E-GROUP
SOFTWARE & BEYOND

Quantum-Safe Cryptography
PQC – where “glitch in the Matrix” is required

Áron Szabó

MELASZ
2026-06-30

PQC vs. classic Cryptography

E-GROUP
SOFTWARE & BEYOND

asymmetric:

one private key, one public key

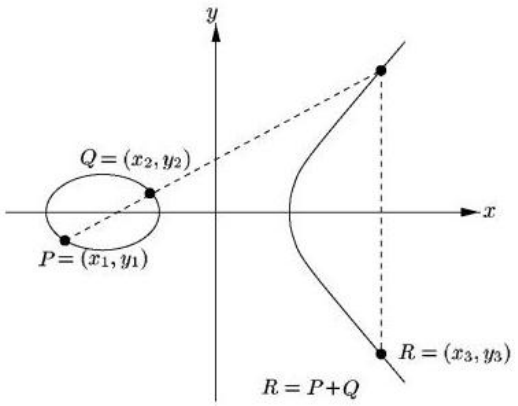
IFP - Integer Factorization Problem
(e.g. RSA)

```
p = 13
q = 11
n = p * q = ?
n = 143

n = 437
p = ? ... (23)
q = ? ... (19)
```

DLP – Discrete Logarithm Problem
(e.g. DSA)

```
a = 10
b = ?
b = 3^a mod 17 =  8

b = 3^a  mod 17 = 14
a = ? ... (9)
```

ECDLP – Elliptic Curve Discrete Logarithm Problem
(e.g. ECDSA)



difficulty and complexity:
class NP-intermediate
(NP-complete is not proven)

PQC vs. classic Cryptography

E-GROUP
SOFTWARE & BEYOND

```
Shor's algorithm

        n = 15                                              // to-be-factorized integer
        a = 7                                               // random number, where "a < n"

        Calculate "r" (period/order):

                f(x) = a^x mod n = f(x + r) = a^(x + r) mod n
                1. 7^1 mod 15 = 7
                2. 7^2 mod 15 = 4
                3. 7^3 mod 15 = 13
                4. 7^4 mod 15 = 1
                5. 7^5 mod 15 = 7                           // f(1) = f(5)
                6. 7^6 mod 15 = 4
                7. ...

        r = 4                                               // period/order
```

PQC vs. classic Cryptography

E-GROUP
SOFTWARE & BEYOND

```
Shor's algorithm

        Calculate "gcd(a^(r/2) +/- 1, n)" (greatest common divisor):

                p = gcd(48, 15) = ?
                q = gcd(50, 15) = ?

                //Euclidean algorithm:
                gcd(48, 15) => 48 = q[0] * 15 + r[0]          // q[0] = 3, r[0] = 3
                gcd(15, 3)  => 15 = q[1] * 3  + r[1]          // q[1] = 5, r[1] = 0
                The final non-zero remainder is r[0] = 3

                gcd(50, 15) => 50 = q[0] * 15 + r[0]          // q[0] = 3, r[0] = 5
                gcd(15, 5)  => 15 = q[1] * 5  + r[1]          // q[1] = 3, r[1] = 0
                The final non-zero remainder is r[0] = 5

        p = gcd(48, 15) = 3                                   // prime factor of "n = 15"
        q = gcd(50, 15) = 5                                   // prime factor of "n = 15"
```

PQC vs. classic Cryptography

E-GROUP
SOFTWARE & BEYOND

ETSI: Quantum Safe Cryptography v1.0.0 (2014-10)

„[...] symmetric key algorithms like AES that can be broken faster by a quantum computer running Grover's algorithm than by a classical computer. [...] This is to say that AES-128 is as difficult for a classical computer to break as AES-256 would be for a quantum computer."

Table 1 – Comparison of conventional and quantum security levels of some popular ciphers.

Algorithm	Key Length	Effective Key Strength / Security Level	
		Conventional Computing	Quantum Computing
RSA-1024	1024 bits	80 bits	0 bits
RSA-2048	2048 bits	112 bits	0 bits
ECC-256	256 bits	128 bits	0 bits
ECC-384	384 bits	256 bits	0 bits
AES-128	128 bits	128 bits	64 bits
AES-256	256 bits	256 bits	128 bits

Note : Effective key strength for conventional computing derived from NIST SP 800-57 "Recommendation for Key Management"

That gives us the intuition for why Grover's algorithm doesn't parallelize.

