Discussion on Quantum-resistance

John Preuß Mattsson
Expert Cryptographic Algorithms and Security Protocols, Ericsson
MSc Engineering Physics/Theoretical Computer Science
MSc Business Administration and Economy

IETF 122 Bangkok, March 18, 2025

Discussion on quantum-resistance

- If large robust quantum computers are ever built, they will break RSA and ECC in a few hours or days.
- Quantum computer attacks will have no practical effect on symmetric crypto (like AES-128 and SHA-256). All strong 128-bit symmetric cryptography is quantum-resistant.
- NIST has finalized standardization of the most important new quantum-resistant algorithms. ML-KEM (FIPS 203), ML-DSA (FIPS 204), and SLH-DSA (FIPS 205).
- PQC standardization will continue in NIST, IETF, etc., in the next five years.
- When to migrate depends for how long protection is required as well as the value of the target.
- General recommendations: Migrate as soon as possible, prioritized systems by 2030, and all systems by 2035

IETF and 3GPP statements on symmetric cryptography

IETF Statement:

— "**The idea that symmetric cryptography will be practically affected by CRQCs is now seen as a misconception**. The "bits of security" concept does not work with algorithms that are not parallelizable and NIST is therefore transitioning to quantum-resistant security levels based on symmetric algorithms where level 1 is equivalent with AES-128, level 2 is SHA-256, etc. UK government assesses that "symmetric algorithms with at least 128-bit keys (such as AES) can continue to be used". **While classical supercomputers might be able to brute force AES-128 around the year 2090**, a huge cluster of one billion CRQCs (according to one estimate costing one billion USD each) would take a million years of uninterrupted calculation to find a single AES-128 key. **Algorithms with quadratic (n^2) speedup like Grover's algorithm (which is proven to be optimal) will not provide any practical quantum advantage for breaking symmetric cryptography and likely not for any other problems.**"

3GPP Statement:

— "**A very good summary of the impact of Cryptographically Relevant Quantum Computers (CRQCs) on symmetric cryptography was recently given in a statement by the Internet Engineering Task Force (IETF)**. The IETF statement refers to UK NCSC whitepaper that says symmetric algorithms with at least 128-bit keys (such as AES) can continue to be used. **SA3 agrees with IETF's analysis**. Most other 128-bit algorithms such as SNOW 3G, ZUC, TUAK, KMAC128, Ascon, etc. are likely to have similar quantum overhead for Grover's algorithm which is known to be optimal. Even if an algorithm has slightly lower quantum overhead than AES-128, SA3 believes the algorithm would still fulfill the requirement (comparable to key search on AES-128) for quantum resistance category 1"

(but 6G will likely use high-performance 256-bit algorithms)

Sam Jaques keynote at CHES 2024: "Qubits would cover the moon"

NIST 2024: "All NIST-approved symmetric primitives that provide at least 128 bits of classical security are believed to meet the requirements of at least Category 1 security"

New quantum-resistance security levels

— “bits of security” does not correlate with practical security when attacks cannot be effectively parallelized.
A CRQC that can break RSA in a few hours would not pose any practical threat at all to symmetrical algorithms such as AES-128 and SHA-256. Even with millions of CRQCs it would take millions of years.

— AES-128, Keccak, SHA-256, and Ascon will remain secure for the foreseeable future.

— NIST has defined five security levels for quantum-resistance:

Level	Security Description
I	At least as hard to break as AES128 (exhaustive key search)
II	At least as hard to break as SHA256 (collision search)
III	At least as hard to break as AES192 (exhaustive key search)
IV	At least as hard to break as SHA384 (collision search)
V	At least as hard to break as AES256 (exhaustive key search)

Discussion on quantum-resistance

- For EDHOC, quantum-resistance means not relying on ECC.
- EDHOC as specified in RFC 9528 uses ECC for key exchange, static DH authentication, and signature authentication.
- draft-ietf-lake-edhoc-psk uses PSK for authentication and PSK+ECC for key exchange.
- All standardized PQC algorithms are KEMs or Signatures.
- While ECC can be implemented with a DH or KEM interface, none of the standardized PQC algorithms can be implemented with a DH interface.
- No ongoing standardization of PQC algorithms with a DH interface, and no algorithms in literature with good performance. CSIDH has small sized but is quite slow.

Quantum-resistance key exchange

- ML-KEM, X-Wing (ML-KEM-768+X25519+SHA-3), Classic McEliece, HQC, and FrodoKEM are quantum-resistant KEMs that can be used as a drop-in replacement for key exchange in EDHOC method 0 (Signature-Signature) and EHDOC with PSK authentication (draft-ietf-lake-edhoc-psk).
- Cannot be used in method 1, 2, and 3, as the static DH authentication interacts with the key exchange and requires an DH interface.
- Great performance but the large sizes are problematic for many constrained use cases.

