

POST-QUANTUM
CRYPTOGRAPHY.

Cryptographic algorithms designed to resist attacks from a **cryptographically relevant quantum computer**. RSA and ECC remain safe today, but harvest-now-decrypt-later capture means the migration has to start before that computer ever exists.

NIST FIPS 203 / 204 / 205

CRQC

HARVEST NOW, DECRYPT LATER

CRYPTO-AGILITY

01 THE MECHANISM



The quantum threat
Shor's algorithm factors large integers and solves discrete logarithms exponentially faster on a sufficiently large quantum computer — breaking RSA, ECC/ECDSA, and Diffie-Hellman entirely, not just weakening them.



Lattice-based hardness
Most NIST-selected algorithms rely on **Module-LWE lattice problems** — believed hard for both classical and quantum computers, with no known efficient quantum algorithm that solves them.



Hybrid deployment
Classical and post-quantum algorithms run **side by side in the same handshake**, so the connection stays secure even if one of the two is later broken.

02 THE FLOW

Key Establishment ML-KEM · EVERY SESSION

1

The server publishes a **lattice-based public key** alongside its classical key share.

2

The client **encapsulates a shared secret** against that public key, producing a ciphertext only the matching private key can open.

3

Both sides combine the **classical and post-quantum secrets** into one session key — the connection is now quantum-resistant.

Digital Signatures ML-DSA · EVERY SIGNED TRANSACTION

1

The signer generates a **lattice-based or hash-based key pair** and publishes only the public key.

2

A message is signed by producing a proof from the private key, with **no known quantum shortcut for forgery**.

3

Verifiers check the signature against the public key using **purely public math** — no shared secret required.

03 WHAT A QUANTUM COMPUTER BREAKS

ALGORITHM	QUANTUM IMPACT
RSA (2048 / 4096-bit)	Fully broken by Shor's algorithm — the private key is recoverable from the public key.
ECC / ECDSA / ECDH	Fully broken by Shor's algorithm — the elliptic-curve discrete log is solved efficiently.
Diffie-Hellman (finite field)	Fully broken by Shor's algorithm — the same discrete-log weakness applies.
AES-256	Weakened, not broken — Grover's algorithm roughly halves the effective key strength.
SHA-256 / SHA-3	Weakened, not broken — Grover's algorithm gives only a quadratic speedup on preimage search.

04 CLASSICAL VS. POST-QUANTUM ALGORITHMS

ALGORITHM	QUANTUM-SAFE	KEY / SIGNATURE SIZE	STANDARDIZATION
RSA-2048	No	Small	Legacy — deprecation path
ECDSA (P-256)	No	Very small	Legacy — deprecation path
ML-KEM (Kyber)	Yes	Larger	NIST FIPS 203 (2024)
ML-DSA (Dilithium)	Yes	Larger	NIST FIPS 204 (2024)
SLH-DSA (SPHINCS+)	Yes	Largest, slowest	NIST FIPS 205 (2024) — most conservative

POST-QUANTUM.

OPERATE · GOVERN · COMMUNICATE

05

MIGRATION & LIFECYCLE

Harvest now, decrypt later is already happening.

Adversaries capture encrypted traffic today to decrypt once a quantum computer exists — long-lived data is at risk now, not later.

Inventory before you migrate.

A cryptographic bill of materials must catalog every algorithm, key size, and library in use before a swap is even possible.

Hybrid mode first.

Pair classical and post-quantum algorithms during the transition so security never regresses below today's baseline.

Crypto-agility is the real deliverable.

Systems must be able to swap algorithms without a full re-architecture as standards continue to evolve.

06

CONTROL MAPPING

NIST FIPS 203 · 204 · 205

Formal federal standards for **ML-KEM, ML-DSA, and SLH-DSA**, finalized in August 2024.

NSA CNSA 2.0

Mandates PQC algorithms for **National Security Systems**, with a phased migration target through 2033.

NIST SP 1800-38

Practical migration guidance for **crypto-agility** and cryptographic bill-of-materials development.

WHITE HOUSE NSM-10

Directs federal agencies to **inventory and prioritize migration** of quantum-vulnerable systems.

07

WHAT PQC DOES NOT SOLVE

Implementation bugs

New algorithms mean new code — and new code means new **side-channel and implementation flaws**, regardless of the underlying math.

Harvested legacy data

Data captured before migration completes **remains exposed forever**, no matter when the switch eventually happens.

Algorithm uncertainty

Lattice assumptions are well-studied but newer than RSA/ECC — a future cryptanalytic break is **not impossible**.

Legacy & IoT devices

Constrained hardware may lack the compute or memory for **larger PQC keys**, delaying migration for years.

08

MYTHS VS. REALITY

MYTH	REALITY
"Quantum computers can break encryption today."	No existing quantum computer has the qubit count or error correction to run Shor's algorithm against real-world keys — but harvest-now-decrypt-later capture is happening right now.
MYTH	REALITY
"AES and SHA are quantum-broken too."	Symmetric crypto is only weakened, not broken — doubling AES key length restores its security margin against Grover's algorithm.
MYTH	REALITY
"We can wait until a quantum computer actually exists."	Migration takes years across an entire supply chain, and data captured today can be decrypted retroactively the moment a quantum computer arrives.
MYTH	REALITY
"One algorithm swap and we're done."	Standards will keep evolving — the real goal is crypto-agility : the ability to swap algorithms without re-architecting systems.
MYTH	REALITY
"This is only a government or military concern."	Any organization protecting long-lived secrets — health records, IP, financial data — faces the same harvest-now-decrypt-later exposure.

09

SPEAK THE MATH — KEY TERMS

CQRC

Cryptographically Relevant Quantum Computer — powerful enough to **break RSA/ECC-scale keys** with Shor's algorithm.

SHOR'S ALGORITHM

A quantum algorithm that efficiently **factors integers and solves discrete logarithms**, breaking RSA, ECC, and DH.

GROVER'S ALGORITHM

A quantum algorithm giving a **quadratic speedup** on brute-force search, halving effective symmetric key strength.

LATTICE PROBLEM

A hard math problem, e.g. **Module-LWE**, believed resistant to both classical and quantum attacks.

HARVEST NOW, DECRYPT LATER

Storing **encrypted data today** to decrypt it once a CQRC becomes available.

CRYPTO-AGILITY

The capacity to **swap cryptographic algorithms** without a system-wide redesign.