Lessons from a Decade of Post-Quantum Authentication

Masoud Asadi, Ehsan Ghasemi, Mohsin Kahn, Ferhat Karakoç, Anna Kåhre, Andreas Ljunggren, Santeri Paavolainen, John Preuß Mattsson, Göran Selander, Sini Ruohomaa, Erik Thormarker, Juha Sääskilahti

Ericsson

Abstract: Ericsson began its PQC migration project in 2015. From the outset, authentication has been a major part. This position paper distills a decade of research and development into concrete recommendations: avoid stateful signatures and unnecessary complexity; prefer SHA-3; standardize seed-only private-key formats; prefer hedged signatures; avoid floating-point arithmetic; recognize practical quantum attacks on symmetric cryptography are a misconception; and treat composite signatures with caution because of their malleability and compliance risks. Internal benchmarking shows that ML-DSA and SLH-DSA meet telecom performance requirements and are the only sufficiently mature options for the 2030–2031 migration deadlines. Constrained IoT systems, however, still require lightweight KEM- and NIKE-based authentication that current standardized PQC algorithms do not provide.

Introduction

Ericsson started its PQC migration project in 2015, shortly after the US government announced that it would initiate a transition to quantum-resistant algorithms. From the outset, a major part of the project focused on authentication, covering new theoretical constructions [1], survey and overview papers [2], constrained radio systems [3], KEM-based authentication [4], public-key infrastructure [5], and extensive internal studies of the viability of PQC authentication in real-world products. This position paper shares the lessons learned over the past decade, including Ericsson's current plans and the path forward:

- **Avoid stateful signatures**. Stateful signatures such as XMSS [6] and LMS [7] require the signer to securely maintain and update state for every signature. While this is suboptimal, the associated security risks can be managed in some use cases, such as certificate signing. NIST identified in SP 800-208 [8] that backups of private keys pose a major risk to state management and therefore prohibited private-key backup. Due to this, Ericsson product roadmaps forbid the use of XMSS and LMS due to the significant security and business continuity risks associated with state mismanagement and private-key loss.

- **Prefer SHA-3 over SHA-2.** Even before the publication of SHA-3 [9–10], Ericsson contributed to the design of the standardized Keccak-based TUAK algorithm [11] for 4G and has been encouraging suppliers to support SHA-3 ever since. Unfortunately, CNSA has been a major obstacle to supplier adoption of SHA-3. The theoretical weaknesses of SHA-2 make secure use significantly more complex and have contributed to several practical vulnerabilities, including side-channel vulnerabilities. A recent example is the vulnerabilities in SPHINCS+ [12], demonstrating that even cryptography experts can make security-critical mistakes when using SHA-2. For an example of the complexity introduced by SHA-2, see Section 11 of FIPS 205 [13]. We use the SHAKE256-based Ed448 rather than Ed25519 to avoid SHA-2 and believe that all future systems should use (Turbo)SHAKE [9–10, 14]. Our strong preference for the SHA-3 family also means that we have no plans to use HashML-DSA, which is additionally considered harmful [15].

- **Private-key formats and seed-only representations.** We determined already in 2016 that seeds should be the canonical private-key format [1]. Unfortunately, ML-DSA specifies multiple private-key formats [16–17], which are now used in practice, creating unnecessary complexity and interoperability problems. Some libraries support only the 32-byte seed ξ, while others support only the expanded private key. While we prefer seed-only representations, a single expanded-key format would have been preferable to the current situation. We encourage SDOs to specify a single, seed-only private-key format for all future standards. Seed-only formats not only minimize persistent secret state but also provide stronger binding properties.

- **Be careful with composite signatures.** Composite signatures move complexity from the protocol layer into the PKI. For an organization that controls the entire system, this is a poor trade-off. Poorly designed composites, such as [18], not only inherit malleability vulnerabilities from their components but also introduce new malleability vulnerabilities and destroy the beyond-unforgeability (BUFF) properties of ML-DSA. Malleability and the lack of BUFF properties have caused many practical vulnerabilities in the past [19]. Restricting use to a single protocol or applications not requiring SUF-CMA and BUFF is not a solution. Developers cannot be expected to understand subtle security properties, and signatures are routinely reused beyond their original protocol, for file encryption, allowlists, logging, incident response, and more. Signatures should therefore provide strong security properties by default. Composite signatures as long-term roots of trust create legal and regulatory uncertainty. The weak composites in [18] do not qualify as hybrids under many definitions [20–25], which require a hybrid construction to provide security at least as strong as its strongest component. EU documentation further states that, unless explicitly specified otherwise, all components must be agreed for a construction, as a whole, to be considered agreed [26]. Non-composite signatures avoid these problems: they do not create a new algorithm, and the quantum-vulnerable component can easily be removed. As correctly identified by NIST [27], all future general-purpose signatures should provide SUF-CMA. Several European agencies are recommending against committing to weak composites such as [18], and ANSSI is driving the standardization of strong, well-designed composites [28]. Such composites are particularly well-suited for protocols such as SSH that do not rely on PKI. Ericsson internal guidelines forbid the use of composites that significantly decrease the security properties of ML-DSA and forbid the use of all composites for long-term roots of trust.