/ Filippo Valsorda, ex-Google/ex-Cloudflare, 2026,
Quantum Computers Are Not a Threat to 128-bit Symmetric Keys, https://words.filippo.io/128-bits /

PQC vs. classic Cryptography

E-GROUP
SOFTWARE & BEYOND

2017-12-04 – NIST PQC 69 submissions accepted
2022-07-05 - NIST PQC Winners (Round 3)
2024-08-13 – NIST FIPS 203 (ML-KEM based on CRYSTALS-KYBER)
2024-08-13 – NIST FIPS 204 (ML-DSA based on CRYSTALS-DILITHIUM)
2024-08-13 – NIST FIPS 205 (SLH-DSA based on SPHINCS+)
2025-03-11 – NIST PQC Winners (Round 4)

crypto libraries: OpenSSL 3.5, Java 24, Bouncy Castle 1.81, .NET 10.0 rc1, ...
CA softwares, HW device drivers, product/service certifications, ...
SSL/TLS protocol, blind signatures, homomorphic encryptions (CKKS => ISO/IEC DIS 28033-3), IoT adaptations ...

	Public-Key Encryption Key-Encapsulation Mechanism		Digital Signature	
	Winner	Round 4	Winner	Round 4
code-based		HQC		
lattice-based	CRYSTALS-KYBER		CRYSTALS-DILITHIUM FALCON	
isogeny-based				
multivariate				
zero-knowledge				
hash-based			SPHINCS+	

PQC - ML-KEM

E-GROUP
SOFTWARE & BEYOND

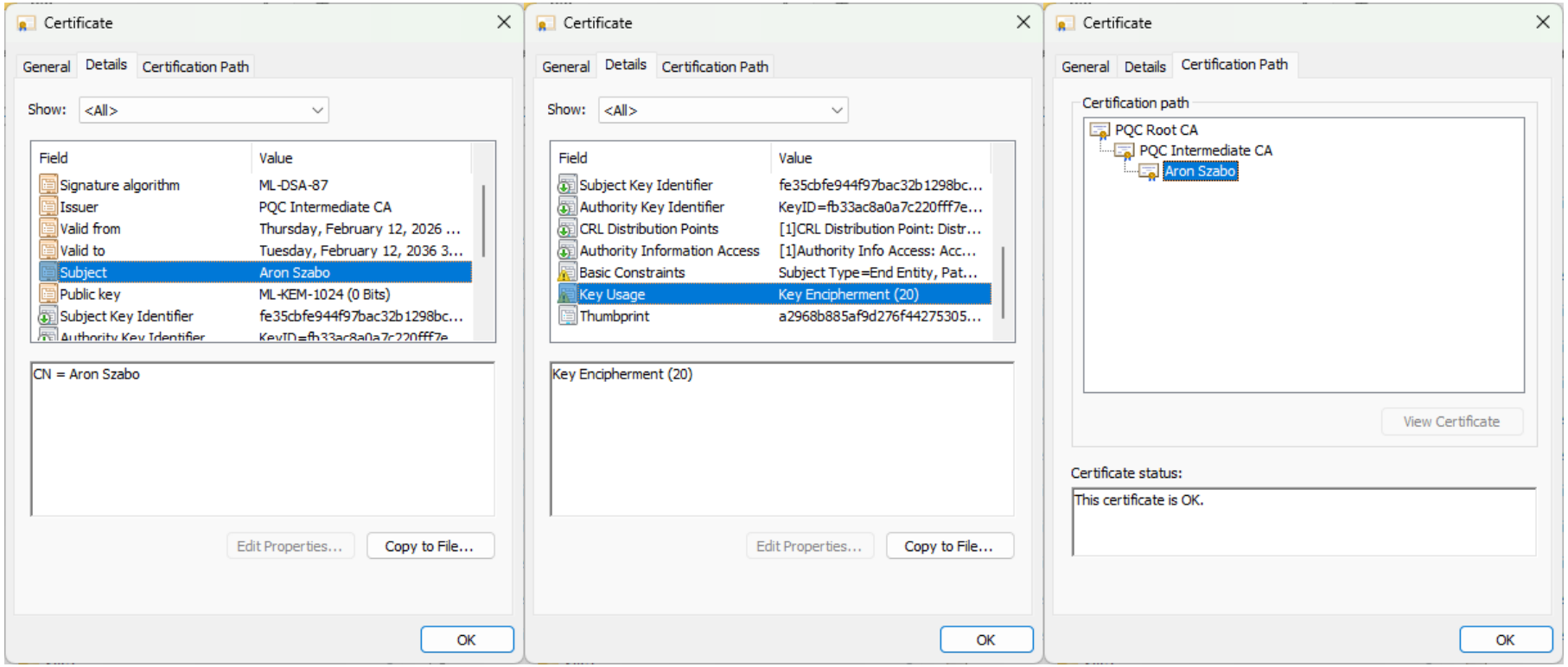
2024-08-13 - NIST FIPS 203 (ML-KEM based on CRYSTALS-KYBER)

