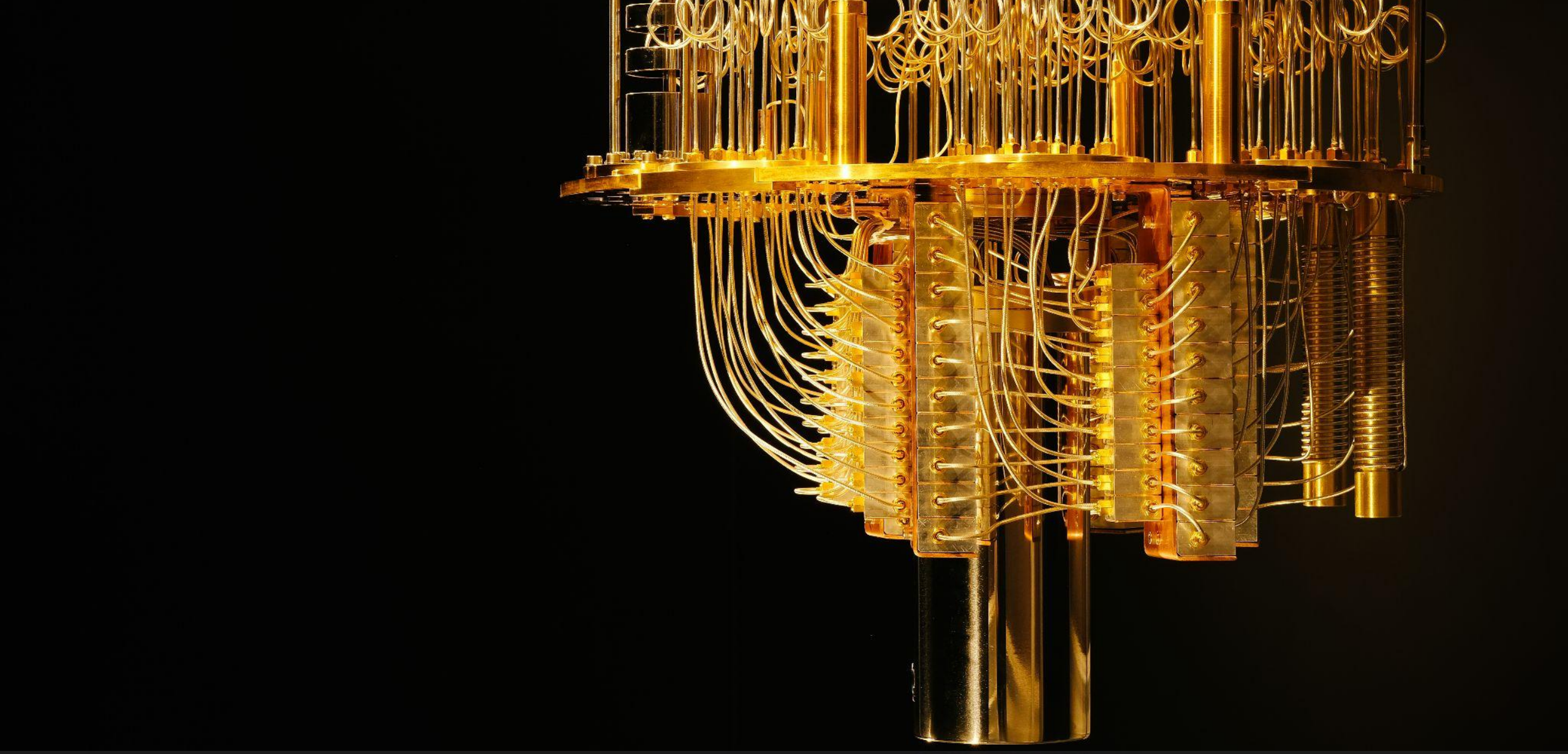
Red Hat
Enterprise Linux

Red Hat's Path to PQC with OpenSSL

