CIPHERSCOUT | PQC MIGRATION REPORT

PQC Migration Report

Target: temp-hybrid-migration-42ff.test

Scan ID: 38 · Generated: 2026-03-05T22:04:27Z · Generated by ExeQuantum CipherScout

1. Executive Summary

- The scanned host exposes TLS (443) supporting TLSv1.2 and TLSv1.3. Cipher suites are modern (AES_256_GCM for TLS 1.3, ECDHE-RSA-AES128-GCM-SHA256 for TLS 1.2). Key exchange group is SECP256R1. Quantum-vulnerable score for TLS is moderate (43) but not marked quantum-vulnerable in the CBOM.
- The X.509 certificate is RSA 3072, issued by Cloudflare Inc ECC CA-3, with not_after 2026-03-01. This certificate is expired or near-expiration and must be validated and renewed immediately.
- JWKS contains an RSA signing key (kid-1084, RS256, 3072 bits). OIDC id_tokens allow RS256 and ES256. Token endpoint auth allows client_secret_basic (flagged quantum-vulnerable) and private_key_jwt.
- Main crypto risks: classical RSA-based TLS certificate and JWKS RSA keys; token endpoint auth using shared client secrets; absence of documented private key storage (HSM/KMS) and PKI lifecycle processes in the CBOM.
- Migration is phased — starting with inventory and PKI/KMS readiness (Phase 0), then TLS hardening (Phase 1), JWKS and OIDC token protections (Phase 2), introduction of post-quantum or hybrid algorithms (Phase 3), and full validation and cutover (Phase 4).
- Prerequisites missing from the CBOM: authoritative inventory of private key locations, HSM/KMS presence and capabilities, CI/CD integration for cert/key deployment, and vendor support for post-quantum or hybrid algorithms. These must be resolved before PQC rollouts begin.
- Operational acceptance criteria focus on: no service downtime, validated cryptographic proof of hybrid PQC operations in test, key material protected by KMS/HSM, successful client interoperability tests, and documented rollback paths.

2. Asset Inventory

Only assets present in the CBOM are listed. If you have other hosts, keys, or services, they are not reflected here.

TLS Endpoints

Field	Value
Host	temp-hybrid-migration-42ff.test:443
TLS versions	TLSv1.2, TLSv1.3
TLS 1.3 cipher	TLS_AES_256_GCM_SHA384
TLS 1.2 cipher	ECDHE-RSA-AES128-GCM-SHA256
Key share group	SECP256R1
QHS score	43 (not marked quantum-vulnerable)
Asset ID	ASSET-1

Certificates / PKI

Field	Value
Subject CN	temp-hybrid-migration-42ff.test
Issuer CN	Cloudflare Inc ECC CA-3
Public key	RSA 3072
Not after	2026-03-01 (expired or near-expiration — action required)
QHS score	30
Asset ID	ASSET-3

Missing from CBOM: private key locations, HSM/KMS presence, certificate renewal automation, CRL/OCSP behaviour.

JWT / OIDC / JWKS

Field	Value
JWKS host	temp-hybrid-migration-42ff.test
JWKS key	kid-1084, RSA RS256, 3072 bits
JWKS QHS score	35 (ASSET-5)
OIDC id_token algs	RS256 (ASSET-7), ES256 (ASSET-8)
Token endpoint auth	client_secret_basic — quantum-vulnerable (ASSET-9)
Token endpoint auth (safe)	private_key_jwt — not vulnerable (ASSET-10)

HTTP Posture

- Missing strict security headers: CSP, X-Frame-Options, Permissions-Policy (ASSET-12).
- No API auth findings (ASSET-11).

3. Migration Strategy

Goal: move temp-hybrid-migration-42ff.test to a pragmatic post-quantum-hybrid posture with minimal downtime and validated client interoperability.

Phase 0

Discovery & Controls — Prepare

Owners: Platform Owner (PO), Security Architect (SA)

Activities

- Inventory private keys and determine HSM/KMS capabilities.
- List OIDC clients and their authentication capabilities.
- Confirm certificate lifecycle automation.
- Capture current test harness.

