The migration to post-quantum cryptography

Peter Schwabe

Max Planck Institute for Security and Privacy

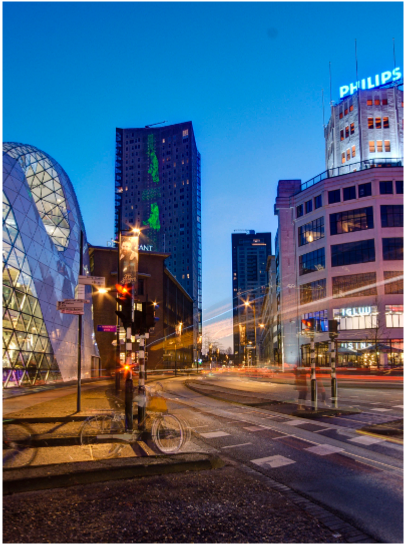
August 29, 2025

About me

- **2001–2007: Aachen**
  Studied Computer Science (Diplom)

About me

- **2001–2007: Aachen**
  Studied Computer Science (Diplom)
- **2008–2011: Eindhoven**
  Ph.D. in Department of Mathematics



About me

- 2001–2007: Aachen
  Studied Computer Science (Diplom)
- 2008–2011: Eindhoven
  Ph.D. in Department of Mathematics
- 2011–2012: Taipei
  Postdoc at Academia Sinica and NTU

About me

- **2001–2007: Aachen**
  Studied Computer Science (Diplom)
- **2008–2011: Eindhoven**
  Ph.D. in Department of Mathematics
- **2011–2012: Taipei**
  Postdoc at Academia Sinica and NTU
- **Since 2013: Nijmegen**
  From Assistant to Full Professor

Since Sep. 2020: MPI-SP

- Located in **Bochum**
- Founded in 2019
- Currently 13 PIs

- Aim to have
  - 6 Departments
  - 12 Research Groups
  - Around 250 people total

Since Sep. 2020: MPI-SP

[A small demo]

DLP and factoring

Discrete Logarithms

- X25519 is Diffie-Hellman key exchange
- (More specifically, elliptic-curve DH)
- Relies on hardness of **discrete-logarithm problem (DLP)**
- Also signature algorithms from (EC)DLP: DSA, ECDSA, EdDSA

DLP and factoring

Discrete Logarithms

- X25519 is Diffie-Hellman key exchange
- (More specifically, elliptic-curve DH)
- Relies on hardness of **discrete-logarithm problem (DLP)**
- Also signature algorithms from (EC)DLP: DSA, ECDSA, EdDSA

Factoring

- RSA is “Rivest-Shamir-Adleman” signatures (or encryption)
- Relies on hardness of **factoring** large integers

DLP and factoring

Discrete Logarithms

- X25519 is Diffie-Hellman key exchange
- (More specifically, elliptic-curve DH)
- Relies on hardness of **discrete-logarithm problem (DLP)**
- Also signature algorithms from (EC)DLP: DSA, ECDSA, EdDSA

Factoring

- RSA is “Rivest-Shamir-Adleman” signatures (or encryption)
- Relies on hardness of **factoring** large integers

- Most of today’s key agreement and signatures use (EC)DLP or factoring-based schemes

DLP and factoring

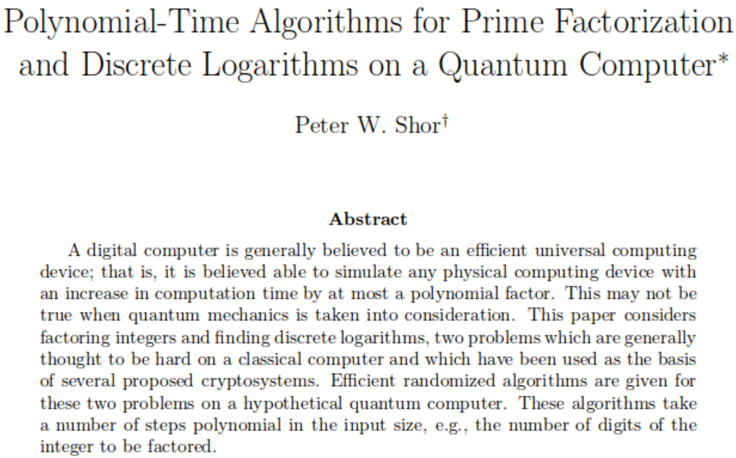
Discrete Logarithms

- X25519 is Diffie-Hellman key exchange
- (More specifically, elliptic-curve DH)
- Relies on hardness of **discrete-logarithm problem (DLP)**
- Also signature algorithms from (EC)DLP: DSA, ECDSA, EdDSA