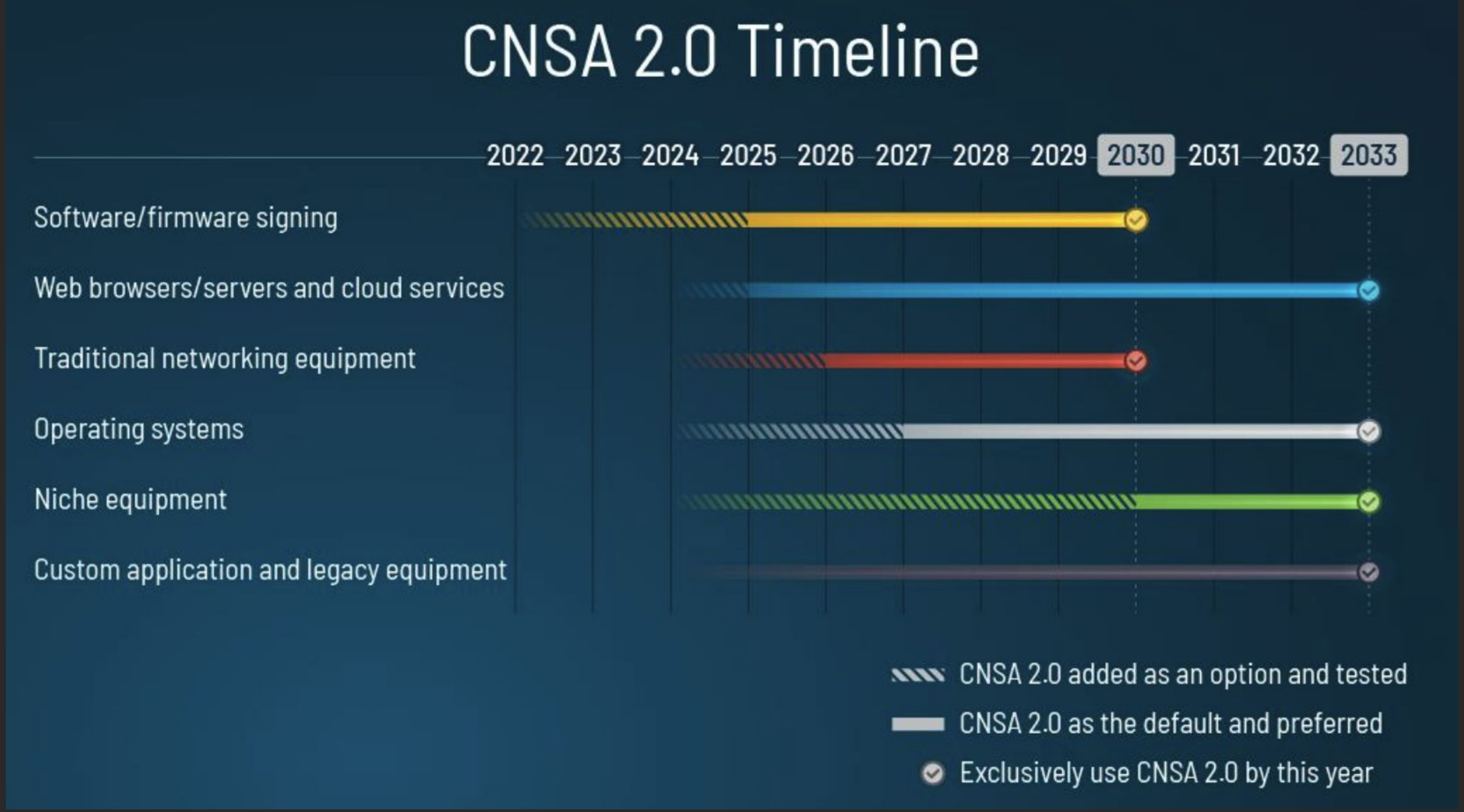
Clemens Lang
RHEL Crypto Team Product Owner

