Q-Day survival guide
What the post-quantum cryptography transition means for confidential computing

Arthur Savage
Red Hat - Emerging Tech Security Team

Outline

- What is PQC? Why care?
- Old timelines + current state of industry
- Research developments (quantum computing and NIST candidates)
  - Separate fact from fiction from unknown
- Where is cryptography in confidential computing?
  - Hardware vs software
  - Victories + pain points
- How to migrate?
- Open forum

The world runs on cryptography

- Secure exchange of information
  - **Networking**
- Verifying authenticity or authorship
  - **Digital signatures**
- Confidentiality
  - **Data encryption**
- Trusted Execution Environments
  - **Attestation**
  - **Memory integrity**

What is post-quantum cryptography (PQC)?

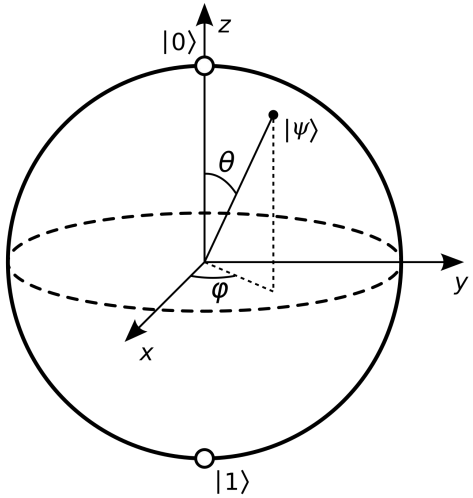
- Classical asymmetric cryptography can be broken by a large quantum computer
- Vulnerable classical cryptography must be replaced with post-quantum equivalents before a large quantum computer exists

The Quantum Threat Model

Quantum Computers: How do they work?!

Quantum computers are **non-deterministic**:

- Perform calculations by leveraging quantum mechanical phenomena (superposition, entanglement, etc.) and sampling.
- Good at calculating many possibilities instantaneously, NOT general-purpose computing
  - Ideal for parameter tuning, complex simulations, **breaking encryption**
- Very hard to build



Shor's Algorithm

A quantum algorithm that finds the prime factors of an integer in polynomial time, which undermines the hardness of factoring/discrete log problems upon which most classical cryptography relies, and is highly parallelizable.

This means:

A sufficiently-sized quantum computer can break classical cryptography (and has already done so on a small scale).

Quantum attacks

Harvest Now; Decrypt Later (HN;DL) - **Happening right now**

Scrape the Internet of communications and data to decrypt when quantum computers become large enough.

Digital Forgery - **Requires quantum computer**

After Q-day, it will become impossible to verify authorship or authenticity of data, which destroys the software supply chain.

- **Critically**: we won't know when this begins.

Types of quantum computers:

1. **Superconducting** - fast gate operations
2. **Photonic** - great variety, no supercooling needed
3. **Neutral atom** - suspended by optical tweezers
4. **Trapped ion** - electromagnetic fields manipulate ions
5. **Quantum dot** - silicon coupled qubits

Good reference: The Quantum Insider

Algorithms and regulations

Vulnerable:

Anything reliant on hardness of prime factoring or discrete log.

- RSA
- Elliptic curve
- Diffie-Hellman
- Ed25519
- ECDSA
- etc…

Secure:

- **Lattice-based cryptography**
- Symmetric cryptography
  - AES
- Cryptographic hashes
  - SHA and SHAKE
  - Merkle trees
  - **Hash-based signatures**

Who decides?

NIST competitions

- Cryptographers developed dozens of proposed algorithms over several rounds of submission
- Winning algorithms become FIPS standards

NIST
National Institute of Standards and Technology

Commercial National Security Algorithm Suite 2.0 (CNSA 2.0)

- Requirements for U.S. National Security Software
- Also general guidance, **matters for open source**.

