WHITEPAPER

Achieving CNSA 2.0 Compliance at the Edge

A practical guide to managing ML-KEM and ML-DSA cryptography in disconnected and constrained deployments

April 2026

nova8 Technologies

nova8.io

Table of Contents

1. Executive Summary

The NSA Commercial National Security Algorithm Suite 2.0 (CNSA 2.0) establishes a timeline and algorithm selection for transitioning national security systems to quantum-resistant cryptography. Published in September 2022 and updated in 2024, the advisory specifies ML-KEM (FIPS 203) for key establishment and ML-DSA (FIPS 204) for digital signatures, with deadlines that vary by system category but converge on full enforcement by 2033 for software and firmware signing, and complete transition by 2035.

For organizations operating edge infrastructure in defense, critical infrastructure, and industrial environments, CNSA 2.0 compliance is not primarily an algorithm selection exercise. The algorithms are specified. The challenge is operational: sustaining correct algorithm selection, key rotation cadence, signature verification policy, and cryptographic library currency across every device in a fleet, for years, through hardware refresh cycles, software updates, and evolving threat guidance.

This whitepaper examines the specific operational challenges that edge deployments face in achieving and maintaining CNSA 2.0 compliance. It addresses why disconnected operation, long deployment lifetimes, and constrained compute budgets change the requirements for cryptographic policy management, how immutable images and atomic updates can simplify compliance, what evidence teams need to produce for auditors and authorizing officials, and what practical steps organizations should take to begin the transition. The paper draws exclusively on publicly available standards and guidance from NIST, NSA, IETF, and the UAPI Group. It is intended for security leads, platform engineers, and compliance teams evaluating post-quantum readiness for edge deployments.

2. The Operational Problem: CNSA 2.0 Is Lifecycle Management

NSA's CNSA 2.0 advisory establishes a timeline for transitioning national security systems to quantum-resistant cryptography. The guidance specifies ML-KEM (FIPS 203) for key establishment and ML-DSA (FIPS 204) for digital signatures, with deadlines that vary by system category but converge on full enforcement by 2033.

What makes CNSA 2.0 harder than prior algorithm transitions is scope. It touches every cryptographic dependency in the stack simultaneously: TLS session establishment, firmware signing, boot integrity measurement, key encapsulation for data at rest, certificate issuance, and update authentication. Enabling one algorithm in one subsystem is a necessary first step, but it is not compliance.

The real operational challenge is lifecycle management: sustaining correct algorithm selection, key rotation cadence, signature verification policy, and cryptographic library currency across every device in a fleet, for years, through hardware refresh cycles, software updates, and changing threat guidance. CNSA 2.0 compliance is a continuous property, not a point-in-time certification event.

Prior algorithm transitions (such as the migration from SHA-1 to SHA-256, or from RSA-1024 to RSA-2048) affected individual subsystems and could be managed incrementally. CNSA 2.0 requires simultaneous attention to key exchange, digital signatures, and hash functions across every protocol layer. The interdependencies between these transitions create coordination challenges that grow with fleet size and deployment complexity.

The CNSA 2.0 timeline provides distinct milestones. New system acquisitions should prefer post-quantum algorithms by 2027. All software and firmware signing must use post-quantum algorithms by 2033. Full transition to quantum-resistant cryptography must be complete by 2035 (see Figure 1). For systems being fielded today with expected operational lifetimes of ten years or more, these deadlines are not distant planning horizons; they fall within the deployed lifecycle of current hardware.

