Demystifying Post-Quantum Cryptography: The Hybrid Approach

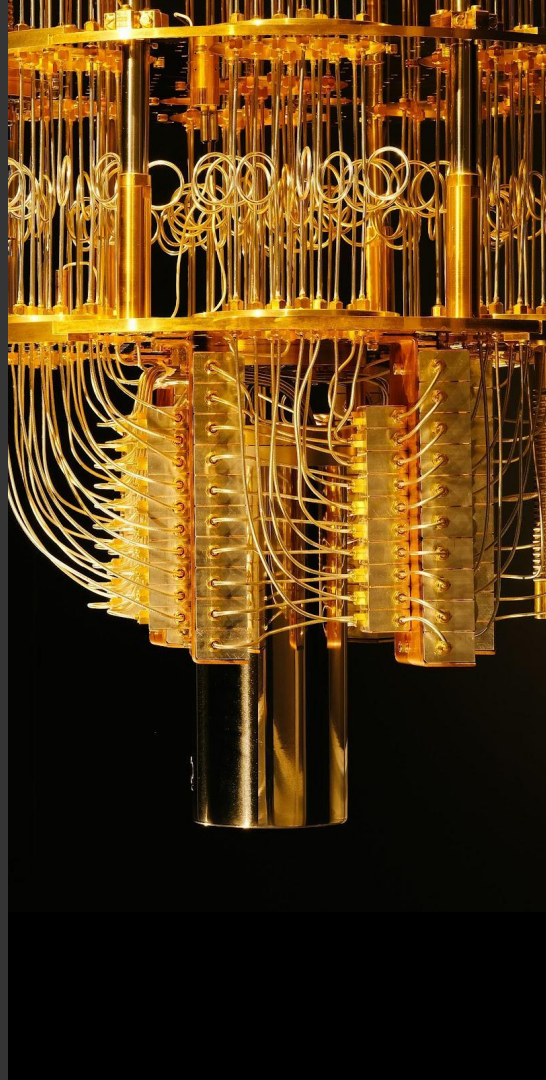
Rutvik Kshirsagar
Red Hat

Clemens Lang
Red Hat

Red Hat

Quantum Computers & PKI

- PKI underpins trust everywhere: TLS, SSH, code signing, identity.
- Security today relies on factorization and discrete log hardness.
- Quantum computers leveraging Shor’s algorithm can compromise RSA, Diffie-Hellman, and elliptic-curve cryptosystems.
- Practical attacks require large, fault-tolerant quantum systems



Is transition to PQC is even necessary at the moment?

Why is this seen as Y2Q / Q-Day problem?

Red Hat

Timelines

Time to Transition

Long (standards → vendor support → product upgrades → system replacement → decommissioning legacy)

Time You Need Data to Remain Secure

Short-lived → Long-lived → Archive /High-value

Time to Build a Quantum Computer

When will attackers have a quantum computer available?

Red Hat

Hybrid Crypto = Defense in Depth

OpenSSL 3.5+(with TLS 1.3) natively implements and supports the first set of **NIST-standardized PQC algorithms**:

- **Module-lattice key encapsulation**

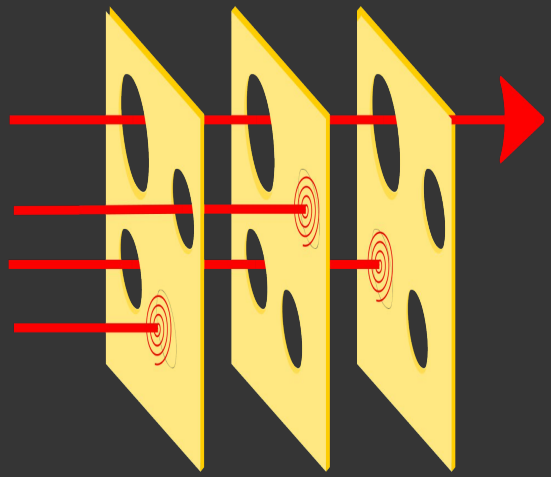
  ML-KEM *(FIPS 203)*

- **Hybrid Key Exchange**

  X25519MLKEM768 *(classical X25519+ML-KEM-768)*

- **Post Quantum Signatures**

  ML-DSA (FIPS 204) or SLH-DSA(FIPS 205)