- **NSA was right about the implementation risks of hybrid signatures.** Non-composite hybrid signatures should be straightforward to implement: the signer generates signatures using two independent algorithms, and the verifier accepts the signature if and only if both

component signatures are valid. Nevertheless, Ericsson has now encountered two non-composite hybrid signature implementations from different suppliers that fail to meet this basic requirement. This aligns with NSA warnings [29] and is also consistent with a broader pattern in which suppliers frequently overlook cryptographic requirements, such as assurance of public-key validity, private-key possession, and key-owner identity, that are not captured by interoperability test vectors. Ericsson has also implemented a non-composite hybrid signature solution internally for a platform that does not support SLH-DSA; it behaves as expected, with verification succeeding if and only if both component signatures are valid.

- **SLH-DSA and ML-DSA performance is adequate.** The performance of SLH-DSA is adequate for all our TLS, DTLS, QUIC, IKEv2, and SSH authentication use cases in telecom [30]. We strongly prefer accepting a modest performance penalty over introducing the additional complexity of hybrid signatures. ML-DSA therefore also clearly meets our performance requirements. Merkle-Tree Certificates (MTC) [31] introduce substantial additional complexity into the PKI and should be considered only if they become broadly available and the performance of conventional ML-DSA certificates proves problematic.

- **Prefer hedged signatures.** NIST added hedged signing to ML-DSA [16] and made it the default, based on a suggestion from Ericsson. NIST has also stated that it intends to do the same for FN-DSA [32], again based on a suggestion from Ericsson. Hedged signing is strongly preferred because it provides robust protection against two different classes of implementation failures. The fresh randomness used during signing helps protect against side-channel and fault-injection attacks, while the deterministic component derived from the signing key and message provides robustness against a weak or compromised random-number generator.

- **Security Levels 1 and 2 are secure.** The idea that symmetric cryptography will be practically affected by CRQCs is a misconception [33]. Quantum algorithms offering only a quadratic speedup, such as Grover's algorithm, which is proven to be optimal, do not provide a practical quantum advantage against modern symmetric cryptography. The focus that some European agencies place on upgrading symmetric cryptography [26] is distracting from the urgent PQC migration.

- **Avoid floating-point.** We have always been cautious about the use of floating-point arithmetic in cryptography, including FN-DSA. Floating-point arithmetic is a non-starter in constrained environments and, even on laptops and servers, implementations are notoriously difficult to reason about across different architectures, FPUs, and compiler environments. Recent work on floating-point error sensitivity in Falcon [34], together with the development of a practical, high-performance fixed-point implementation [35], have convinced us that FIPS 206 should eliminate floating-point altogether. Ericsson will likely prohibit the use of FN-DSA implementations relying on floating-point arithmetic in our products.

- **ML-DSA and SLH-DSA are the only mature options.** For systems subject to the 2030 and 2031 migration deadlines [36–38], only ML-DSA and SLH-DSA are sufficiently mature for deployment, including non-composite authentication. FN-DSA, multivariate signatures, and SQIsign should, for these use cases, be regarded as potential future optimizations, with the latter two still in research. In addition to the cryptographic vulnerabilities and legal uncertainties associated with some composite constructions, the implementation maturity of composite signatures is many years behind that of ML-DSA and SLH-DSA. As shown in Table 1, at least nine interoperable TLS implementations support ML-DSA authentication, and at least four support SLH-DSA. To our knowledge, no TLS implementation currently

supports composite signatures. See [39] for an excellent overview of potential future algorithms and the reasons why the Web will deploy ML-DSA.

- **Long transition periods demand early action.** Systems such as public-key infrastructures (PKIs) and devices with long lifetimes require long transition periods. The EU roadmap on PQC explicitly recognizes this challenge [20]. CNSA has similarly prioritized software and firmware signing over other use cases [29], because roots of trust, once deployed, may be difficult to replace. Moreover, experience shows that even roots of trust in consumer devices that are relatively easy to update are often never replaced. Early cryptographically relevant quantum computers (CRQCs) will most likely be expensive and scarce resources, making it reasonable to assume that adversaries will have to prioritize which targets to attack [40]. Long-term authentication keys and roots of trust are particularly valuable targets, as compromising them can enable active attacks and potentially provide access to significantly more data than compromising a single ephemeral key. Protecting systems against CRQCs over the coming decades therefore requires starting these long transitions now.

