

Sanctum SecOps LLC Certification Practice Statement (CPS)

Status: DRAFT — v0.1 **Document OID:** 1.3.6.1.4.1.65953.23.1.2 **Version:** 0.1 **Effective date:** DRAFT — not yet effective. Pending Brian Vicente review and approval. **Document date:** 2026-07-27

DRAFT NOTICE. This document is version 0.1 and is a DRAFT. It describes the practices used to build and operate the LAB/TEST demonstration PKI hierarchy ([sanctum-root-ca-g1](#) , [sanctum-issuing-ca-g1](#) , [sanctum-pqc-ca-g1](#)) generated by [scripts/gen-ca.sh](#) for the Cygnus PQC console repository build. It implements the companion Certificate Policy ([docs/CP.md](#) , OID 1.3.6.1.4.1.65953.23.1.1). Section numbering follows RFC 3647 verbatim, including sections with "No stipulation," because auditors check for the presence of every numbered section.

1. Introduction

1.1 Overview

1.1.1. This Certification Practice Statement ("CPS") describes the practices that Sanctum SecOps LLC follows (for this demonstration hierarchy) in issuing, managing, revoking, and renewing public key certificates.

1.1.2. This CPS implements the Certificate Policy at [docs/CP.md](#) (OID [1.3.6.1.4.1.65953.23.1.1](#)). Where the CP states a requirement, this CPS states how that requirement is met in practice by [scripts/gen-ca.sh](#) and the surrounding repository tooling.

1.1.3. As with the CP, this CPS covers the LAB/TEST three-tier hierarchy only. See the DRAFT NOTICE above.

1.2 Document name and identification

1.2.1. Document name: **Sanctum SecOps LLC Certification Practice Statement (CPS)**.

1.2.2. Document OID: [1.3.6.1.4.1.65953.23.1.2](#) .

1.2.3. Version: 0.1 (DRAFT).

1.3 PKI participants

1.3.1 Certification authorities

1.3.1.1. `sanctum-root-ca-g1` — self-signed root, RSA-4096/SHA-384, 20-year validity, generated once via `openssl req -new -x509`.

1.3.1.2. `sanctum-issuing-ca-g1` — classical-algorithm issuing CA, ECDSA P-384/SHA-384, 10-year validity, signed by the root via `openssl x509 -req -CA`.

1.3.1.3. `sanctum-pqc-ca-g1` — PQC-flagged issuing CA (see Section 6.1.1 in the CP for the current signing-algorithm limitation), ECDSA P-384/SHA-384 today, 10-year validity, signed by the root.

1.3.2 Registration authorities

1.3.2.1. No stipulation. No separate RA exists; `scripts/gen-ca.sh` performs both CSR generation and signing in a single automated flow for this demo.

1.3.3 Subscribers

1.3.3.1. The sample leaf `demo.pki.sanctumsecops.com`, issued by `sanctum-issuing-ca-g1`.

1.3.4 Relying parties

1.3.4.1. See CP Section 1.3.4.

1.3.5 Other participants

1.3.5.1. No stipulation.

1.4 Certificate usage

1.4.1. See CP Section 1.4. This CPS adds no further stipulation.

1.5 Policy administration

1.5.1. See CP Section 1.5. Custodian: Brian Vicente. This CPS is maintained alongside the CP in `docs/CPS.md` and rendered via `scripts/render-docs.py`.

1.6 Definitions and acronyms

1.6.1. See CP Section 1.6.

2. Publication and repository responsibilities

2.1 Repositories

2.1.1. Practice: the repository directory layout is `public/{aia,crl,docs,oid,policy,pem}/`, created and populated by `scripts/gen-ca.sh`, `scripts/gen-oid-html.py`, and `scripts/render-docs.py`. `public/index.html` is the repository landing page.

2.2 Publication of certification information

2.2.1. Practice: DER-encoded certificates are published under `public/aia/` (one file per CA/leaf, matching the AIA `caIssuers` URI path `/aia/<name>.crt`); PEM copies are published under `public/pem/` for human/tooling convenience; DER CRLs are published under `public/crl/` (matching the CDP URI path `/crl/<name>.crl`).

2.3 Time or frequency of publication

2.3.1. Practice: publication happens at generation time when `scripts/gen-ca.sh` runs. No automated re-issuance cron job exists in v0.1.

2.4 Access controls on repositories

2.4.1. Practice: `scripts/fix-perms.sh` sets all directories under `public/` to `o+x` (world-traversable) and all files to `o+r` (world-readable), then verifies readability from an unprivileged (`nobody`-equivalent) account, per Hard Rule 4 in the build directive.

3. Identification and authentication

3.1 through 3.4. Practice matches CP Sections 3.1–3.4 exactly; this CPS adds no additional practice detail because the demonstration hierarchy performs no real-world identity verification. All CSR subject fields are populated directly by `scripts/gen-ca.sh` from constants in the script, not from any external identity-proofing workflow. No stipulation beyond that for v0.1.

