OpenSSL Conference 2026

Post-Quantum Cryptography

Why moving too fast can be as risky as moving too slowly

Marcel Dasen | Executive VP Engineering | Securosys

© Securosys 2026, Marcel Dasen

securosys

About the speaker

Executive VP Engineering Securosys SA

Swiss manufacturer for cryptographic security appliances (HSM).

Responsibilities:

Head of Software, Hardware and chip design, Product compliance and safety.

Security certifications of products: FIPS 140-3 and Common Criteria,

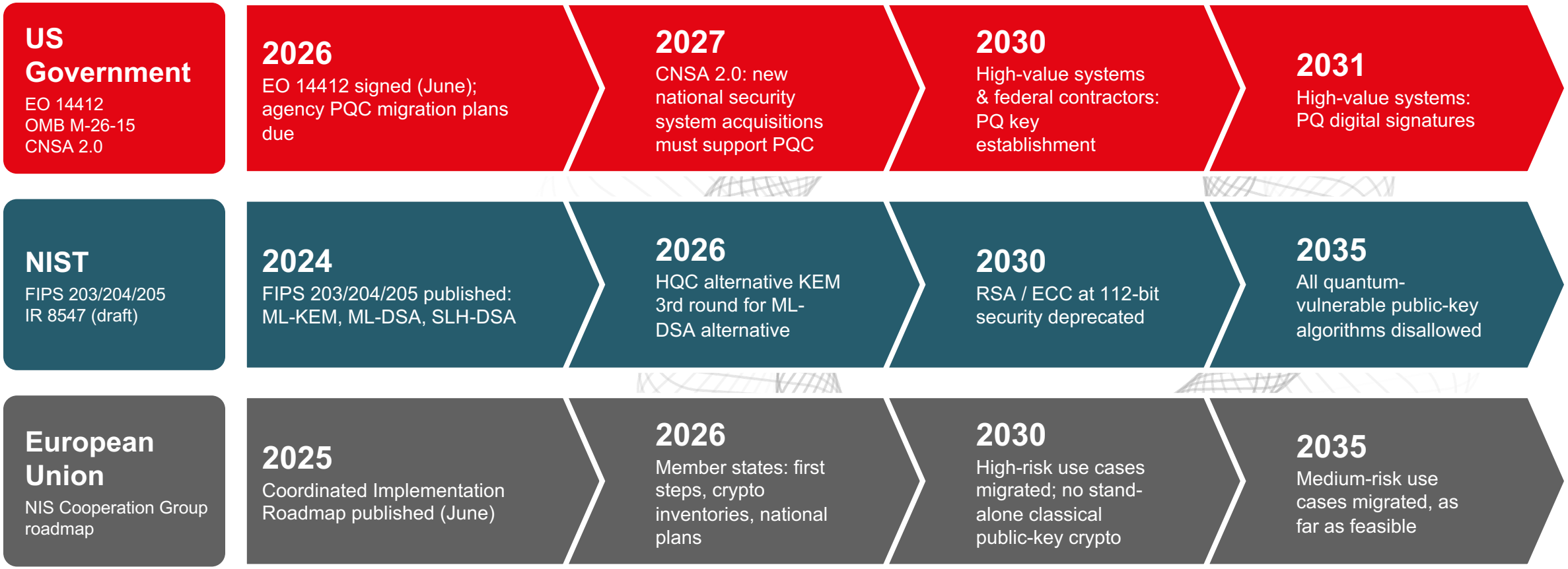
Background:

MS in computer science and engineering from ETH Zürich



© Securosys 2026, Marcel Dasen

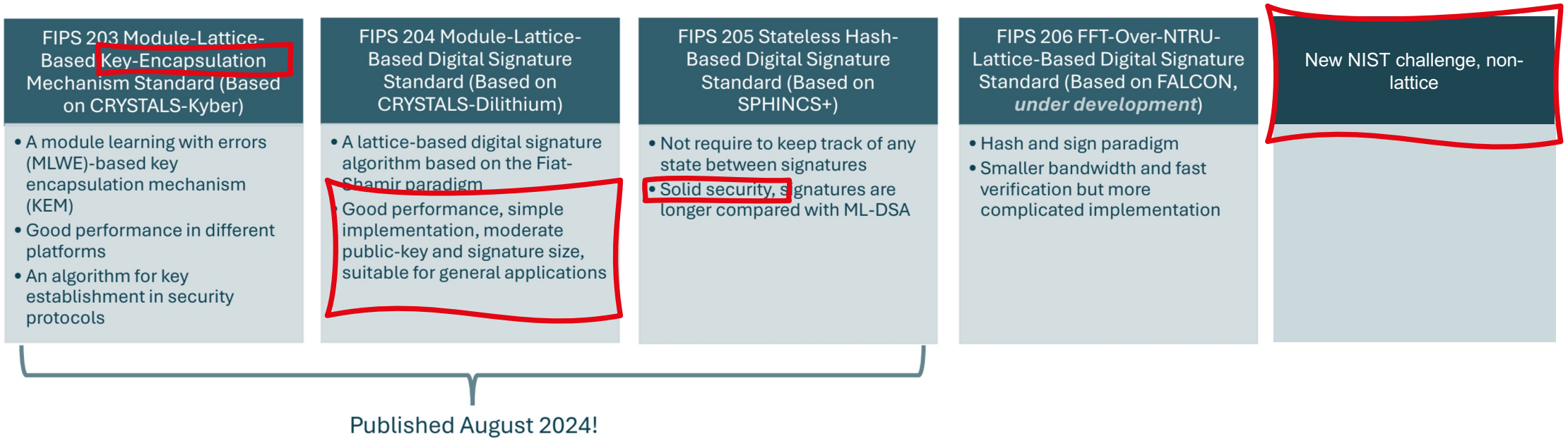
PQC Roadmaps: US, NIST and EU



Sources: White House EO 14412 (22 Jun 2026); OMB M-26-15 (24 Jun 2026); NSA CNSA 2.0 FAQ v2.1; NIST FIPS 203/204/205 (Aug 2024), NIST IR 8547 ipd (Nov 2024); NIS CG Coordinated Implementation Roadmap for PQC (Jun 2025)

© Securosys 2026, Marcel Dasen

Algorithm Standardization



Source: Andy Regenscheid, Manager Cryptographic Technology Group, NIST, PQC conference 2024

© Securosys 2026, Marcel Dasen

PQ Algorithms and Deadlines by Application

APPLICATION AREA	ALGORITHM (CNSA 2.0 / EU / ICAO)[1]	STRENGTH	2024	2026	2028	2030	2032	2034	2036
Firmware signing Secure boot, BMC, device firmware	**LMS / XMSS or ML-DSA-87** LMS/XMSS recommended (state in HW)	Hash / Cat. 5		support & prefer 2025		exclusive 2030			
Software signing Code signing, packages, updates	**ML-DSA-87 or LMS / XMSS** "ML-DSA approved for all signing use cases"	Category 5		support & prefer 2025		exclusive 2030			
TLS certificates Web servers, browsers, cloud	**ML-DSA-87** FIPS 204	Category 5		support & prefer 2025			exclusive 2033		
Key exchange (web) TLS 1.3, cloud services	**ML-KEM-1024** FIPS 203	Category 5		support & prefer 2025			exclusive 2033		
Key exchange (network) VPN, IPsec, routers	**ML-KEM-1024** FIPS 203	Category 5		support & prefer 2026		exclusive 2030			
eID / EUDI wallet (EU) eIDAS 2.0, qualified signatures	**ML-DSA / SLH-DSA** ETSI ESI migrating EN 319 411/412, AdES [2]	EU		EUDI wallets 2026 · migrate		EU: high-risk by 2030			
Passports (ICAO 9303) eMRTD passive authentication	**RSA / ECDSA only** no PQ algorithm in ICAO 9303 yet [3]	Classical		passports issued 2026 stay valid to ~2036 — classical signatures					

CNSA 2.0, all areas: AES-256 · SHA-384/512 · Category 5 parameter sets only

SLH-DSA (FIPS 205): not part of CNSA 2.0, not approved for NSS[1] — stateless hash-based option outside NSS

