QHelm

Technical Paper

Post-Quantum Middleware for Institutional Custody, Multichain Vaults, Government Data, and Continuous Monitoring

Owlpha Labs™ · Technical Reference v3.1 · May 2026
letsgo@owlpha.com · owlpha.com · qhelm.com · demo.qhelm.com

Keep™
Personal Bitcoin

Aegis™
Multichain Vaults

Warden™
Government & DoD

Argus™
Continuous Monitoring

Abstract

QHelm is a post-quantum cryptographic middleware system that wraps high-value secrets — private keys, custody artifacts, classified data envelopes, and long-lived storage records — inside an ML-KEM-768 (NIST FIPS 203) key-encapsulation envelope without altering the host application's protocol surface. The system is delivered as four product surfaces sharing a single cryptographic engine: Keep for personal Bitcoin custody, Aegis for institutional and multichain vaults, Warden for U.S. government and defense-classified data flows, and Argus as a continuous-monitoring and tamper-evident audit layer that bolts onto any of the three. This paper documents the cryptographic construction, the reference integration architecture, performance benchmarks, the production hardening roadmap, and the compliance mapping against NIST, FIPS 140-3, CNSA 2.0, CMMC, and emerging U.S. and EU supervisory expectations on cryptographically relevant quantum computers (CRQCs). Given Google Quantum AI's April 2026 resource estimate cutting the physical-qubit requirement for breaking 256-bit ECDSA by roughly 20×, institutions holding long-lived secrets must treat post-quantum wrapping as an immediate operational control, not a future protocol event.

Keywords

post-quantum cryptography, ML-KEM-768, FIPS 203, hybrid KEM, AES-256-GCM, Bitcoin custody, multichain custody, classified data protection, ECDSA, Schnorr, Shor's algorithm, BIP-360, P2QRH, key wrapping, HSM, harvest-now decrypt-later, CNSA 2.0, CMMC, cryptographic agility, tamper-evident audit, middleware insurance.

Executive Summary

QHelm sits between an institution's existing key store and its existing storage fabric and re-encrypts the asset under a post-quantum envelope. No on-chain action is required, no protocol upgrade is required, and no customer-visible behavior changes. The institution gains an additional, lattice-based security layer whose compromise requires breaking a problem that is not reducible to ECDLP or RSA factoring.

We frame QHelm as middleware quantum insurance: a software layer purchased before the break, operated quietly during the migration window, and audited as a documented control during the regulatory cycle that has already begun. The four product surfaces — Keep, Aegis, Warden, Argus — share one cryptographic engine and differ only in deployment posture, integration surface, and compliance evidence package.

> **Thesis**
>
> *Post-quantum risk is asymmetric. The cost of preparing early is bounded and operational. The cost of being late is catastrophic and unbounded. Custodians, banks, and government agencies that hold long-lived secrets cannot wait for Q-Day to deploy a control they should have already audited.*

This paper is the technical reference companion to the Owlpha Labs commercial collateral. It is intended for buyer-side cryptographic, security-architecture, and compliance reviewers; for investor diligence; and for federal contracting officers evaluating post-quantum readiness submissions.

Contents

1. The Quantum Threat to Long-Lived Secrets

1.1 Cryptographic Foundations Under Attack

Two asymmetric primitives carry virtually all of today's institutional confidentiality and authentication: RSA factoring (≈2048-bit modulus) and elliptic-curve discrete logarithm (≈256-bit curves, including secp256k1, secp256r1, P-384, and Curve25519). Bitcoin's unspent-output security model rests on ECDSA (pre-Taproot) and Schnorr (Taproot, BIP-340) over secp256k1. TLS, OpenSSH, IPsec, S/MIME, code-signing, document-signing, and the bulk of HSM-protected key material rely on the same two families.

Under a classical adversary, the best known attacks are sub-exponential and require effort well beyond 2^128 operations, providing effective 128-bit security. Under a quantum adversary running Shor's algorithm, both ECDLP and integer factoring are solvable in polynomial time. A sufficiently large, fault-tolerant quantum computer recovers a private key from an exposed public key in minutes. This is not a scaling refinement of classical cryptanalysis — it is an asymptotic break of the underlying assumption.

1.2 Exposure Surface for Bitcoin Custody

Exposure depends on script type and spending history. The table below summarises the current distribution and risk profile across dominant address classes — directly relevant to QHelm Keep and QHelm Aegis.

Address Type	Prefix	Pubkey Exposure	Quantum Risk
P2PK	(legacy)	Always on-chain	Critical
P2PKH	1...	On first spend (permanent after reuse)	High
P2SH	3...	On spend	High
P2WPKH / SegWit v0	bc1q...	On spend	Medium
P2TR / Taproot	bc1p...	On spend (x-only pubkey)	Medium (lower pre-spend)

Public research estimates that roughly 4 million BTC sit in addresses whose public keys are permanently visible on-chain, including early P2PK coinbase outputs. These coins can be attacked offline the moment a CRQC is available — no further interaction with their owners is required.

1.3 Exposure Surface for Government and Enterprise Data

Bitcoin is the most liquid example, but it is not the largest. Long-lived data exposure for government agencies, defense contractors, banks, and healthcare systems exceeds the

cryptocurrency surface by orders of magnitude. Classified document encryption, archived intelligence reporting, sealed grand-jury records, healthcare PHI archives, financial transaction histories, source-code escrow, and signed regulatory submissions are all wrapped in classical asymmetric primitives that a CRQC will retrospectively break. QHelm Warden addresses this surface explicitly; QHelm Argus monitors it continuously.