Factoring

- RSA is “Rivest-Shamir-Adleman” signatures (or encryption)
- Relies on hardness of **factoring** large integers

- Most of today’s key agreement and signatures use (EC)DLP or factoring-based schemes
- DLP and Factoring are related → we have a **crypto monoculture**

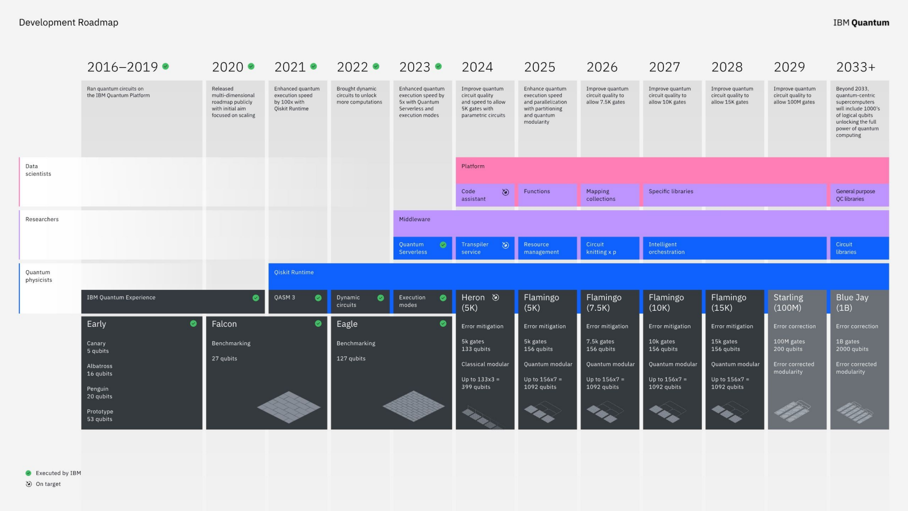


Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*

Peter W. Shor†

Abstract

A digital computer is generally believed to be an efficient universal computing device; that is, it is believed able to simulate any physical computing device with an increase in computation time by at most a polynomial factor. This may not be true when quantum mechanics is taken into consideration. This paper considers factoring integers and finding discrete logarithms, two problems which are generally thought to be hard on a classical computer and which have been used as the basis of several proposed cryptosystems. Efficient randomized algorithms are given for these two problems on a hypothetical quantum computer. These algorithms take a number of steps polynomial in the input size, e.g., the number of digits of the integer to be factored.