NIST-selected algorithms

Lattice-based cryptography (gold standard):

- **ML-KEM** - FIPS 203, key encapsulation
- **ML-DSA** - FIPS 204, digital signature

Hash-based signatures:

- SLH-DSA - FIPS 205, stateless but large
- LMS/XMSS - NIST standard (not FIPS), stateful, not for general use but considered safe

Future algorithms:

- FN-DSA - formerly Falcon, controversial digital signature
- HQC - key encapsulation, backup to ML-KEM

CNSA 2.0

Suite of algorithms permitted for U.S. National Security Software

All use cases:

- ML-KEM-1024
- ML-DSA-87
- AES-256
- SHA2 (384 or 512)

Some use cases:

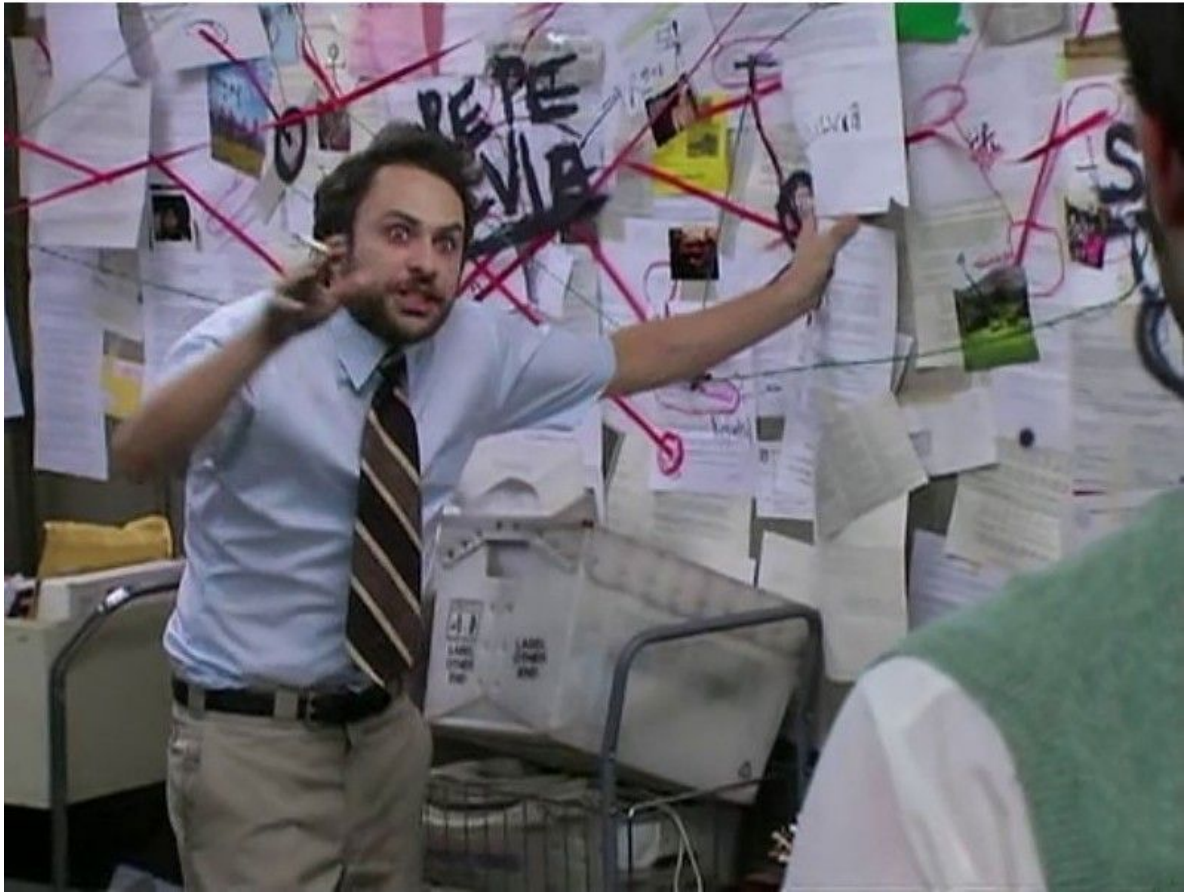
- LMS/XMSS - primarily firmware
- SHA3 (384 or 512) - internal hardware use only

Not allowed by CNSA 2.0

- AES-128
- SHA2-256
- SHA3-256
- Hybrid PQC + classical

This matters for confidential computing!