		TLS key share size (in bytes)		Operation per s (higher is better)	
Algorithm	Security Level	**Client**	**Server**	**Client**	**Server**
X25519	❌	32	32	19,000	19,000
ML-KEM-512	I	800	768	45,000	70,000
ML-KEM-768	III	1,184	1,088	29,000	45,000
ML-KEM-1024	V	1,568	1,568	20,000	30,000
mceliece348864	I	261,120	96	62,000	14,000

Inspired by Bas Westerbaan (Cloudflare)

Quantum-resistance key exchange

- Table 1.KEM Public key and ciphertext sizes in bytes. Total size is public key size plus ciphertext size which is a relevant measure when KEMs are used for ephemeral key exchange is protocols like TLS 1.3 and IKEv2.
- Table 2. KEM performance in cycles (50%) on 2023 AMD Ryzen 7 7700. Performance number from eBACS: ECRYPT Benchmarking of Cryptographic Systems by Bernstein and Lange (editors) https://bench.cr.yp.to/results-kem/amd64-hertz.html. Total is cycles for key gen + encapsulation + decapsulation.
- HQC and FrodoKEM are slower and have larger public key and ciphertext sizes.

Name	Category	Public key	Ciphertext	Total
ML-KEM-512	1	800	768	1568
HQC-128	1	2249	4481	6730
ML-KEM-512+HQC-128	1	3049	5249	8298
FrodoKEM-640	1	9616	9720	19336

Name	Category	Key Gen	Encapsulation	Decapsulation	Total
ML-KEM-512 (kyber512)	1	15420	24443	18693	58556
HQC-128 (hqc128round4)	1	61311	170433	283249	514993
ML-KEM-512+HQC-128	1	76731	194876	301942	573549
FrodoKEM-640 (frodokem640shake)	1	2084314	2265633	2222733	6572680

Quantum-resistance signatures

- ML-DSA, SLH-DSA, FN-DSA, XMSS, and LMS are quantum-resistant signatures that can be used as a replacement for ECC signatures in EDHOC method 0 (Signature-Signature).
- XMSS and LMS (not shown in table) are stateful, so not 100% drop-in replacements from system view.
- Large sizes and slow performance are problematic in some use cases.
- It will take many years until MAYO, UOV, SQISign, HAWK, etc. will be standardized.

		Sizes (bytes)		CPU time (lower is better)	
	Security level	Public key	Signature	Signing	Verification
Ed25519	✗	32	64	1 (baseline)	1 (baseline)
RSA-2048	✗	384	384	70	0.3
ML-DSA-44	II	1,312	2,420	4.8	0.5
FN-DSA-512	I	897	666	8*	0.5
SLH-DSA-128s	I	32	7,856	8,000	2.8
SLH-DSA-128f	I	32	17,088	550	7
$MAYO_1$	I	1,168	321	4.7	0.3
$MAYO_2$	I	5,488	180	5	0.2
UOV Is	I	66,576	96	2.5	2
SQISign I	I	64	177	60,000	500
HAWK-512	I	1,024	555	2	1

Inspired by Bas Westerbaan (Cloudflare)

Discussion on quantum-resistance

- Standardized PQC algorithms are drop-in replacements for key exchange and signatures in EDHOC method 0 and draft-ietf-lake-edhoc-psk. Performance is good, but the large sizes are problematic. COSE work needed. No LAKE work.
  - LMS already standardized in COSE. Work on ML-KEM, X-Wing, ML-DSA, SLH-DSA, and FN-DSA ongoing.
- None of the standardized PQC algorithms can be used in method 1, 2, or 3 (Static DH authentication), not even for key exchange. KEMs can be used for authentication, but this would require new EDHOC methods with more roundtrips.
- If the attacker has a CRQC, the theoretical security of EDHOC method 0–3 with ECC is completely lost. In practice, early CRCRs will likely be very expensive and only used in targeted attacks to recover keys and ciphertexts that are of particular interest.
- A CRQC would not affect the security of the PSK in draft-ietf-lake-edhoc-psk. The PSK is used for both authentication and key agreement. The theoretical security (PFS, identity protection) of ECDHE would be lost. What is left is a simple symmetric key agreement, where the G_X and G_Y are just used as random values.

Method Type Value	Initiator Authentication Key	Responder Authentication Key
0	Signature Key	Signature Key
1	Signature Key	Static DH Key
2	Static DH Key	Signature Key
3	Static DH Key	Static DH Key

Is something missing? Adding PSK to method 0-3?