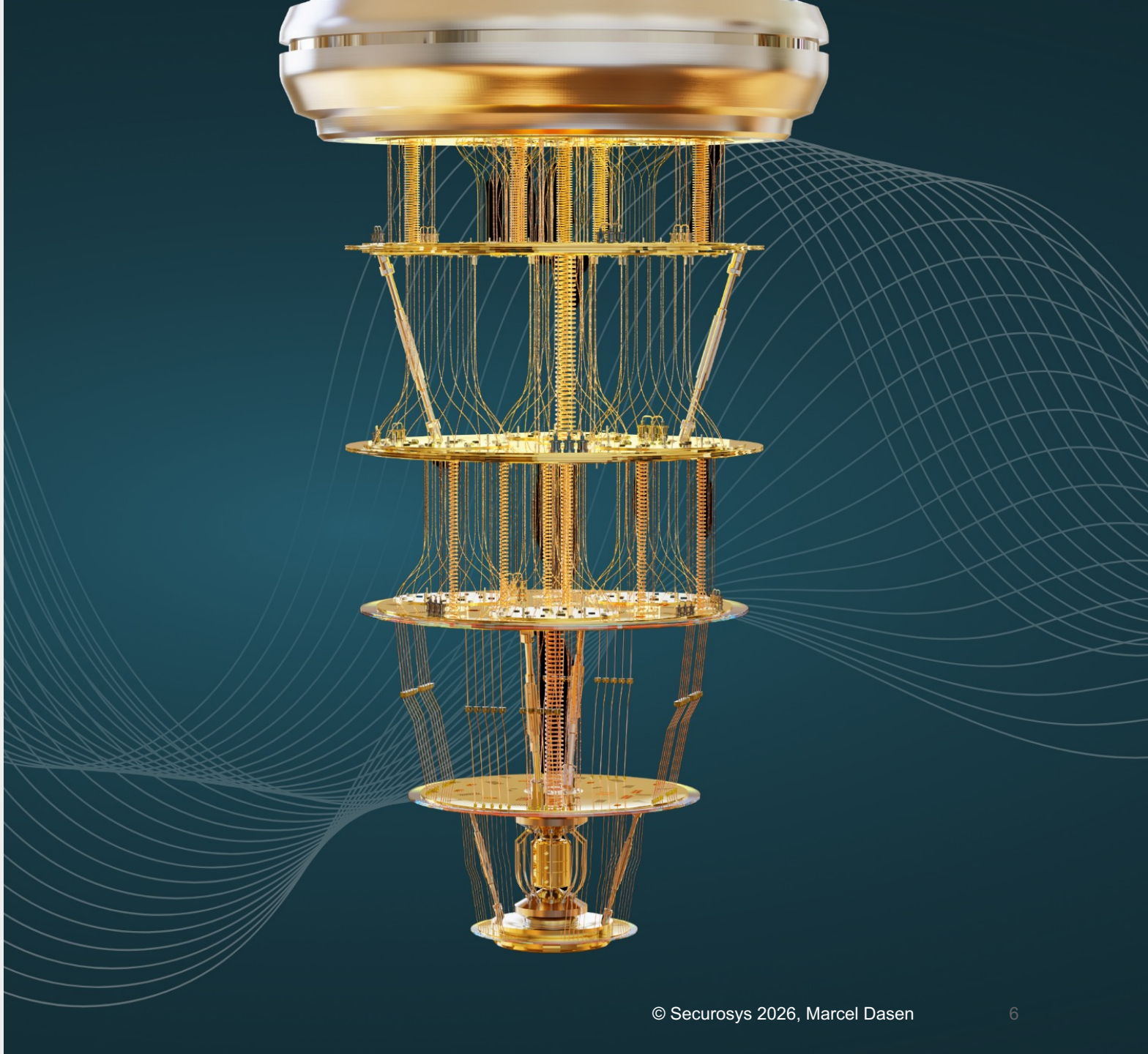
Note: X25519MLKEM768 (today's TLS default) is not CNSA 2.0 compliant

[1] NSA CNSA 2.0 FAQ v2.1 (Dec 2024), p. 6 and p. 9; NSA CSA (Sep 2022). Dashed line: from 1 Jan 2027 new NSS acquisitions must be CNSA 2.0 compliant.
[2] Regulation (EU) 2024/1183 (eIDAS 2.0); EU Coordinated PQC Roadmap (Jun 2025); ETSI ESI quantum-safe migration plan (2025).
[3] ICAO Doc 9303: passive authentication with RSA, DSA, ECDSA only; passports valid ~10 years (S. Lepstra, published by ICAO, 2025).

© Securosys 2026, Marcel Dasen

The Quantum Computer

An Update



© Securosys 2026, Marcel Dasen

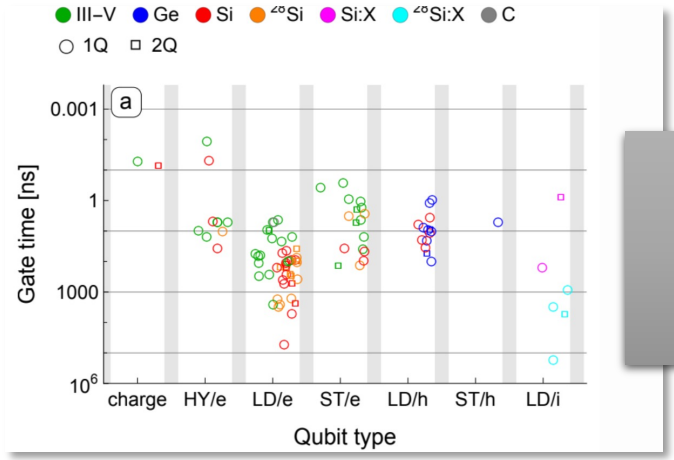
Looking at the Q Computer Roadmap

2023	2025	2026	2027	2029	2033+
Introduce parallelization of quantum computations.	**Extend algorithms on quantum computing + HPC and demonstrate error correction code.**	**Demonstrate first example of scientific quantum advantage and a fault-tolerant module.**	**Diversify quantum advantage and entangle fault-tolerant modules.**	**Deliver the first fault-tolerant quantum computer.**	**Unlock the full power of quantum computing at scale.**
2023 is all about pushing speed in quantum workflows by introducing parallelization in the Qiskit Primitives.	We will release Quantum + HPC tools that will leverage Nighthawk, a new higher-connectivity quantum processor able to execute more complex circuits.	We will demonstrate the first examples of quantum advantage using a quantum computer with HPC.	The scale, quality, speed of the quantum computer will improve to allow executing quantum circuits at a scale of 10K gates on a 1000+ qubits.	The first fault-tolerant quantum computer will be available to clients and allow execution of 100M gates on 200 qubits.	Scale fault-tolerant quantum computers to run circuits of 1 billion gates on up to 2000 qubits, unlocking the full power of quantum computing.

Source: https://www.ibm.com/roadmaps/quantum/

10K gates → 100M gates
x 10'000

2000 qubits
1000m gates



Qgates are not fast (10 - 100ns range)
qbits decoherence O(1 ms)

=> O(10'000) gates maximum

Stano, P., Loss, D. Review of performance metrics of spin qubits in gated semiconducting nanostructures. *Nat Rev Phys* **4**, 672–688 (2022)

© Securosys 2026, Marcel Dasen

Quantum System 2 Nighthawk



https://www.ibm.com/quantum/blog/swiss-innovation-hub

© Securosys 2026, Marcel Dasen

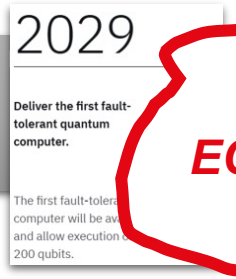
How Many Gates Do We Need

Can a Q computer of sufficient quality be built for SHOR?	Qbits need 2 O(N) Q-Gates need O(N^2)*	Cai, Jin-Yi (2024). "Shor's algorithm does not factor large integers in the presence of noise"
Can a Q computer of sufficient size be built for GROVER?	AES 128, n O(N)	Grassl, M., Langenberg, B., Roetteler, M., Steinwandt, R. (2016). Applying Grover's Algorithm to AES: Quantum Resource Estimates.
Are lattice problems really this hard? Is there really no quantum algorithm break Lattice LWE?		Daniel J. Bernstein et al. S-unit attacks, D.J. Bernstein and T. Lange (2021) "Non-randomness of S-unit lattices" H. Bambury (2022), "S-unit attacks on structured lattice-based cryptosystems"

* Based on QFT, Dewang Sun et al. Optimization and Performance Analysis of Shor's Algorithm in Qiskit

Qgates are not fast (10 - 100ns
qbits decoherence O(1 ms)

=> O(10'000) gates maximum



ECC256: O(256) qbits
O(65K) gates

O(n^12) qates, 3-7K qbits

© Securosys 2026, Marcel Dasen

A Layered Challenge

Base IC level	Component	Infrastructure	Control	Application
TPM	Disk encryption	**Ingress Control / TLS termination**	ssh	Any Win / Azure app, Authenticode
Smartcard, eID, access systems	MACSec EAP-TLS	VPN, IKE	**https / TLS**	Signtools (e.g. code signature)
Secure Enclave	Bootloaders / BIOS	**OpenSSL / TLS**	syslog	**PKI**
CPU crypto accelerators	IEEE 802.1X EAP	CMS / sMIME	SNMP	**Public CAs**
		Oauth, JWT, SAML		DBs
		Code signature / e.g. PGP RPM packages		IAM / Access control
		Crypto libraries		

0-1 years

1-3 years

1-3 years

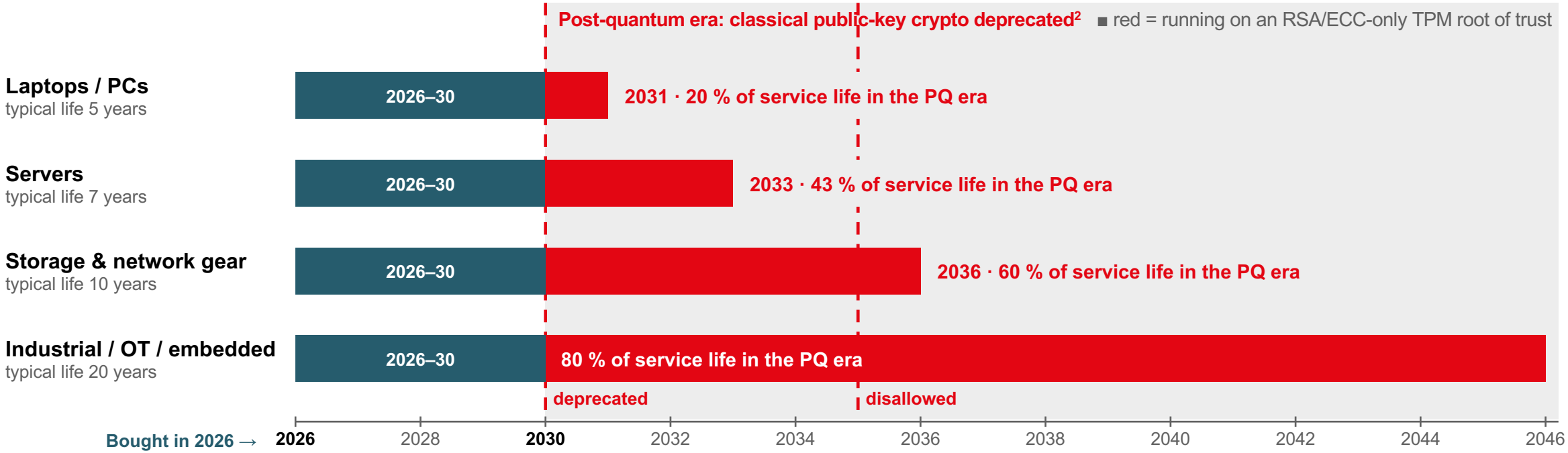
0-2 years

0-2 years

0-1 years

© Securosys 2026, Marcel Dasen

Hardware Bought Today Lives the PQ Era



The gap

PQC-capable TPMs only now specified (TCG TPM 2.0 v185, Mar 2026)[1] — today's platforms ship with RSA/ECC-only TPMs that cannot be upgraded.