Development Roadmap

IBM **Quantum**

	2016–2019 ✓	2020 ✓	2021 ✓	2022 ✓	2023 ✓	2024	2025	2026	2027	2028	2029	2033+
	Ran quantum circuits on the IBM Quantum Platform	Released multi-dimensional roadmap publicly with initial aim focused on scaling	Enhanced quantum execution speed by 100x with Qiskit Runtime	Brought dynamic circuits to unlock more computations	Enhanced quantum execution speed by 5x with Quantum Serverless and execution modes	Improve quantum circuit quality and speed to allow 5K gates with parametric circuits	Enhance quantum execution speed and parallelization with partitioning and quantum modularity	Improve quantum circuit quality to allow 7.5K gates	Improve quantum circuit quality to allow 10K gates	Improve quantum circuit quality to allow 15K gates	Improve quantum circuit quality to allow 100M gates	Beyond 2033, quantum-centric supercomputers will include 1000's of logical qubits unlocking the full power of quantum computing
Data scientists						Platform						
						Code assistant ◎	Functions	Mapping collections	Specific libraries			General purpose QC libraries
Researchers					Middleware							
					Quantum Serverless ✓	Transpiler service ◎	Resource management	Circuit knitting x p	Intelligent orchestration			Circuit libraries
Quantum physicists			Qiskit Runtime									
	IBM Quantum Experience ✓		QASM 3 ✓	Dynamic circuits ✓	Execution modes ✓	Heron (5K) ◎ Error mitigation 5k gates 133 qubits Classical modular Up to 133x3 = 399 qubits	Flamingo (5K) Error mitigation 5k gates 156 qubits Quantum modular Up to 156x7 = 1092 qubits	Flamingo (7.5K) Error mitigation 7.5k gates 156 qubits Quantum modular Up to 156x7 = 1092 qubits	Flamingo (10K) Error mitigation 10k gates 156 qubits Quantum modular Up to 156x7 = 1092 qubits	Flamingo (15K) Error mitigation 15k gates 156 qubits Quantum modular Up to 156x7 = 1092 qubits	Starling (100M) Error correction 100M gates 200 qubits Error corrected modularity	Blue Jay (1B) Error correction 1B gates 2000 qubits Error corrected modularity
	Early ✓ Canary 5 qubits Albatross 16 qubits Penguin 20 qubits Prototype 53 qubits	Falcon ✓ Benchmarking 27 qubits		Eagle ✓ Benchmarking 127 qubits								

✓ Executed by IBM

◎ On target

See **https://www.ibm.com/quantum/blog/ibm-quantum-roadmap-2025**

“Our conservative estimate is that cryptographically relevant quantum computers are likely to be available within 16 years.”

—BSI: The status of quantum computer development, Jan. 2025

Post-quantum crypto (PQC)

Definition

Post-quantum crypto is (asymmetric) crypto that resists attacks using classical *and quantum* computers.

Post-quantum crypto (PQC)

Definition

Post-quantum crypto is (asymmetric) crypto that resists attacks using classical *and quantum* computers.

5 main directions

- Lattice-based crypto (PKE and Sigs)
- Code-based crypto (mainly PKE)
- Multivariate-based crypto (mainly Sigs)
- Hash-based signatures (only Sigs)
- Isogeny-based crypto (it’s complicated. . .)

Should you care now?

“Harvest now, decrypt later”

https://en.wikipedia.org/wiki/Utah_Data_Center#/media/File:EFF_photograph_of_NSA's_Utah_Data_Center.jpg

Should you care now?

"Harvest now, decrypt later"

https://en.wikipedia.org/wiki/Utah_Data_Center#/media/File:EFF_photograph_of_NSA's_Utah_Data_Center.jpg

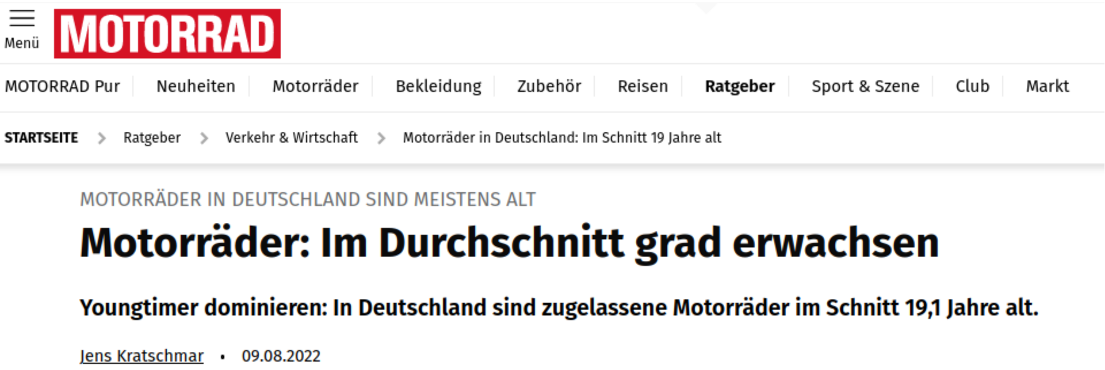
Mosca's theorem

$$X + Y > Z$$

- X: For how long do you need encrypted data to be secure?
- Y: How long does it take you to migrate to PQC
- Z: Time it will take to build a cryptographically relevant quantum computer

If $X + Y > Z$, you should worry.