- IKEv2 and TLS 1.3 specifies how to use a PSK together with ECDHE and Signature authentication to provide quantum-resistance. Used by several governments for national security systems.
- https://www.rfc-editor.org/rfc/rfc8784.html
- https://www.rfc-editor.org/rfc/rfc8773.html
- https://datatracker.ietf.org/doc/html/draft-ietf-tls-8773bis-05
- https://diva-portal.org/smash/get/diva2:1902626/FULLTEXT01.pdf
- This can be done quite easily by combining RFC 9528 method 0-3 with draft-ietf-lake-edhoc-psk and registering four new methods.
  - Could consider allowing a group PSK in these case. The PSK is then providing quantum-resistant group security.
- Could also consider four methods where one part authenticates with only PSK and the other with Signature+PSK or Static DH+PSK.

Is something missing? Adding PSK to method 0-3?

- **Quite simple to add PSK to the RFC 9528 methods. Just add wrap message_3 in ENC(ID_CRED_PSK, ….)**
- **Worth doing?**

```
Initiator                                                      Responder
|                METHOD, SUITES_I, G_X, C_I, EAD_1                     |
+--------------------------------------------------------------------->|
|                           message_1                                  |
|                                                                      |
|       G_Y, Enc( C_R, ID_CRED_R, Signature_or_MAC_2, EAD_2 )          |
|<---------------------------------------------------------------------+
|                           message_2                                  |
|                                                                      |
| Enc( ID_CRED_PSK, AEAD( ID_CRED_I, Signature_or_MAC_3, EAD_3 ) )     |
+--------------------------------------------------------------------->|
|                           message_3                                  |
|                                                                      |
|                         AEAD( EAD_4 )                                |
|<- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -+
|                           message_4                                  |
```

Further reading

- Ericsson Technology Review, “Quantum technology and its impact on security in mobile networks”
  https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/ensuring-security-in-mobile-networks-post-quantum

- Constrained Radio Networks, Small Ciphertexts, Signatures, and Non-Interactive Key Exchange
  https://csrc.nist.gov/csrc/media/Events/2022/fourth-pqc-standardization-conference/documents/papers/constrained-radio-networks-pqc2022.pdf

- New Scientist, “Cryptographers bet cash on when quantum computers will beat encryption”
  https://www.newscientist.com/article/2370022-cryptographers-bet-cash-on-when-quantum-computers-will-beat-encryption/

- NSA, “Announcing the Commercial National Security Algorithm Suite 2.0”
  https://media.defense.gov/2022/Sep/07/2003071834/-1/-1/0/CSA_CNSA_2.0_ALGORITHMS_.PDF

- BSI, ANSSI, Dutch and Swedish NCSA, ”Position Paper on Quantum Key Distribution”
  https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Crypto/Quantum_Positionspapier.pdf

- Ericsson Blog, “Migration to quantum-resistant algorithms in mobile networks”
  https://www.ericsson.com/en/blog/2023/2/quantum-resistant-algorithms-mobile-networks

- arXiv, “Quantum-Resistant Cryptography”
  https://arxiv.org/abs/2112.00399

- NISTIR 8547, ”Transition to Post-Quantum Cryptography Standards”
  https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf

- ANSSI, “Guide des Mécanismes cryptoraphiques”
  https://cyber.gouv.fr/sites/default/files/2021/03/anssi-guide-mecanismes_crypto-2.04.pdf

Further reading

- Status of quantum computer development
  https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Studien/Quantencomputer/Entwicklungstand_QC_V_2_0.html?nn=916616

- ANSSI plan for post-quantum transition
  https://pkic.org/events/2023/pqc-conference-amsterdam-nl/pkic-pqcc_jerome-plut_anssi_anssi-plan-for-post-quantum-transition.pdf

- Landscape of Quantum Computing in 2024
  https://sam-jaques.appspot.com/quantum_landscape_2024

- On factoring integers, and computing discrete logarithms and orders, quantumly
  http://kth.diva-portal.org/smash/get/diva2:1902626/FULLTEXT01.pdf

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

- 3GPP Statement on PQC Migration
  https://www.3gpp.org/ftp/tsg_sa/WG3_Security/TSGS3_118_Hyderabad/docs/S3-244307.zip

- Sam Jaques, "Quantum Attacks on AES"
  https://www.youtube.com/watch?v=eB4po9Br1YY&t=3227s

- FIPS 203 (ML-KEM), FIPS 204 (ML-DSA), and FIPS 205 (SLH-DSA) approved
  https://csrc.nist.gov/news/2024/postquantum-cryptography-fips-approved

- ML-KEM is Great. What's Missing?
  https://emanjon.github.io/Publications/ML-KEM%20is%20Great!%20What's%20Missing.pdf