Red Hat



Source:
https://newsroom.ibm.com/download/quantum-computers-and-accelerated-discovery_40645906341_o.jpg
https://quantum.ibm.com/services/resources?limit=50&view=table
https://security.stackexchange.com/q/87345
https://crypto.stackexchange.com/q/436

CNSA 2.0 Timeline



Source:
https://media.defense.gov/2025/May/30/2003728741/-1/-1/0/CSA_CNSA_2.0_ALGORITHMS.PDF
https://www.gsma.com/newsroom/post-quantum-government-initiatives-by-country-and-region/

Red Hat

What we'll cover today

- Timelines vs. RHEL release cycle
- PQC in RHEL 10.0 & CentOS 10 Stream
- PQC signatures over RPM packages
- PQC and FIPS
- PQC in OpenSSL via PKCS#11

Timelines vs. RHEL release cycle

2025 **RHEL 10.0 Release**

2035 **RHEL 10.10 End of Life**

End of standard support

Some time after **2040** **RHEL 10 Extended Update Support EOL**

End of extended support

ML-DSA/ML-KEM Standards

August 2024

Standards for PQC in TLS

2026+

Regulators require PQC

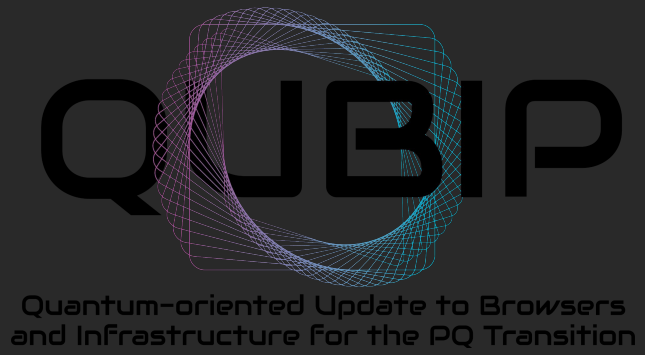
2030 - 2035

Sources:
https://access.redhat.com/support/policy/updates/errata#RHEL10_Planning_Guide

Red Hat

PQC Tech Preview

RHEL 10.0, non-default



Enabling

```
# dnf install crypto-policies-pq-preview
# update-crypto-policies --set DEFAULT:TEST-PQ
```

Result

- hybrid key exchange in TLS
- pure ML-DSA

OpenSSL win: Providers

Under the hood

OpenSSL 3.2.2

oqsprovider 0.8.0 with liboqs 0.12.0

Sources:
https://www.redhat.com/en/blog/post-quantum-cryptography-red-hat-enterprise-linux-10
https://github.com/open-quantum-safe/oqs-provider

```
[root@rhel-10-0 ~]# openssl s_client \
   -CAfile CA.crt \
   -connect test.openquantumsafe.org:6217 </dev/null \
      |& grep -A1 'Peer signature type'
Peer signature type: mldsa87
Negotiated TLS1.3 group: X25519MLKEM768
```