Swiss Cheese Security

Red Hat

Variant Significance - Security Level 1/3/5

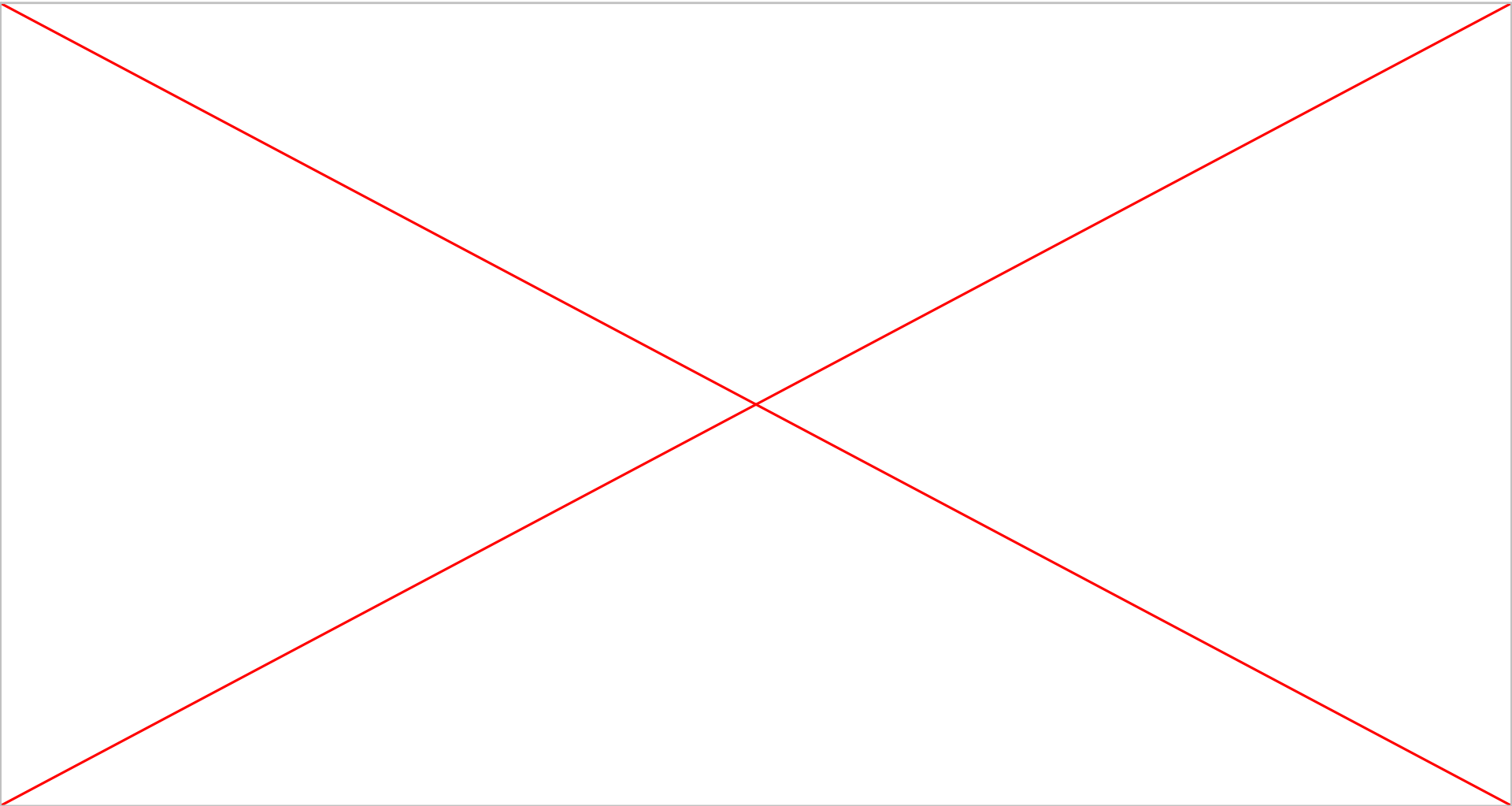
ML-KEM (512, 768, 1024)

- Lower variants: smaller keys and faster handshakes; suitable for short-lived sessions.
- Higher variants: larger keys; provide higher security margins for long-term confidentiality.