strengths and sizes

```
mlkem512  = ML-KEM-512    - NIST Security Level 1 (~ AES-128)
mlkem768  = ML-KEM-768    - NIST Security Level 3 (~ AES-192)
mlkem1024 = ML-KEM-1024   - NIST Security Level 5 (~ AES-256)

  strength  | secret key | public key | ciphertext |  shared
------------+------------+------------+------------+---------
ML-KEM-512  | 1632 bytes |  800 bytes |  768 bytes | 32 bytes
ML-KEM-768  | 2400 bytes | 1184 bytes | 1088 bytes | 32 bytes
ML-KEM-1024 | 3168 bytes | 1568 bytes | 1568 bytes | 32 bytes
```

RSA 3072 ciphertext
384 bytes

HQC 256 ciphertext
14421 bytes



PQC - ML-KEM

E-GROUP
SOFTWARE & BEYOND

2024-08-13 – NIST FIPS 203 (ML-KEM based on CRYSTALS-KYBER)

experiences

1)
no asymmetric encryption, just key encapsulation (internal random for m_message, no external input)
/dev/qrandom ???, haveged/HAVEGE (HArdware Volatile Entropy Gathering and Expansion) ???

2)
deterministic = same message and same public key produces the same shared secret
(K_shared_secret, r_random_seed) <- HASH(m_message || HASH(ek_public_key))

3)
r_random_seed generation shall follow specification (m_message and ek_public_key)
otherwise nonce reuse may leak information about ciphertext (e.g. same K_shared_secret is used)
Common Criteria EAL4 evaluated codes are recommended

4)
SSL/TLS derived shared secret will not rely solely on ML-KEM
AES <- X25519MLKEM768 = Curve25519 ECDH + ML-KEM-768 (NIST Security Level 3)
AES <- SecP384r1MLKEM1024 = NIST P-384 ECDH + ML-KEM-1024 (NIST Security Level 5)