Should you care now? (part II)



MOTORRÄDER IN DEUTSCHLAND SIND MEISTENS ALT

Motorräder: Im Durchschnitt grad erwachsen

Youngtimer dominieren: In Deutschland sind zugelassene Motorräder im Schnitt 19,1 Jahre alt.

Jens Kratschmar • 09.08.2022

The NIST PQC “not-a-competition”

- Inspired by two earlier NIST crypto competitions:
  - AES, running from 1997 to 2000
  - SHA3, running from 2007 to 2012

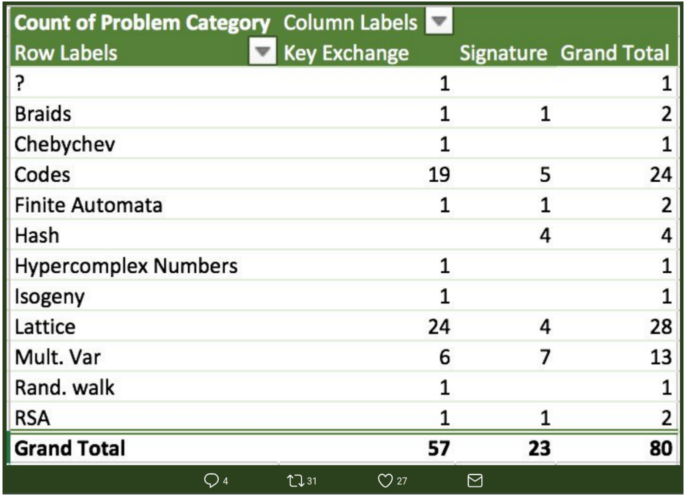
The NIST PQC “not-a-competition”

- Inspired by two earlier NIST crypto competitions:
  - AES, running from 1997 to 2000
  - SHA3, running from 2007 to 2012
- Approach: NIST specifies criteria, everybody is welcome to submit proposals
- Selection through an open process and multiple rounds
- Actual decisions are being made by NIST

The NIST PQC “not-a-competition”

- Inspired by two earlier NIST crypto competitions:
  - AES, running from 1997 to 2000
  - SHA3, running from 2007 to 2012
- Approach: NIST specifies criteria, everybody is welcome to submit proposals
- Selection through an open process and multiple rounds
- Actual decisions are being made by NIST
- PQC project:
  - Announcement: Feb 2016
  - Call for proposals: Dec 2016 (based on community input)
  - Deadline for submissions: Nov 2017

NIST PQC – how it started (Nov. 2017)



Count of Problem Category	Column Labels		
Row Labels	Key Exchange	Signature	Grand Total
?	1		1
Braids	1	1	2
Chebychev	1		1
Codes	19	5	24
Finite Automata	1	1	2
Hash		4	4
Hypercomplex Numbers	1		1
Isogeny	1		1
Lattice	24	4	28
Mult. Var	6	7	13
Rand. walk	1		1
RSA	1	1	2
Grand Total	**57**	**23**	**80**

Overview tweeted by Jacob Alperin-Sheriff on Dec 4, 2017.

NIST PQC – first results (Jul. 2022)

4 schemes selected for standardization

- **CRYSTALS-Kyber**: lattice-based key agreement
- **CRYSTALS-Dilithium**: lattice-based signatures
- Falcon: lattice-based signatures
- **SPHINCS$^+$**: hash-based signatures

4 schemes advanced to round 4

- Classic McEliece: code-based key agreement
- BIKE: code-based key agreement
- HQC: code-based key agreement
- SIKE: isogeny-based key agreement (✝ 30.07.2022)

NIST PQC – first results (Jul. 2022)

4 schemes selected for standardization

- **CRYSTALS-Kyber**: lattice-based key agreement
- **CRYSTALS-Dilithium**: lattice-based signatures
- Falcon: lattice-based signatures
- **SPHINCS**$^+$: hash-based signatures

4 schemes advanced to round 4

- Classic McEliece: code-based key agreement
- BIKE: code-based key agreement
- HQC: code-based key agreement
- SIKE: isogeny-based key agreement (**†** 30.07.2022)
- **Additionally (June 2023):** 40 new signature submissions