Recommendation

Build infrastructure security on roots of trust that already support PQC (e.g. HSMs with ML-DSA / ML-KEM) and reduce dependence on the TPM as the root of trust.

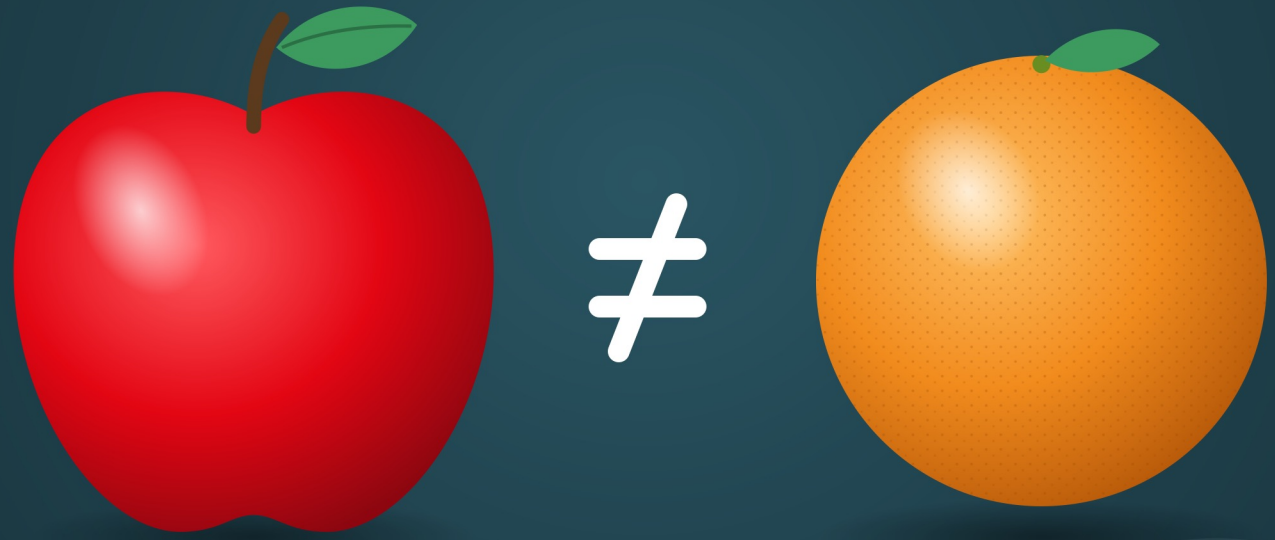
[1] Trusted Computing Group, "New computing specification implements PQC measures…", 22 Mar 2026 (TPM 2.0 Library v185: ML-KEM endorsement keys, ML-DSA attestation keys); first PQC TPM samples announced by SEALSQ, Sep 2025.
[2] NIST IR 8547 ipd (Nov 2024): RSA/ECC deprecated after 2030, disallowed after 2035; EU roadmap: high-risk use cases by 2030. Service lives are typical, indicative values.

© Securosys 2026, Marcel Dasen

PQ algorithms are NOT a 1:1 replacement

An Update

Comparing apples and oranges



RSA · ECC · ECDH

classical, mature —
one key type for everything

ML-KEM · ML-DSA

KEM ≠ signature —
new sizes, APIs, failure modes

PQC Sizes and Performance

Algorithm	Security	Standards	Pub key (bytes)	Private key (bytes)	Signature size	Sig time O(ms)
RSA 3072	128b	FIPS 186-5	384	384	384	10
ECC (384)	192b	FIPS 186-5	96	48	256	.5
ML-DSA	L5	FIPS 204	2,6KB	4,8KB	4.6KB	.5
SLH-DSA (256)	L2-L5	FIPS 205	32- 64	O(n*pk)	50.0KB	100
LMS/XMSS	L2-L5	SP800-208	32-128	O(n*pk)	Log2(n)*32	100
ML-KEM	L5	FIPS 203	2,6KB	4,8KB	4.6KB	.5
FN-DSA	L5	*FIPS 206**	1,8KB	2KB	1.3KB	.5

Performance

Message size

© Securosys 2026, Marcel Dasen

Multi-Recipient Encryption: RSA Key Wrapping vs KEM

System / protocol	Today: RSA key wrapping	Post-quantum path	What must change	Not possible / limits
Data encryption for a group of recipients				
S/MIME, CMS (e-mail, encrypted PDF)	Content key RSA-OAEP-wrapped once per recipient	STANDARD ML-KEM via KEMRecipientInfo[1]	New ML-KEM encryption certificate per user; mail clients & gateways	**Old mail stays RSA-wrapped; KEM key cannot sign**
OpenPGP	Session key RSA-encrypted per recipient	STANDARD ML-KEM + X25519, v6 keys[2]	New keys / subkeys for every user	**Interop gaps: GnuPG follows LibrePGP**
JWE / COSE (tokens, APIs)	RSA-OAEP key management, multi-recipient JSON	DRAFT ML-KEM for JOSE/COSE[3]	Libraries, JWKS key sets	**No final standard yet**
XML Encryption (SAML, WS-Security)	EncryptedKey with RSA-OAEP	NONE no KEM in W3C XML Encryption	Spec, IdPs and service providers	**Not possible today**
EFS, Purview / RMS file protection	File key wrapped per user RSA key + recovery agent	NONE no published PQ mechanism	Vendor products, recovery-agent certs	**Not possible today**
HSM / KMS key transfer, BYOK	Key exported RSA-OAEP + AES-KWP	STANDARD ML-KEM + AES-KW (PKCS#11 v3.2)	HSM firmware, cloud import APIs	**Cloud BYOK APIs largely RSA-only**
Authentication by "proof of decryption"				
Kerberos PKINIT (smart-card logon)	KDC wraps reply key to client's RSA key	DRAFT ML-KEM path, RSA mode deprecated[4]	KDC, clients, smart cards	**No standard yet**
TLS 1.2 RSA key exchange	Server proves its key by decrypting	NONE removed in TLS 1.3	Move to TLS 1.3: ML-KEM + ML-DSA	

KEM ≠ RSA
ML-KEM cannot sign: separate encryption and signing certificates, new proof-of-possession at the CA.

3× per recipient
512 B (RSA-4096) → 1,568 B (ML-KEM-1024) per recipient; 1,000 recipients ≈ 0.5 → 1.6 MB.

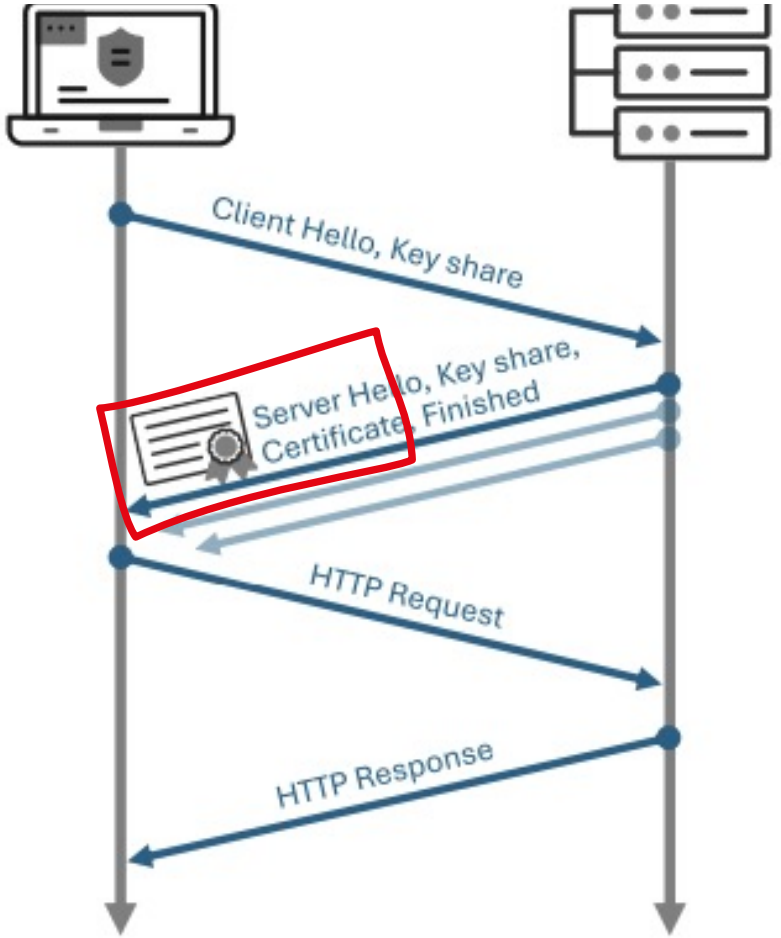
Archives stay exposed
RSA-wrapped data must be re-wrapped while the old private keys still exist.

[1] RFC 9629 (KEM in CMS, 2024); RFC 9936 (ML-KEM in CMS). 2 RFC 9980, "Post-Quantum Cryptography in OpenPGP", Jun 2026. 3 draft-ietf-jose-pqc-kem-05 (Dec 2025). 4 draft-bokovoy-kitten-pkinit-pqc-00 (Jun 2026). Status as of Oct 2026, to our knowledge.