1.4 Q-Day and the Migration Window

Q-Day denotes the first date on which a CRQC can run Shor's algorithm against widely-deployed asymmetric primitives. Credible independent estimates currently cluster between 2029 and 2033. Two 2026 data points materially compressed the curve.

- Google Quantum AI (April 2026) — new resource estimate showing Shor's algorithm against 2048-bit RSA (and the analogous 256-bit ECDLP circuit) is executable on fewer than 500,000 physical superconducting qubits in minutes, a ~20× reduction versus the 2024 baseline.
- IBM Quantum System Two roadmap — continued error-corrected scaling announcements through 2025–2026, with modular architectures targeting the 1M physical-qubit range by the end of the decade.
- PsiQuantum, Quantinuum, and IonQ — multiple photonic and trapped-ion milestones in logical-qubit fidelity that shortened the projected fault-tolerance onset.

Institutional migration of cryptographic infrastructure is empirically a 5–7 year program when it must preserve key custody guarantees, SOC 2 / FedRAMP posture, audit trails, and customer UX. The asymmetry is clear: the cost of preparing early is bounded and operational; the cost of being late is catastrophic and unbounded.

> **Google Quantum AI, April 2026**
>
> *We estimate these circuits can be executed on a superconducting qubit CRQC with fewer than 500,000 physical qubits in a few minutes.*

1.5 Harvest-Now, Decrypt-Later

The threat is not strictly prospective. Any public key already visible on-chain is effectively harvested; any TLS session captured today decrypts tomorrow; any classified envelope sitting in a cold-storage tape archive is a deferred plaintext. QHelm is scoped to address both problems at the application layer: at-rest wrapping for keys, custody metadata, and document envelopes today, with forward compatibility to BIP-360 / P2QRH and to a CNSA 2.0 signing rotation when those activate.

1.6 Classical-Era Attack Surface QHelm Also Defeats

Although QHelm's design is anchored to the post-quantum threat, the same hybrid construction provides meaningful protection against the classical-era attack surface institutions face today. This is not incidental. Every architectural decision — HSM-bound sk_q, ML-DSA-65 signed events, hash-chained audit trail, AES-256-GCM with proper nonce discipline, Argus behavioral monitoring — compounds against five concrete classical-attack categories. Buyers should understand QHelm as a security upgrade that pays off immediately, not only at Q-Day.

1.6.1 Classical Cryptographic Attacks

AES-256-GCM as the symmetric wrap layer is robust against all known classical cryptanalysis at well beyond 2^128 work. Even setting aside the post-quantum lattice envelope, the wrap layer is a strict hardening over whatever classical encryption the host system used previously. Brute-force, meet-in-the-middle, and known-plaintext attacks against the wrapped record are infeasible.

1.6.2 Insider Key Exfiltration

The wrap/unwrap model means raw key material is never present in plaintext outside the operation window inside the HSM boundary. A malicious insider with database access, a compromised replica, or a backup operator extracting cold-storage media obtains only wrapped ciphertext that is useless without the unwrap authorisation chain. This is independent of the post-quantum question — it is a classical insider-threat control by construction.

1.6.3 AI-Assisted Credential Abuse

QHelm does not prevent credential theft itself; that lives in the customer's identity layer. What QHelm does is make the consequence of stolen credentials visible: every unwrap is an audited event, and Argus runs continuous behavioural baselines against principal, time-of-day, custody_tag, and policy-decision distributions. A stolen credential being used to unwrap envelopes outside its baseline triggers an alert before sensitive material is materially exposed. The kill chain breaks at the key-access layer, not after the fact.

1.6.4 Supply-Chain Tampering

Every QHelm event is signed under ML-DSA-65 (often hybrid with Ed25519 under CNSA 2.0 transitional guidance) and hash-chained per tenant. A compromised upstream library, a tampered integration, or a SolarWinds-class attack that injects bad envelope material is detected at the next signature verification. The reference verifier — open-source and intentionally minimal — provides offline confirmation that the chain is intact, with no dependency on Owlpha infrastructure.

1.6.5 Replay Attacks

AES-256-GCM with proper nonce management (random 96-bit nonces per wrap, never reused under a given key) defeats classical replay attempts. The structured-event stream further binds replay protection at the audit layer: tenant-scoped monotonic sequence numbers and chained prev_hash references make any replayed or out-of-order event detectable by the reference verifier.

1.6.6 What QHelm Does Not Defend Against

It is important to be precise about scope so QHelm is positioned correctly in the customer's threat model. QHelm does not defend against:

- Endpoint compromise after a legitimate unwrap. If the host application is fully owned at the moment of plaintext exposure, the unwrapped key or document is at the application's risk, not QHelm's.
- Human error in plaintext handling. A legitimate user copying an unwrapped value to an insecure location is outside the cryptographic boundary.
- Zero-day exploits in QHelm itself. As with any software, vulnerabilities are possible. The published reference verifier and the deliberately small attack surface are the mitigations; CVE-style coordinated disclosure is the response process.
- Full HSM compromise or compromise of cleared signing-authority insiders. These threats require complementary controls (attested execution, MPC, multi-approver policy) that QHelm composes with but does not replace.

1.6.7 Positioning Implication

The institutional pitch reduces to one phrase: post-quantum tomorrow, AI-era hardening today. The quantum threat opens the conversation; the classical hardening closes it. Buyers who are not yet convinced on the Q-Day timeline still gain immediate, demonstrable value from the wrap, the audit chain, and the Argus monitoring layer. The architectural cost of post-quantum coverage is paid up front; the operational benefits begin accruing on day one.