- **Lightweight PQC is needed for IoT.** ML-KEM and ML-DSA are unsuitable for constrained IoT radio systems [3], particularly those with payloads of only a few tens of bytes, low transmission rates, and strict duty-cycle limits. This is especially evident for mutually authenticated key exchange. KEM- and NIKE-based authentication can provide message and code size advantage over signature-based authentication. However, signatures may still be required for other device functions, such as firmware and software signing. To our knowledge, the IETF LAKE WG is the first SDO to have adopted work on KEM-based authentication [41]. Since current PQC algorithms are not well suited to constrained IoT environments, LAKE will ask CFRG to consider researching and specifying more lightweight quantum-resistant key-exchange algorithms, including lattice-based KEMs and isogeny-based non-interactive key exchanges (NIKEs) [42]. As shown in Table 2, NIKE-based approaches can significantly reduce the message size of mutually authenticated AKEs. LAKE has already standardized NIKE-based authentication [43].

Summary and Conclusions

A decade of PQC authentication work has taught us the importance of simplicity and maturity. Stateful signatures, composite signatures, and floating-point schemes introduce operational or compliance risks that outweigh their benefits, while ML-DSA and SLH-DSA are already mature and performant enough for telecom deployment. Uneven implementation quality, even for "simple" non-composite hybrids, shows that conformance testing must go beyond basic algorithm compliance. Constrained IoT systems remain underserved and require lightweight KEM- and NIKE-based authentication. With migration deadlines approaching in 2030–2035, these choices are becoming urgent. We share our experience, including our missteps, to help others avoid repeating them.

Table 1. ***CipherSnake PQC capability test results as of 27 August 2026.*** *CipherSnake is an internal Ericsson test suite for evaluating TLS 1.3 libraries, including interoperability and support for PQC key exchange, PKI, and authentication. Libraries are scored based on standard compliance and support for ML-DSA, SLH-DSA-SHAKE, X25519MLKEM768 (*), and MLKEM1024 (**). Some libraries were tested using pre-release versions (†).*

	Library	ML-DSA (PKI)	ML-DSA (auth)	SLH-DSA (PKI)	SLH-DSA (auth)	ML-KEM hybrid*	ML-KEM standalone**
★★★★	**bouncycastle:1.85**	✓	✓	✓	✓	✓	✓
★★★★	**erlang-otp:29.0.4**	✓	✓	✓	✓	✓	✓
★★★★	**openssl:pre-4.1.0-fd2e286†**	✓	✓	✓	✓	✓	✓
★★★★	**wolfssl:pre-5.9.3-1e5216e†**	✓	✓	✓	✓	✓	✓
★★★	**openssl:4.0.1**	✓	✓	✓	✗	✓	✓
★★★	**wolfssl:5.9.2**	✓	✓	✓	✗	✓	✓
★★	**boringssl:0.20260803.0**	✓	✓	✗	✗	✓	✓
★★	**gnutls:3.8.13**	✓	✓	✗	✗	✓	✗
★★	**go-crypto:1.27**	✓	✓	✗	✗	✓	✓
★★	**nss:3.128**	✓	✓	✗	✗	✓	✓
★★	**rustls:0.23.43**	✓	✓	✗	✗	✓	✓
★	**openjdk:27-rc†**	✗	✗	✗	✗	✓	✗
—	**mbedtls:4.2.0**	✗	✗	✗	✗	✗	✗
—	**openjdk:26**	✗	✗	✗	✗	✗	✗

Table 2. ***Comparison of message and authentication key sizes for Authenticated Key Exchange.*** *The comparison highlights the importance of standardizing lightweight KEMs and NIKEs for constrained IoT radio systems. KEM- and NIKE-based authentication have the additional benefit that the same algorithm can be used for both authentication and key exchange, whereas provisioning and managing PSKs is significantly more complex. DAWN [44], CTIDH [45], and MIKE [46–47] are examples of lightweight research algorithms. The total message size is an approximate estimate for the LAKE protocol.*

