Generating a post-quantum OpenPGP key with Sequoia on Fedora

Stavros Kousidis

Published 2026-09-26
Tested on Fedora 45 · Sequoia PGP sq

This guide records a minimal, reproducible path for generating a hybrid post-quantum OpenPGP certificate with Sequoia PGP. It is deliberately marked **experimental**: client support for the algorithms and certificate profile is still evolving, and every desktop client must be tested separately.

Goal

Generate a password-protected transferable secret key with:

- ML-DSA-65 combined with Ed25519 for signatures;
- ML-KEM-768 combined with X25519 for encryption;
- a five-year validity period; and
- a separate revocation certificate.

Prerequisites

Use a build of Sequoia PGP whose `sq` command exposes the required cipher suite and encryption algorithm:

```
sq --version
sq key generate --help
```

Before continuing, confirm that the help output lists `mldsa65-ed25519` and `mlkem768-x25519`. Package availability and command-line details may differ between Fedora releases and development builds.

Generate the key

Replace the example identity and output filenames before running the command. One option is to create a single explicit User ID containing both name and e-mail address:

```
sq key generate \
  --own-key \
  --userid "YOUR NAME <you@example.org>" \
  --profile rfc9580 \
  --cipher-suite mldsa65-ed25519 \
  --encryption-algorithm mlkem768-x25519 \
  --expiration 5y \
  --output openpgp-pq-secret.pgp \
  --rev-cert openpgp-pq.rev
```

Alternatively, replace the `--userid` option with separate identity arguments:

```
--name "YOUR NAME" \
--email "you@example.org"
```

Choose the identity representation that fits the intended application; you do not need to generate both certificates. Both forms worked in the Fedora 45 KMail and Evolution proof-of-concept tests. Evolution integrated more cleanly with the explicit `--userid` form. See the Evolution integration guide for that application-specific observation.

Enter a strong, unique passphrase when prompted. The secret-key file is sensitive even when encrypted.

The `rfc9580` profile selects the modern OpenPGP packet and certificate profile. The hybrid post-quantum algorithms are specified separately through the cipher-suite and encryption-algorithm options.

Inspect the result

```
sq inspect openpgp-pq-secret.pgp
```

Confirm at least the following properties:

- the fingerprint is present and recorded through an independent channel;
- the public-key algorithm is `ML-DSA-65+Ed25519`;
- the encryption algorithm is `ML-KEM-768+X25519`;
- the secret key is encrypted; and
- the creation and expiration times match the intended validity period.

Verify the output files

Also verify that both files exist and are non-empty:

```
ls -lh openpgp-pq-secret.pgp openpgp-pq.rev
```

Store the key safely

Keep the encrypted secret key and the revocation certificate in separate, access-controlled locations. Do not publish either file. Only the public certificate belongs in a public repository, WKD deployment or key-distribution service.

Current limitations

Generating a certificate does not imply that desktop mail clients can use it. In particular, private-key operations, middleware integrations and OpenPGP implementations may reject the algorithms or the certificate format. Test signing, verification, encryption and decryption with every component in the intended workflow.

Next steps

Continue with the Fedora 45 mail-client proof-of-concept guides for KMail, Claws Mail and Evolution. Their shared setup records the common Chameleon environment, disposable test certificate and verification methodology.