4. Certificate life-cycle operational requirements

4.1 Certificate application

4.1.1. Practice: `scripts/gen-ca.sh` generates each CSR in-line via `openssl req -new`, immediately followed by signing. There is no separate submission/ review queue in v0.1.

4.2 Certificate application processing

4.2.1. No stipulation beyond Section 4.1.

4.3 Certificate issuance

4.3.1. Practice — root: `openssl req -x509 -days 7300 -sha384`. Issuing CAs: `openssl x509 -req -CA <root.pem> -CAkey <root.key> -CAcreateserial -days 3650 -sha384 -extfile <openssl.cnf> -extensions v3_intermediate_ca`. Leaf: `openssl x509 -req -CA <issuing-ca-g1.pem> -CAkey <issuing-ca-g1.key> -CAcreateserial -days 825 -sha256 -extfile <openssl.cnf> -extensions v3_leaf`.

4.3.2. Practice: every issued CA certificate and the leaf are immediately exported to DER (`openssl x509 -outform DER`) and copied to PEM, then placed under `public/aia/` and `public/pem/` respectively, in the same script run.

4.4 Certificate acceptance

4.4.1. No stipulation.

4.4.2. Practice: publication is automatic and immediate; see Section 4.3.2.

4.4.3. No stipulation.

4.5 Certificate key and certificate usage

4.5.1. See CP Section 4.5. No stipulation beyond that.

4.6–4.8 Certificate renewal / re-key / modification

4.6.1 through 4.8.7. No stipulation for v0.1. `scripts/gen-ca.sh` is idempotent-by-refusal: re-running it without `--force` exits non-zero if key material already exists (see Section 6.5), rather than silently renewing or modifying anything.

4.9 Certificate revocation and suspension

4.9.1. Practice: `scripts/gen-ca.sh` writes a `V` (valid) entry for the sample leaf into `sanctum-issuing-ca-g1`'s OpenSSL CA `index.txt` database, which is the file `openssl ca -revoke` would need to flip to `R` in a future revocation event. No certificate is revoked as part of a normal `gen-ca.sh` run.

4.9.2 through 4.9.6. No stipulation for v0.1 beyond the CP.

4.9.7. Practice — **CRL issuance:**

```
openssl ca -config <issuing-ca-g1/openssl.cnf> -gencrl \  
-keyfile <issuing-ca-g1.key> -cert <issuing-ca-g1.pem> \  
-out <issuing-ca-g1.crl.pem>  
openssl crl -in <issuing-ca-g1.crl.pem> -outform DER \  
-out public/crl/sanctum-issuing-ca-g1.crl
```

A second CRL is generated after bumping `crlnumber` to model the delta artifact `sanctum-issuing-ca-g1+.crl`, with a shorter (1-day) validity window. The root CA's CRL uses a 365-day validity window, reflecting offline-root operational practice.

4.9.7.1. **CDP/AIA URIs are embedded as plain `http://`, never `https://`**, matching RFC 5280 guidance and the non-negotiable build constraint. Verified by direct extraction:

```
$ openssl x509 -inform DER -in public/aia/leaf-demo.crt -noout -text | grep -A4 "CRL  
Distribution"  
    X509v3 CRL Distribution Points:  
        Full Name:  
            URI:http://crl.sanctumsecops.com/crl/sanctum-issuing-ca-g1.crl  
        Full Name:  
            URI:http://crl.sanctumsecops.com/crl/sanctum-issuing-ca-g1+.crl  
$ openssl x509 -inform DER -in public/aia/leaf-demo.crt -noout -text | grep -A2  
"Authority Information"  
    Authority Information Access:  
        CA Issuers - URI:http://pki.sanctumsecops.com/aia/sanctum-issuing-ca-g1.crt
```

4.9.8 through 4.9.16. No stipulation for v0.1.

4.10 Certificate status services

4.10.1. Practice: CRL only; see Section 4.9.7. No OCSP responder is implemented.

4.11 End of subscription

4.11.1. No stipulation.

4.12 Key escrow and recovery

4.12.1. Practice: no escrow. Private keys are generated locally by `openssl genpkey` inside `scripts/gen-ca.sh`'s working directory and never transmitted elsewhere.

5. Facility, management, and operational controls

5.1 through 5.8. No stipulation for v0.1. Practice note: this repository's private key material lives only under `pki-repo/ca/*/private/` (mode 700 directories, mode 600 key files) inside the build sandbox — it is not deployed to any HSM, YubiKey, or physical facility. Production key custody practices (offline air-gapped root, YubiKey/HSM-backed intermediates) are described at the architecture level in the `sanctum-pki-cert-ops` skill and are deferred to a future CPS version once the production hierarchy (distinct from this LAB/TEST material) is built.