Let's talk timelines



CNSA 2.0 timelines (released December 2024)

- **January 1st, 2027** - "All new software acquisitions must be CNSA 2.0 compliant"
- **December 31st, 2030** - "All equipment that cannot support CNSA 2.0 must be phased out"
- "**NSA intends <u>all software</u> be quantum-resistant by 2035**"

Industry timelines

Accelerating much faster than government regulations.

- **Let'sEncrypt**: in prod by 2027
- **Google**: 2029
- **Cloudflare**: Full PQC by 2029
- **IBM**: Alongside academia, helped create ML-KEM and ML-DSA
- **Red Hat:** Some PQC in-prod now

Recent advances push big tech closer to the Q-day danger zone

As of last Monday (June 22nd, 2026):

U.S. President Trump signs two executive orders:

- Transition "high-value assets" in federal agencies and contractors
- **PQC KEM by Dec. 31st, 2030 and signatures by Dec. 31st, 2031**
- **Excludes NSS.** CNSA 2.0 still applies.
- Engage foreign governments and industry to use NIST algorithms
- Prioritizing "commercialization and deployment" and "increasing the scale and performance of commercial quantum computers"

References: NBC news article, Cloudflare blog, the PQC executive order and quantum computing executive order

International perspective

G7 cybersecurity working group statement (June 1st, 2026) - recommendations for financial sector, risk-based prioritization with **most-sensitive systems being transitioned by 2030-2032**. Still takes 2035 as final date.

PQC migration timelines for most countries have transition complete **between 2030 and 2035**.

No affiliation with Discovery Communication, LLC

Myth: "We can just make RSA bigger, right?"

Reality: quantum computers break classic cryptography in polynomial time, not exponential.

And the same effort you might put into making RSA bigger could be spent implementing PQC instead.

Myth: "Q-day is always a decade away!"

Reality: quantum computing advancements have accelerated dramatically.

Most estimates now put Q-day at or before 2030 because:

1. Improvements in quantum error correction
2. >6,000 qubit computers
3. More efficient Shor's algorithm

Let's break down the specifics.

Shor's algorithm is possible with as few as 10,000 reconfigurable qubits - Cain et al. (March, 2026)

Still Shor's, but with new speedup techniques.

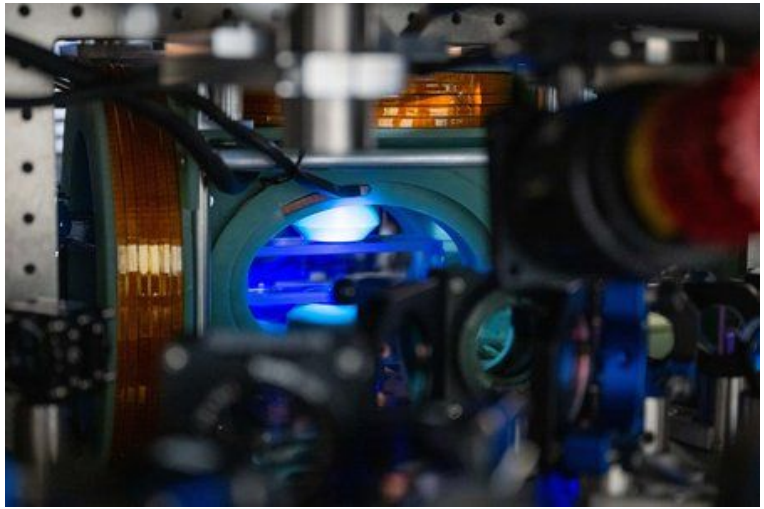
- Implements high-rate **quantum error-correcting codes**, and introduces efficient logical instruction sets and circuit design.

If speedup is successful:

- ECC-256 can be broken with 26,000 qubits in a few days time.
- That's **a lot less** than the original estimates of 20 million qubits.

CalTech team sets new record with 6,100 qubit array (September, 2025)

- Largest stable qubit array ever demonstrated
- Neutral-atom qubits trapped by a grid of lasers
- **Not all operations needed for Shor's algorithm possible yet** with this style of computer.



A new trick brings stability to quantum operations (April, 2026)

- Improving stability of quantum gates in neutral atom quantum computers
- **Not a 17,000-qubit quantum computer**; paper states this technique can be applied to that number of qubits

ETH zürich

How does this alter threat models?