The SIKE shock

Castryck, Decru, 2022: *An efficient key recovery attack on SIDH*

The SIKE shock

Castryck, Decru, 2022: *An efficient key recovery attack on SIDH*

- SIDH was “A decade unscathed” (Craig Costello, ePrint 2021/543)

The SIKE shock

Castryck, Decru, 2022: *An efficient key recovery attack on SIDH*

- SIDH was “A decade unscathed” (Craig Costello, ePrint 2021/543)
- SIKE **lowered** parameters during NIST PQC (following Jaques, Schanck: *Quantum cryptanalysis in the RAM model: Claw-finding attacks on SIKE* (ePrint 2019/103))

The SIKE shock

Castryck, Decru, 2022: *An efficient key recovery attack on SIDH*

- SIDH was “A decade unscathed” (Craig Costello, ePrint 2021/543)
- SIKE **lowered** parameters during NIST PQC (following Jaques, Schanck: *Quantum cryptanalysis in the RAM model: Claw-finding attacks on SIKE* (ePrint 2019/103))
- Competent, smart people tried to break it (e.g., Martindale, Panny: *How to not break SIDH* (ePrint 2019/558))

The SIKE shock

Castryck, Decru, 2022: *An efficient key recovery attack on SIDH*

- SIDH was “A decade unscathed” (Craig Costello, ePrint 2021/543)
- SIKE **lowered** parameters during NIST PQC (following Jaques, Schanck: *Quantum cryptanalysis in the RAM model: Claw-finding attacks on SIKE* (ePrint 2019/103))
- Competent, smart people tried to break it (e.g., Martindale, Panny: *How to not break SIDH* (ePrint 2019/558))

Yet, **full break without any “warning”**

More NIST PQC

- First three standards released in August 2024:
  - ML-KEM (CRYSTALS-Kyber)
  - ML-DSA (CRYSTALS-Dilithium)
  - SLH-DSA (SPHINCS^{+})
- October 2024: 14 on-ramp signatures advanced to round 2
- March 2025: HQC selected for standardization (concludes round 4)
- FN-DSA (Falcon) standard draft almost ready

Summary of NIST PQC (so far)

Key agreements standards

- **ML-KEM**
- HQC

Signature standards

- **ML-DSA**
- **SLH-DSA**
- FN-DSA

Summary of NIST PQC (so far)

“The public-key encryption and key-establishment algorithm that will be standardized is CRYSTALS-KYBER. The digital signatures that will be standardized are CRYSTALS-Dilithium, FALCON, and SPHINCS$^+$. While there are multiple signature algorithms selected, NIST recommends CRYSTALS-Dilithium as the primary algorithm to be implemented”

—NIST IR 8413-upd1

BSI recommendations

Key agreements standards

- ML-KEM
- Classic McEliece (code-based, in standardization by ISO)
- FrodoKEM (lattice-based, in standardization by ISO)

Signature standards

- ML-DSA
- SLH-DSA
- XMSS and LMS (*stateful*, also standardized by IETF & NIST)

> *“Post-quantum schemes should only be used in combination with classical schemes (“hybrid”) if possible.”*
>
> —Recommendations by the BSI

https://www.bsi.bund.de/EN/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Quantentechnologien-und-Post-Quanten-Kryptografie/quantentechnologien-und-post-quanten-kryptografie_node.html

Motivation for hybrid deployments

Don’t make systems less secure in the attempt to make them more secure against future quantum attackers!

Motivation for hybrid deployments

Don’t make systems less secure in the attempt to make them more secure against future quantum attackers!

- Cryptanalysis of PQ schemes is not as stable as for ECC
  - SIKE. . . (was deployed, **hybrid**, by Google and Cloudflare)
  - Late breaks of GeMSS and Rainbow

Motivation for hybrid deployments

Don’t make systems less secure in the attempt to make them more secure against future quantum attackers!

- Cryptanalysis of PQ schemes is not as stable as for ECC
  - SIKE. . . (was deployed, **hybrid**, by Google and Cloudflare)
  - Late breaks of GeMSS and Rainbow
- Implementation security of PQ schemes is not as mature as for ECC
  - Side-channel protection for ECC based on rich algebraic structure
  - For lattices: mostly masking + shuffling
  - Continued successful SCA against *protected* implementations
  - Compilers screwing with code in new ways (“Kyberslash”)

Isn’t hybrid to expensive?

