Post-Quantum Transition: Standards, Effects on Protocols

RIPE89, October 29, 2024, Prague

Dmitry Belyavskiy
Principal Software Engineer

Red Hat

Who am I

Dmitry Belyavskiy
Red Hat Principal Software Engineer
Maintain: OpenSSL, OpenSSH

OpenSSL Technical Committee member since 2021

Current work: Post-Quantum transition in Red Hat

I am not
...a cryptographer
...a network engineer

Red Hat

QUBIP Consortium

Quantum oriented update to **B**rowsers and **I**nfrastructure for the **P**Q transition, QUBIP.EU

Fondazione Links – Passion for Innovation
Coordinator

Politecnico di Torino

Red Hat

Security Pattern

CSIC – Consejo Superior de Investigaciones Científicas

Telefónica

Universidad Politécnica de Madrid

Tampere University

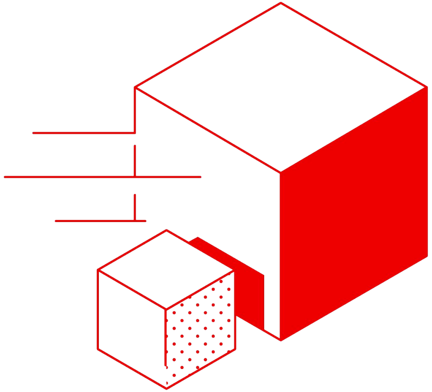
Smart Factory

Ciber Voluntarios.org

Disclaimer: QUBIP project is co-funded by the European Union under the Horizon Europe framework programme [grant agreement no. 101119746].


RIVEST CREATED US ALL THOSE YYEARS, BUT IN A POST QUANTUM WORLD, CAN WE REALLY TRUST EACH OTHER ANYMORE?

Quantum vs Post-Quantum



Quantum Cryptography

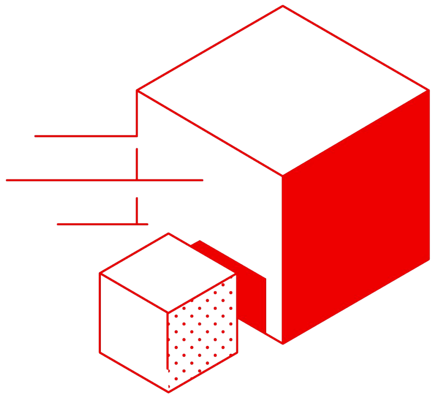
Cryptography based on Quantum Mechanics

Post-Quantum Cryptography

Also: Quantum-Safe, Quantum-Resistant

Cryptography resistant to Quantum Computers

Why Post Quantum transition?



Quantum Threats

Quantum Computers will break traditional cryptography

Shor algorithm to break RSA, (EC)DSA, (EC)DH

Quantum computers are in future

Post-Quantum algorithms are here

Timeline: circa 2030

Red Hat

NIST PQ contest

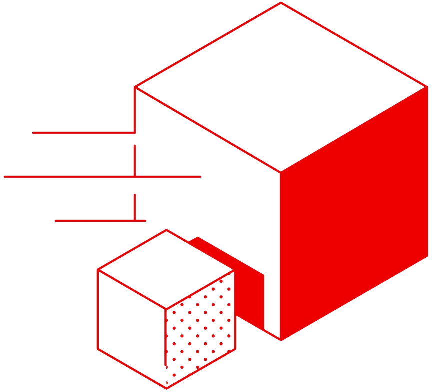
Announcement: 2016
69 participants in round 1

Chosen for standardisation: 2022
1 algorithm for Key Exchange, 3 for signature

Final standards: 2024
1 algorithm for Key Exchange, 2 for signature

Ongoing process
4 algorithms, 1 was successfully attacked
Additional Digital Signature Schemes

PQC: Standard bodies



Algorithms: NIST

Signature: ML-DSA (ex-Dilithium), SLH-DSA (ex-SPHINX+)

Key Establishment: ML-KEM (ex-Kyber)

Protocols: IETF

Post-Quantum Use In Protocols (pquip)

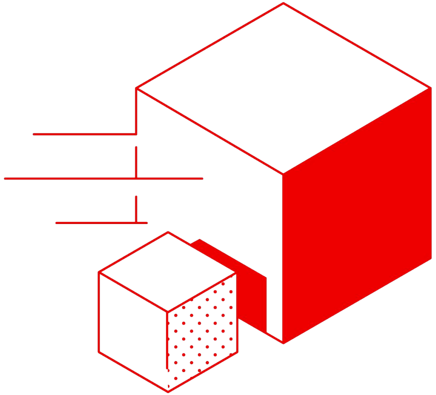
IETF Security Area

Hardware

OASIS group

Red Hat

PQ Math



Series of Red Hat blog posts

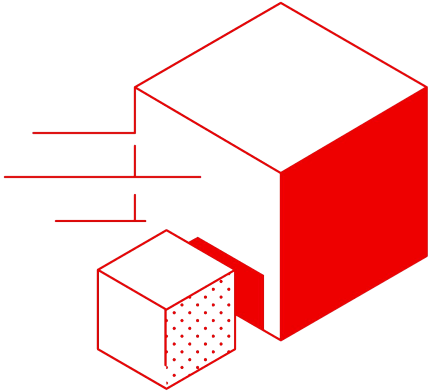
Post-quantum cryptography: An introduction