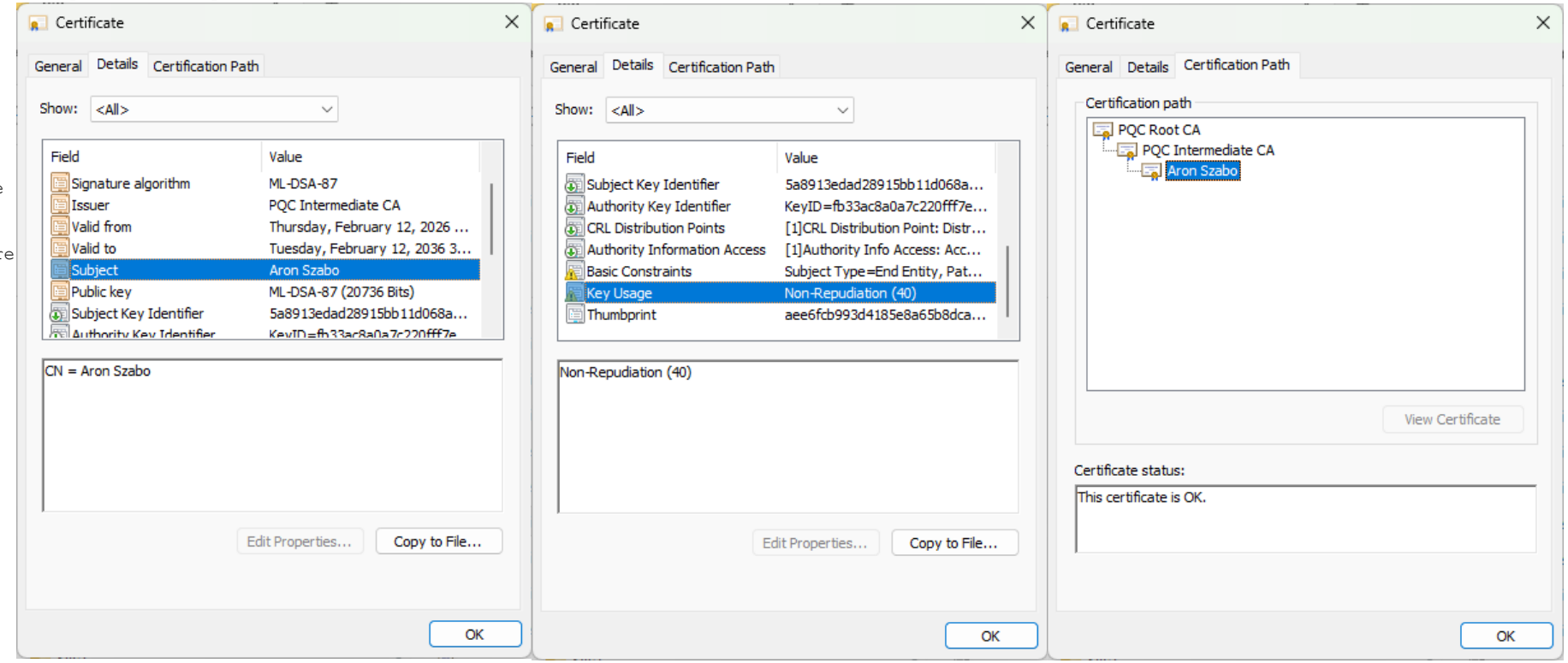
PQC - ML-DSA

E-GROUP
SOFTWARE & BEYOND

2024-08-13 - NIST FIPS 204 (ML-DSA based on CRYSTALS-DILITHIUM)

strengths and sizes

```
mldsa44   = ML-DSA-44     - NIST Security Level 2 (~ AES-128)
mldsa65   = ML-DSA-65     - NIST Security Level 3 (~ AES-192)
mldsa87   = ML-DSA-87     - NIST Security Level 5 (~ AES-256)

  strength   | secret key | public key | signature
-------------+------------+------------+-----------
ML-DSA-44    | 2560 bytes | 1312 bytes | 2420 bytes
ML-DSA-65    | 4032 bytes | 1952 bytes | 3309 bytes
ML-DSA-87    | 4896 bytes | 2592 bytes | 4627 bytes

RSA 3072 signature
384 bytes

ECC 256 signature (raw)
64 bytes
(NIST P-256, ECDSA)

FALCON padded 512 signature
666 bytes

FALCON padded 1024 signature
1280 bytes
```



PQC - ML-DSA

2024-08-13 – NIST FIPS 204 (ML-DSA based on CRYSTALS-DILITHIUM)

experiences

1)
one signature creation may require many iterations to meet all requirements (e.g. 7 iterations -> 1 signature)

2)
source of all uniqueness is y_random_seed based on
- the Kappa counter (starting from 0) and
- 32 bytes value (0x00 for "deterministic" and random for "hedged")

3)
y_random_seed generation shall follow specification (tr_public_key_hash and M'_message)
otherwise nonce reuse may leak information about secret key (s1_secret_value)
Common Criteria EAL4 evaluated codes are recommended

```
z1 = y + c1 x s1
z2 = y + c2 x s1
(z1 - z2) = (y + c1 x s1) - (y + c2 x s1)
s1 = (z1 - z2)/(c1 - c2)
```

4)
"internal μ" or "external μ" – calculation of HASH(tr_public_key_hash || M'_message)
performance differences exist whether hashing is performed internally or externally