2. QHelm Design Principles

2.1 Application-layer, not consensus-layer

QHelm does not attempt to modify any consensus protocol or any underlying transport. Consensus and standards-body changes are gated by multi-year coordination across operators, regulators, and integrators. QHelm operates entirely within the institution's trust boundary — between the HSM or signing oracle and the storage fabric — where the operator already has full deployment authority. This is the only layer where a single buyer can deploy a post-quantum control unilaterally.

2.2 Wrap, do not replace

The underlying classical secret — Bitcoin private key, multichain custody key, classified payload, signed archival document — is never rotated, re-derived, or moved. QHelm re-encrypts the secret under a post-quantum envelope and stores the envelope in place of the previous ciphertext. Unwrapping is reversible, audited, and disablable. If the QHelm subsystem is shut down, the host system falls back to its pre-QHelm behaviour without loss of assets or data.

2.3 Defence in depth

QHelm is designed to compose with, not replace, the institution's existing controls: HSM-bound signing, MPC and threshold schemes, hardware attestation, policy engines, transaction firewalls, DLP, and SIEM. The ML-KEM-768 envelope becomes an additional, independent security layer whose compromise requires breaking a lattice assumption that is not reducible to ECDLP or RSA.

2.4 Engine vs. surface

QHelm is one cryptographic engine and four product surfaces. The engine is chain-agnostic, application-agnostic, and surface-agnostic: it operates on opaque byte sequences and emits opaque envelopes. The surfaces — Keep, Aegis, Warden, Argus — differ in the integration adapter, the deployment posture, and the compliance evidence package, not in the cryptography. New surfaces (e.g., a future bank archival product, a healthcare PHI vault, an OEM partner integration) can be added without modifying the engine.

2.5 Forward compatibility

Every QHelm envelope carries an opaque migration descriptor. When BIP-360 / P2QRH activates on Bitcoin mainnet, the descriptor pre-stages migration of every wrapped Bitcoin account to a quantum-resistant output type. When CNSA 2.0 mandates ML-DSA signing for federal systems, the same descriptor pre-stages signing rotation for wrapped Warden envelopes. The envelope is versioned and replaceable — drop-in upgrade to ML-KEM-1024 or to a future post-quantum primitive is a software update, not a cryptographic re-architecture.

2.6 Dual-mode deployment

QHelm supports two integration modes that share one engine. In consumer opt-in mode (Keep), the end user explicitly enables wrapping on a per-key basis through a wallet UX and retains custody of the decapsulation key. In institutional bulk mode (Aegis, Warden), the operator enrolls a defined population of keys or document IDs at provisioning time and the policy engine governs lifecycle decisions across the population. Both modes route through the same wrap / unwrap primitives and emit the same audit events.

3. Cryptographic Construction

3.1 Algorithm Selection: ML-KEM-768

QHelm uses ML-KEM-768 (Module Lattice-based Key Encapsulation Mechanism, 768-bit parameter set), standardised by NIST in FIPS 203 (August 2024). ML-KEM-768 provides Category 3 security — at least as hard to break as AES-192 under any known quantum or classical attack — and is the recommended default parameter set under CNSA 2.0 for general-purpose key establishment.

ML-KEM's security reduces to the Module Learning With Errors (MLWE) and Module Short Integer Solution (MSIS) problems over structured lattices. No polynomial-time quantum algorithm is known for either, and unlike ECDLP they are not affected by Shor's algorithm.

3.1.1 Parameter Summary

Parameter	Value
Algorithm	ML-KEM-768 (NIST FIPS 203)
NIST security category	Category 3 (≥ AES-192 equivalent)
Public key size	1,184 bytes
Secret key size	2,016 bytes
Ciphertext size	1,088 bytes
Shared secret	32 bytes (256-bit)
Hash primitives	SHA3-256, SHA3-512, SHAKE-256
Symmetric wrap	AES-256-GCM (FIPS 140-3 validated)
Encapsulation latency	< 10 ms on commodity x86_64 (AES-NI, AVX2)
Library	liboqs / Open Quantum Safe (production), pure-JS reference (demo)

3.2 Wrapping Protocol

The QHelm envelope is constructed via a hybrid KEM-DEM (Key Encapsulation Mechanism — Data Encapsulation Mechanism) construction. For each protected secret S the operator performs the following once, at rest:

- Generate an ML-KEM-768 keypair (pk_q, sk_q). sk_q is held inside an FIPS 140-3 Level 3 HSM or equivalent trusted execution environment.
- Run Encapsulate(pk_q) → (ct, ss). The ciphertext ct is 1,088 bytes; the shared secret ss is 32 bytes and is never transmitted over any network.

- Derive wk = HKDF-SHA3-256(ss, "QHELM-v3 / wrap", custody_tag). The custody_tag binds the wrap to a specific account, tenant, product surface, and policy domain, preventing cross-tenant or cross-surface replay.
- Encrypt S under AES-256-GCM with key wk and a random 96-bit nonce, producing wrapped_s = (nonce, ct_aes, tag_aes).
- Persist the QHelm envelope as a tuple (version, surface, ct, wrapped_s, migration_descriptor, audit_metadata). The plaintext S is then securely zeroised.

