

Post-Quantum

Cryptography Conference

Why the Internet isn't ready for Post-Quantum Certificates

NIST has finalized the first set of post-quantum algorithms, and post-quantum key agreement has been enabled by default in browsers for over a year. Why are signatures lagging behind? This talk provides the latest updates in a fast-moving ecosystem, a recap of the challenges in migrating to post-quantum certificates, and an overview of ongoing efforts to make post-quantum signatures practical in the WebPKI. In a followup breakout session, we go into detail into some of the more promising proposals for coping with post-quantum certificates.



Luke Valenta
Research Engineer at Cloudflare



KEYFACTOR



January 15 and 16, 2025 - Austin, TX (US) | Online

PKI Consortium Inc. is registered as a 501(c)(6) non-profit entity ("business league") under Utah law (10462204-0140) | pkic.org



PKI
Consortium



Why the Internet *still* isn't ready for post-quantum certificates

Luke Valenta, Cloudflare Research

lvalenta@cloudflare.com

PKI Consortium Post-Quantum Cryptography Conference, January 15, 2025

Cloudflare's role in the WebPKI

We run a **global network** spanning 330+ cities in 120+ countries.

We serve nearly **20% of all websites** (63+ million HTTP requests per second).

We care deeply about a **private, secure** and **fast** Internet, helping design, and adopt, among others:

- Free SSL (2014), TLS 1.3 and QUIC
- DNS-over-HTTPS
- Private Relay / OHTTP
- Encrypted ClientHello

Today's topic:

- Migrating to PQ cryptography



Changing the Internet / WebPKI is hard

- **Very diverse.** Many different users / stakeholders with varying (performance) constraints and update cycles.

We can't assume everyone is on fiber, or uses modern CPU, can store state, or can update at all.

- **Protocol ossification.** Despite being designed to be upgradeable, any flexibility that isn't used in practice is probably broken, because of faulty implementations.