Before:

- Greater emphasis on HN;DL, less on signatures

Now:

- PQC signatures are **more urgent** and **further behind**
- Hardware updates must happen sooner

Myth: PQC is too new to trust!

Reality: lattice cryptography has been around a while.

Earliest lattice cryptography schemes were developed in 1996, with much improvement since ← That's 30 years ago!

ML-KEM is derived from Kyber (2005).

Kyber was first submitted to NIST in 2017.

FIPS-203 and FIPS-204 released in August, 2024.

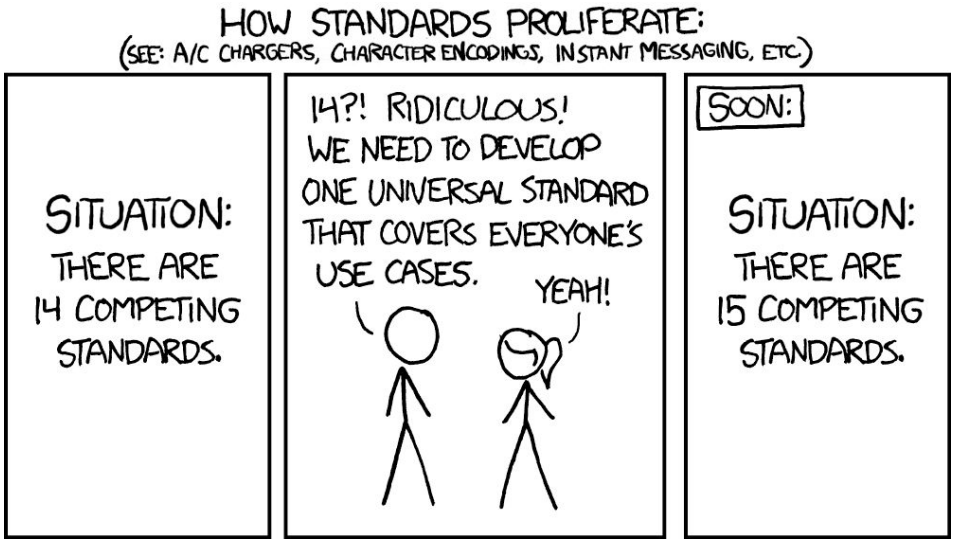
But how much do we trust it?

Hybrid vs. pure PQC

Benefits of hybrid:

- Classical encryption is still safe from classical computers

Detriments:



Source

Respond quickly to vulnerabilities

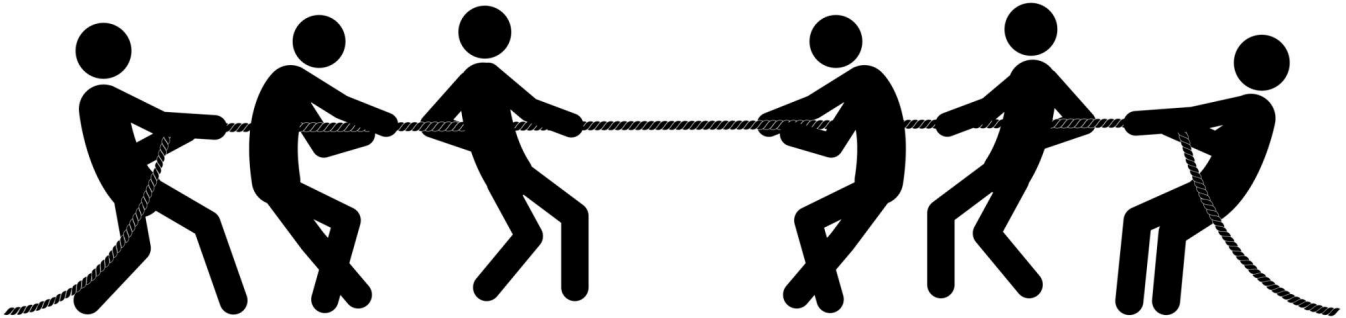
Effectively adapt to changing regulations

Competitive advantage if locked-in crypto breaks

Less potential for misuse

Easier for maintainers

Avoid tech debt, breaking changes, and bloat



CRYPTOAGILITY

LOCKED-IN CRYPTO

Myth: AES & SHA security halves against quantum computers

Reality:

Quantum computers are not a threat to 128-bit symmetric keys - Filippo Valsorda

Grover's algorithm cannot be easily parallelized, so cryptographic consensus (including NIST) is that it is not an imminent threat.