Post-quantum cryptography: Hash-based signatures

Post-quantum cryptography: Lattice-based cryptography

Post-quantum cryptography: Code-based cryptography

PQ transition challenges - I



Secure solution from insecure components

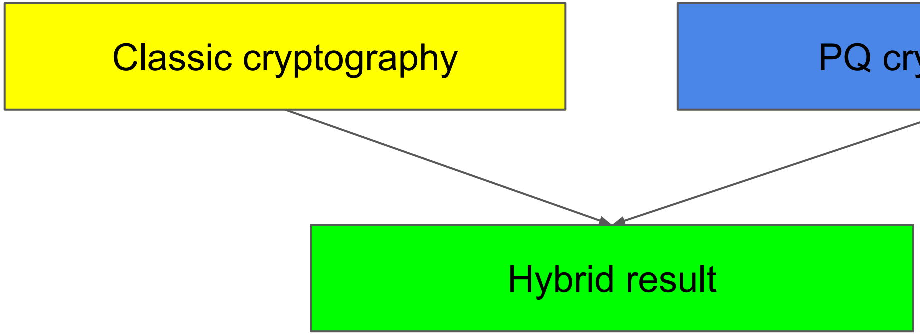
We can't trust classical algorithms

We can't trust new algorithms

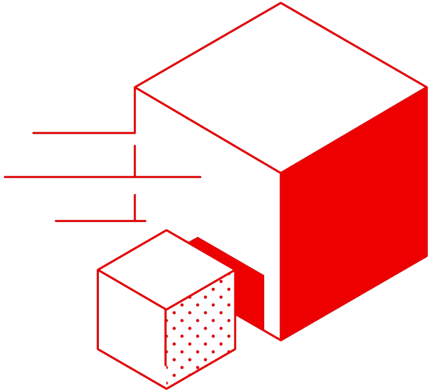
Temporary(?) solution

Hybrid solutions: combinations of classical and new algorithms

Hybrid solutions



PQ transition challenges – II



Size matters

Big keys/signatures

- RSA-3072 (classic): 387/384 bytes
- ML-DSA (PQ): 1312/2420 bytes

Other issues

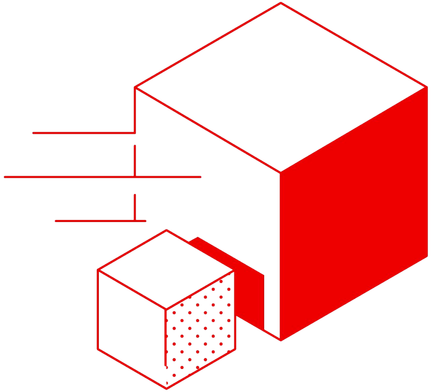
Performance problems

Compatibility problems

Network specific problems:

- Extra round trips
- UDP amplification
- DNSSec

DSA and KEM



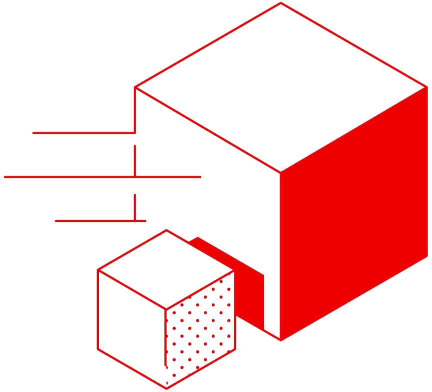
DSA: digital signature algorithms

Did you connect to a proper peer?
Was the email from a proper person?
Is your firmware issued by a proper source?