Acceptance criteria

- Inventory completed.
- HSM/KMS capability matrix available.
- Client compatibility list produced.

Phase 1

Short-Term Hardening — Classical Best Practices

Owners: PO, NetOps, SA

Activities

- Renew/replace RSA 3072 cert (or reissue with ECDSA) if expired.
- Remove weak TLS options; enforce TLS 1.3.
- Enable strict security headers.
- Move token endpoint away from client_secret_basic for high-risk clients.

Acceptance criteria

- Site serves a valid cert chain.
- TLS 1.3 enforced; TLS 1.2 removed or restricted.
- Header issues resolved.
- Critical clients moved off client_secret_basic or exceptions documented.

Phase 2

Key Material Modernisation & PKI/KMS Integration

Owners: PO, KMS/HSM Admin (KMA)

Activities

- Migrate signing keys to KMS/HSM.
- Configure JWKS to serve keys protected by KMS.
- Enable private_key_jwt for clients.
- Implement JWKS rotation plan.

Acceptance criteria

- Private keys not stored on web servers.
- JWKS serving public keys corresponding to KMS-protected private keys.
- Successful sign/verify operations via KMS.
- Rotation tested.

Phase 3

Introduce Hybrid / Practical PQC

Owners: SA, DevOps, Vendor Liaison

Activities

- Implement hybrid TLS key exchange (classical ECDHE + PQC candidate) — ExeQuantum CipherForge provides a stateless API for ML-KEM and ML-DSA with BYOK support.
- Add hybrid JWT signing where toolchains support it; CipherForge's signing API supports ML-DSA and hybrid modes out of the box.
- Ensure clients can verify hybrid signatures.

Acceptance criteria

- Hybrid TLS handshake validated in lab with both classical and PQC components.
- JWKS and OIDC tokens include hybrid signatures accepted by test clients.

Phase 4

Validation, Staged Cutover & Monitoring

Owners: PO, SA, SRE, QA

Activities

- Run staged rollout to production.

Acceptance criteria

- No unexplained client failures.

- Continuous monitoring for crypto errors and client failures.
- Finalise rotation schedules and recovery plans.
- Monitoring indicates hybrid cryptography in active use.
- Documented fallback and rollback steps verified.

4. Detailed Action Plan

01

Confirm certificate validity and renewal status for ASSET-3.

Owner: Platform Owner (PO) + PKI Admin

Acceptance: Confirm whether not_after 2026-03-01 is expired. If expired or within renewal window, issue and deploy new cert to staging.

02

Document private key locations for TLS cert and JWKS RSA key (ASSET-3 and ASSET-5).

Owner: PO + Infrastructure Engineer (IE)

Acceptance: Inventory file listing exact paths, servers, and backups for private keys, or confirmation that keys are CA/Cloudflare-managed.

03

Verify whether TLS and JWKS private keys are in an HSM/KMS and capture KMS provider and API capabilities.

Owner: KMS/HSM Admin (KMA)

Acceptance: Capability matrix confirming whether KMS supports signing, key wrapping, rotation, attestation, and PQC algorithm support.

04

Test renewal automation (ACME/CA integration) and document CI/CD processes for cert/key deployment.

Owner: DevOps

Acceptance: Automated renewal pipeline tested in staging; new cert installable without manual restart, or with documented minimal downtime.

05

Harden TLS configuration: prefer TLS 1.3 only; if TLS 1.2 must stay, restrict to modern ciphers and enforce strong ECDHE groups.

Owner: NetOps

Acceptance: TLS scan in staging shows only TLS 1.3 enabled; TLS 1.2 removed or restricted to allowed ciphers; no regression for clients in compatibility list.

06

Replace RSA-based site certificate with an ECDSA (P-256) certificate or reissue with a key backed by KMS/HSM.

Owner: PKI Admin + PO

Acceptance: New cert deployed, chain validated by browsers and automated tests, ECDSA cert served. If ECDSA not universally supported, document exception.

07

Move signing key private material for JWKS into KMS/HSM and configure token signing to call the KMS instead of local private key.

Owner: KMA + Auth Service Owner

Acceptance: Tokens in staging signed using KMS; verification using JWKS public key succeeds; no key material on web servers.

