The Post-Quantum Apocalypse Is Already Upon Us

Dr. Michael B. Jones

Principal, Self-Issued Consulting

Trends in Digital Identity (TDI)
April 21, 2026, Verona

William Gibson observed:

“The future is already here — it's just not evenly distributed”

That’s an apt description of the impact of quantum computers on cryptography and its use in our identity systems

The Future's Impact on the Present

Quantum computers are predicted to be able to break the cryptographic algorithms used in today's identity systems

- RSA, Elliptic Curve, etc.

We don't know when

This possibility has huge implications now

All software using cryptography has to be updated before Cryptographically Relevant Quantum Computers (CRQCs) are created

Disruptive is an understatement!

Why is action needed now?

"Store now – decrypt later" attacks a reason

- An adversary can store encrypted traffic now and decrypt it once CRQCs exist
- *What are you sending now that you wouldn't want to be revealed later?*
- Thwarting this attack requires using quantum-safe encryption now

Why is action needed now?

Long-lived digital signatures another reason

- Signed code – particularly for IoT devices
- Signed trust anchors – for example, for PKI certificate chains and federations
- Any signatures you want to be able to determine validity of in the future need to use post-quantum-safe algorithms now

Kinds of Post-Quantum-Safe Algorithms

Post-Quantum-Safe Algorithms Come in Two Flavors

- Pure post-quantum algorithms
- Hybrid post-quantum algorithms

Pure post-quantum algorithms designed to resist attacks by CRQC's on their own

- E.g., ML-DSA for signatures
- E.g., ML-KEM for encryption

Hybrid algorithms combine a pure-PQ algorithm and a classical algorithm

- E.g., ML-DSA-44 with ECDSA P-256
- *Combination unbroken unless both broken*

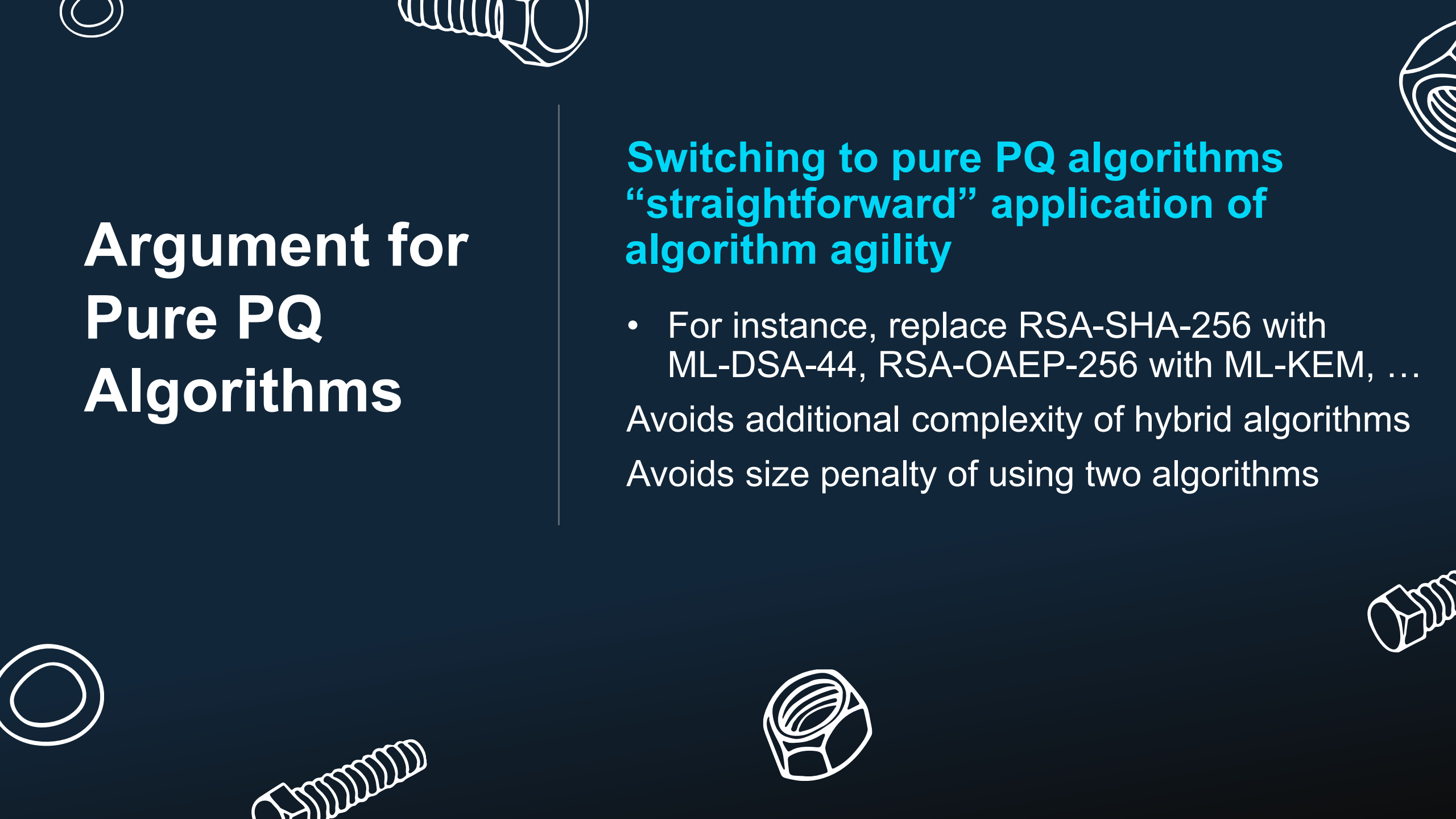
Argument for Pure PQ Algorithms

Switching to pure PQ algorithms "straightforward" application of algorithm agility

- For instance, replace RSA-SHA-256 with ML-DSA-44, RSA-OAEP-256 with ML-KEM, …

Avoids additional complexity of hybrid algorithms

Avoids size penalty of using two algorithms



Argument for Hybrid PQ Algorithms

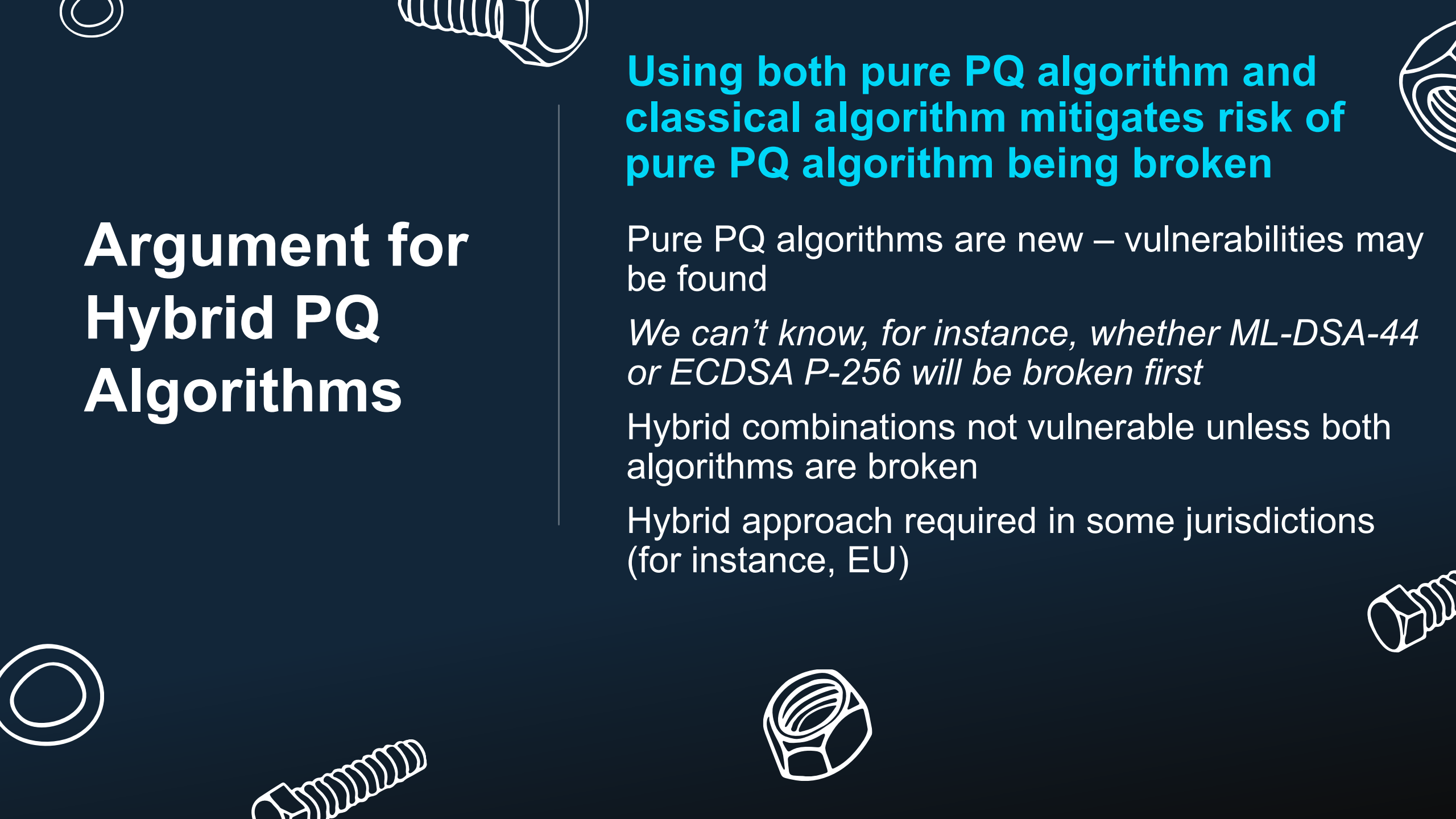
Using both pure PQ algorithm and classical algorithm mitigates risk of pure PQ algorithm being broken