ML-DSA (44, 65, 87)

- Lower variants: smaller signatures and faster verification; useful for high-volume signing (e.g. TLS, logs).
- Higher variants: larger signatures; better suited for long-lived trust artifacts (e.g. firmware, root certs).

HYBRID PQC TLS DEMO



TLS 1.3 was explicitly redesigned so that:

- Key exchange can evolve independently of cipher suites
- Crypto agility is preserved

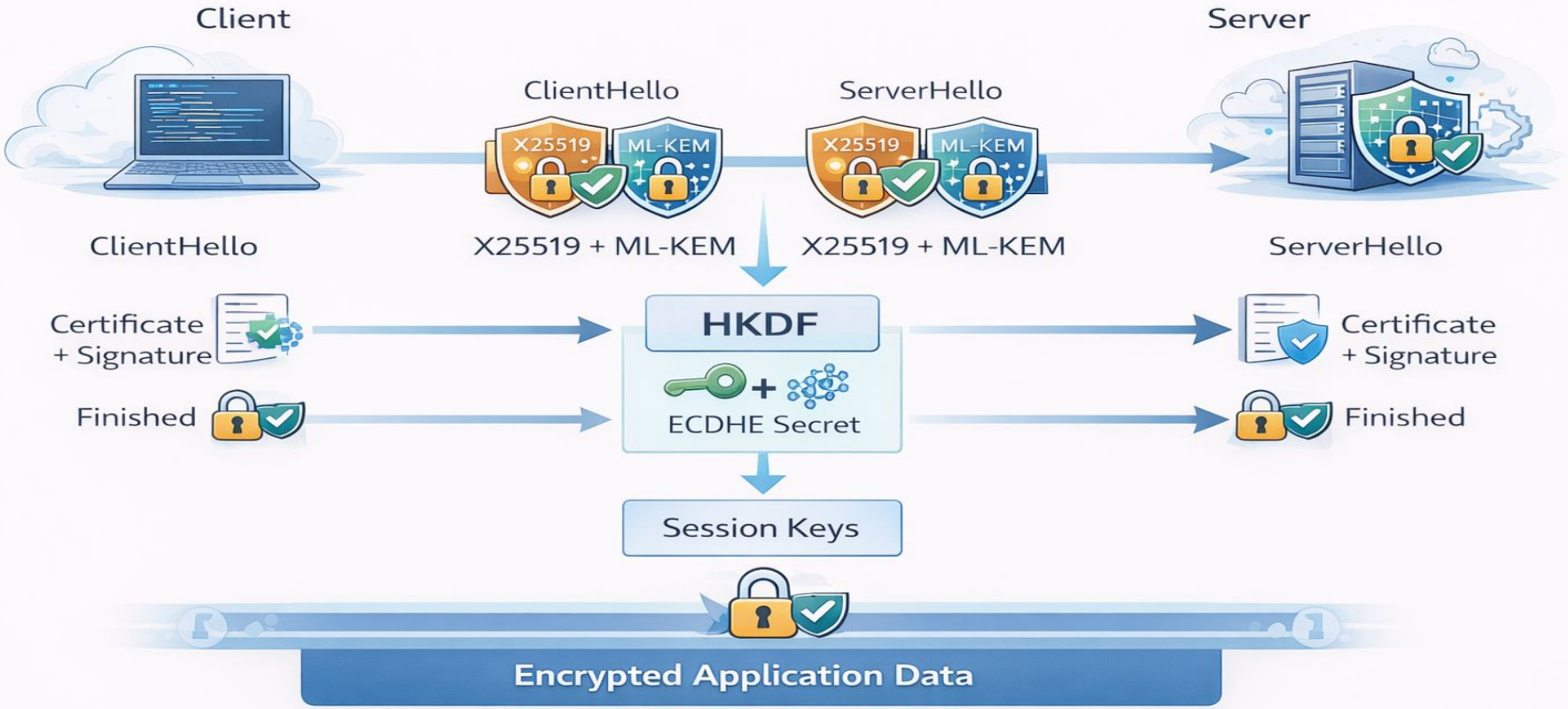
Hybrid PQC operates only in the key exchange, enabling safe incremental deployment in TLS 1.3.

X25519MLKEM768 determines how secrets are created.

MLDSA-65 / SLH-DSA determines who is trusted to create them.

Cipher suites are about how data is protected afterward.