Cryptography in the WebPKI

Key agreement/public-key encryption

RSA/(EC)DHE

Establish a **secure session key** for symmetric encryption

Symmetric-key encryption

AES

Encrypt data to ensure **confidentiality**

Digital signatures/certificates

RSA/(EC)DSA

Verify **authenticity** of messages

Hash functions

SHA

Verify **integrity** of data

Cryptography in the WebPKI

Key agreement/public-key encryption

RSA/(EC)DHE

Establish **secret** key for
symmetric encryption



**Shor's
algorithm:
complete
breakage**

Verify **authenticity** of messages

Symmetric-key encryption

AES

Encrypt data to ensure **confidentiality**

Hash functions

SHA


Verify **integrity** of data

Cryptography in the WebPKI

Key agreement/public-key encryption

RSA/(EC)DHE

Establish *secret key* for
symmetric encryption



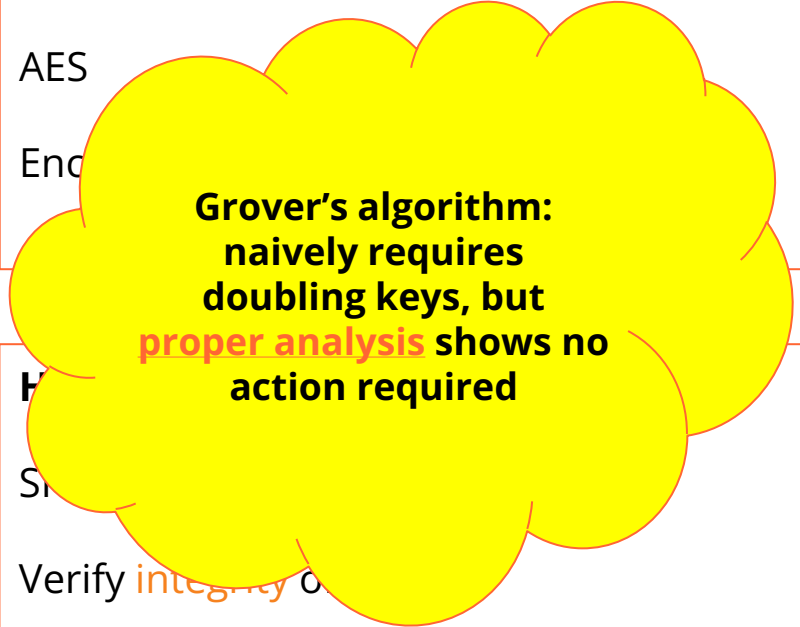
**Shor's
algorithm:
complete
breakage**

Verify *authenticity* of messages

Symmetric-key encryption

AES

Enc



**Grover's algorithm:
naively requires
doubling keys, but
proper analysis shows no
action required**

Dec

St

Verify *integrity* of

This talk

Update¹ on the challenges in migrating the **Internet / WebPKI** to post-quantum certificates.

1. Bas Westerbaan, [The dawn of the post-quantum Internet](#), PKIC PQ Conference, AMS, Nov. 2023

There will be *two* post-quantum migrations.

1. Key agreement 🤝

Communication can be recorded today and decrypted in the future. We need to upgrade **as soon as possible**.

1. Signatures

Less urgent: need to be replaced **before** the arrival of cryptographically-relevant quantum computers.

Key agreement 🤝

Urgent, and the *easier* one.

Key agreement: easier, but with challenges

#1, larger key sizes trigger software bugs (e.g., split ClientHello)

- But we can mitigate with early testing and careful deployment

#2, diverse set of clients and servers on the Internet to upgrade

- But only two parties involved in key agreement, so upgrading a few popular clients and servers results in significant deployment

#3, establishing trust in new algorithms and implementations takes time

- But we can deploy in hybrid mode (e.g., X25519 + ML-KEM768) with minimal overhead

Timeline: PQ key agreement on the Web

2022

NIST announced algorithm selection.

Coordinating at IETF, Cloudflare enabled post-quantum key agreement.

2024

NIST published FIPS 203 (ML-KEM).

Implementations move to final standard.

Browser support (Chrome, Firefox, etc.) ramps up.

TLS library support on the rise 📈.

2019

Feasibility study w/ Chrome

TL;DR

- Lattice-based KEMs perform well.
- Lots of broken connections due to ossification.

2023

Chrome enabled at 1%.

Google enabled support server-side.

Cloudflare added support for internal connections, and for connections to customer origin servers (see [Suleman's talk tomorrow](#) for lessons learned).

2025

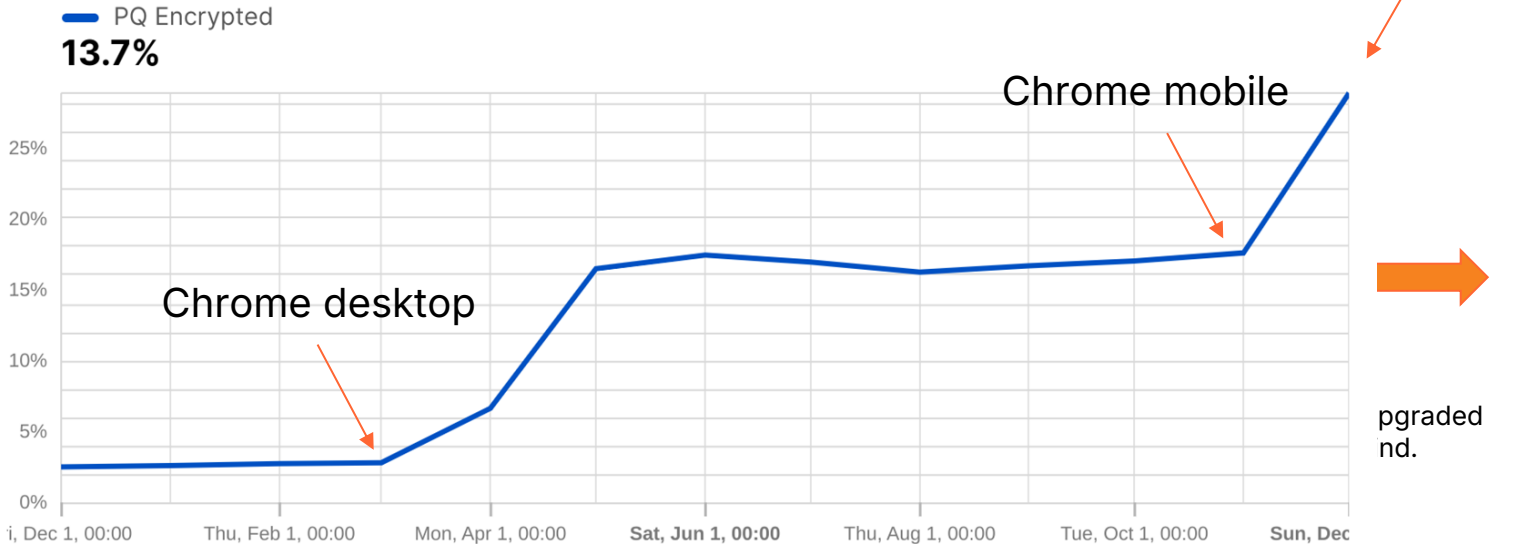
If you haven't upgraded yet, you're behind.



