp would reflect the retroactive signing time rather than the original proof generation time. Organisations requiring strict signing latency guarantees for compliance purposes should plan for a dual-region HSM configuration with automatic failover. A US-region CloudHSM cluster with a separate ML-DSA-65 key pair is planned for AffixIO's second operational phase.

SECTION 16

Conclusion

The case for quantum-safe attestation is built on three facts that are now matters of public record rather than projections. The quantum computing resource estimates for breaking classical cryptography have fallen by a factor of two hundred in seven years, driven by engineering rather than theory. Regulatory mandates for post-quantum migration are in force for national security systems and spreading to regulated commercial sectors via CNSA 2.0, NIST IR 8547, NIS2, and EU AI Act requirements. And ML-DSA-65, standardised as NIST FIPS 204 after eight years of public cryptanalysis, is deployable in production today using available hardware, open libraries, and standard cloud infrastructure.

For organisations generating AI governance records, compliance attestations, and cryptographic audit trails in 2026, the question is no longer whether to adopt post-quantum signatures but when. The answer that this paper makes the case for is: from the point of generation of records that must remain valid for years or decades. Every month of delay extends the compliance-creation gap, the period during which classically-signed records accumulate that will eventually be quantum-vulnerable.

AffixIO's production deployment demonstrates three things that are worth stating plainly:

- Quantum-safe attestation is operational today, not aspirational. ML-DSA-65 runs in the AffixIO production proof pipeline at governance-relevant latencies on standard AWS infrastructure.
- The Merkle tree architecture efficiently amortises the signature size overhead of ML-DSA-65 across large sets of governance records, making the storage cost practical even at AI governance volumes.
- Third-party verification without vendor dependency is achievable. Self-contained verification bundles give any party with a FIPS 204-compliant implementation the ability to verify a governance record offline, for as long as the open standard remains supported, regardless of AffixIO's operational status.

The limitations described in Section 15 are real and should be evaluated honestly. The underlying ZK proof construction is classically secure but not quantum-safe. HSM vendor support for ML-DSA-65 is not yet universal. Signature sizes are substantially larger than classical alternatives. These are engineering trade-offs with known mitigation paths, not fundamental blockers. The cost of adopting ML-DSA-65 now is an increase in signature size, an HSM procurement decision, and a library dependency. The cost of deferring is a growing body of classically-signed governance records against a future in which those signatures are forgeable and the audit trail they compose is worthless as evidence.

PQC readiness in 2026 is not a future compliance exercise. It is the present engineering decision that determines whether the AI governance records being generated today will still be trustworthy in the proceedings and audits of ten years from now. The infrastructure to get this right exists and is production-proven. The regulatory direction is clear. The time is now.

AffixIO's quantum-safe attestation infrastructure is available for evaluation. Proof records, signed Merkle roots, and ML-DSA-65 attestations from the production system are publicly readable at a production deployment. The published public key and verification methodology described in Section 11 are sufficient for any third party to verify any production record without contacting AffixIO.

Related reading

- WP-019: Post-Quantum PKI Migration: ML-KEM and ML-DSA in Production
- WP-011: Merkle Tree Audit Architecture for AI Decision Systems
- WP-003: The Proof-Not-Log Paradigm for AI Audit Trails

Frequently asked questions

Why ML-DSA-65 specifically?

It is NIST's module-lattice standard with conservative parameter sets suited to long-lived governance records.

When does quantum break RSA for archives?

Nobody knows the exact date, but captured signatures are already at risk under harvest-now-decrypt-later.

Where are keys stored?

a certified HSM in eu-west-2 with private keys non-exportable from hardware.