Hybrid PQC TLS 1.3 Handshake



Where are we and why is this important?

- ~60% of Cloudflare's human TLS traffic uses hybrid ML-KEM key exchange
- IETF working groups are adding PQC (e.g. TLS, IPsec, SSH, ...)
- Chrome 116+ negotiates PQ handshakes by default.
- Fedora 43 supports PQC signatures and key exchange
- IBM z16 ships with lattice-based signatures and KEMs baked into firmware & boot.

https://github.com/ietf-wg-pquip/state-of-protocols-and-pqc

Red Hat

Cloudflare Radar

Search for locations, autonomous systems, reports, domains and more...

Adoption & Usage

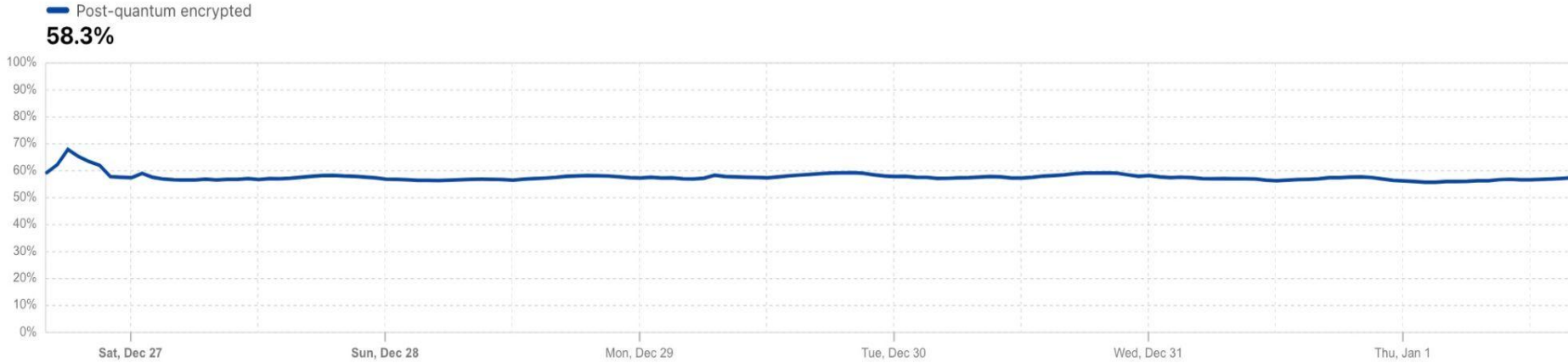
Worldwide

Post-quantum encryption

On essentially all domains served through Cloudflare, including this one, we have enabled hybrid post-quantum key agreement.
Check out our blog post *The state of the post-quantum Internet* for more context.

Post-quantum encryption adoption

Post-quantum encrypted share of HTTPS request traffic



Browser support

Check your browser for post-quantum encryption support

✓ PQ Your browser is connecting using the *X25519MLKEM768* key agreement, which is post-quantum secure.

PQC in Software Supply Chain

Problem: Long-lived systems (IoT, embedded, enterprise) require signatures that remain secure decades after release.