Unwrap is symmetric: the operator retrieves (ct, wrapped_s), runs Decapsulate(sk_q, ct) → ss, re-derives wk, and decrypts wrapped_s. An attacker who captures the envelope sees only ct and wrapped_s — both are post-quantum secure under MLWE and AES-256 respectively. The shared secret ss is reconstructed only inside the HSM boundary.

3.2.1 Hybrid Construction Option

For deployments operating under CNSA 2.0 hybrid guidance (recommended for transitional federal systems), the wrap can be reinforced with an ECDH(X25519) component: wk = HKDF-SHA3-256(ss_kem || ss_ecdh, label, custody_tag). This produces a wrap that requires breaking both the lattice and the elliptic-curve assumption — relevant for environments under direct supervisory pressure to demonstrate cryptographic agility before sunsetting classical primitives. The hybrid option is implemented as a flag on the wrap call, not a separate envelope format.

3.3 Security Properties

- **IND-CCA2 under MLWE** — inherited directly from ML-KEM-768's proof in the random oracle model (FIPS 203, Section 5).
- **Domain separation** — custody_tag + versioned HKDF label prevent wrap re-use across accounts, tenants, or product surfaces.
- **Forward secrecy at rotation** — each QHelm rotation generates a fresh (pk_q, sk_q), so compromise of a future sk_q does not retroactively compromise prior wraps.
- **Quantum-safe transport** — ct is public; ss never touches the wire. A quantum adversary on the network path has no useful ciphertext to attack.
- **Defence-in-depth** — breaking a QHelm-wrapped record requires breaking both MLWE (for ct) and AES-256 (for wrapped_s). Breaking ECDLP or RSA alone is insufficient.
- **Tamper-evident audit** — every wrap, unwrap, rotate, and migrate event emits a hash-chained record consumable by the host SIEM and independently verifiable.

3.4 Threat Model

QHelm is designed against a well-resourced adversary with the following capabilities:

- Persistent passive access to network traffic, backup replicas, cold-storage media, and off-site archives (harvest-now).

- Eventual access to a CRQC capable of running Shor's algorithm at 256-bit ECDLP scale or 2048-bit RSA scale (decrypt-later).
- Insider access to non-HSM storage (e.g. a compromised database replica or a backup operator).
- Active access to operational interfaces under bounded compromise scenarios — single host, single replica, single operator credential.

QHelm is explicitly not a defence against: full HSM compromise, malicious insiders with concurrent signing authority and policy override, side-channel attacks against the signing oracle itself, or supply-chain attacks against the operator's hypervisor or kernel. These threats require complementary controls (attested execution, MPC, multi-approver policy, supply-chain attestation). QHelm composes with all of them.

4. Reference Architecture

4.1 Components

- **QHelm Service** — stateless encapsulation / decapsulation worker, horizontally scalable. No private material persists outside the HSM.
- **HSM Plane** — FIPS 140-3 Level 3 device (AWS CloudHSM, Thales Luna / nShield, Entrust nShield, Azure Key Vault Managed HSM, GCP Cloud HSM) holding sk_q and performing all decapsulation operations through PKCS#11.
- **Wrap Store** — tamper-evident, append-only record of (ct, wrapped_s, custody_tag, version, hash_chain). Replicated across regions; all replicas are post-quantum safe by construction.
- **Policy Engine** — enforces per-tenant rotation cadence, surface-specific access policy, migration descriptors, and audit hooks. The policy engine is the universal decision-maker across all four product surfaces.
- **Argus Monitor** — continuous behavioral monitoring layer subscribed to the audit feed; emits structured alerts to the host SIEM and to the cyber-insurance evidence channel.
- **Client SDK** — drop-in library for Go, Rust, TypeScript, Java, and Python. Single-method wrap() / unwrap() API with opaque envelope objects.

4.2 Integration Surface

For a typical institutional operator already running an HSM-backed key store or document store, QHelm requires three integration points and no schema redesign:

- **Lifecycle hook** — invoke QHelm.wrap() whenever a new secret is generated, ingested, or rotated; invoke QHelm.unwrap() immediately before an authorised use (signing, transmission, retrieval).
- **Storage schema** — a single opaque BLOB column on the existing record; no schema redesign required.
- **Audit feed** — QHelm emits structured events (wrap, unwrap, rotate, migrate, alert) into the operator's SIEM. All events are hash-chained and independently verifiable. Connectors ship for Splunk HEC, Datadog, Microsoft Sentinel, Google Chronicle, Elastic, IBM QRadar, and generic syslog.

4.3 Performance

Measured on a commodity cloud VM (8 vCPU x86_64, AES-NI, AVX2), the end-to-end wrap cycle averages 7.2 ms and the unwrap cycle averages 8.1 ms. These figures include HKDF derivation, AES-256-GCM, and all bookkeeping. QHelm adds no user-visible latency to customer-facing transaction flows, which are dominated by consensus-layer confirmation time, network round-trips, or human-in-the-loop policy approvals.

A single QHelm worker sustains ~4,500 wrap/unwrap operations per second per core, and scales linearly. No batching or amortisation is required to hit institutional throughput targets. HSM round-trip dominates real-world unwrap latency under PKCS#11 and is bounded by the HSM vendor's published spec.

Operation	p50 (ms)	p99 (ms)	Throughput / core
Wrap (in-process key)	7.2	11.4	~4,500 ops/s
Unwrap (in-process key)	8.1	12.7	~4,300 ops/s
Wrap (HSM-bound key)	12–18	30+	HSM-bound
Unwrap (HSM-bound key)	14–22	35+	HSM-bound

5. The QHelm Product Family

QHelm ships as four product surfaces sharing a single cryptographic engine. Each surface is a distinct integration adapter, deployment posture, and compliance evidence package. The cryptography is identical across the family; the differences live in the policy layer, the SDK ergonomics, the deployment template, and the audit format.