CNSA 2.0 is dissenting opinion.

Hardware: where does PQC matter?

- **Trusted Execution Environments**
  - AMD SEV-SNP, Intel's TDX
- Hardware Security Modules (HSMs)
- Trusted Platform Modules (TPMs)
- Roots of trust (CPU, GPU, Edge)
- Resource-constrained environments

[1] Hardware-Enabled Security: Confidential Computing of Data in Cloud Workloads - NIST IR 8320E ipd, May 2026

Existing progress

TPM 2.0 specification v185 updated to include ML-KEM and ML-DSA

- Reference code still in-progress, then manufacturing.
- Projects downstream of reference code need updating

Entrust HSM

- PQC available

Existing progress - processors

Intel

- Intel has PQC accelerators (QAT), LMS/XMSS in cryptogrpahic primitives libraries, and partnership with Arquit
- **No published transition timelines for PQC** in root-of-trust
  - Keynote yesterday stated all offerings fully PQC by 2029

AMD

- PQC secure boot on FPGA and some vendor solutions
- Website discusses hybrid quantum computing, but **not PQC root-of-trust, and no public transition timeline**.

NVIDIA

- Has SDK for PQC acceleration (OQS)
- **No public PQC timelines for confidential computing.**

This is concerning because:

All current CoCo hardware must be replaced within 2-4 years!

- Anything that cannot be updated in software or firmware
- Endorsement keys and burned-in certs for TPMs
- **Attestation keys for AMD's SEV-SNP and Intel's TDX are ECDSA**

Prepare for:

- Hardware shortages and/or higher costs
- Delays could run past Q-day (if we are unlucky)

Discrepancies between CNSA 2.0 and current hardware

Memory integrity protection is AES-128

- AMD's SEV-SNP
- Intel's TDX

Intel TDX uses SHA3-256 for MAC generation (too short)

These are not vulnerabilities, just not compliant.