Timeline: PQ key agreement on the Web

Post-Quantum Encryption Adoption Worldwide

Post-Quantum encrypted share of human HTTPS request traffic



2019

Feasibility study

TL;DR

- Lattice-based
- Lots of broken

Cloudflare Radar

Dec 25, 2023, 00:00 UTC → Dec 23, 2024, 16:45 UTC

Key agreement

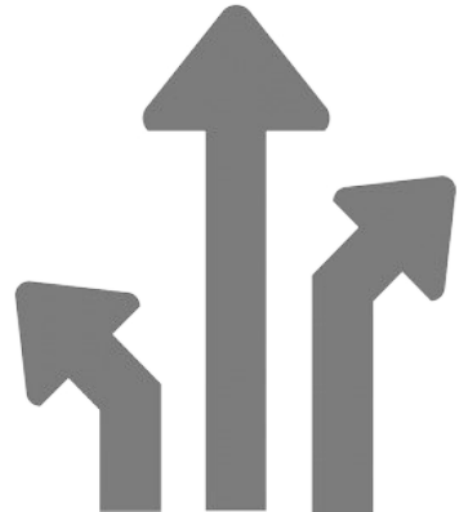
Urgent and the **easier** of the two to deploy. As of January 2025 we see **30+%** client-side deployment. That took over 5 years.

Signatures

Less urgent, but **much more challenging**.

#1, many more parties involved:

Cryptography library developers, browsers, certificate authorities, HSM manufacturers, CT logs, and every server admin that cobbled together a PKI script.



#2, there is **no all-around great** PQ signature

	PQ	Sizes (bytes)		CPU time (lower is better)	
		Public key	Signature	Signing	Verification
Ed25519	✗	32	64	0.15	1.3
RSA ₂₀₄₈	✗	256	256	80	0.4
ML-DSA ₄₄	✓	1,312	2,420	1 (baseline)	1 (baseline)
SLH-DSA _{128s}	✓	32	7,856	14,000	40
SLH-DSA _{128f}	✓	32	17,088	720	110
LMS _{M4_H20_W8}	✓	48	1,112	2.9 ⚠	8.4
Falcon ₅₁₂ (soon FN-DSA)	✓	897	666	3 ⚠	0.7