6. Technical security controls

6.1 Key pair generation and installation

6.1.1. Practice — exact commands run by `scripts/gen-ca.sh`:

```
# Root
openssl genpkey -algorithm RSA -pkeyopt rsa_keygen_bits:4096 -out root.key

# Issuing CAs (classical + PQC-flagged)
openssl genpkey -algorithm EC -pkeyopt ec_paramgen_curve:P-384 -out issuing-ca-g1.key
openssl genpkey -algorithm EC -pkeyopt ec_paramgen_curve:P-384 -out pqc-ca-g1.key

# Sample leaf
openssl genpkey -algorithm EC -pkeyopt ec_paramgen_curve:P-256 -out leaf-demo.key
```

6.1.2. Practice note on `sanctum-pqc-ca-g1`: this CA's own signing key is ECDSA P-384 today, not ML-DSA, because OpenSSL 3.5.5 as configured in this sandbox lacks a default provider exposing ML-DSA key generation through the `openssl genpkey / openssl req / openssl ca` CLI surface used

by this script. The CA's *name and policy intent* (PQC profile target) are documented here and in the CP (Section 7.1.3.2) even though today's implementation is classical-only. This gap must be closed before any production PQC CA claims conformance with FIPS 204.

6.1.3 through 6.1.7. See CP Section 6.1 for the corresponding policy statements; practice matches exactly.

6.2 Private key protection and cryptographic module engineering controls

6.2.1. Practice: `chmod 700` on each CA's `private/` directory, `chmod 600` on each private key file, performed inline by `scripts/gen-ca.sh` immediately after key generation. No HSM or hardware token is used for this lab hierarchy.

6.3 Other aspects of key pair management

6.3.1. No stipulation.

6.3.2. Practice — validity periods: root 7300 days, issuing CAs 3650 days, leaf 825 days, encoded as `DAYS_ROOT`, `DAYS_ICA`, `DAYS_LEAF` constants at the top of `scripts/gen-ca.sh`.

6.4 through 6.6

No stipulation for v0.1.

6.7 Network security controls

6.7.1. Practice: `crl.sanctumsecops.com` and `pki.sanctumsecops.com` must be configured with no HSTS header and no HTTP→HTTPS redirect. This is enforced at the web-server-configuration layer outside this script's scope, but is called out here as a hard operational requirement that this CPS depends on.

6.8 Time-stamping

6.8.1. No stipulation.

7. Certificate, CRL, and OCSP profiles

7.1 through 7.3. Practice matches the CP's profile definitions exactly (Section 7 of `docs/CP.md`). This CPS adds the following verification practice: after every `scripts/gen-ca.sh` run, the following commands are executed and their output reviewed before the repository is considered ready to publish:

```
openssl crl -inform DER -in public/crl/sanctum-issuing-ca-g1.crl -noout -text
openssl crl -inform DER -in public/crl/sanctum-issuing-ca-g1+.crl -noout -text
openssl x509 -inform DER -in public/aia/sanctum-root-ca-g1.crt -noout -subject -issuer
-dates
openssl x509 -inform DER -in public/aia/sanctum-issuing-ca-g1.crt -noout -subject
-issuer -dates
openssl x509 -inform DER -in public/aia/sanctum-pqc-ca-g1.crt -noout -subject -issuer
-dates
openssl x509 -inform DER -in public/aia/leaf-demo.crt -noout -subject -issuer -dates
openssl x509 -inform DER -in public/aia/leaf-demo.crt -noout -text | grep "CRL
Distribution"
openssl x509 -inform DER -in public/aia/leaf-demo.crt -noout -text | grep "Authority
Information"
```

These are also captured verbatim in [scripts/verify-pki-repo.sh](#) (Section 10.8 of the build directive).

8. Compliance audit and other assessments

8.1 through 8.6. No stipulation for v0.1.

9. Other business and legal matters

9.1 through 9.17. See the corresponding CP sections (Section 9 of [docs/CP.md](#)); this CPS adds no further practice detail because this is a non-commercial, DRAFT, LAB/TEST hierarchy. No stipulation beyond the CP for each of: 9.1 Fees, 9.2 Financial responsibility, 9.3 Confidentiality, 9.4 Privacy (see [public/policy/privacy-policy.md](#)), 9.5 IP rights, 9.6 Representations and warranties, 9.7 Disclaimers, 9.8 Limitations of liability, 9.9 Indemnities, 9.10 Term and termination, 9.11 Notices, 9.12 Amendments, 9.13 Dispute resolution, 9.14 Governing law, 9.15 Compliance with law, 9.16 Miscellaneous provisions, 9.17 Other provisions.

Generated from a single Markdown source ([docs/CPS.md](#)) by [scripts/render-docs.py](#) , which produces both [public/docs/CPS.html](#) and [public/docs/CPS.pdf](#) so the two can never diverge. Do not hand-edit the HTML or PDF outputs.

Rendered 2026-07-27 11:56 UTC by `scripts/render-docs.py` from a single Markdown source. HTML and PDF outputs are generated from the same source and can never diverge.