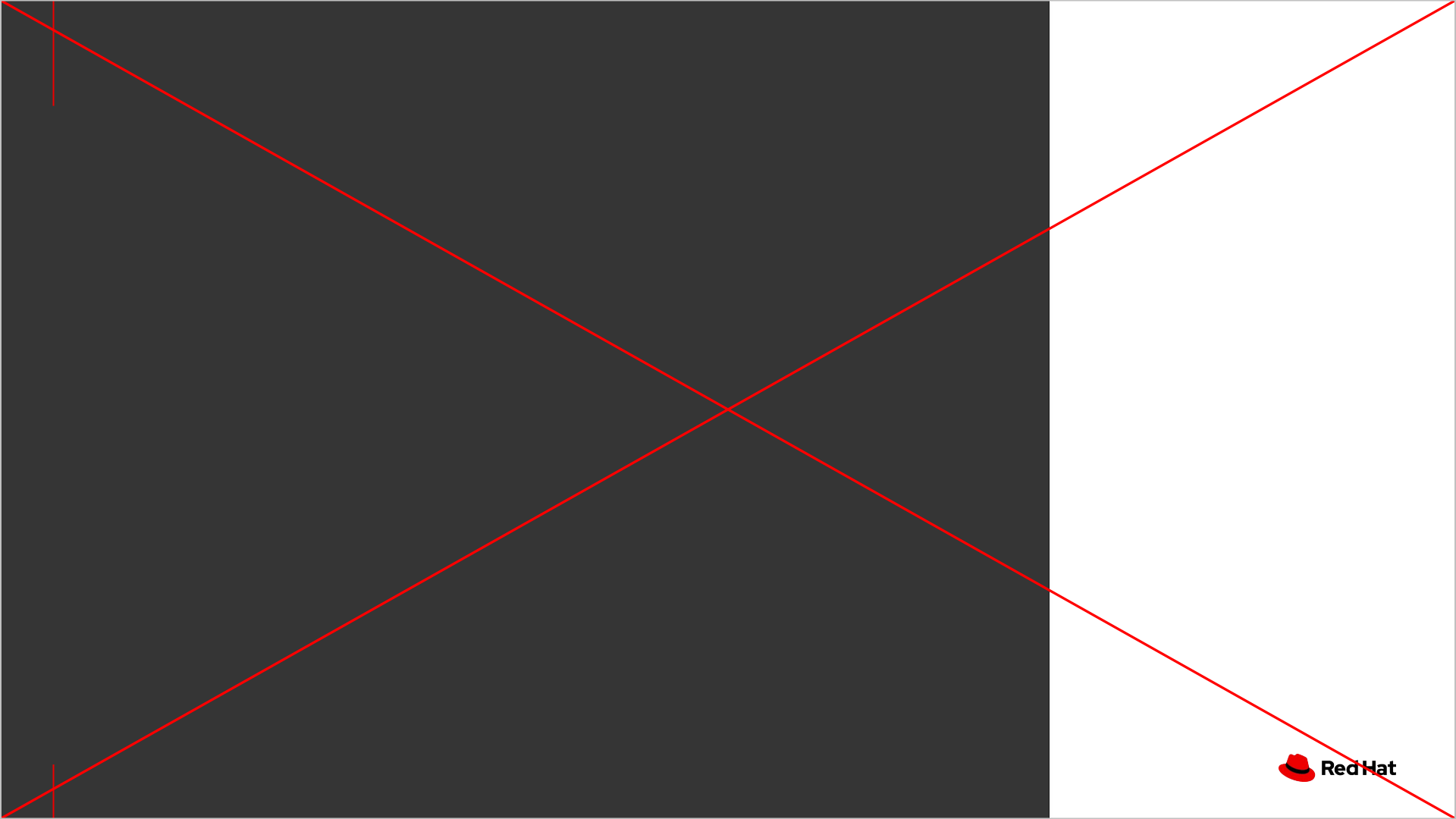
Solution: PQC signatures over RPMs

- OpenPGP draft openpgp-pqc with EdDSA/ML-DSA hybrid
- No support in GnuPG/LibrePGP
- Sequoia, rpgpie, GopenPGP, rnp are working on it

Demo: Sequoia-PGP with PKCS#11 backend using the Kryoptic software token signing an RPM

github.com/neverpanic/fosdem-rpm-pqc-signing-demo/

Red Hat



Summary

- Defense in Depth: Hybrid crypto (classical/PQC) combines X25519 and ML-KEM to protect against "Harvest-now-decrypt-later."
- Protocol Readiness: TLS 1.3 easily integrates PQC into the key exchange layer. SSH also has hybrid PQC support in recent versions.
- Global Momentum: ~60% of human TLS traffic, driven by Chrome and Cloudflare, uses hybrid KEM.
- Supply Chain: Hybrid signing ensures software remains verifiable post-classical crypto deprecation.

Questions?

Thank you for the attention.

Resources

- Post Quantum Cryptography in Sequoia PGP
  - draft-ietf-openpgp-pqc-16 - Post-Quantum Cryptography in OpenPGP
  - github.com/neverpanic/fosdem-rpm-pqc-signing-demo/
- NIST IR 8547 initial public draft, Transition to Post-Quantum Cryptography Standards
- Signing RPM packages using quantum-resistant cryptography | Red Hat Developer
- The Features of 3.5: Hybrid ML-KEM | OpenSSL Foundation
- Prototyping post-quantum and hybrid key exchange and authentication in TLS and SSH
- Post-quantum cryptography in Red Hat Enterprise Linux 10