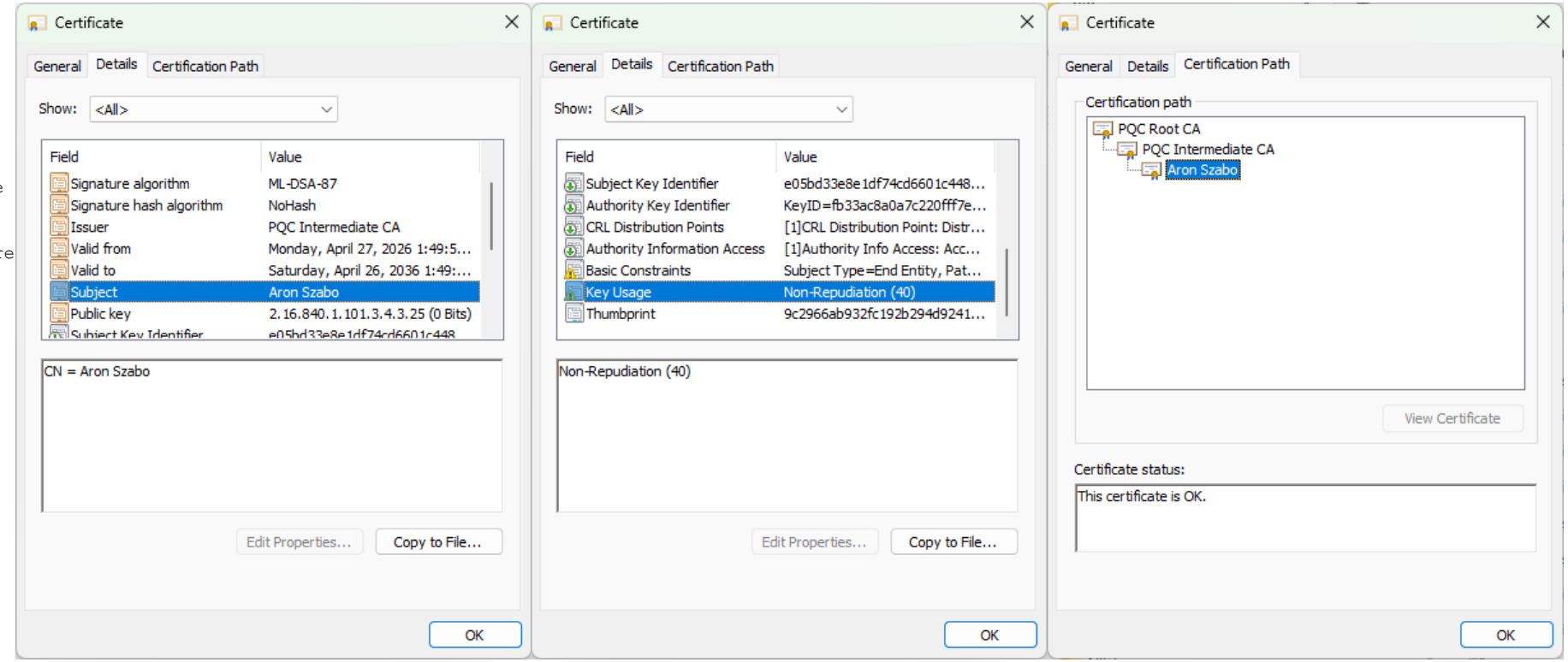
PQC - SLH-DSA

E-GROUP
SOFTWARE & BEYOND

2024-08-13 - NIST FIPS 205 (SLH-DSA based on SPHINCS+)

strengths and sizes

```
SLH-DSA-SHA2/SHAKE-128s/f - NIST Security Level 1 (~ AES-128)
SLH-DSA-SHA2/SHAKE-192s/f - NIST Security Level 3 (~ AES-192)
SLH-DSA-SHA2/SHAKE-256s/f - NIST Security Level 5 (~ AES-256)

  strength   | secret key | public key |     signature      |    hash     | small/fast
-------------+------------+------------+--------------------+-------------+-----------
SLH-DSA-128 |    64 bytes |   32 bytes |  7856/17088 bytes | SHA2/SHAKE |    s/f
SLH-DSA-192 |    96 bytes |   48 bytes | 16224/35664 bytes | SHA2/SHAKE |    s/f
SLH-DSA-256 |  128 bytes |   64 bytes | 29792/49856 bytes | SHA2/SHAKE |    s/f

RSA 3072 signature
384 bytes

ECC 256 signature (raw)
64 bytes
(NIST P-256, ECDSA)

FALCON padded 512 signature
666 bytes

FALCON padded 1024 signature
1280 bytes
```