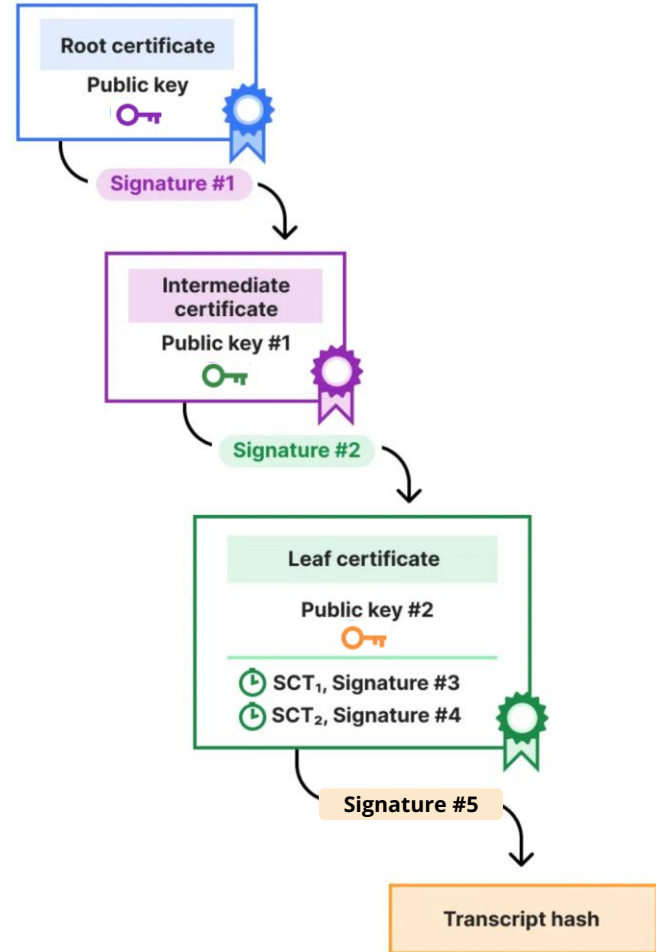
Online signing — Falcon's Achilles' heel

- For fast signing, Falcon requires a **floating-point unit** (FPU).
- We do not have enough experience running cryptography securely (**constant-time**) on the FPU.
- On commodity hardware, **Falcon should not be used when signature creation can be timed**, eg. TLS handshake.
- Not a problem for signature verification.