© Securosys 2026, Marcel Dasen

TLS Handshake

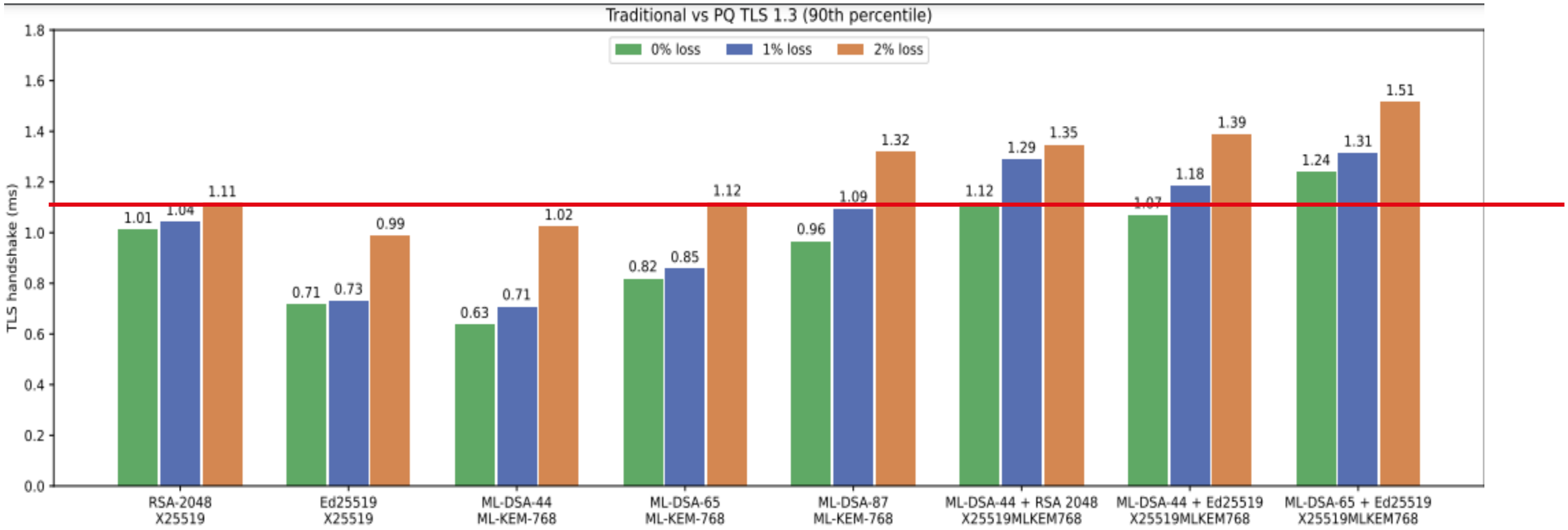
Component	Cryptographic Elements	ML-DSA-87 (Bytes)	ML-DSA-44 (Bytes)	ECC256 (Bytes)*
Server Certificate	1 Public Key	2,592	1,312	64
	1 Certificate Signature	4,627	2,420	71
	2 SCT Signatures	9,254	4,840	142
Intermediate CA Cert	1 Public Key	2,592	1,312	64
	1 Certificate Signature	4,627	2,420	71
TLS Handshake	1 Handshake Signature	4,627	2,420	71
Total Transmission	2 Public Keys, 5 Signatures	28,319	14,724	483
		2 RTT	1 RTT	0 RTT



© Securosys 2026, Marcel Dasen

OpenSSL: Larger packets = higher loss probability

TLS handshake time

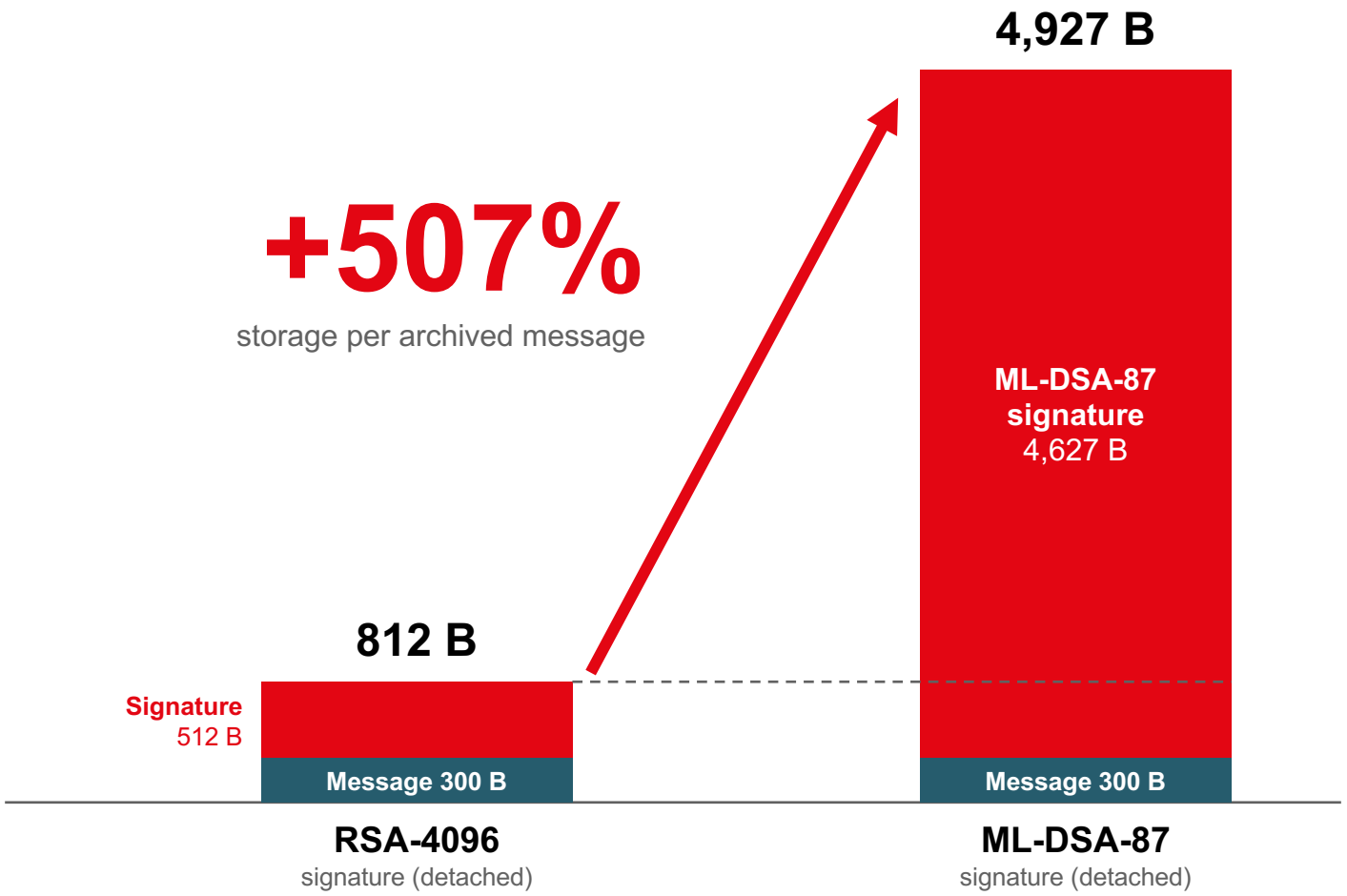


Source: Project Quantum-safe Hardware Security Module, HSLU and Securosys SA, supported by Innosuisse

HSLU Hochschule Luzern

© Securosys 2026, Marcel Dasen

Example Archiving Signed ISO 20022 Messages



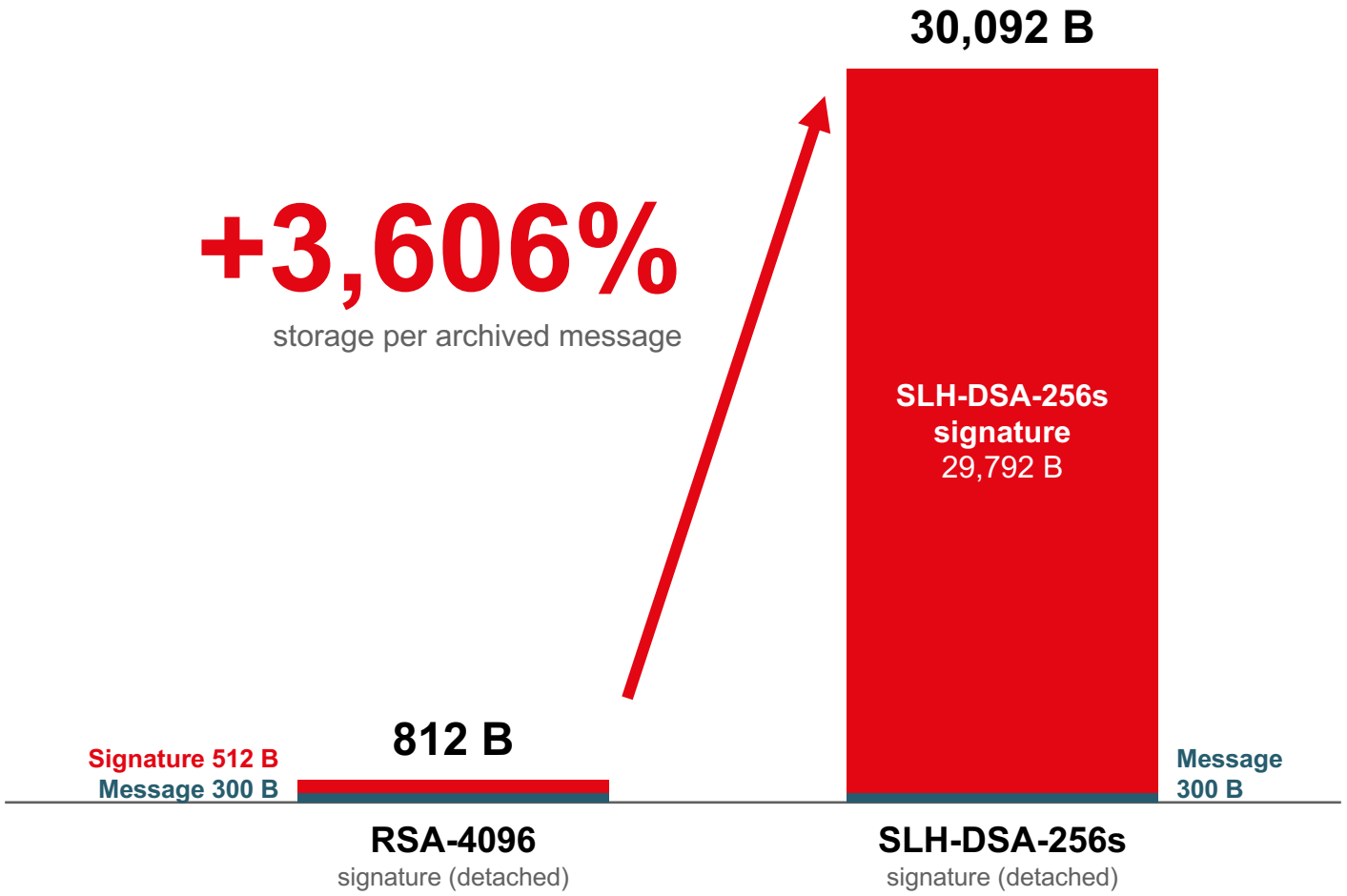
What it means for the archive

- **Signature vs. message:** 1.7× with RSA-4096, 15.4× with ML-DSA-87
- **Per 100 million messages:** 81 GB → 493 GB
- **Incl. public key for verification:** 1,324 B → 7,519 B (+468%)
- **The signature, not the payment, now dominates archive size and retention cost**

Assumptions: ISO 20022 message 300 B; raw signatures per FIPS 186-5 / FIPS 204 (RSA-4096: 512 B, ML-DSA-87: 4,627 B); no CMS/XAdES container overhead.