PQC - SLH-DSA

E-GROUP
SOFTWARE & BEYOND

2024-08-13 – NIST FIPS 205 (SLH-DSA based on SPHINCS+)

experiences

1)
no signature algorithm (no NP-hard problem), just hash algorithm

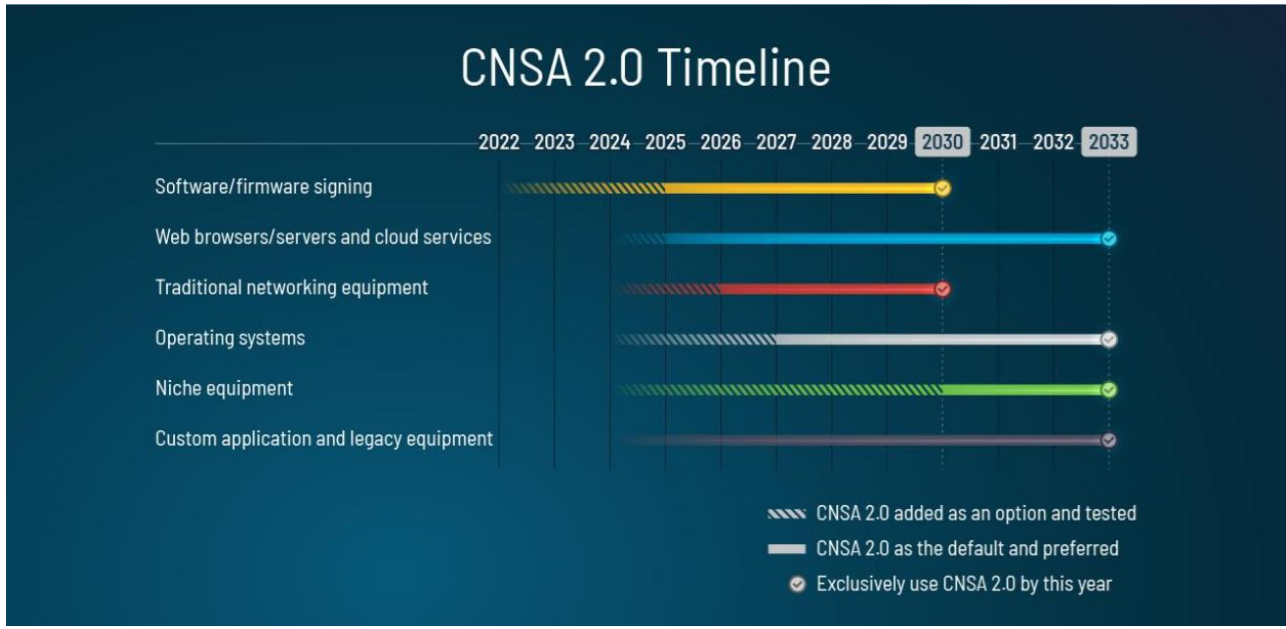
2)
small signature size
SLH-DSA-SHA2-256s = 29792 bytes
SLH-DSA-SHA2-256f = 49856 bytes
RSA 3072 = 384 bytes
ECC 256 = 64 bytes

3)
fast signature creation (on average Intel Core i7 or AMD Ryzen 7)
SLH-DSA-SHA2-256f = 300 ms
SLH-DSA-SHA2-256s = 5000 ms
RSA 3072 = 5 ms
ECC 256 = 0,1 ms

=> human scenarios, code signing, archiving

PQC transition

2022-09-07
NSA (National Security Agency)
Announcing the Commercial National Security Algorithm Suite 2.0
deadline: 2030/2033



2026-03-25
Google
Quantum frontiers may be closer than they appear
deadline: 2029

PQC transition

E-GROUP
SOFTWARE & BEYOND

For example, our simulation of the point addition quantum circuit for the NIST standardized curve P-256 needs 2330 logical qubits [...]. In comparison, Shor's factoring algorithm for a 3072-bit modulus needs 6146 qubits [...].
/ Microsoft, 2017, Quantum Resource Estimates for Computing Elliptic Curve Discrete Logarithms, https://eprint.iacr.org/2017/598.pdf /