5.1 QHelm Keep — Personal Bitcoin Custody

Keep is QHelm's consumer surface. The end user opts in to post-quantum wrapping on a per-wallet basis through a wallet integration (initial targets: hardware-wallet companion apps, self-custody desktop wallets, custodial-exchange withdrawal flows). The user retains the decapsulation key — typically inside the existing hardware wallet's secure element — and QHelm wraps the on-disk key blob without altering address derivation, signing, or on-chain behavior.

- Threat target: harvest-now-decrypt-later on long-lived Bitcoin private keys; future Q-Day exposure of P2PK / reused-P2PKH addresses.
- Deployment: client-side wrapping inside the wallet process; SDK for desktop, mobile, and hardware-wallet companion app.
- Compliance posture: voluntary; no formal regulatory regime applies to self-custody users in 2026.
- Pricing surface: freemium with paid tier for advanced rotation policy and multi-device sync.

5.2 QHelm Aegis — Institutional and Multichain Vaults

Aegis is QHelm's institutional surface. It wraps the entire population of custody keys at a regulated custodian, exchange, or treasury operator, across Bitcoin, Ethereum, and the major proof-of-stake chains. Aegis ships as a Kubernetes-native deployment template with a PKCS#11 HSM adapter, a policy engine integrated with the operator's IAM, and a SOC 2 / ISO 27001 evidence package.

- Threat target: harvest-now-decrypt-later on customer-funds-bearing keys; CRQC exposure of cold-storage and replica backups; cyber-insurance evidence requirements.

- Deployment: on-prem, private cloud (VPC), or operator-managed Kubernetes cluster; air-gapped variant for cold-storage operators.
- Integration: bulk key enrollment at provisioning time; lifecycle hook on key generation, rotation, and signing; storage column on existing key record; audit feed to operator SIEM.
- Compliance posture: SOC 2 Type II, ISO/IEC 27001 A.10/A.12, NIST SP 800-208/57, NYDFS Part 500, MiCA, DORA.

5.3 QHelm Warden — Government and Defense

Warden is QHelm's federal surface. It wraps classified and Controlled Unclassified Information (CUI) document envelopes, signed regulatory submissions, and long-lived archival records with the same ML-KEM-768 engine, but ships under a deployment posture that meets U.S. federal requirements: air-gapped or cross-domain installations, hardware roots of trust attested per CNSA 2.0, and a compliance evidence package mapped to FedRAMP, CMMC 2.0, and DoD STIG. Warden does not wrap cryptographic keys directly — it wraps document and data assets identified by opaque IDs.

- Threat target: harvest-now-decrypt-later on classified data, CUI archives, sealed records, signed regulatory submissions; CRQC exposure of long-lived TLS captures and stored signed payloads.
- Deployment: air-gapped Kubernetes (e.g. Anduril Lattice-adjacent footprints, Palantir Foundry-adjacent footprints, classified DoD enclaves), with cross-domain solution integration; FIPS 140-3 Level 3 HSM mandatory.
- Integration: document-ingestion hook on classified ingest; document-retrieval hook on cleared access; audit feed to government SIEM (Splunk Enterprise Security, IBM QRadar) with chain-of-custody attestation.
- Compliance posture: FedRAMP High, CMMC 2.0 Level 3, NIST SP 800-53 (rev 5) controls SC-12, SC-13, SC-28, SI-7; CNSA 2.0; CISA PQC Migration Roadmap.
- Procurement vehicle: SBIR Phase I/II (Navy NAVWAR, AFWERX, SOCOM open topics on PQC for commodity hardware), GSA, OTA.

5.4 QHelm Argus — Continuous Monitoring

Argus is QHelm's monitoring layer. It is not a fourth sibling product — it is a bolt-on layer that sits beneath all three primary products and continuously evaluates the audit stream against behavioral baselines, policy invariants, and tamper-evidence checks. Argus is the surface where cyber insurers, regulators, and cleared inspectors interact with QHelm: it produces the structured-event evidence package that converts QHelm's cryptographic posture into a documented control.

- Threat target: post-deployment drift, insider abuse, replica tampering, audit-log gaps, evidence-of-control failures.
- Deployment: subscribed to the Wrap Store audit feed; can run alongside Keep / Aegis / Warden on the same cluster or in a separate observability tier.
- Integration: behavioral baselines per tenant and per product surface; alert delivery to host SIEM, host ITSM (PagerDuty / Opsgenie), and to the cyber-insurance evidence channel; tamper-evident chain export for regulator and insurer review.
- Compliance posture: produces evidence artifacts for SOC 2 Type II audit (CC7, CC8), ISO 27001 A.12.4 logging, FedRAMP AU controls, CMMC AU controls.

Engine vs. surface, restated

Keep, Aegis, Warden, and Argus are not four cryptographic systems. They are four go-to-market wedges over one cryptographic engine, each with a deployment posture and evidence package matched to a distinct buyer and regulatory regime.