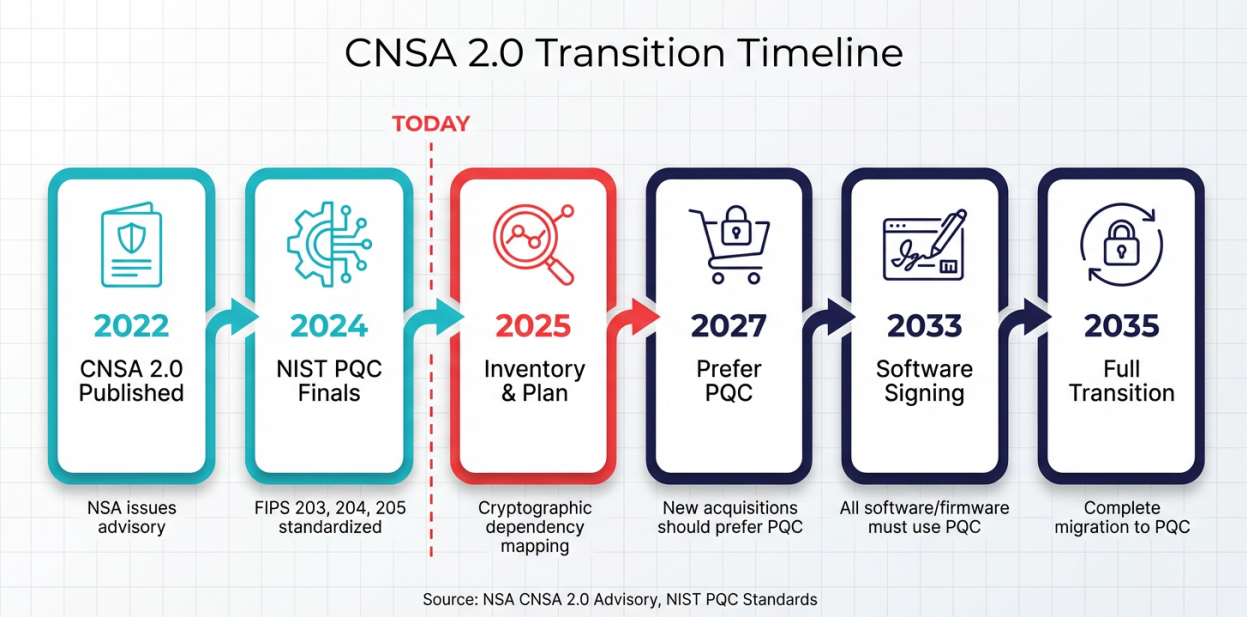

Figure 1: CNSA 2.0 transition timeline from advisory publication through full post-quantum migration.

3. Why Edge Constraints Make CNSA 2.0 Harder

Edge deployments introduce three compounding constraints that enterprise data-center transitions do not face. First, devices are often disconnected or intermittently connected, meaning cryptographic policy changes cannot depend on real-time coordination with a central authority. Second, edge hardware is long-lived: military, industrial, and critical-infrastructure systems routinely operate for 10 to 15 years, spanning multiple algorithm transition cycles. Third, compute and memory budgets are constrained, which limits which post-quantum parameter sets are practical.

Size and Bandwidth Implications

ML-KEM-768 ciphertexts are roughly 30 times larger than equivalent X25519 key shares, and ML-DSA-65 signatures are roughly 50 times larger than Ed25519. On high-throughput data-center links these differences are negligible, but on bandwidth-constrained tactical or satellite links they affect protocol round-trip time, certificate chain size, and session establishment latency. Teams must verify that the chosen parameter sets perform acceptably under realistic network conditions, not just in lab benchmarks.

An ML-DSA-65 public key is 1,952 bytes (compared to 32 bytes for Ed25519), and a signature is 3,309 bytes (compared to 64 bytes). For devices that must verify multiple signatures at boot time (boot image, container images, configuration manifests), the aggregate verification overhead must be measured under realistic hardware conditions.

Disconnected Operation

Disconnected operation means cryptographic policy must be self-contained in the deployed image, not fetched at runtime. A device that depends on a network-accessible policy engine to determine which algorithms to use, which certificates to trust, or which key rotation cadence to enforce will fail to maintain its cryptographic posture during connectivity gaps.

This constraint is fundamental: the device must carry its complete cryptographic policy as part of its deployed image, and that policy must remain enforceable for the duration of any expected disconnection period. Devices deployed on tactical networks, maritime platforms, or remote industrial sites may operate for weeks or months without reliable connectivity to a central management plane.

Physical Exposure

Physical exposure adds a further dimension. Devices deployed in contested or uncontrolled environments face risks of key extraction, side-channel observation, and physical tampering that are less relevant in controlled facilities. Cryptographic key material management must account for the possibility that an adversary has sustained physical proximity to the device.

The combination of long deployment lifetimes, constrained compute budgets, intermittent connectivity, and physical exposure creates a compliance environment where standard enterprise approaches to cryptographic lifecycle management are insufficient.

- Disconnected operation means cryptographic policy must be self-contained in the deployed image, not fetched at runtime.
- Long deployment lifetimes require a credible plan for multiple algorithm transitions on the same hardware.
- Constrained compute budgets may rule out larger PQC parameter sets or require hardware acceleration planning.
- Bandwidth-limited links change the practical cost of larger key shares, ciphertexts, and signatures.

4. Implementation Approach: Immutable Images and Atomic Crypto Policy

The most defensible implementation pattern for CNSA 2.0 at the edge binds cryptographic policy to the system image itself. When the operating system is distributed as an immutable, signed image, the algorithm configuration, cryptographic library versions, and TLS policy travel as part of the verified artifact rather than being configured independently per device.