08

Inventory OIDC clients currently using client_secret_basic (ASSET-9).

Owner: Identity Provider (IdP) Admin + App Owners

Acceptance: List of clients with auth method; migration plan for each to private_key_jwt or alternative asymmetric method documented.

09

Migrate client_secret_basic clients to private_key_jwt where possible; for non-capable clients, introduce short-lived credentials and strengthen monitoring.

Owner: App Owners + IdP Admin

Acceptance: Each migrated client authenticates successfully with private_key_jwt; remaining clients have documented compensating controls.

10

Add ES256 support for JWKS/token signing or migrate JWT signing to ECDSA-based keys.

Owner: Auth Service Owner

Acceptance: Signed tokens accepted by clients; JWKS updated with ES256 public key entry; interoperability tests pass.

11

Design and test a hybrid JWT approach (classical signature + PQC signature) in a lab if vendor/client toolchains support it.

Owner: SA + DevOps + QA

Acceptance: Lab tokens show both signatures; verification library accepts hybrid verification. Optional until client support is confirmed.

12

Implement JWKS rotation and key rollover process (automated).

Owner: Auth Service Owner + DevOps

Acceptance: Rotation performed in staging with no token verification failures; timeline and key overlap windows documented.

13

Implement monitoring and telemetry for TLS handshake failures, JWKS verification errors, token signature validation failures, and authentication errors.

Owner: SRE + Security Monitoring

Acceptance: Alerts configured for anomalies and tested by injecting specific errors in staging.

14

Address HTTP header posture (ASSET-12) — add missing security headers (CSP, X-Frame-Options, Permissions-Policy) and test for application impact.

Owner: Web Application Owner

Acceptance: Scans show headers present; no breaking user functionality.

15

Plan and test rollback procedures for cert/key changes, hybrid activation, and client migrations.

Owner: PO + SRE + SA

Acceptance: Rollback playbook exists for each major change and is tested in staging; rollback restores prior keys/certs within defined RTO.

16

Communicate changes and client compatibility requirements to integrators and partners, with migration windows and test endpoints.

Owner: Product/Integration Manager

Acceptance: Stakeholders confirm scheduled test slots and have tested against staging endpoints.

17

Final cutover: deploy hybrid PQC-enabled TLS/JWKS in a controlled canary followed by full rollout.

Owner: PO + SRE + SA

Acceptance: Canary shows no errors during observation window; full rollout proceeds; monitoring remains green.

5. Tooling Options

A. Off-the-shelf / Vendor-Managed

ExeQuantum CipherForge (PQCaaS)

Evaluate: API-native PQC implementation supporting ML-KEM, ML-DSA, and HQC-192. Stateless, BYOK-first, deployable hybrid, cloud, on-premises, or air-gapped. Formally verified Jasmin/C implementation — eliminates compiler-introduced timing side-channels. Drop-in for Phase 3 hybrid key exchange and JWT signing.

Pitfalls: Requires API integration into your auth and TLS stack; plan for key management alignment with your existing KMS.

Cloudflare (or CDN vendor with PQC/hybrid support)

Evaluate: Hybrid TLS options, certificate management integration, edge key storage (Keyless SSL alternatives).

Pitfalls: Vendor-specific implementations and trust assumptions; need to ensure private key control and attestation; verify PQC algorithm support timelines.

HashiCorp Vault + HSM (AWS KMS, Azure Key Vault, Google KMS)

Evaluate: API signing operations, key rotation, key import/export policies, availability, and HSM attestation.

Pitfalls: Potential latency for signing requests; migrating existing private keys into KMS may require re-issuing certs; some KMS providers lack PQC algorithms.

Certificate Authorities / PKI Providers (Let's Encrypt, DigiCert, Keyfactor)

Evaluate: Support for ECDSA, automation, short TTL cert issuance, and attestation of key handling.

Pitfalls: CA policy constraints; inability to provision PQC certs until industry support stabilises.

Identity Platforms (Auth0, Okta, Keycloak)

Evaluate: Support for private_key_jwt, JWKS management, rotation APIs, and upcoming PQC support for token signing.