Pure PQ algorithms are new – vulnerabilities may be found

We can't know, for instance, whether ML-DSA-44 or ECDSA P-256 will be broken first

Hybrid combinations not vulnerable unless both algorithms are broken

Hybrid approach required in some jurisdictions (for instance, EU)



Caveats

I am not a cryptographer

I'm therefore not going to critique the proposed PQ algorithms themselves

I will not be commenting on validity or strengths of particular PQ algorithms

I will not be recommending which PQ algorithms to choose in what cases

- E.g., making tradeoffs between ML-DSA and SLH-DSA

I assume you will consult cryptographers and engineering experts to choose appropriate PQ algorithms for your uses

Caveats

I do not have a crystal ball

I'm therefore not going to predict when Cryptographically Relevant Quantum Computers (CRQCs) will be developed

NIST, EU, others have done risk assessments and produced PQ migration recommendations based on them

I assume you will utilize these professional assessments and resulting recommendations and legislation to inform your decision making

Costs to Note

PQ algorithms have different sizes and performance than classical algorithms

ECDSA P-256

- Public key 64 bytes, private key 32 bytes
- Signature 64 bytes

ML-DSA-44

- Public key 1,312 bytes, private key 2,560 bytes
- Signature 2,420 bytes

ML-DSA-87 is faster than ECDSA P-521

Which algorithms to use out of scope for this talk

State of the Standards

National Post-Quantum Standards

Several US NIST standards done

- ML-DSA, SLH-DSA for signatures
- ML-KEM for encryption

Other NIST standards in progress

- FN-DSA for signatures

- See https://csrc.nist.gov/projects/post-quantum-cryptography

State of the Standards

National Post-Quantum Standards

Non-US countries also developing PQ standards

EU Coordinated Implementation Roadmap for the Transition to Post-Quantum Cryptography

- https://digital-strategy.ec.europa.eu/en/library/coordinated-implementation-roadmap-transition-post-quantum-cryptography

China's Next-generation Commercial Cryptographic Algorithms Program (NGCC)