This approach eliminates the most common source of compliance failure in distributed systems: configuration drift. When cryptographic policy is managed through per-device configuration files, package updates, or operator-driven changes, the fleet inevitably diverges. Some devices may be running updated algorithm configurations while others remain on prior versions, creating a fleet state that is difficult to audit and impossible to certify as uniformly compliant.

Atomic Updates and Partial-Migration Prevention

Atomic updates ensure that cryptographic policy changes are applied as whole-image replacements. This eliminates the risk of partial migration, a state where some subsystems have transitioned to ML-KEM while others still negotiate classical key exchange, creating a device that is neither fully classical nor fully post-quantum.

The systemd Automatic Boot Assessment specification (as referenced by the UAPI Group) provides a mechanism for health-checking new images and reverting to the previous known-good state if the update fails. This is particularly important for cryptographic transitions: if a new image with updated PQC algorithms fails to establish connectivity with the fleet management plane (perhaps because the management plane has not yet been upgraded to support the new algorithms), the device automatically reverts to the previous image with its prior cryptographic configuration.

Signed Release Channels

Signed release channels provide the trust anchor for update authenticity. Each image is signed before distribution, and the device verifies the signature before promotion. This ensures that the cryptographic policy embedded in the image has not been tampered with during transit or storage, and that the signing authority is traceable back to the organization's release governance.

For CNSA 2.0 compliance, the signing mechanism itself should use post-quantum algorithms. An update channel that authenticates releases with classical-only signatures is vulnerable to the same quantum threat that CNSA 2.0 is designed to address. The transition to post-quantum signing of release artifacts should be among the earliest milestones in the migration plan.

- Cryptographic policy versioned inside the image eliminates per-device configuration drift.
- Atomic image replacement prevents partial-migration states that are difficult to audit.
- Signed release channels ensure update authenticity without depending on operator-side verification steps.
- Rollback to the previous image preserves the prior cryptographic posture as a complete, known-good state.

5. What Teams Need to Prove for Compliance

CNSA 2.0 compliance programs need to produce artifactable evidence, not just architectural assertions, that the deployed fleet enforces the declared cryptographic policy. Auditors and authorizing officials need to see that the running image on each device matches a specific, signed release; that the signing chain traces back to an authorized build; that key material is managed within the declared rotation cadence; and that the update mechanism itself uses quantum-resistant authentication.

Fleet-Wide Enforcement

Fleet-wide enforcement is the critical proof point. A platform that supports ML-KEM on one device but cannot demonstrate consistent deployment across all devices in a cohort does not meet the intent of the advisory. Evidence must show that policy enforcement is systematic (applied through the release process) rather than dependent on per-device operator action.

The compliance question is not "does the platform support ML-KEM?" but rather "can you demonstrate that every device in the deployed fleet is running an image that enforces the declared cryptographic policy, and that no device has drifted from that policy?" This distinction between capability and enforcement is where many compliance programs fall short.

Rollback Safety

Teams should also be prepared to demonstrate rollback safety. If an image update fails or a post-quantum parameter set proves incompatible with a specific hardware variant, the device must revert to a complete, auditable prior state rather than falling into an undefined configuration. The rollback image must itself carry a valid cryptographic posture, so reverting does not inadvertently regress the device to a pre-CNSA-2.0 state.