© Securosys 2026, Marcel Dasen

Using the safe bet SLH-DSA ….



What it means for the archive

- **Signature vs. message:** 1.7× with RSA-4096, 99× with SLH-DSA
- **Per 100 million messages:** 81 GB → 3.0 TB
- **vs. ML-DSA-87:** 6.1× larger; fast variant 256f: 49,856 B (+6,077%)
- **Most conservative assumption (hash functions only), at the highest storage cost**

Assumptions: ISO 20022 message 300 B; raw signatures per FIPS 186-5 / FIPS 205 (RSA-4096: 512 B, SLH-DSA-SHA2-256s, NIST category 5 like ML-DSA-87: 29,792 B); no CMS/XAdES container overhead.

© Securosys 2026, Marcel Dasen

Is Your Infrastructure Ready?

STORAGE & DATABASES

Can your storage system or database cope with such a size increase?

+507%
ML-DSA-87

+3,606%
SLH-DSA-256s

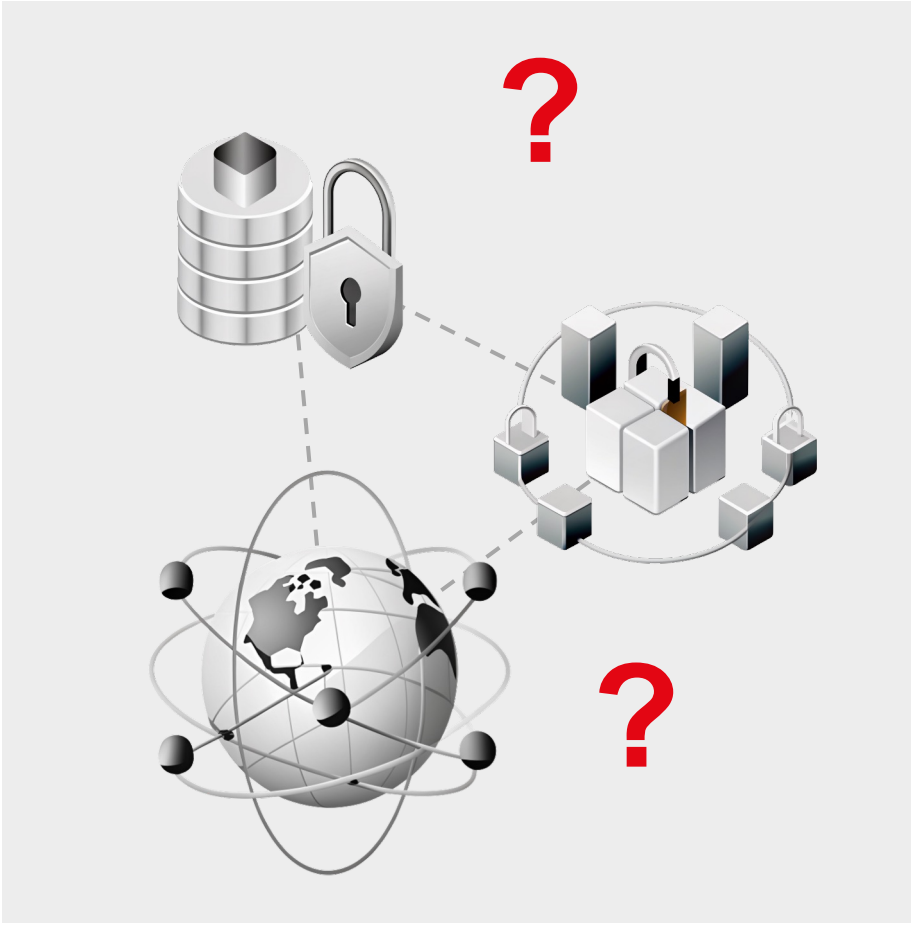
more storage per signed record, for the full retention period

NETWORKS

Can your networks cope with the additional traffic?

483 B ➡ **+5700 %**
ECC-256 → ML-DSA-87

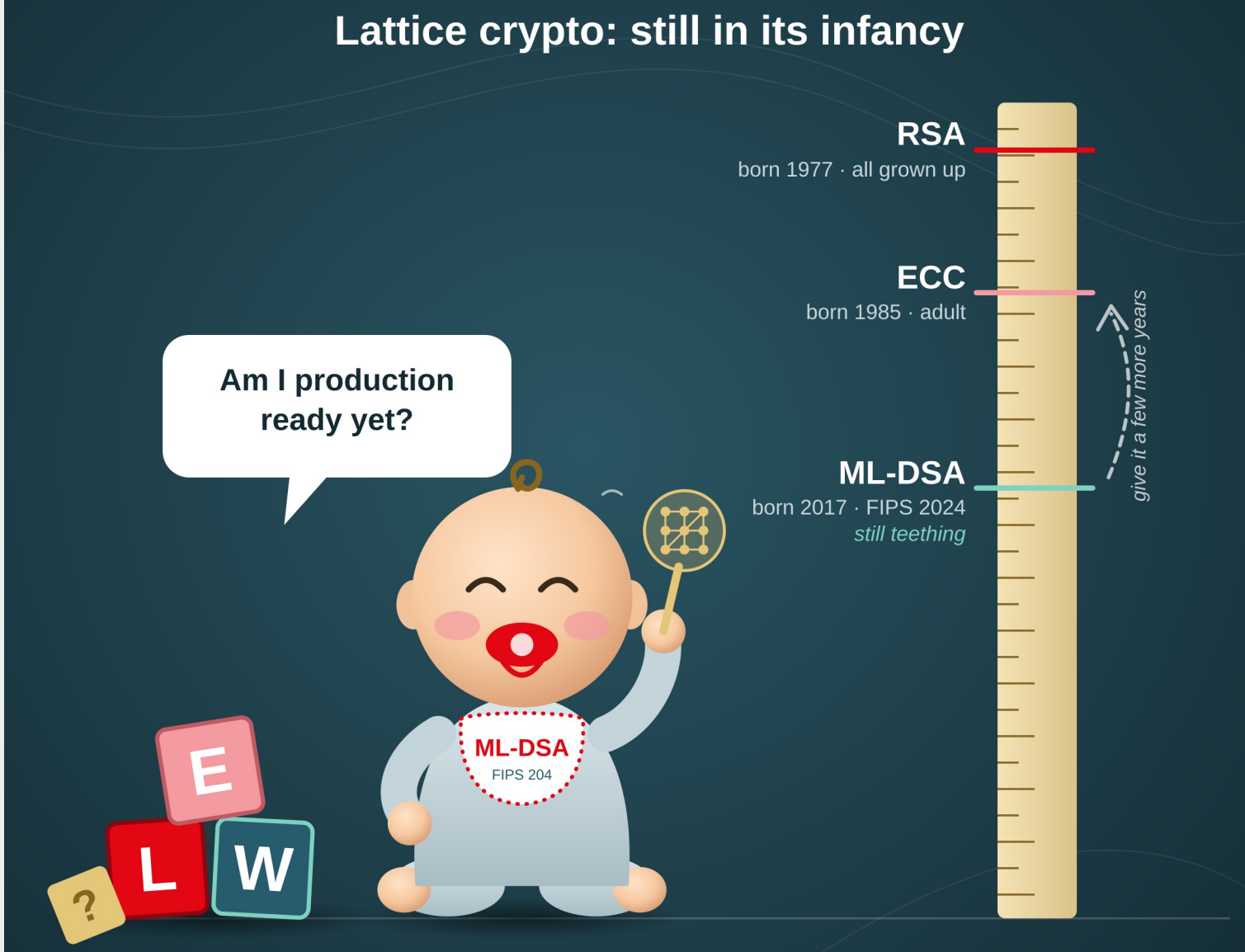
certificate chain + handshake signatures per TLS connection