Computational complexity

- Today’s systems use ECC
- ML-KEM is about as costly as ECC
- Hybrid costs about 2× slowdown

Isn’t hybrid to expensive?

Computational complexity

- Today’s systems use ECC
- ML-KEM is about as costly as ECC
- Hybrid costs about $2\times$ slowdown
- Argument needs some more care with HW acceleration
- Anyway already have ECC
- Anyway will need PQC

Isn’t hybrid to expensive?

Computational complexity

- Today’s systems use ECC
- ML-KEM is about as costly as ECC
- Hybrid costs about 2× slowdown
- Argument needs some more care with HW acceleration
- Anyway already have ECC
- Anyway will need PQC

Sizes

- PQC cryptographic objects are much bigger than for ECC
- X25519 PK: 32 B
- Additing 32 Bytes to 1KB makes a small difference

Hybrid and the NIST competition

"NIST recognizes that some users may wish to deploy systems that use "hybrid modes," which combine post-quantum cryptographic algorithms with existing cryptographic algorithms (which may not be post-quantum). These "hybrid modes" are outside of the scope of this document, which is focused on post-quantum cryptographic algorithms only.

—NIST PQC Call for Proposals, 2016

Hybrid and the NIST competition

> *“NIST recognizes that some users may wish to deploy systems that use “hybrid modes,” which combine post-quantum cryptographic algorithms with existing cryptographic algorithms (which may not be post-quantum). These “hybrid modes” are outside of the scope of this document, which is focused on post-quantum cryptographic algorithms only.*
>
> —NIST PQC Call for Proposals, 2016

Consequences

- Reduce complexity and probably discussions

Hybrid and the NIST competition

“NIST recognizes that some users may wish to deploy systems that use “hybrid modes,” which combine post-quantum cryptographic algorithms with existing cryptographic algorithms (which may not be post-quantum). These “hybrid modes” are outside of the scope of this document, which is focused on post-quantum cryptographic algorithms only.

—NIST PQC Call for Proposals, 2016

Consequences

- Reduce complexity and probably discussions
- Non-mandatory hybrid deployment lead to other discussions:
  - Long discussions if Kyber512 meets level-1 security
  - No question if Kyber512+X25519 meets level-1 security

Hybrid and the NIST competition

> *"NIST recognizes that some users may wish to deploy systems that use "hybrid modes," which combine post-quantum cryptographic algorithms with existing cryptographic algorithms (which may not be post-quantum). These "hybrid modes" are outside of the scope of this document, which is focused on post-quantum cryptographic algorithms only.*
>
> —NIST PQC Call for Proposals, 2016

Consequences

- Reduce complexity and probably discussions
- Non-mandatory hybrid deployment lead to other discussions:
  - Long discussions if Kyber512 meets level-1 security
  - No question if Kyber512+X25519 meets level-1 security
- For targeted hybrid deployment, designs could have (and would have!) made other choices

Summary so far

- There are several standards for PQC
- There are existing implementations, integrated in libraries

Summary so far

- There are several standards for PQC
- There are existing implementations, integrated in libraries
- Need to add these schemes to classical schemes (hybrid)
- Update protocols, applications, systems

Summary so far

- There are several standards for PQC
- There are existing implementations, integrated in libraries
- Need to add these schemes to classical schemes (hybrid)
- Update protocols, applications, systems

How hard can this be?

[Answer 1: Back to our demo]

Answer 1: more positive examples

- Signal is using PQC (ML-KEM) since 2023

Answer 1: more positive examples

- Signal is using PQC (ML-KEM) since 2023
- Apple's iMessage uses PQC (ML-KEM)

Answer 1: more positive examples

- Signal is using PQC (ML-KEM) since 2023
- Apple’s iMessage uses PQC (ML-KEM)
- AWS is using PQC “across several key services”

Answer 1: more positive examples

- Signal is using PQC (ML-KEM) since 2023
- Apple’s iMessage uses PQC (ML-KEM)
- AWS is using PQC “across several key services”
- Certified Infineon smartcard supports ML-KEM