KEM: key establishment mechanism

Symmetric keys to protect communication

DSA



Threat model

Restore private key by the public one

Impersonate well-known site

Extract secrets in real-time

Countermeasures: rebuild chain-of trust

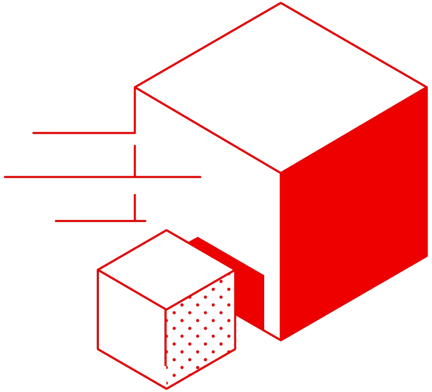
New hardware (CA/Browser forum requirements)

New trusted roots

New end-user certificates

Red Hat

KEM



Threat model

Collect data now

Restore symmetric keys later

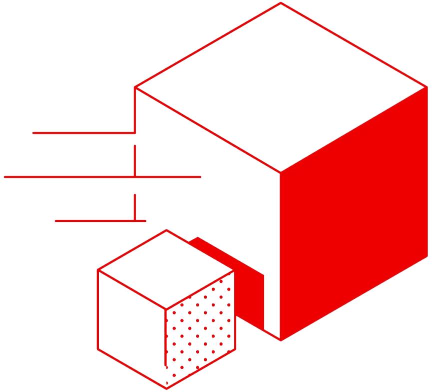
Extract secrets

Countermeasures

Use new software implementing PQ algorithms

Red Hat

TLS now and tomorrow



Pre-standard adoption

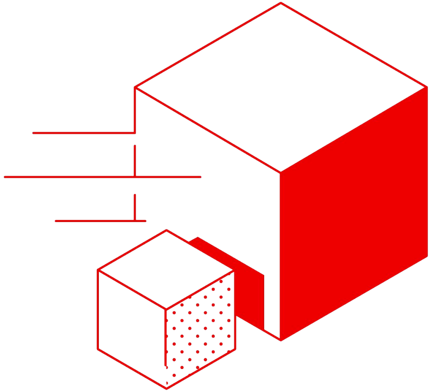
Key establishment: Kyber-based hybrids

Browsers, CDNs

Moving to standards

Kyber => ML-KEM

Traditional problems: extra round trips



Large certificates chains

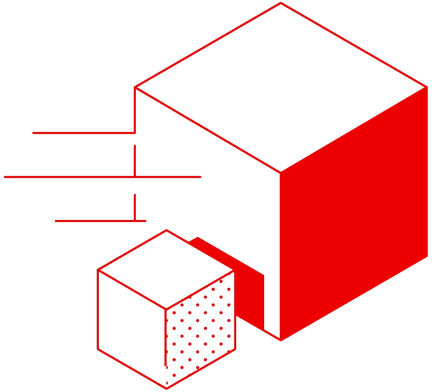
4k RSA => 22k ML-DSA

Response/request ratio limitations

QUIC: spec-level limitations 3x

DTLS: spec-level recommendation 3x, nobody implements

Traditional problems: TCP slowstart



Too small to fit certificate chain

TCP initial send window: 10 Maximum Segment Size

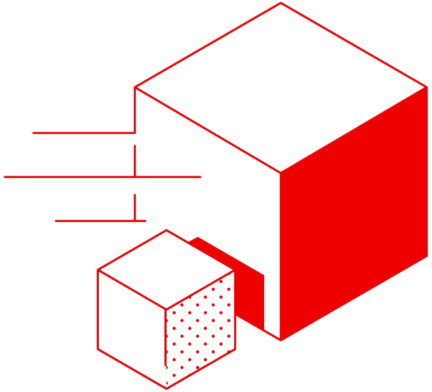
To avoid extra round-trips, 25 MSS is worth investigation

UDP based protocols

QUIC: has its own congestion control, worth investigating

DTLS: doesn’t have its own congestion control

DNSSec



All problems in one protocol

Small request, big response => amplification

Too big RRSIGs => don’t fit one packet

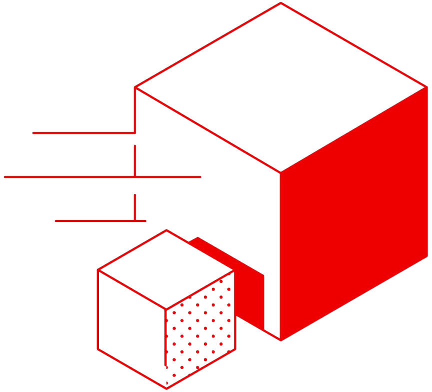
ARRF: a proposal to split RRs at application level