Software

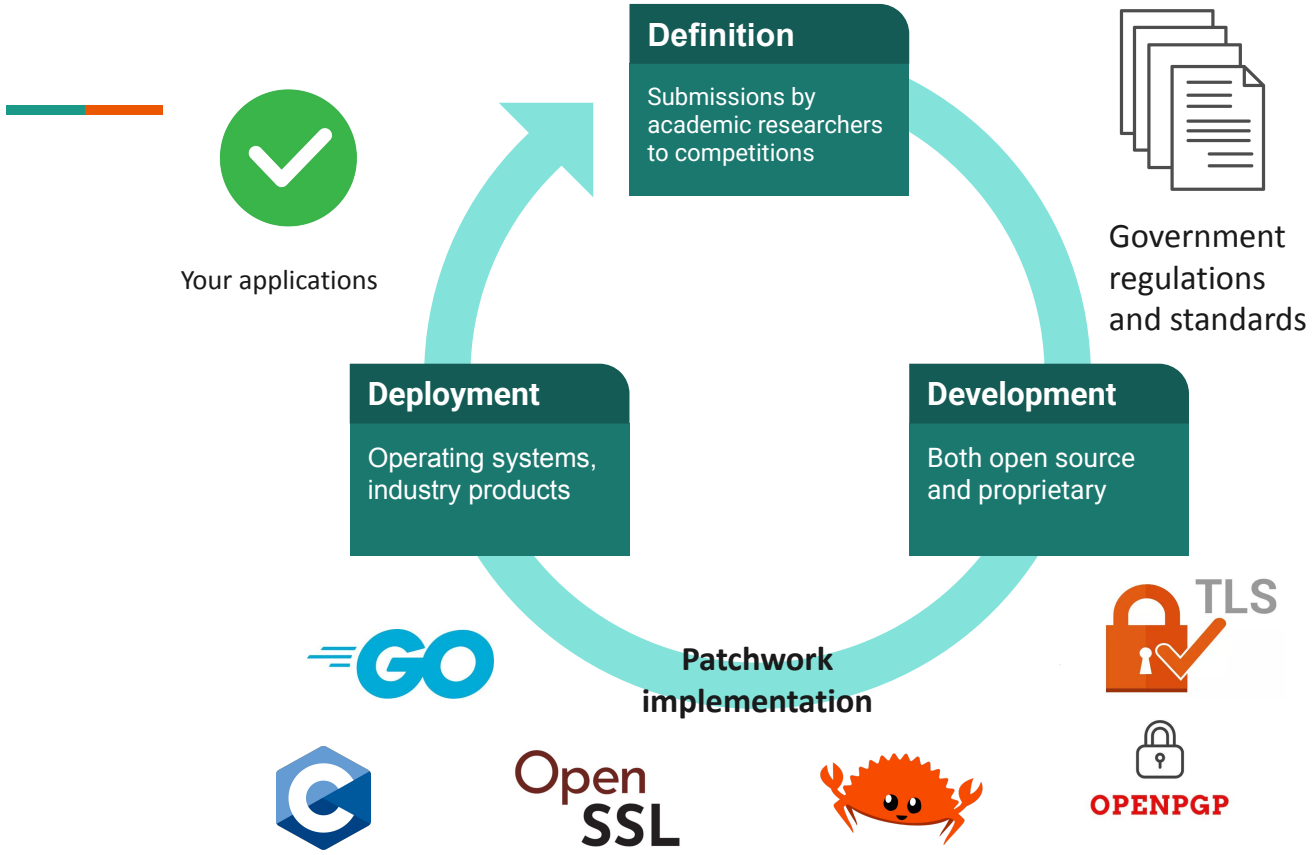
Heatmap

Standard	Overall Range	Pure PQC encrypt	Hybrid PQ encrypt	Pure PQ sig	Hybrid PQ sig
SSH	3 to 7	3	7	3	3
TLS 1.2[1]	0 to 0	0	0	0	0
TLS 1.3[2]	3 to 6	4	6	4	3
X.509[3]	5 to 7	7	5	7	6
S/MIME	5 to 7	7	5	7	6
OpenPGP	2 to 6	2	6	6	6
IKE/IPSec	3 to 6	6	6	5	3
MLS	4 to 4	4	4	4	4
TPM	2 to 7	7	2	7	2
DNSSec	1 to 1	-	-	1	1

Key

0	Consensus Against Inclusion
1	Blocked / Stalled
2	In Progress / Chartered
3	Unofficial Draft(s)
4	Official Draft(s)
5	Progress to Finalization
6	Near-Finalized
7	Finalized / Approved
-	Unknown / NA

Post Quantum Cryptography Coalition April 2026


Definition
Submissions by academic researchers to competitions
Development
Both open source and proprietary
Deployment
Operating systems, industry products
Patchwork implementation
Your applications
Government regulations and standards
TLS
OPENPGP
OpenSSL

Project-specific example #1: Trustee

Key broker and attestation service for confidential containers projects

- Needs hardware upgrade for PQC
- Token signing and key transport are Ed25519 and RSA-OAEP (awaiting standardization for JOSE/JWT)
- TLS transport needs hybrid key exchanges
- Alignment with OCI standards and confidential containers
- Language-specific blockers

This list is not exhaustive

Project-specific example #2: Keylime

TPM-based remote attestation, vTPM applications, some experimental TEE support

- Linux IMA
  - Signature size is an issue
- TPM
  - Hardware availability AND software libraries
- SHA256
  - Problem for CNSA 2.0
- Agent, Client, and Verifier all need updates

And some issues from the last slide as well…

Open forum:

- **What project are you working on?** Where do you foresee the cryptographic migration impacting your team?
- **Concerns about industry direction?** Hybrid vs. Pure, etc.
- **Success stories?**
- **Difficulties you've experienced firsthand?**

Questions for me?

Thank you to all my colleagues at Red Hat, especially at Emerging Tech and the PQC Program;

Thank you to the upstream communities who make all these technologies possible;

Thank you to the event organizers;

And to you, for attending this talk!