In this work, we propose FTQC architectures based on reconfigurable atom arrays [...] that Shor's algorithm can be implemented at cryptographically relevant scales using as few as 10,000 atomic qubits [...] our most time-efficient architectures can potentially enable runtimes of 10 days for ECC-256 with ≈ 26,000 qubits, and 97 days for RSA-2048 with ≈ 102,000 qubits [...] FIG. 3. Resources required for Shor's algorithm [...] RSA-2048 6,144 log. qubits [...] ECC-256 1,450 log. qubits
/ Caltech/Oratomic, 2026, Shor's algorithm is possible with as few as 10,000 reconfigurable atomic qubits, https://arxiv.org/pdf/2603.28627 /

we do not expect meaningful scaling challenges between a quantum computer with 1200 logical qubits and one with 1450, so, [...] we assume that first-generation fast-clock CRQCs may be able to solve ECDLP on secp256k1 and similar elliptic curves in about 9 minutes on average.
/ Google, 2026, Securing Elliptic Curve Cryptocurrencies against Quantum Vulnerabilities: Resource Estimates and Mitigations, https://quantumai.google/static/site-assets/downloads/cryptocurrency-whitepaper.pdf /

=> affects IoT signature layers and other systems where ECDSA is applied

PQC in legislation

2022-07-01 - 2024-12-31: Act L of 2013
2025-01-01 - ... : Act LXIX of 2024

Application of PQC is mandatory to
- SP/RP itself (e.g. government),
- client/customer of SP/RP (e.g. government).

Application of PQC is validated by
- 3rd party auditor.

Application of PQC ensures
- Confidentiality of data (encryption),
- Integrity, Authenticity of data (signature). ???

Application of PQC affects
- SSL/TLS protocol,
- other protocols (end-to-end encrypted data in JWE at OIDC or in XML Encryption at SAML). ???

2013. évi L. törvény

Hatályos: 2022. 07. 01. – 2022. 12. 19.

49.[24] *poszt-kvantumtitkosítás:* a matematikailag valószínűsíthetően igazolható, kvantumszámítógép által megvalósított támadás ellen a hagyományos kriptográfiai alkalmazáson felüli poszt-kvantum alkalmazást, illetve megoldást nyújtó titkosítás, amely során a két végpont közötti kommunikáció felhasználásával, az adatátvitellel megosztott kulcsot hoz létre a két végfelhasználó között, anélkül, hogy a kulcsot jogosulatlan harmadik fél megismerné;

50.[25] *poszt-kvantumtitkosítás alkalmazásra kötelezett szervezet:* a Szabályozott Tevékenységek Felügyeleti Hatósága (a továbbiakban: SZTFH) elnökének rendeletében meghatározott,

a) a kormányzati célú hálózatokról szóló kormányrendelet szerinti igénybevételre kötelezett szervezet,

b) a hitelintézetekről és a pénzügyi vállalkozásokról szóló törvény szerinti bank, valamint

c) az 1. mellékletben meghatározott törvények hatálya alá tartozó közműszolgáltató és az 1. mellékletben meghatározott törvények felhatalmazása alapján kiadott jogszabályok hatálya alá tartozó közszolgáltatást nyújtó szervezet.

2024. évi LXIX. törvény

Közlönyállapot 2025. 01. 01.

(8) E törvény poszt-kvantumtitkosításra vonatkozó rendelkezéseit a Szabályozott Tevékenységek Felügyeleti Hatósága (a továbbiakban: SZTFH) elnökének rendeletében meghatározott, a következő szervezetekre (a továbbiakban: poszt-kvantumtitkosítás alkalmazására kötelezett szervezet) és hatósági felügyeletükre irányuló tevékenységekkell alkalmazni:

a) a kormányzati célú hálózatokról szóló kormányrendelet szerinti igénybevételre kötelezett szervezet, valamint

b) a következő törvények hatálya alá tartozó közműszolgáltató és a következő törvények felhatalmazása alapján kiadott jogszabályok hatálya alá tartozó, közszolgáltatást nyújtó szervezet:

ba) a földgázellátásról szóló törvény,

bb) a földgáz biztonsági készletezéséről szóló törvény,

bc) a villamos energiáról szóló törvény,

bd) a távhőszolgáltatásról szóló törvény,

be) a víziközmű-szolgáltatásról szóló törvény, valamint

bf) a hulladékról szóló törvény.