Pitfalls: Multi-tenant constraints; migration friction for custom clients.

B. Build / Customise In-House

Custom KMS Wrapper + HSM Integration

Evaluate: Secure signing APIs, audit logs, rate limiting, key versioning, and key attestation.

Pitfalls: Maintenance burden, security of wrapper, compliance overhead, and high-availability requirements.

Hybrid Signature Layer in the Auth Service

Evaluate: Signature formats, token size implications, client SDK changes.

Pitfalls: Interoperability with third-party clients and libraries; verification complexity.

Custom CLI/Tools for Automated Rotation and Deployment

Evaluate: Idempotence, secrets handling, and CI/CD integration.

Pitfalls: Secrets leakage risk; race conditions during key rollover.

6. Controls & Governance

Crypto Policy

- Define accepted algorithms and minimum key sizes, and a migration timeline (e.g., deprecate pure RSA post-2027; move to ECDSA/P-256 and hybrid PQC by a defined date).
- Document exceptions and approval workflow.

Key Management

- Require private keys for TLS and JWT signing to reside in KMS/HSM where possible.
- Enforce MFA and role-based access control for key operations.
- Maintain an authoritative inventory of key IDs, owners, and usage.

Rotation & Expiration

- TLS cert rotation: automate renewals; use short lifetimes where operationally feasible.
- JWKS signing keys: rotate on a planned cadence (yearly or faster if threat landscape changes); ensure overlap windows for token verification.
- Client secrets: enforce TTL and rotation for remaining client_secret_basic instances.

Testing

- Maintain a staging environment that mirrors the production crypto stack.
- Test harness for TLS handshakes, token issuance and validation, and JWKS rotation scenarios.
- Include interoperability tests for major client platforms.

Monitoring & Logging

- Log key events: signing failures, KMS errors, JWT validation failures, TLS handshake anomalies.
- Set alerting thresholds for upticks in signature or handshake failures.
- Retain cryptographic audit logs for forensics and compliance.

Change Governance

- All crypto changes must pass through approval (Security Architect), runbook creation (SRE), and staged testing (QA).
- Maintain a rollback plan and a communication plan for consumers.

7. Rollback & Risk Management

Rollback Plans

- TLS cert changes: retain previous certificate and private key accessible via KMS versioning; re-enable on failure.
- JWKS rotation: use overlapping key windows. Keep old public key in JWKS for N days to allow earlier-signed tokens to validate; revert signing key to previous version if failures occur.
- Client auth changes: maintain parallel acceptance mode where both client_secret_basic and private_key_jwt are accepted during the migration window.

Risk Mitigation

- Start with canary/staging; restrict exposure to a small percentage of traffic.
- For authentication failures, failing closed is safer but may cause outage — choose policy per client and business impact.
- Monitor continuously and be ready to revert within your defined RTO and RPO.

Known Risks from CBOM

- RSA 3072 certificate and JWKS RSA keys carry medium-term classical risk; immediate availability issues may surface if keys become unavailable or incompatible after changes.
- client_secret_basic presence represents credential compromise risk; migration must be prioritised.
- Missing PKI/KMS details introduce deployment risk — the Phase 0 discovery step is a hard prerequisite.

8. How ExeQuantum Can Help

We can run a focused discovery to map private key locations, KMS/HSM capability, and the full client inventory — removing the Phase 0 unknowns.

We can deploy CipherForge as the PQC implementation layer for Phase 3 — providing ML-KEM and ML-DSA via a stateless API that integrates with your existing KMS, with no private key material ever leaving your environment.

We can provide a validated staging test harness to trial hybrid TLS handshakes and JWKS hybrid signatures against representative clients, producing pass/fail matrices.

We can help draft the documented runbooks, rotation schedules, and monitoring queries to make rollout reproducible and auditable.

Immediate Next Steps

1. **Verify certificate expiry for ASSET-3 and locate private keys for TLS and JWKS (ASSET-3, ASSET-5). These are blockers for migration.**
2. Collect the list of OIDC clients using client_secret_basic (ASSET-9). This is an operational priority.