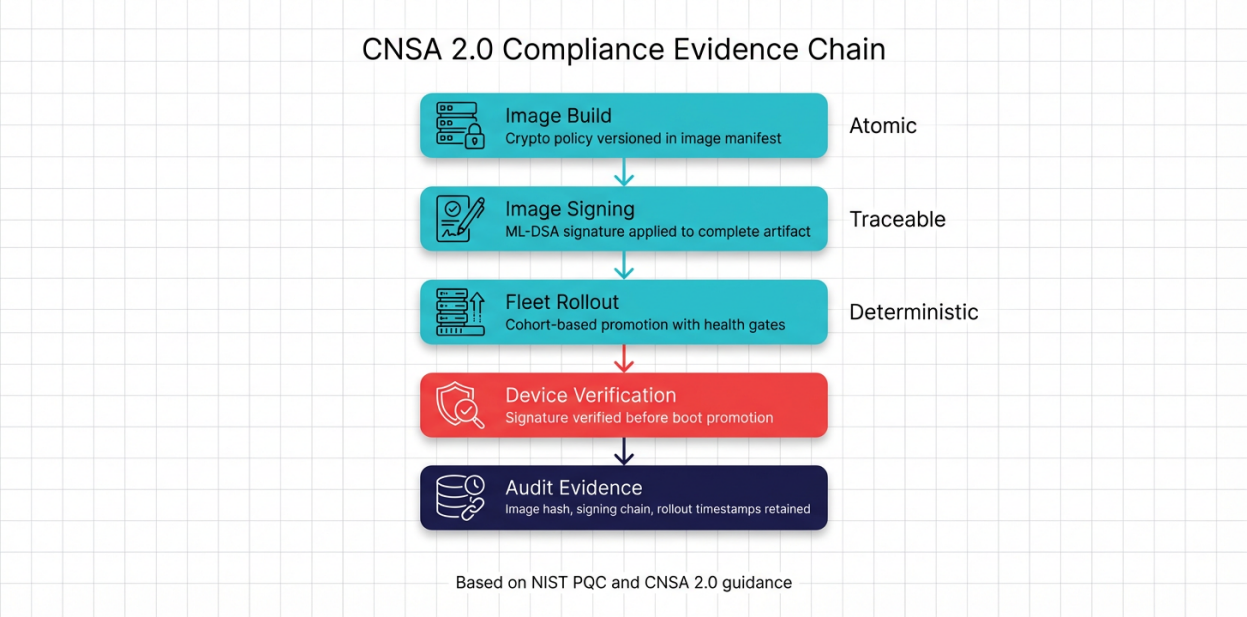

Figure 2: Compliance evidence chain from image build through fleet-wide audit evidence retention.

- Every deployed image must be traceable to a signed build with known cryptographic policy.
- Fleet-wide consistency must be demonstrable through rollout records, not assumed from architecture.
- Key rotation evidence must show cadence compliance under realistic connectivity constraints.
- Rollback images must preserve a valid cryptographic posture, not regress to classical-only configuration.

6. Cryptographic Dependency Inventory

The foundation of any CNSA 2.0 migration plan is a comprehensive cryptographic dependency inventory that maps every algorithm usage across the platform stack. This inventory must go beyond TLS cipher suite configuration to capture every subsystem that depends on cryptographic operations.

Scope of the Inventory

A complete inventory covers at minimum the following categories: TLS session establishment (cipher suites, key exchange groups, certificate verification), boot integrity (Secure Boot signing, measured boot, boot artifact verification), image and update authentication (release signing, signature verification at promotion time), certificate infrastructure (CA hierarchies, certificate issuance algorithms, revocation mechanisms), data protection at rest (encryption algorithms for persistent storage, key wrapping), and inter-device authentication (mutual TLS, device identity certificates, fleet enrollment protocols).

Each entry in the inventory should document the current algorithm, the target CNSA 2.0 algorithm, the library or implementation providing the algorithm, the update mechanism for changing the algorithm, and any dependencies on external services (certificate authorities, key management systems, signing infrastructure) that must also be migrated.

Why the Inventory Matters

The inventory serves two purposes. First, it is the planning foundation: the migration plan is derived from the inventory by ordering transitions according to dependency relationships, risk exposure, and operational impact. Second, it is the evidence foundation: the inventory and subsequent migration records become the documentation that auditors and authorizing officials will review to assess compliance posture.

Without a comprehensive inventory, teams risk discovering unmigrated cryptographic dependencies late in the transition, potentially after compliance deadlines have passed. The most common oversights are embedded cryptographic usages in provisioning protocols, device enrollment flows, and inter-device authentication mechanisms that were not part of the initial TLS-focused migration scope.

7. Fleet-Wide Enforcement and Audit Evidence

CNSA 2.0 compliance is not a property of a single device; it is a property of the entire deployed fleet. Demonstrating compliance requires evidence that every device in the fleet is running the declared cryptographic policy, that the policy was deployed through an authenticated release process, and that deviations (if any) are documented and justified.

Rollout Records and Cohort Tracking

Image-based platforms that track rollout state per device can produce rollout records showing when each device received and promoted a specific image version. These records, combined with the image manifest (which documents the cryptographic policy embedded in the image), provide the evidence chain from policy declaration through fleet-wide deployment.

Cohort-based rollout adds additional evidence structure. When images are deployed in stages (canary groups, then broader cohorts, then fleet-wide promotion), the rollout records document the progression and any issues encountered. This staged evidence is more persuasive to auditors than a single fleet-wide push because it demonstrates deliberate risk management.

Key Rotation Under Connectivity Constraints

Key rotation cadence is one of the harder compliance requirements to satisfy at the disconnected edge. If the declared rotation cadence is 90 days but a device is disconnected for 120 days, the device will exceed its rotation window. Teams must document how the platform handles this scenario: does the device continue operating with expired key material (and if so, under what justification), or does it enter a restricted mode until connectivity is restored?