DNSSec field experiments

See presentation today later

Red Hat

Linux for PQ experiments



Fedora choice

liboqs by Open Quantum Safe

Low-level implementations

OpenSSL provider

Includes post-quantum crypto policy

Container

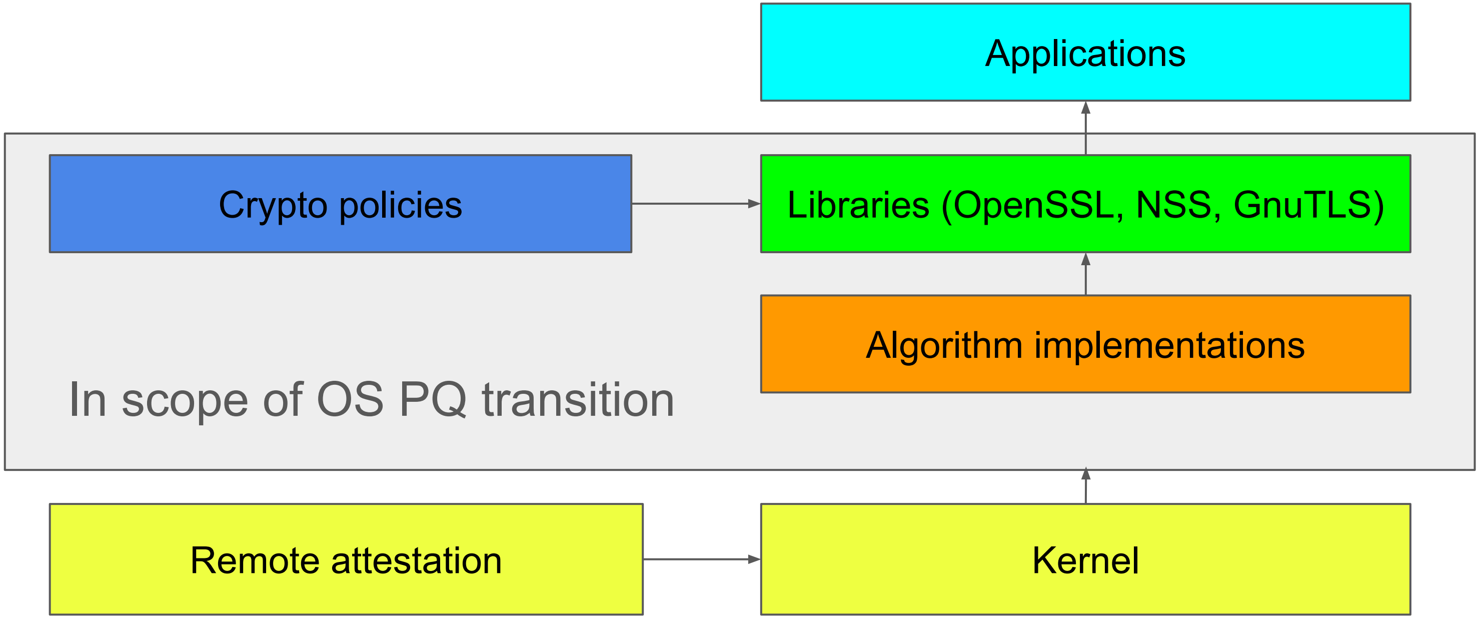
https://github.com/QUBIP/pq-container

Upstream work

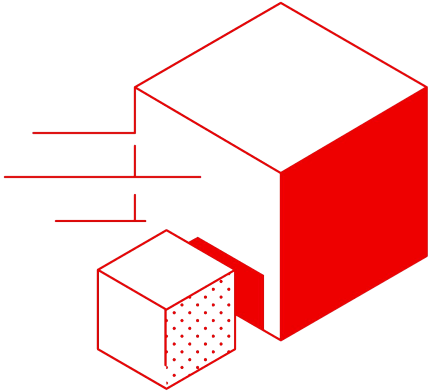
OpenSSL, NSS, GnuTLS

Red Hat

OS PQ transition: scope



Which algorithm to choose



Our algorithm choice

NIST standards

Kyber-based hybrids => ML-KEM based hybrids

Experimental status

We expect incompatibilities

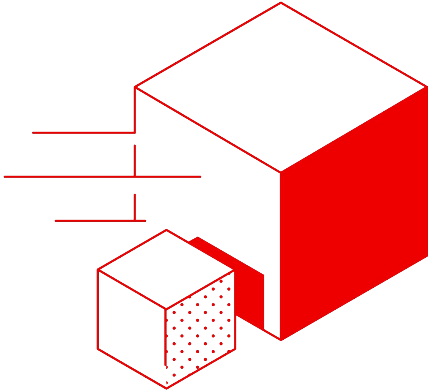
OpenSSH

NTRU algorithm

ML-KEM (9.9+)

Red Hat

What can you do for PQ transition



Networks

Test your systems

Applications

Identify hard-coded limitations

Raise issues upstream

Protocols

Participate in IETF working groups

RPKI?

Useful links

[Post-Quantum Cryptography for Engineers]

[Vision Paper: Do we need to change some things?]

[Research Agenda for a Post-Quantum DNSSEC]

[Field Experiments on Post-Quantum DNSSEC]

Red Hat

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make
Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat

Red Hat