© Securosys 2026, Marcel Dasen

New math, new risks

Maturity of lattice-based crypto

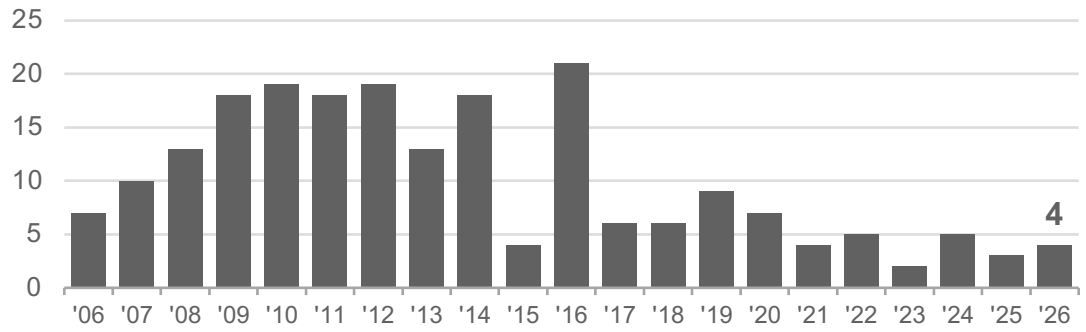


© Securosys 2026, Marcel Dasen

Lattice Crypto is not yet mature

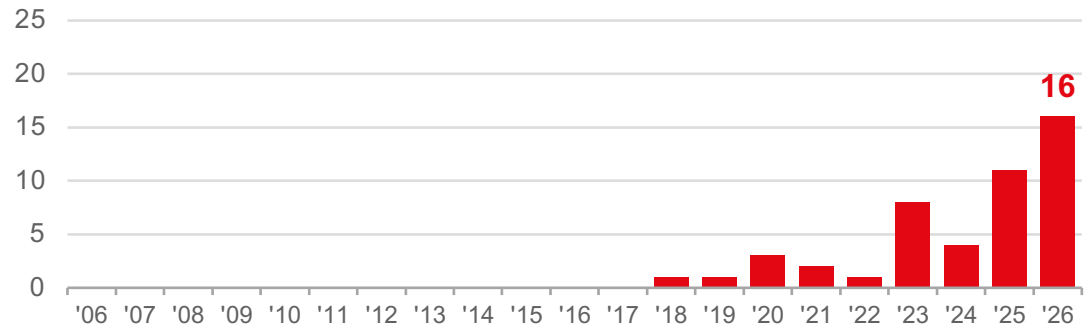
Published side-channel, fault and timing-attack papers per year, from a public scholarly database[1]

RSA since 1977 — **≈ 4 / year**



down from ~19 / year in 2009–2012: attacks understood, countermeasures standard

ML-DSA / Dilithium since 2017 — **16 in 2026**



Jan–Oct alone, ×16 since 2018 — and still accelerating

SIDE CHANNEL

RSA: 211 papers in 20 years, now ~4 / year: settled
ML-DSA: **47 since 2018, 27 in 2025–26 alone[1]**

MATHEMATICS

RSA: No new real shortcut against RSA in 20 years
ML-DSA: **Lattice attack advances 2020, Oct 2025, Feb 2026[2]**

IMPLEMENTATIONS

RSA: Bugs well understood, fixes standard
ML-DSA: **Severe bugs still found in 2026, even in a "verified" library[2]**

[1] OpenAlex (openalex.org), title search, publication years 2006–2026, retrieved 9 Oct 2026; RSA: title contains "RSA" AND (side-channel OR fault OR timing OR leakage), 211 works; ML-DSA: title contains "Dilithium" OR "ML-DSA" AND the same terms, 47 works. Keyword match incl. countermeasure papers and some duplicates; 2026 is a partial year.
[2] D. J. Bernstein, "Exploiting ML-DSA bugs", June 2026, §1.2 (2018 Dilithium 1.0 bug) and §10.2 (Karenin et al., Asiacrypt 2025; Ogilvie, ePrint 2026/279); N. Kobeissi, "Verification theatre", ePrint 2026/192.

© Securosys 2026, Marcel Dasen

SPA & DPA: Lattice Crypto Leaks

WITH SECUROSYS SA · PQCrypto 2020

Defeating NewHope with a Single Trace[1]

"The complete shared secret can be extracted from data of one single trace only."

- Leak in message encoding (poly_frommsg), the same pattern used in lattice KEMs such as Kyber/ML-KEM
- Cortex-M4, NIST reference C code: secret visible on the oscilloscope (-O0); ≥ 96% single-trace success (-O3)
- Masking alone insufficient → combine with shuffling

SPA single trace

Single-trace attacks on **masked** lattice encryption via NTT leakage[2]

Masking does not stop leakage from one execution

DPA many traces

Generic chosen-ciphertext side-channel attacks on CCA-secure lattice KEMs (Kyber, Saber, …)[3]

Timing software

KyberSlash: secret-dependent division timings recover Kyber keys[4]

Found in reference and widely deployed implementations

> **… each new ML-DSA library will have about a 1/4 chance of a severe vulnerability. Presumably the side-channel risks are also higher …**
>
> Daniel J. Bernstein, "Exploiting ML-DSA bugs", June 2026[5]

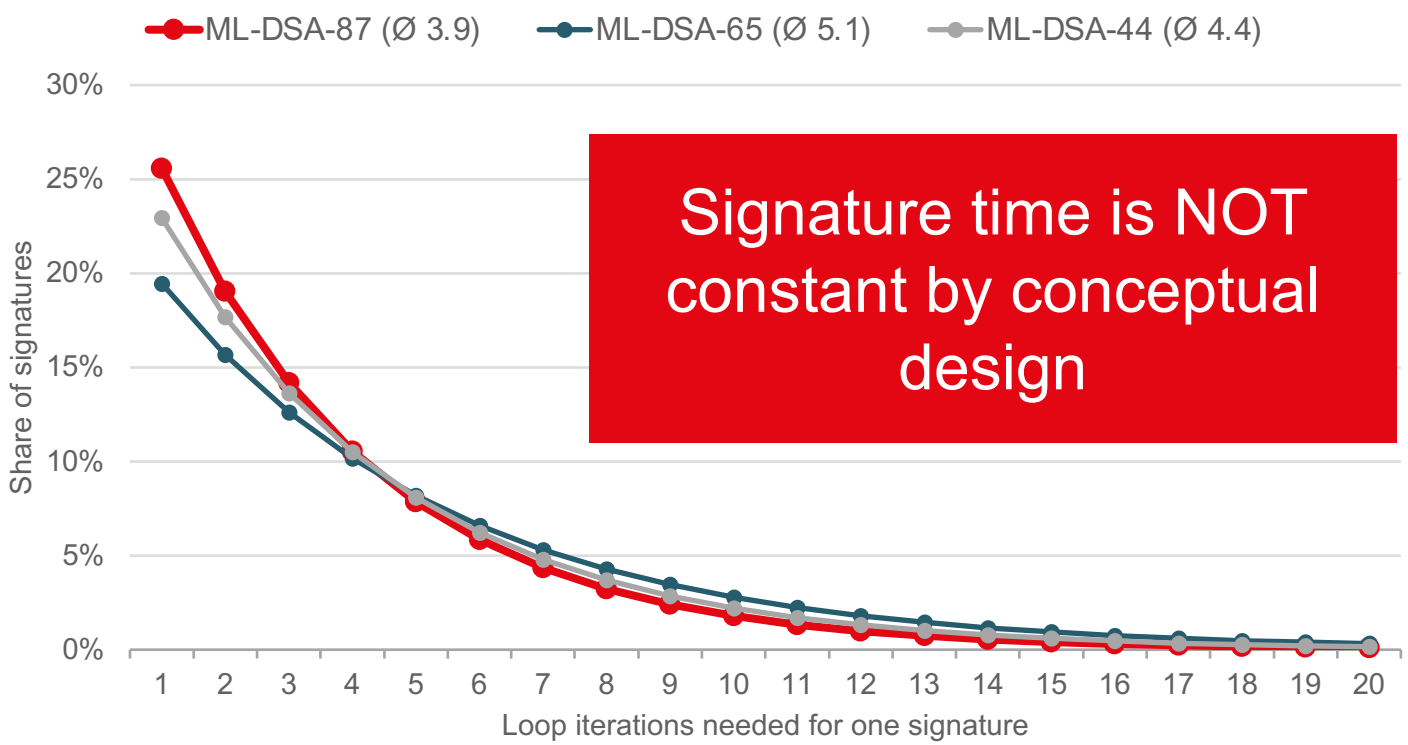
His recommendation: *"deploy PQ systems as an extra layer of security beyond ECC, rather than as a replacement for ECC."*[5]

[1] D. Amiet, A. Curiger (Securosys SA), L. Leuenberger, P. Zbinden: "Defeating NewHope with a Single Trace", PQCrypto 2020, LNCS 12100, ePrint 2020/368.
[2] R. Primas, P. Pessl, S. Mangard: "Single-Trace Side-Channel Attacks on Masked Lattice-Based Encryption", CHES 2017, LNCS 10529.
[3] P. Ravi, S. S. Roy, A. Chattopadhyay, S. Bhasin: "Generic Side-channel Attacks on CCA-secure Lattice-based PKE and KEMs", IACR TCHES 2020(3).
[4] D. J. Bernstein, K. Bhargavan, S. Bhasin, A. Chattopadhyay et al.: "KyberSlash: Exploiting secret-dependent division timings in Kyber implementations", 2025.
[5] D. J. Bernstein: "Exploiting ML-DSA bugs", 1 June 2026, cr.yp.to/papers/mldsa-20260601.pdf, Sections 1.4, 6.2 and 7.5.

© Securosys 2026, Marcel Dasen

Rejection Sampling: Every Signature Is a Lottery

Each attempt: sample y → z = y + c·s_1 → **reject** if z or hint could reveal s_1 → retry[3]



ML-DSA-87: iterations per signature[1,2]

3.9	3	16	24
mean	median	p99	p99.9

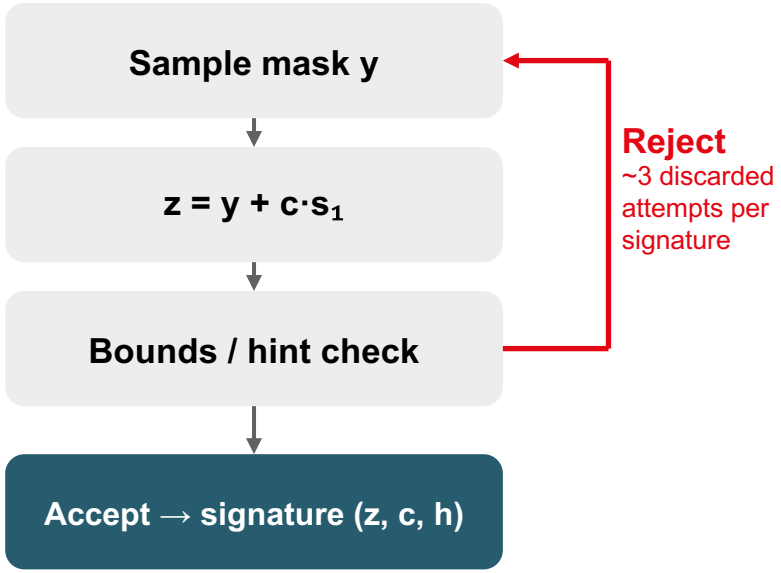
- **Latency:** p99 signing time ≈ 4× the average; 1 in 20 signatures needs >10 loops
- **Throughput:** set by the mean (~3.9 loops, plus ~1% hint rejections), not by one attempt
- **Sizing:** plan HSM capacity and timeouts for the tail, not the average

[1] NIST FIPS 204, Module-Lattice-Based Digital Signature Standard (ML-DSA), Aug 2024, Table 1: expected repetitions 4.25 / 5.1 / 3.85 (ML-DSA-44 / -65 / -87).
[2] NIST pqc-forum, "ML-DSA signing failure rate": measured incl. hint rejection 4.36 / 5.14 / 3.91 (H. Becker, M.-J. O. Saarinen, K. Kwiatkowski).
[3] Distribution: geometric model, P(k) = p·(1−p)^(k−1) with p = 1 / E[iterations] from²; percentiles derived from the same model.