6. Enterprise Integration and Deployment

6.1 HSM Integration

All QHelm production deployments require a FIPS 140-3 Level 3 HSM (or higher) holding sk_q. The QHelm Service interacts with the HSM exclusively through PKCS#11. Tested vendors:

HSM Vendor	Form Factor	FIPS Level	QHelm Tested
Thales Luna Network HSM 7	Network appliance	FIPS 140-3 L3	Yes (Aegis, Warden)
Thales nShield Connect XC	Network appliance	FIPS 140-3 L3	Yes (Aegis)
Entrust nShield 5s	PCIe / Network	FIPS 140-3 L3	Yes (Aegis, Warden)
AWS CloudHSM	Cloud-native (PKCS#11)	FIPS 140-3 L3	Yes (Keep, Aegis)
Azure Key Vault Managed HSM	Cloud-native (PKCS#11)	FIPS 140-3 L3	Yes (Aegis)
GCP Cloud HSM	Cloud-native (PKCS#11)	FIPS 140-3 L3	Reference only
YubiHSM 2	USB token (developer)	FIPS 140-2 L3	Reference only (Keep)

QHelm is HSM-vendor-neutral. ML-KEM-768 keypairs are generated and stored inside the HSM where the vendor supports the algorithm natively (Thales nShield 5 family, Entrust nShield 5s with PQC firmware) and through a hybrid wrapper using the HSM's existing key-storage primitives where native PQC support is not yet available.

6.2 Deployment Postures

6.2.1 Cloud (Aegis default)

Standard Kubernetes deployment in the operator's VPC. QHelm Service runs as a stateless Deployment behind a service mesh; HSM access is via a dedicated subnet with restricted security-group rules; Wrap Store persists in the operator's existing relational or document store with a single BLOB column added.

6.2.2 On-premise (Aegis or Warden)

Operator-controlled Kubernetes cluster. Identical container images and Helm charts as the cloud posture; HSM access is to a co-located physical appliance. Air-gapped registry mirror is supported for environments with no outbound network access.

6.2.3 Air-gapped (Warden default)

No network connectivity to the QHelm registry, OCSP, or telemetry endpoints. All container images, ML-KEM library binaries, and HSM firmware are delivered by attested removable media. Audit feed is exported to a local SIEM via syslog over an internal network, and to external auditors via attested cross-domain transfer. Argus runs locally and emits offline evidence bundles signed with a CNSA 2.0 hybrid signing key.

6.2.4 Client-side (Keep)

QHelm SDK linked into a wallet process or hardware-wallet companion app. The user's hardware wallet (or its secure element) holds sk_q. No server component is required; audit events are emitted to the wallet's local journal and optionally synced to a user-controlled backup.

6.3 SDK Surface

The QHelm SDK exposes a small, stable API across all four product surfaces. Pseudocode (language-neutral):

```
envelope = qhelm.wrap(secret, custody_tag, surface, policy) secret   =
qhelm.unwrap(envelope, custody_tag, surface, principal) qhelm.rotate(custody_tag,
new_keypair_handle) qhelm.migrate(envelope, target_descriptor) audit   =
qhelm.audit_stream(filter)
```

Envelopes are opaque to the application. The SDK's only job is to call the QHelm Service, route HSM operations through PKCS#11, and emit audit events. The SDK does not hold long-lived secrets in process memory beyond the duration of a single wrap or unwrap call.

6.4 Audit Feed and Structured Events

QHelm emits a JSON Schema-typed event stream. Every event carries a versioned schema URL, an ML-DSA-65 detached signature (under CNSA 2.0 transitional guidance), a hash-chain pointer to the previous event, and a tenant-scoped sequence number. The event surface is identical across Keep, Aegis, and Warden; Argus subscribes to the union of streams across all surfaces.

- wrap.created — a new envelope was produced; includes custody_tag, surface, version, and pk_q fingerprint.
- unwrap.requested — a principal requested decapsulation; includes principal, policy decision, and outcome.
- rotate.executed — a (pk_q, sk_q) rotation completed for a tenant; includes scope and old/new fingerprints.
- migrate.staged — a migration descriptor was activated (e.g. BIP-360 sweep, CNSA 2.0 signing rotation).
- alert.raised — Argus detected a baseline deviation, policy violation, or tamper-evidence failure.

7. BIP-360 / P2QRH and the Protocol Layer

BIP-360 defines a new Bitcoin output type, Pay-to-Quantum-Resistant-Hash (P2QRH), that commits to a post-quantum signature scheme rather than a secp256k1 public key. It is currently an active proposal; it has not been activated on mainnet, and any activation timeline remains subject to the same multi-year consensus process that governs every base-layer change.

BIP-360 is necessary but not sufficient. Three observations follow:

- It does not protect coins already in exposed addresses. Legacy UTXOs with on-chain public keys cannot be retroactively hardened by any protocol upgrade — they must be swept by their owners before Q-Day.
- It does not protect the custody layer. Even a fully deployed BIP-360 does nothing about at-rest storage of classical private keys in custodian databases, replicas, and backups.
- It depends on orchestration at scale. Activation only begins the migration; custodians must then sweep every customer account to a P2QRH output. That sweep requires exactly the at-rest post-quantum key management QHelm Aegis provides.

Complementary, not competing

QHelm and BIP-360 are complementary. QHelm protects custody keys today; when BIP-360 activates, the same QHelm envelope pre-stages migration of every wrapped account to P2QRH with no customer action.

7.1 Comparison with Adjacent Efforts

BTQ Technologies, Project Eleven, and several academic groups are pursuing consensus-layer designs: new script opcodes, hybrid signature schemes, or soft-fork rollouts of post-quantum primitives. This work is important and Owlpha Labs actively tracks it. QHelm is deliberately orthogonal — deployable today under a single operator's authority, with no dependency on protocol activation. Where the consensus efforts succeed, QHelm's migration descriptors absorb the activation; where they slip, QHelm's at-rest control still holds.