- https://www.niccs.org.cn/

Others, including South Korea, Japan, Russia

- See https://postquantum.com/post-quantum/sovereignty-quantum-pqc/

State of the Standards

IETF Post-Quantum Standardization

PQ standardization happening in many WGs

- TLS, LAMPS, COSE, JOSE, IPSecME, CFRG, HPKE, Plants, PQuip

TLS standardizing both pure and hybrid PQ algs

LAMPS (X.509 WG) is the farthest along

COSE standardized ML-DSA for COSE and JOSE

JOSE standardizing hybrid PQ algs, also for COSE

Still a lot of work in progress

State of the Standards

IETF Decisions on Hybrid Combinations

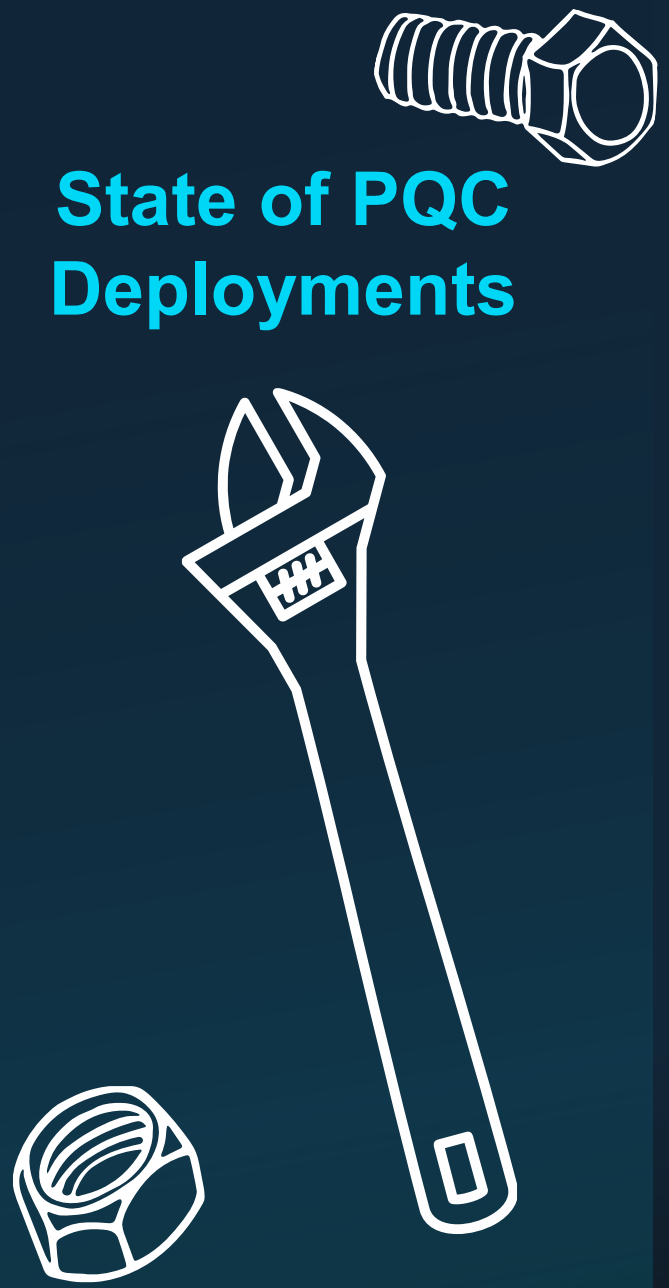
IETF working groups debating which hybrid combinations to standardize

In theory, for instance, could standardize all combinatorial combinations of 3 ECDSA or 2 EdDSA classical algorithms with 3 PQ ML-DSA algorithms

- Result would be 15 hybrid combinations
- Consensus to standardize only a few combinations

Deciding which hybrid combinations make sense a focus of IETF PQ standardization work