© Securosys 2026, Marcel Dasen

Rejected Signatures may Leak the Key

The signing loop



Rejected attempts are never output — but they are computed on the device and leak through power and EM.

TCHES 2025

Rejected Signatures' Challenges Pose New Challenges: Key Recovery of CRYSTALS-Dilithium via Side-Channel Attacks[1]

Single-trace leakage of rejected challenges and responses → private key in seconds to minutes, 100% success, all three levels (Cortex-M4).

ePrint 2025

Release the Power of Rejected Signatures: An Efficient Side-Channel Attack on the ML-DSA Cryptosystem[2]

Rejected signatures alone recover the key with < 30 traces; adding them cuts traces needed by ≥ 50% (reference and ASM implementations).

DATE 2026

Rejection Matters: Efficient Non-Profiling Side-Channel Attack on ML-DSA via Exploiting Public Templates[3]

No clone device needed; ~300 traces for key recovery (~10× fewer than prior art). Hedged (randomized) signing does not help.

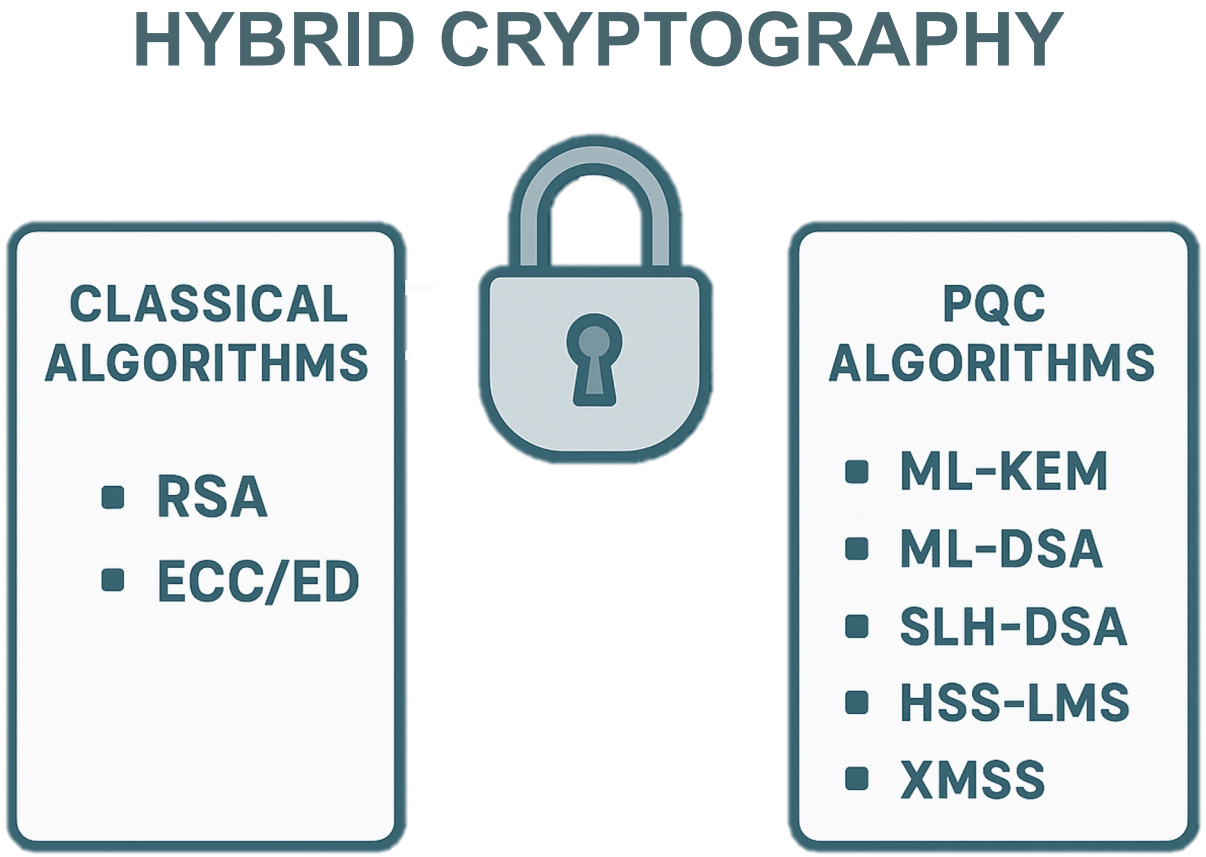
Mitigation: mask and protect every iteration incl. rejected ones; use SCA-evaluated implementations (e.g. certified HSMs)

[1] Y. Zhou, W. Wang, Y. Sun, Y. Yu: "Rejected Signatures' Challenges Pose New Challenges: Key Recovery of CRYSTALS-Dilithium via Side-Channel Attacks", IACR TCHES 2025 · ePrint 2025/214.
[2] Z. Liu, A. Wang, C. Wei, Y. Ding, J. Zhang, A. Liu, L. Zhu: "Release the Power of Rejected Signatures: An Efficient Side-Channel Attack on the ML-DSA Cryptosystem", IACR ePrint 2025/582.
[3] Y. Zhao, W. Cheng, Z. Qiao, Y. Liu, Y. Zhou: "Rejection Matters: Efficient Non-Profiling Side-Channel Attack on ML-DSA via Exploiting Public Templates", IEEE DATE 2026 · ePrint 2026/056.

© Securosys 2026, Marcel Dasen

Hybrid to the rescue ?

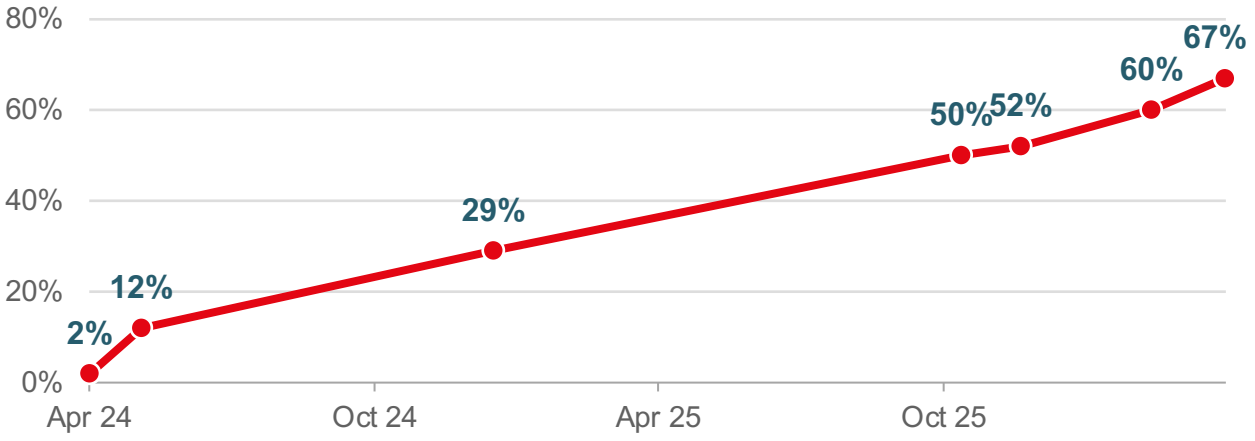
It depends…



© Securosys 2026, Marcel Dasen

Hybrid KEM: PQ safety at little Cost

Adoption: share of HTTPS requests to Cloudflare using hybrid PQ key exchange[1]



Drivers: Chrome 124 default (Apr 2024) · Apple iOS 26 (Sep 2025) · server defaults (OpenSSL 3.5, BoringSSL, Go)

Cost: X25519 → X25519MLKEM768[2,3]

Handshake bytes on the wire

1.0 kB → 3.3 kB **+2.3 kB**

Client CPU per full handshake

0.69 ms → 0.75 ms **+8 %**

Key-exchange op: ECDH vs. ML-KEM encaps

10.4 k/s 26.3 k/s **2.5× faster**

Typical network round trip: 20–100 ms → +0.06 ms CPU is invisible to users

Hybrid makes perfect sense for the session secret: keys are ephemeral, so hybrid buys protection against harvest-now-decrypt-later today, X25519 remains a safety net, and nothing long-lived has to be migrated again.