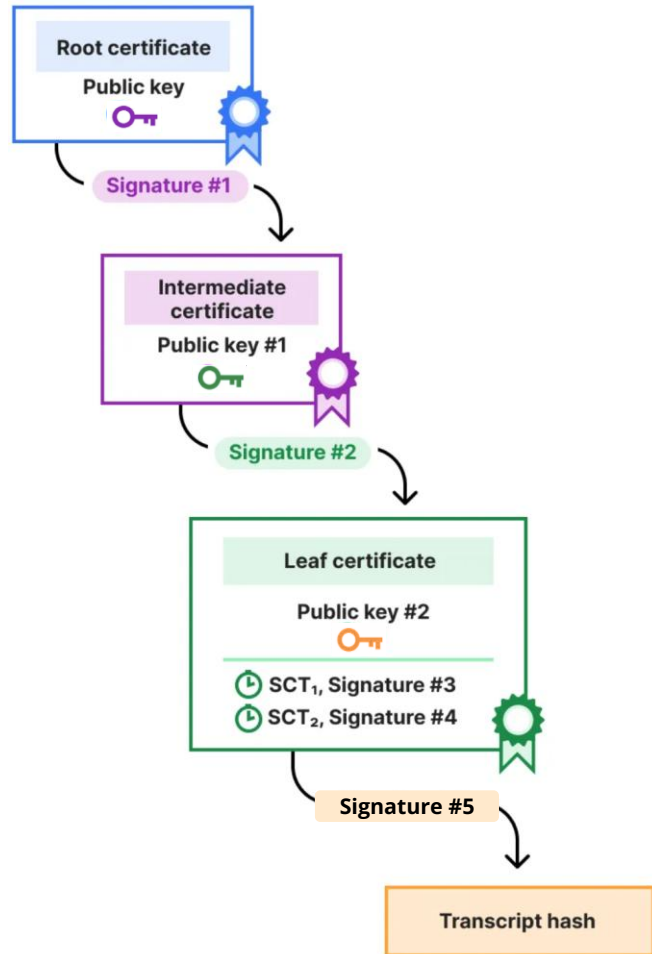


#3, there are **many** signatures on the Web

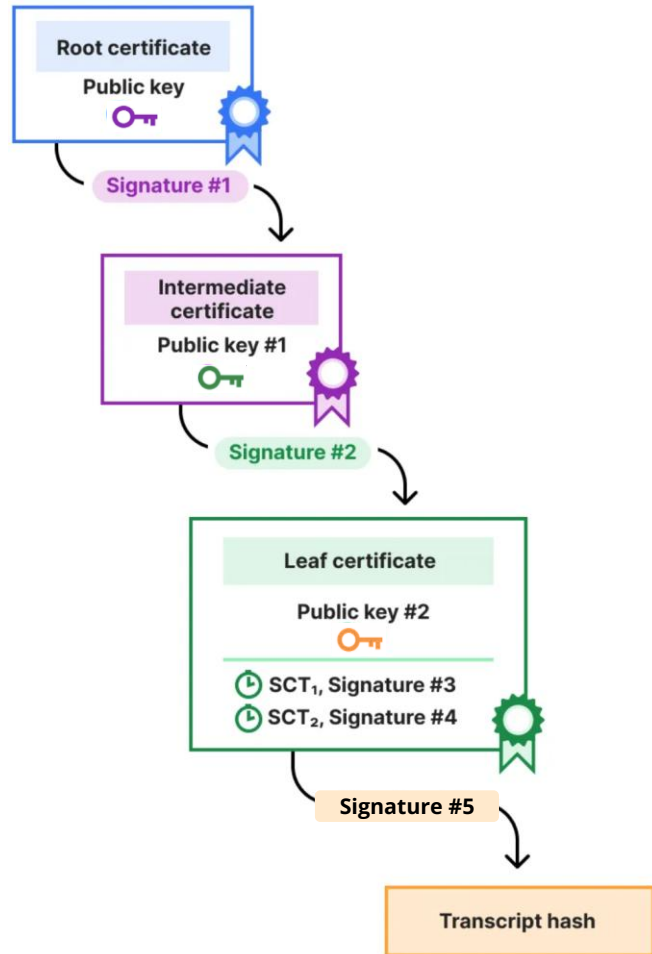
Typically **5 signatures** and **2 public keys** when visiting a **website**.



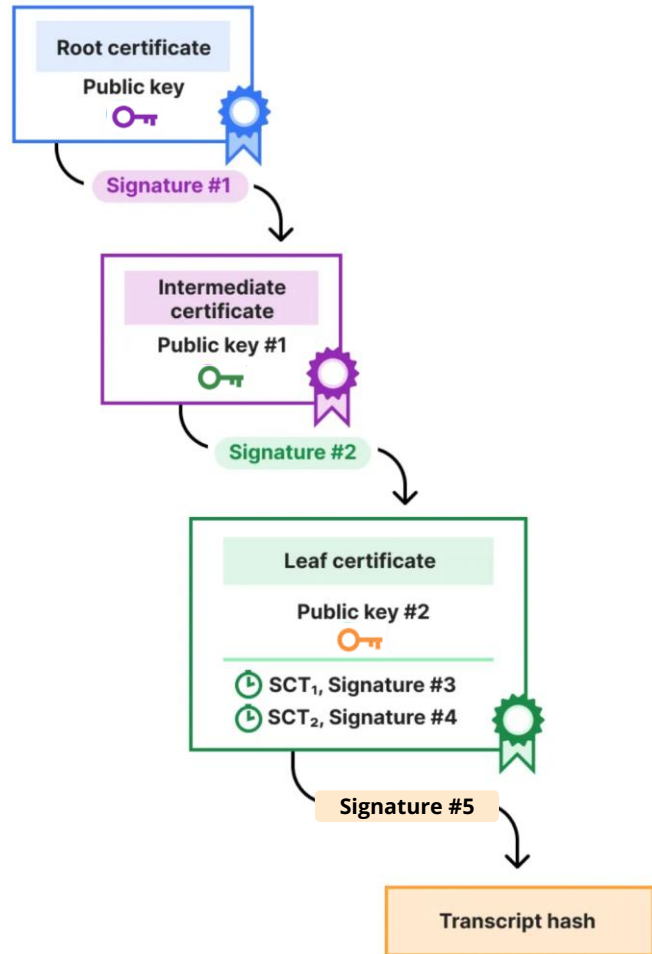
Classical (✘)	Algorithm	Size (bytes)
Signature #1 (root on intermediate)	RSA ₄₀₉₆	512
Public key #1 (intermediate)	RSA ₂₀₄₈	256
Signature #2 (intermediate on leaf)	RSA ₂₀₄₈	256
Public key #2 (leaf)	P-256	32
Signature #5 (leaf on transcript)	P-256	64
Signature #3 (signed certificate timestamp)	P-256	64
Signature #4 (signed certificate timestamp)	P-256	64
Total		1,248