CentOS 10 Stream: PQC by Default

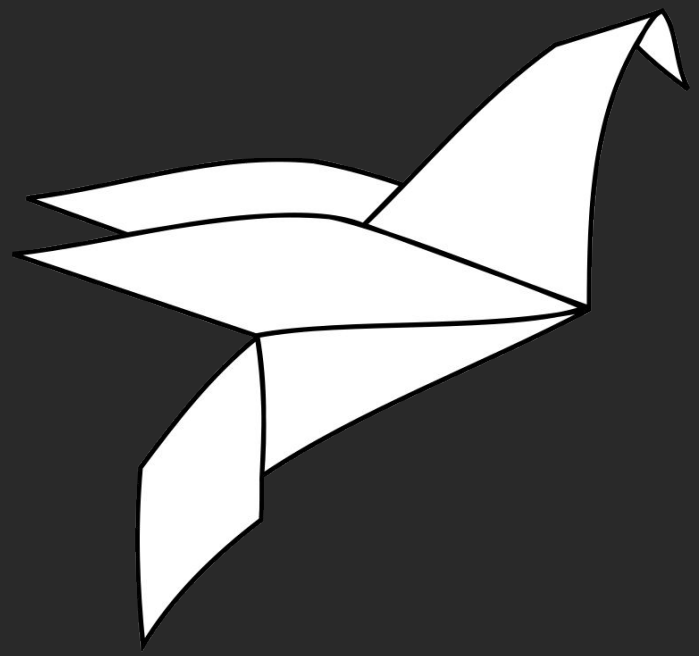
- OpenSSL 3.5.1, PQC by default
- We're testing for
  - PQC in TLS key exchange
  - Self-signed pure ML-DSA certs
  - Dual stack certs RSA/ML-DSA
- Code written against oqsprovider in 10.0 works

OpenSSL win: Compatibility

Sources:
https://github.com/openssl/openssl/issues/26326
https://github.com/openssl/openssl/pull/26853
https://gitlab.com/redhat/centos-stream/rpms/openssl/-/commit/98cf25a4c0d4b9fa03944c92a8d366621250985e
https://gitlab.com/redhat/centos-stream/rpms/crypto-policies/-/commit/20e322ef5720fa51b6056dae87f5b2b18a406854

Post-Quantum RPM Signatures

Using OpenSSL's primitives



- IETF WG is adding PQC to OpenPGP
- GnuPG unlikely to implement RFC 9580
- RPM transitioned to Sequoia-PGP
- Sequoia-PGP is adding PQC support with OpenSSL as backend

OpenSSL win: Reusability

Sources:
https://datatracker.ietf.org/doc/html/draft-ietf-openpgp-pqc-12

Post-Quantum Crypto and FIPS

An unresolvable dilemma?

- OpenSSL's FIPS provider allows continued use of older certified module
- Customers that need FIPS also the ones shouting loudest for PQC 😄
- FIPS certification takes years due to backlog

OpenSSL win: Providers

- Hybrids can be a stopgap measure
- NIST SP 800-227 Section 4.6 approves hybrid constructions
  - Secp256r1MLKEM768: Usable now with validated secp256r1
  - X25519MLKEM768: Usable in the future with validated ML-KEM

Sources:
https://openssl-communities.org/d/3cXGJCEx/enabling-hybrid-key-exchange-in-fips-mode-for-old-fips-providers
https://github.com/openssl/openssl/pull/27728 or https://github.com/openssl/openssl/pull/28103
https://csrc.nist.gov/pubs/sp/800/227/final

PQC Hardware Tokens

PKCS#11 3.2

Why PKCS#11?

Software Signing Keys in HSMs

HSMs typically support PKCS#11

PKCS#11 PQC Standardization

PKCS#11 version 3.2 adds PQC support

OpenSSL win: Providers

pkcs11-provider

Red Hat's pkcs11-provider supports ML-DSA through PKCS#11 3.2 in version 1.1

Sources:
https://github.com/latchset/pkcs11-provider/releases/tag/v1.1.0

Red Hat + OpenSSL = PQC

Rapid Prototyping to Full Support

- oqsprovider in 10.0
- OpenSSL 3.5 in CentOS 10 Stream

Reuse in Other Protocols

- RPM signatures using Sequoia-PGP
- Uses PQC from OpenSSL 3.5

Hybrids for FIPS Needs

- SP 800-227 enables hybrid key exchange in TLS

Hardware Security Modules

- pkcs11-provider allows using PQC from HSMs in OpenSSL
- Interesting for Software Signing

Thank you!
Questions?

I would appreciate feedback.

What's next in your org's PQ transition?

@neverpanic@chaos.social

Red Hat