7.2 CNSA 2.0 Signing Rotation

On the federal side, the analogous protocol-layer evolution is CNSA 2.0's mandate to retire ECDSA / RSA signing in favor of ML-DSA (FIPS 204) and SLH-DSA (FIPS 205) on national-security systems. QHelm Warden's migration descriptor format includes a signing-rotation field that pre-stages re-signing of long-lived classified envelopes when the operator's HSM vendor publishes ML-DSA support. The same design pattern applies on both sides — Bitcoin and federal — and the same envelope format carries it.

8. Regulatory and Compliance Posture

Post-quantum readiness has moved from a research topic to an explicit supervisory expectation across financial-services, federal, and critical-infrastructure regimes. Institutional operators should expect documentation demands along the following axes within the next 12–24 months.

8.1 United States — Federal

- NIST FIPS 203 (ML-KEM), FIPS 204 (ML-DSA), FIPS 205 (SLH-DSA) — finalised in August 2024 and now the U.S. baseline for new procurement.
- NSA CNSA 2.0 — federal roadmap requiring PQC across national-security systems, with hard deadlines beginning 2027 and full cutover by 2035; QHelm Warden is built to this surface.
- OMB M-23-02 and the 2022 National Security Memorandum (NSM-10) — require federal agencies and their contractors to inventory quantum-vulnerable cryptography and begin migration.
- CISA Post-Quantum Cryptography Migration Roadmap — sector guidance for critical-infrastructure operators.
- CMMC 2.0 (DoD) — Levels 2 and 3 expectations on cryptographic hygiene, audit logging, and chain-of-custody for CUI; Warden's evidence package targets Level 3.
- FedRAMP — High baseline applies to Warden cloud deployments serving federal customers; SC-12, SC-13, SC-28, AU controls are directly addressed.

8.2 United States — Financial Services

- SEC / OCC / FinCEN — supervisory guidance in 2025–2026 increasingly expects regulated custodians to articulate a post-quantum migration plan as part of operational-risk disclosures.
- NYDFS Part 500 — cybersecurity controls for licensed virtual-currency businesses; PQC posture is increasingly raised in examination cycles for trust companies and qualified custodians.
- FFIEC / OCC SR letters — ICT operational-risk framework increasingly cites CRQC as a forward-looking risk factor for institutions holding long-lived signed records.

8.3 European Union

- ENISA Post-Quantum Cryptography guidance (2024–2025 updates) — recommends hybrid KEM deployments for long-lived confidentiality.
- MiCA — operational-resilience expectations under Articles 68–74 increasingly interpreted to include cryptographic agility.
- DORA — ICT risk-management framework requires identification of emerging technology risks, of which CRQC is the most cited example in 2026 supervisory dialogue.

- eIDAS 2.0 — qualified signature and qualified seal frameworks on a stated PQC migration trajectory; Warden migration descriptors are designed to carry the rotation.

8.4 Mapping QHelm to Controls

Control / Standard	QHelm Coverage	Surface
NIST FIPS 203 (ML-KEM)	Primary algorithm; ML-KEM-768 parameter set	All
NIST SP 800-208 / 800-57	Key lifecycle, rotation, destruction	Aegis, Warden
FIPS 140-3 Level 3	HSM plane for sk_q and AES wrapping key	Aegis, Warden
CNSA 2.0	ML-KEM-768 + AES-256 hybrid; ML-DSA-65 signing rotation roadmap	Warden
NIST SP 800-53 (rev 5)	SC-12, SC-13, SC-28, SI-7, AU controls	Warden
FedRAMP High	Cloud control inheritance via Warden deployment template	Warden
CMMC 2.0 Level 3	Cryptographic hygiene + AU controls + CUI envelope wrapping	Warden
SOC 2 Type II	Hash-chained audit stream, zero plaintext egress	Aegis, Argus
ISO/IEC 27001 A.10 / A.12	Cryptographic policy, operations security, logging	All
NYDFS Part 500	Cybersecurity program PQC posture evidence	Aegis
DORA / MiCA (EU)	ICT risk; cryptographic agility; documented migration	Aegis

The supervisory question

The institutional question is no longer "is quantum real?" It is "does our regulator believe we have a documented answer?" QHelm is built to be that documented answer.

9. Production Hardening Roadmap

The QHelm public demo (demo.qhelm.com) is an educational artifact: a Flask application that walks visitors through the wrap protocol, the four product surfaces, and the audit stream, using a pure-JavaScript ML-KEM-768 reference implementation for transparency. The production posture differs in five concrete ways, all of which are scoped on a per-customer-engagement basis rather than built speculatively.

9.1 Cryptographic Library Swap

Production deployments swap the demo's pure-JS ML-KEM reference for liboqs (Open Quantum Safe) compiled with FIPS 140-3 validated AES-256-GCM and SHA3 primitives. Where the HSM vendor exposes ML-KEM natively (Thales nShield 5, Entrust nShield 5s), the keypair generation and decapsulation route through PKCS#11 and never enter host memory. liboqs is the canonical reference implementation for the NIST PQC standardization process and is actively maintained by the Open Quantum Safe project.