	Auth API	Mutual Authentication Algorithm	Ephemeral Key Exchange Algorithm	Total Message Size (bytes)	Authentication Key Size (bytes)
Traditional	Signature	Ed25519	X25519	220	32
	NIKE	X25519	X25519	120	32
	PSK	PSK	X25519	120	16
NIST PQC	KEM	HQC-KEM-128	HQC-KEM-128	15800	2249
	Signature	ML-DSA-44	HQC-KEM-128	11610	1312
	Signature	FN-DSA-512	HQC-KEM-128	8100	897
	PSK	PSK	HQC-KEM-128	6800	16
	Signature	ML-DSA-44	ML-KEM-512	6430	1312
	KEM	ML-KEM-512	ML-KEM-512	3160	800
	Signature	FN-DSA-512	ML-KEM-512	2930	897
	PSK	PSK	ML-KEM-512	1630	16
Other PQC / Research	Signature	FN-DSA-512	DAWN-β-512	2320	897
	KEM	DAWN-β-512	DAWN-β-512	1920	514
	KEM	mceliece348864	ML-KEM-512	1820	261120
	Signature	Mayo 1	DAWN-β-512	1630	1168
	Signature	Mayo 2	DAWN-β-512	1350	5488
	Signature	SQISign-I	DAWN-β-512	1290	65
	KEM	mceliece348864	DAWN-β-512	1210	261120
	Signature	UOV Is	DAWN-β-512	1180	66576
	PSK	PSK	DAWN-β-512	1020	16
	Signature	Mayo 2	CTIDH-2048	900	5488
	Signature	SQISign-I	CTIDH-2048	830	65
	Signature	UOV Is	CTIDH-2048	730	66576
	NIKE	CTIDH-2048	CTIDH-2048	570	256
	PSK	PSK	CTIDH-2048	570	16
	Signature	Mayo 2	MIKE	520	5488
	Signature	SQISign-I	MIKE	450	65
	Signature	UOV Is	MIKE	350	66576
	NIKE	MIKE	MIKE	190	64
	PSK	PSK	MIKE	190	16

References

[1] Post-quantum Lattice-based Cryptography
https://kth.diva-portal.org/smash/get/diva2:935425/FULLTEXT01.pdf

[2] Quantum-Resistant Cryptography
https://arxiv.org/pdf/2112.00399

[3] Constrained Radio Networks, Small Ciphertexts, Signatures, and Non-Interactive Key Exchange
https://csrc.nist.gov/csrc/media/Presentations/2022/constrained-radio-networks-small-ciphertexts/images-media/session6-mattson-constrained-radio-networks-pqc2022.pdf

https://csrc.nist.gov/csrc/media/Events/2022/fourth-pqc-standardization-conference/documents/papers/constrained-radio-networks-pqc2022.pdf

[4] ML-KEM is Great! What's Missing?
https://csrc.nist.gov/csrc/media/Presentations/2025/ml-kem-is-great/images-media/ml-kem-is-great.pdf

https://csrc.nist.gov/csrc/media/Events/2025/workshop-on-guidance-for-kems/documents/papers/ml-kem-is-great-paper.pdf

[5] Migrating telecom to quantum-resistant cryptography on a global scale
https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/migrating-telecom-to-quantum-resistant-cryptography

[6] XMSS: eXtended Merkle Signature Scheme
https://www.rfc-editor.org/rfc/rfc8391.html

[7] Leighton-Micali Hash-Based Signatures
https://www.rfc-editor.org/rfc/rfc8554.html

[8] Recommendation for Stateful Hash-Based Signature Schemes
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-208.pdf

[9] SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

[10] SHA-3 Derived Functions: cSHAKE, KMAC, TupleHash and ParallelHash
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-185.pdf

[11] Specification of the TUAK algorithm set
https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2402

[12] Breaking Category Five SPHINCS+ with SHA-256
https://eprint.iacr.org/2022/1061.pdf

[13] Stateless Hash-Based Digital Signature Standard
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.205.pdf

[14] KangarooTwelve and TurboSHAKE
https://www.rfc-editor.org/rfc/rfc9861.html

[15] HashML-DSA considered harmful
https://keymaterial.net/2024/11/05/hashml-dsa-considered-harmful/

[16] Module-Lattice-Based Digital Signature Standard
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.204.pdf

[17] Algorithm Identifiers for the Module-Lattice-Based Digital Signature Algorithm (ML-DSA)
https://www.rfc-editor.org/rfc/rfc9881.html

[18] Composite Module-Lattice-Based Digital Signature Algorithm (ML-DSA)
https://datatracker.ietf.org/doc/html/draft-ietf-lamps-pq-composite-sigs

[19] Ericsson comments on SP 800-230 Additional SLH-DSA Parameter Sets
https://csrc.nist.gov/files/pubs/sp/800/230/ipd/docs/sp800-230_ipd_comments_received.pdf