State of PQC Deployments



Deployments of Post-Quantum Cryptography

The future is already here — it's just not evenly distributed

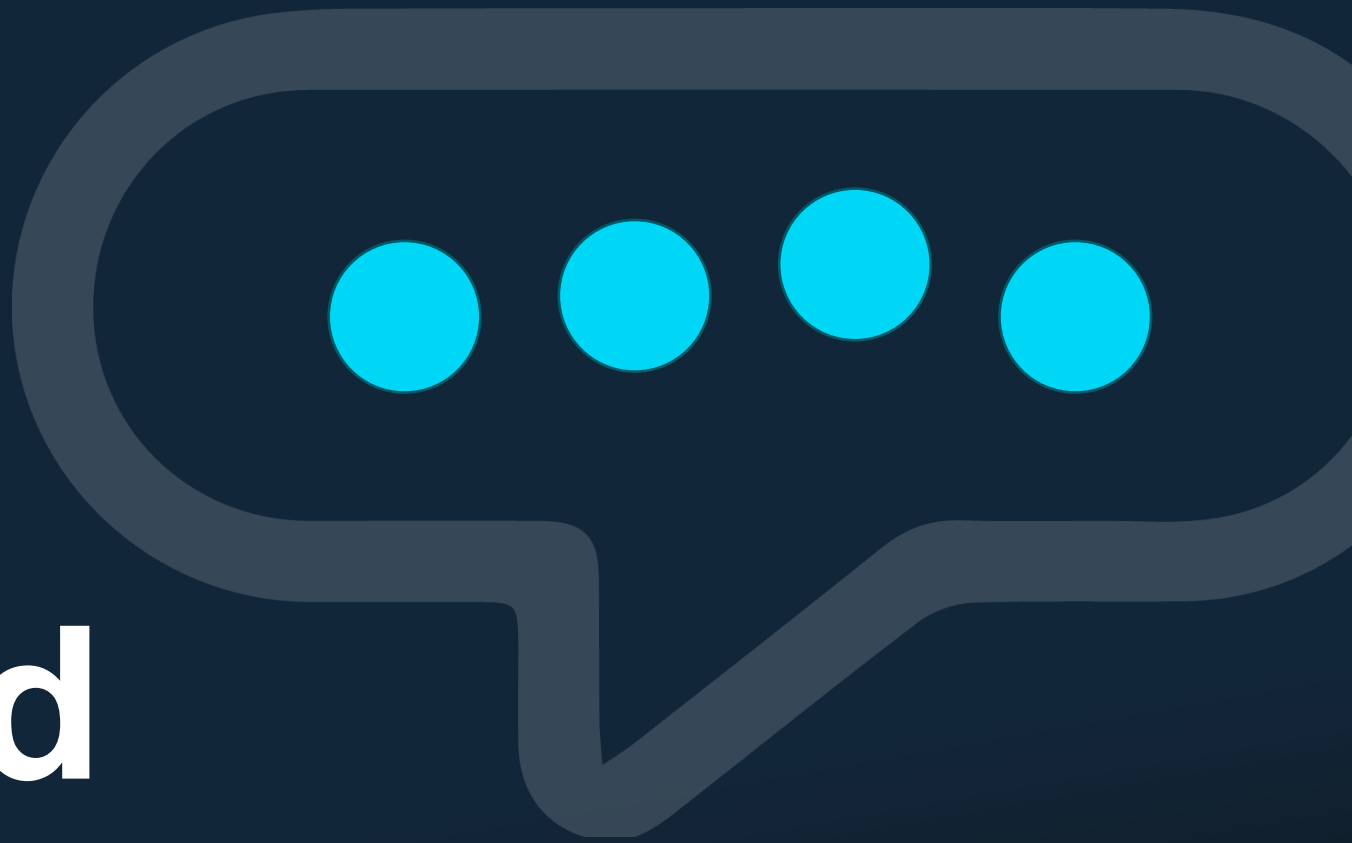
Rollouts gated by standards progress

Some deployments already using completed standards

- At IETF 123 in July 2025, John Mattson reported "*40% of HTTPS client requests use PQC*"
- Yubico has FIDO2 authenticators using ML-DSA

In other cases, the standards needed for deployments are still works in progress

Call to Action and Discussion



Inventory software you're responsible for and make a plan

All software using cryptography has to be updated before Cryptographically Relevant Quantum Computers (CRQCs) are created

Understand your exposure and responsibilities

Your platform vendors will update some of it

- For example, Chrome is already there

Some software and standards are no longer maintained

- Many SAML2 implementations unsupported
- There isn't a PQ plan for XMLDSIG

Determine what you need to update and/or replace in your environment and make a plan

Understand what's truly hard about this

Developing PQ algorithms is hard

- *But that's what cryptographers do*

Creating standards for using PQ algs is hard

- *But that's what standards professionals do*

Updating software to use PQ standards is hard

- *But that's what responsible vendors do*

Deploying the updated software in your environment and replacing that which will not be updated is very hard

- *Doing that is your responsibility*

Acting in the presence of uncertainty can be the hardest thing of all

- *Will you and your organization step up to the challenge and do what's necessary?*

The Post-Quantum Apocalypse is already upon us

It’s time to prepare and act