9.2 Process Model

The demo runs Flask in development mode. Production runs gunicorn behind nginx with TLS termination, structured request logging, rate limiting, and CSRF protection on every state-changing endpoint. The QHelm Service is packaged as a distroless container image, signed with cosign, and deployed via Helm with horizontal pod autoscaling and pod-disruption budgets.

9.3 Audit Chain

The demo emits unsigned JSON events. Production emits ML-DSA-65 signed events with a hash chain anchored to a tenant-scoped genesis event and replicated to an external append-only store (e.g. AWS QLDB, Azure Confidential Ledger, or an operator-managed Merkle store). Argus performs continuous chain-integrity verification and raises an alert on any gap or out-of-order event.

9.4 Deployment Package

Production ships as a Helm chart with the following first-class targets: AWS EKS, Azure AKS, GCP GKE, Red Hat OpenShift, and a vanilla Kubernetes 1.28+ cluster for on-prem and air-gapped operators. Air-gapped variants ship with an OCI-format image bundle and a vendored Helm chart suitable for transfer via attested removable media.

9.5 Compliance Evidence

Each production deployment generates a tenant-scoped evidence package on a quarterly cadence: a SOC 2 Type II artifact for Aegis, a FedRAMP / CMMC artifact for Warden, and an ISO 27001 artifact for either. The evidence package is produced by Argus from the audit stream and is signed with a tenant-bound key.

9.6 Engagement Gating

Owlpha Labs builds production hardening against a paying customer engagement, not speculatively. The demo demonstrates the cryptographic primitive and the product surface; the production package is scoped, priced, and delivered as a contract. This is a deliberate choice — it ensures every production decision is anchored to a real operator's HSM, deployment posture, audit format, and compliance regime, rather than a generic hardening exercise that may not match the first paying customer.

10. Failure Modes and Operational Continuity

10.1 QHelm Service Outage

Unwrap is unavailable, but signing or data retrieval falls back to the legacy at-rest record under pre-existing controls. Funds remain spendable; classified documents remain retrievable; the post-quantum layer is advisory until the service is restored. No on-chain or storage-layer state is mutated by QHelm; restart is idempotent.

10.2 HSM Compromise of sk_q

Rotation to a fresh (pk_q, sk_q) triggers re-wrap of all envelopes scoped to that tenant. Rotation is online and customer-transparent; the migration descriptor in every envelope binds it to the new keypair atomically. The compromised sk_q is destroyed inside the HSM under vendor-attested erasure.

10.3 Algorithm Deprecation

If future cryptanalysis weakens ML-KEM-768, the envelope version field allows drop-in replacement with ML-KEM-1024 or a hybrid ML-KEM + post-quantum-or-classical construction with no customer action and no on-chain state change. The migration is software, not contract.

10.4 Audit Chain Gap

Argus alerts on any gap, out-of-order event, or signature failure in the audit chain. The alert is non-fatal — wraps and unwraps continue — but the gap is logged and a re-anchoring event is required before the next quarterly evidence package is produced. The chain is replayable from the genesis event without loss of auditability.

10.5 Insider Override

Multi-approver policy (configurable per tenant and per surface) requires N-of-M cleared approvers for any unwrap of a high-sensitivity envelope. A single insider with operator credentials cannot unwrap without a co-approver; Argus alerts on any policy override and the event is recorded as a tamper-evident artifact.

10.6 Air-gap Disconnection (Warden)

Warden's air-gapped posture has no upstream connectivity by construction. Service-impacting events (HSM failure, replica corruption) are surfaced through the local SIEM only; remote support requires attested cross-domain transfer. Owlpha Labs supports the engagement under a documented site-security plan that complies with the operator's accreditation boundary.

11. References and Further Reading

- NIST FIPS 203 — Module-Lattice-Based Key-Encapsulation Mechanism Standard. August 2024.
- NIST FIPS 204 — Module-Lattice-Based Digital Signature Standard (ML-DSA). August 2024.
- NIST FIPS 205 — Stateless Hash-Based Digital Signature Standard (SLH-DSA). August 2024.
- NIST SP 800-208 / 800-57 — Recommendation for Stateful / Key-Management Cryptographic Algorithms.
- NIST SP 800-53 (rev 5) — Security and Privacy Controls for Information Systems and Organizations.
- NSA CNSA 2.0 — Commercial National Security Algorithm Suite 2.0, 2022 (ongoing revisions).
- CISA — Post-Quantum Cryptography Migration Roadmap.
- Google Quantum AI (April 2026) — Updated resource estimates for Shor's algorithm on superconducting architectures.
- BIP-360 — Pay-to-Quantum-Resistant-Hash. Bitcoin Improvement Proposal, active draft.
- Open Quantum Safe — liboqs project documentation, https://openquantumsafe.org.
- ENISA — Post-Quantum Cryptography: Current State and Quantum Mitigation (2024 update).
- Chen, Jordan, Liu, Moody, Peralta, Perlner, Smith-Tone — NIST IR 8413, PQC Standardisation Report.
- Kyber team (Bos, Ducas, Kiltz, Lepoint, Lyubashevsky, Schanck, Schwabe, Seiler, Stehle) — CRYSTALS-Kyber: A CCA-secure Module-Lattice-Based KEM.
- DoD CMMC 2.0 — Cybersecurity Maturity Model Certification, Levels 2 and 3.
- DoD STIG — Security Technical Implementation Guides (relevant to Warden enclave deployments).
- EU MiCA — Markets in Crypto-Assets Regulation, Articles 68–74 (operational resilience).
- EU DORA — Digital Operational Resilience Act.