All ML-DSA (✅)	Algorithm	Size (bytes)
Signature #1 (root on intermediate)	ML-DSA ₄₄	2,420
Public key #1 (intermediate)	ML-DSA ₄₄	1,312
Signature #2 (intermediate on leaf)	ML-DSA ₄₄	2,420
Public key #2 (leaf)	ML-DSA ₄₄	1,312
Signature #5 (leaf on transcript)	ML-DSA ₄₄	2,420
Signature #3 (signed certificate timestamp)	ML-DSA ₄₄	2,420
Signature #4 (signed certificate timestamp)	ML-DSA ₄₄	2,420
Total		14,724



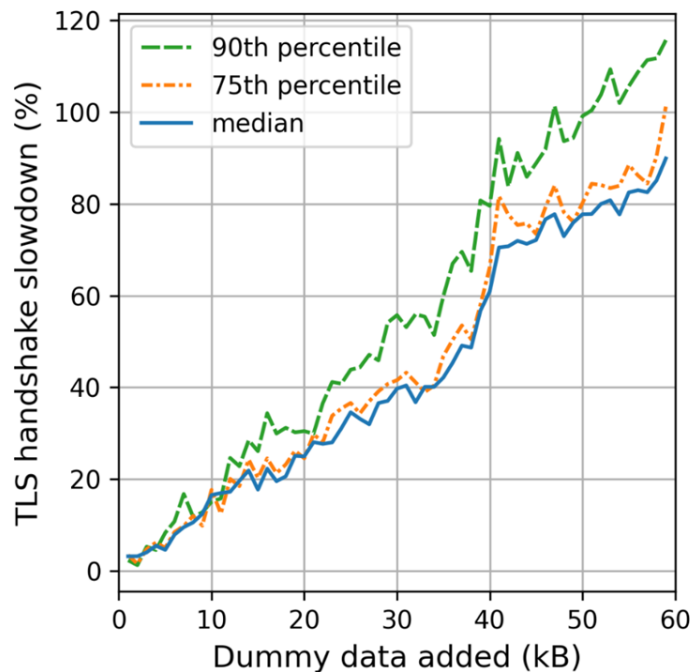
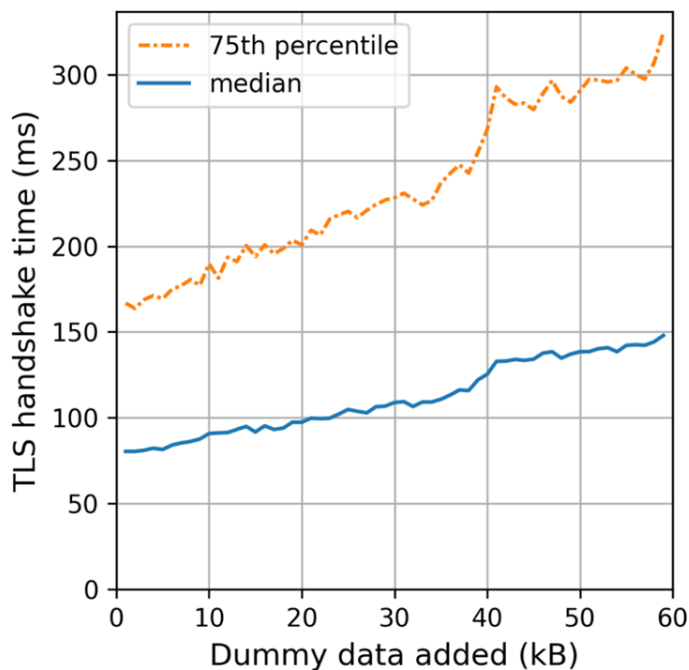
Falcon+ML-DSA (<input checked="" type="checkbox"/>)	Algorithm	Size (bytes)
Signature #1 (root on intermediate)	Falcon ₅₁₂	666
Public key #1 (intermediate)	Falcon ₅₁₂	897
Signature #2 (intermediate on leaf)	Falcon ₅₁₂	666
Public key #2 (leaf)	ML-DSA ₄₄	1,312
Signature #5 (leaf on transcript)	ML-DSA ₄₄	2,420
Signature #3 (signed certificate timestamp)	Falcon ₅₁₂	666
Signature #4 (signed certificate timestamp)	Falcon ₅₁₂	666
Total		7,293