The most defensible approach is to design the rotation cadence around worst-case connectivity assumptions rather than best-case scenarios. If devices may be disconnected for up to 90 days, the rotation cadence should be set so that key material remains valid for the full disconnection period plus a reconciliation margin. The alternative (requiring connectivity for rotation) creates a dependency that conflicts with the disconnected operation requirement.

Audit Event Continuity

Events that occur during disconnected operation (policy enforcement decisions, workload starts, health check results, any security-relevant actions) must be buffered locally and transmitted when connectivity returns. The central management system should be able to reconstruct a complete timeline of device behavior across disconnection gaps, ensuring that audit evidence has no temporal holes.

8. Practical Recommendations for Edge Teams

Teams beginning CNSA 2.0 planning for edge deployments should start with the cryptographic dependency inventory described in Section 6. This inventory becomes the baseline for transition planning and the evidence foundation for compliance documentation. The following recommendations address the practical steps from inventory through fleet-wide enforcement.

- Start with a full cryptographic dependency inventory across all stack layers, not just TLS.
- Validate post-quantum parameter set performance under realistic edge network conditions, including bandwidth-constrained and high-latency links.
- Require that the update mechanism itself uses quantum-resistant signing and verification, not just the workload-level protocols.
- Plan for at least one additional algorithm transition within the expected hardware lifetime, ensuring the platform supports cryptographic agility as an operational property.
- Treat compliance evidence generation as a platform capability, not a post-hoc documentation exercise.
- Set key rotation cadences based on worst-case connectivity assumptions, not best-case scenarios.
- Design rollback behavior so that reverting to a previous image preserves a valid CNSA 2.0 cryptographic posture.
- Document the reconciliation process for devices that exceed key rotation windows during disconnected operation.
- Include the signing infrastructure (certificate authorities, build signing keys) in the migration scope from the beginning.
- Engage authorizing officials early to establish agreement on what evidence artifacts satisfy CNSA 2.0 compliance requirements for edge-specific scenarios.

9. How nova8 Technologies Aligns with These Principles

nova8 Technologies applies publicly documented cryptographic principles, including NIST post-quantum standards, NSA CNSA 2.0 guidance, and image-based cryptographic agility, in its edge platform architecture. This paper describes the public architectural pattern and evaluation criteria, not product implementation details.

The cryptographic direction of the nova8 platform is aligned with the NIST post-quantum standards (FIPS 203, FIPS 204, FIPS 205) and the NSA CNSA 2.0 transition guidance. The platform architecture supports the lifecycle management approach described in this paper: binding cryptographic policy to the system image, deploying policy changes as atomic image replacements, and producing compliance evidence as a byproduct of the release process.

nova8 Technologies is aligning infrastructure and development practices with CMMC Level 2 expectations. The company holds U.S. Provisional Patent Applications 63/897,352, 63/897,609, 63/903,132, 63/903,161, 63/903,164, and 63/903,168 related to edge computing, security architecture, and operational resilience innovations.

For more information, visit nova8.io or contact the team at contact@nova8.io.

10. References

1. NSA, "Commercial National Security Algorithm Suite 2.0," September 2022, media.defense.gov

2. NIST, "FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard," August 2024, csrc.nist.gov/pubs/fips/203/final

3. NIST, "FIPS 204: Module-Lattice-Based Digital Signature Standard," August 2024, csrc.nist.gov/pubs/fips/204/final

4. NIST, "FIPS 205: Stateless Hash-Based Digital Signature Standard," August 2024, csrc.nist.gov/pubs/fips/205/final

5. White House, "Memorandum M-23-02: Migrating to Post-Quantum Cryptography," January 2023, whitehouse.gov

6. NIST, "CSWP 39: Crypto Agility Considerations for Migrating to Post-Quantum Cryptographic Algorithms," nist.gov

7. CISA, NSA, and NIST, "Quantum-Readiness: Migration to Post-Quantum Cryptography," 2023

8. NCCoE, "Migration to Post-Quantum Cryptography," nccoe.nist.gov/crypto-agility-considerations-migrating-post-quantum-cryptographic-algorithms

9. UAPI Group, "Unified Kernel Images (UKI) Specification," uapi-group.org/specifications/specs/unified_kernel_image/

10. systemd, "Automatic Boot Assessment," systemd.io/AUTOMATIC_BOOT_ASSESSMENT/

11. NIST, "Migration to Post-Quantum Cryptography FAQ," March 2026, pages.nist.gov/nccoe-migration-post-quantum-cryptography/

12. Global Risk Institute, "2024 Quantum Threat Timeline Report," globalriskinstitute.org