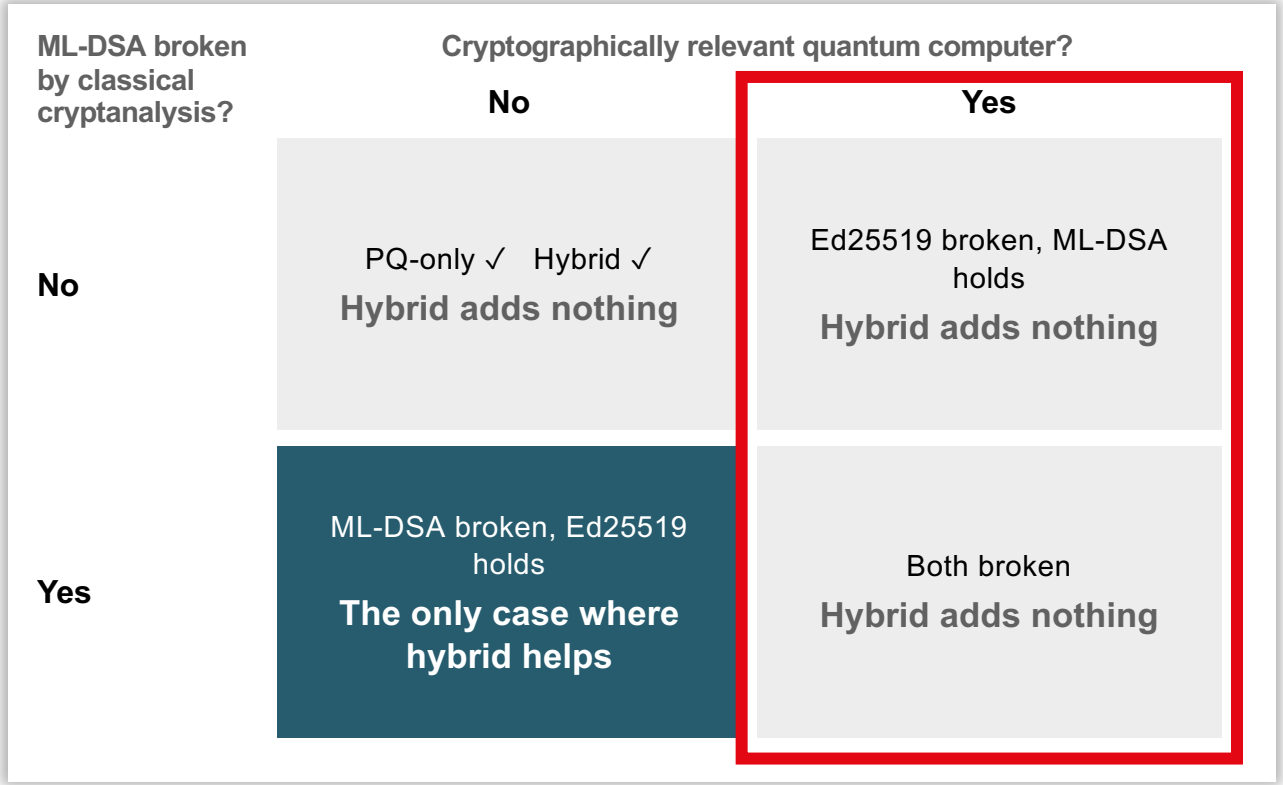
[1] Cloudflare: Radar 2024 Year in Review (Apr–May 2024: 2 %→12 %); Radar 2025 Year in Review (Jan 2025: 29 %, Dec 2025: 52 %); "State of the post-quantum Internet in 2025", 28 Oct 2025 (> 50 %); Radar blog, 27 Feb 2026 (> 60 %); @CloudflareRadar, 15 Apr 2026 (≈ 67 % of human HTTPS traffic). radar.cloudflare.com/post-quantum

[2] Own measurement: OpenSSL 3.5.4, TLS 1.3 full handshakes (s_server / s_time) over loopback, Ed25519 server certificate, 2 vCPU Xeon 2.8 GHz; openssl speed for key-exchange operations.

[3] Key exchange uses X25519 (ECDH on Curve25519), the key-agreement counterpart of the Ed25519 signature scheme.

© Securosys 2026, Marcel Dasen

Hybrid Certificates: Complexity with “no” Benefit



1. **No harvest-now-decrypt-later for signatures**
   A signature only has to hold until it is verified — authentication can switch to PQ when needed, in one step.

2. **Backward compatibility is no argument**
   A legacy verifier only checks the classical signature: it gains zero PQ assurance. Extra security exists only where both signatures are verified.

3. **Hybrid is the most expensive option**
   Two keys, two signatures, composite OIDs, more code paths to get wrong — and TLS bytes ≥ pure PQ.

Recommendation: go straight to PQ for authentication — one migration, one code path, fits existing standards

© Securosys 2026, Marcel Dasen

PQC transition guidance

It seems easy until you do it



© Securosys 2026, Marcel Dasen

Summary

Don’t rush it, plan and manage it

1 SIDE-CHANNEL LEAKAGE — HSM

Run PQC inside a certified HSM

The HSM prevents attackers from collecting the traces they need. It does not protect against a mathematical failure of the new algorithms.

2 STATEFUL HASH SIGNATURES — HSM

LMS / XMSS (SP 800-208) need a hardware module[1]

NSA-recommended for firmware signing — the one-time-key state must never be reused, so signing and state management belong in an HSM.

3 KEY EXCHANGE

Use hybrid: X25519 + ML-KEM

Session keys are ephemeral: hybrid adds protection against harvest-now, decrypt-later at almost no cost.

4 LONG-TERM ENCRYPTED DATA — HSM

Re-encrypt with symmetric-only keys

Replace RSA key wrapping by AES-256 keys held in a secure key store, plus an extra access-control layer that releases keys only for authorised use.

5 CERTIFICATES & AUTHENTICATION

Go straight to pure PQ — with high crypto-agility[2]

If ML-DSA fails completely, fall back to RSA-4096: counter-intuitive, but it needs ~16× the logical qubits of ECC256 and buys a few years.

6 FAILURE NOT AN OPTION — HSM

Use hash-based signatures now

LMS / XMSS or SLH-DSA rely only on the security of hash functions — choose them where you have to be sure today.

Hardware protects the implementation — crypto-agility and hash-based signatures protect against the mathematics.

[1] NSA CNSA 2.0 FAQ v2.1 (Dec 2024), p. 6: LMS/XMSS — “implement signing and state management in hardware”; NIST SP 800-208.
[2] Shor’s algorithm needs about 2n+3 logical qubits for an n-bit modulus (Beauregard, 2003): ~8,200 for RSA-4096 vs ~520 for ECC256.

© Securosys 2026, Marcel Dasen

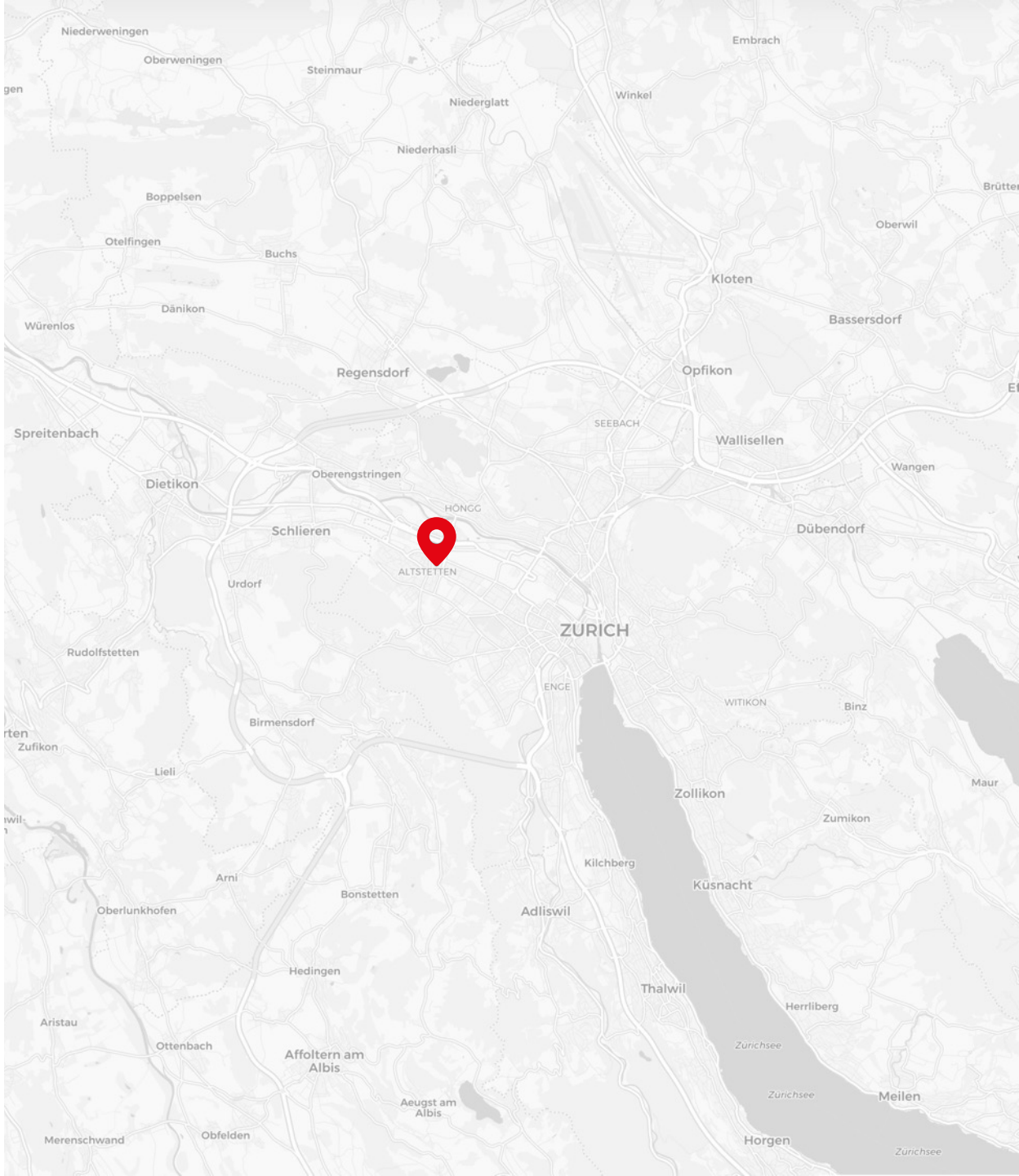
Thank you

Securosys SA
Max-Högger-Strasse 2
8048 Zurich, Switzerland

+41 44 552 31 00

info@securosys.com

Disclaimer: this presentation has been compiled with the aid of AI.



© Securosys 2026, Marcel Dasen