How many (bytes) is **too many**?

How many (bytes) is **too many**?

[Sizing up post-quantum signatures](#), 2021: We found that every 1kB added to the TLS handshake slows it down by about 1.5% at the median.



How many (bytes) is **too many**?

[Sizing up post-quantum signatures](#), 2021: We found that every **1 kB** added to the TLS handshake slows it down by about **1.5%** at the median.

[Chromium Security Design Principles](#), 2024: “Adding **~7kB** is implausible unless a cryptographically relevant quantum computer (CRQC) is **tangibly imminent**.”

How many (bytes) is **too many**?

[Sizing up post-quantum signatures](#), 2021: We found that every **1kB** added to the TLS handshake slows it down by about **1.5%** at the median.

[Chromium Security Design Principles](#), 2024: “Adding **~7kB** is implausible unless a cryptographically relevant quantum computer (CRQC) is **tangibly imminent**.”

[Another look at PQ signatures](#), 2024: Median bytes transferred from server to client for the lifetime of non-resumed QUIC connections to Cloudflare is **4.4kB**.

- Classical signatures and public keys **already** account for about **25%** of all bytes transferred on over half the connections!

Not great, not terrible

It probably won't break the Web, but the performance impact will **delay adoption**.

NIST signature on-ramp

NIST took notice and [has called for new signature schemes](#) to be submitted.

Round 2 candidates were announced in October 2024.

More on these in my [upcoming breakout presentation](#).

The short of it: there are some very promising submissions, but their [security is as of yet unclear](#).

Thus, we cannot assume that a new post-quantum signature will solve our issues.



In the meantime

There are small and larger changes possible to the protocols to **reduce the number of signatures**.

- Leave out intermediate certificates.
- Use key agreement for authentication.
- Overhaul WebPKI, eg. Merkle Tree Certificates.

More on these in my **upcoming breakout presentation**.



Timeline: PQ signatures on the Web

2021

Cloudflare experimented with dummy added data to simulate PQ certs

TL;DR

- 1.5% performance degradation per 1kB added to TLS handshake.
- Ossification still a problem.

2024

NIST published FIPS 204 (ML-DSA), FIPS 205 (SLH-DSA). FIPS 206 (FN-DSA) expected soon.

NIST announced round 2 candidates for on-ramp.

TLS library support on the rise 📈.

Some ML-DSA support in private PKIs.

2026 (?)

Servers will start provisioning PQ certificates, but clients will not use them by default.

2022

NIST announced algorithm selection.

2025 (?)

More agreement on how to do PQ signatures in TLS and the WebPKI.

Some internal deployment from large institutions that want to stay ahead of the curve.



Signatures

Less urgent, but the WebPKI isn't yet ready for broad deployment. Real risk we will start migrating too late.

That's not all: the Internet isn't just TLS

There is much more cryptography out there with their own unique challenges.

- DNSSEC with its harder size constraints
- Research into post-quantum **privacy enhancing techniques**, eg. anonymous credentials, is in the early stages.

Thank you, questions?

References

- Follow along at the [IETF](#)
- Check out our blog, eg.:
 - [2019 TLS experiment](#) with Google
 - [Sizing-up Post-Quantum Signatures](#)
 - [Deploying Kyber worldwide](#)
 - [Another look at PQ signatures](#)
- Reach out: ask-research@cloudflare.com