[20] A Coordinated Implementation Roadmap for the Transition to Post-Quantum Cryptography
https://ec.europa.eu/newsroom/dae/redirection/document/117507

[21] Roadmap on Post-Quantum Cryptography - FAQ
https://ec.europa.eu/newsroom/dae/redirection/document/132120

[22] Transitioning to a Quantum-Secure Economy
https://www3.weforum.org/docs/WEF_Transitioning%20to_a_Quantum_Secure_Economy_2022.pdf

[23] Hybrid PKI Migration Challenges and Solutions
https://project-hapkido.nl/publish/pages/7993/tno_2025_r12487_d6-2_hybrid_pki_migration_challenges_and_solutions.pdf

[24] Nationella rekommendationer för övergången till kvantsäker kryptografi
https://www.ncsc.se/siteassets/publikationer/nationella-rekommendationer-for-overgangen-till-kvantsaker-kryptografi-2026_tga.pdf

[25] Canadian National Quantum-Readiness
https://ised-isde.canada.ca/site/spectrum-management-telecommunications/sites/default/files/documents/Quantum-Readiness%20Best%20Practices%20-%20v04%20-%2010%20July%202024.pdf

[26] ECCG Agreed Cryptographic Mechanisms
https://certification.enisa.europa.eu/document/download/a845662b-aee0-484e-9191-890c4cfa7aaa_en

[27] Status Report on the Second Round of the Additional Digital Signature Schemes Standardization
https://nvlpubs.nist.gov/nistpubs/ir/2026/NIST.IR.8610.pdf

[28] Silithium - A Compact, Efficient and Non-separable Hybrid Signature
https://www.ietf.org/archive/id/draft-devevey-cfrg-silithium-00.html

[29] The Commercial National Security Algorithm Suite 2.0 and Quantum Computing FAQ
https://media.defense.gov/2022/Sep/07/2003071836/-1/-1/0/CSI_CNSA_2.0_FAQ_.PDF

[30] Analysis of signature performance
https://www.3gpp.org/ftp/tsg_sa/WG3_Security/TSGS3_129_Prague/Docs/S3-263235.zip

[31] PKI, Logs, And Tree Signatures (plants)
https://datatracker.ietf.org/group/plants/about/

[32] FIPS 206 Status Update
https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/1HXzjlMUU6Y/m/NEdiz9TqAQAJ

[33] IETF Statement on Quantum Safe Cryptographic Protocol Inventory
https://datatracker.ietf.org/liaison/1942/

[34] Do Not Disturb a Sleeping Falcon: Floating-Point Error Sensitivity of the Falcon Sampler and Its Consequences
https://eprint.iacr.org/2024/1709.pdf

[35] Toward a Secure Fixed-Point Implementation of the Falcon Signature Scheme
https://eprint.iacr.org/2026/1531.pdf

[36] Planning for post-quantum cryptography
https://www.cyber.gov.au/business-government/secure-design/quantum/planning-for-post-quantum-cryptography

[37] Securing the Nation against Advanced Cryptographic Attacks
https://www.whitehouse.gov/presidential-actions/2026/06/securing-the-nation-against-advanced-cryptographic-attacks/

[38] Guide des Mécanismes cryptographiques
https://messervices.cyber.gouv.fr/documents-guides/anssi-guide-mecanismes-crypto-3.00.pdf

[39] Why we cannot wait for better post-quantum signature algorithms
https://blog.cloudflare.com/ml-dsa-will-have-to-do/

[40] On factoring integers, and computing discrete logarithms and orders, quantumly
https://www.diva-portal.org/smash/get/diva2:1902626/FULLTEXT01.pdf

[41] KEM-based Authentication for LAKE
https://datatracker.ietf.org/doc/html/draft-ietf-lake-authkem-edhoc

[42] Post-Quantum (PQ) EDHOC Design Team (DT)
https://datatracker.ietf.org/meeting/126/materials/slides-126-lake-08-pq-edhoc-dt-summary-ietf126presentation-00

[43] Quantum-Resistant Cipher Suites for LAKE
https://datatracker.ietf.org/doc/html/draft-ietf-lake-pqsuites

[44] DAWN: Smaller and Faster NTRU Encryption via Double Encoding
https://eprint.iacr.org/2025/1520.pdf

[45] CTIDH: faster constant-time CSIDH
https://tches.iacr.org/index.php/TCHES/article/view/9069/8657

[46] MIKE (Module Isogeny Key Exchange): An ἰχθύς introduction
https://eprint.iacr.org/2026/640.pdf

[47] PQ KEM comparison tool
https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/1PRO-trRrhc/m/psDaV72DAQAJ