Answer 1: more positive examples

- Signal is using PQC (ML-KEM) since 2023
- Apple’s iMessage uses PQC (ML-KEM)
- AWS is using PQC “across several key services”
- Certified Infineon smartcard supports ML-KEM
- OpenTitan supports SLH-DSA for secure boot

Answer 1: more positive examples

- Signal is using PQC (ML-KEM) since 2023
- Apple’s iMessage uses PQC (ML-KEM)
- AWS is using PQC “across several key services”
- Certified Infineon smartcard supports ML-KEM
- OpenTitan supports SLH-DSA for secure boot
- Automotive industry starts using PQC for software updates

Answer 2 – A bit of history: the case of MD5

- MD5 is a cryptographic hash function
- Hash functions are used as building blocks all over the place

Answer 2 – A bit of history: the case of MD5

- MD5 is a cryptographic hash function
- Hash functions are used as building blocks all over the place
- **1991**: MD5 is proposed by Rivest

Answer 2 – A bit of history: the case of MD5

- MD5 is a cryptographic hash function
- Hash functions are used as building blocks all over the place
- **1991**: MD5 is proposed by Rivest
- **1993**: Collisions in MD5 compression function (den Boer, Bosselaers)

Answer 2 – A bit of history: the case of MD5

- MD5 is a cryptographic hash function
- Hash functions are used as building blocks all over the place
- **1991**: MD5 is proposed by Rivest
- **1993**: Collisions in MD5 compression function
  (den Boer, Bosselaers)
- **1996**: Dobbertin, Bosselaers, Preneel: concerns about MD5

Answer 2 – A bit of history: the case of MD5

- MD5 is a cryptographic hash function
- Hash functions are used as building blocks all over the place
- **1991**: MD5 is proposed by Rivest
- **1993**: Collisions in MD5 compression function (den Boer, Bosselaers)
- **1996**: Dobbertin, Bosselaers, Preneel: concerns about MD5
- **2004**: Wang presents MD5 collisions

Answer 2 – A bit of history: the case of MD5

- MD5 is a cryptographic hash function
- Hash functions are used as building blocks all over the place
- **1991**: MD5 is proposed by Rivest
- **1993**: Collisions in MD5 compression function (den Boer, Bosselaers)
- **1996**: Dobbertin, Bosselaers, Preneel: concerns about MD5
- **2004**: Wang presents MD5 collisions
- **2008**: ***Rogue CA certificate*** using MD5 (Sotirov, Stevens, Appelbaum, Lenstra, Molnar, Osvik, de Weger)

Answer 2 – A bit of history: the case of MD5

- MD5 is a cryptographic hash function
- Hash functions are used as building blocks all over the place
- **1991**: MD5 is proposed by Rivest
- **1993**: Collisions in MD5 compression function (den Boer, Bosselaers)
- **1996**: Dobbertin, Bosselaers, Preneel: concerns about MD5
- **2004**: Wang presents MD5 collisions
- **2008**: ***Rogue CA certificate*** using MD5 (Sotirov, Stevens, Appelbaum, Lenstra, Molnar, Osvik, de Weger)
- **2012**: Flame malware exploits MD5 weaknesses

Answer 2 – A bit of history: the case of MD5

- MD5 is a cryptographic hash function
- Hash functions are used as building blocks all over the place
- **1991**: MD5 is proposed by Rivest
- **1993**: Collisions in MD5 compression function (den Boer, Bosselaers)
- **1996**: Dobbertin, Bosselaers, Preneel: concerns about MD5
- **2004**: Wang presents MD5 collisions
- **2008**: ***Rogue CA certificate*** using MD5 (Sotirov, Stevens, Appelbaum, Lenstra, Molnar, Osvik, de Weger)
- **2012**: Flame malware exploits MD5 weaknesses

Replacing MD5 was “easy”!

What lessons can we learn from this?

1. **Migrating some (many?) applications to PQC is easy.**
   - You are already using PQC, possibly without knowing!
   - Deployment via system updates

What lessons can we learn from this?

1. Migrating some (many?) applications to PQC is easy.
   - You are already using PQC, possibly without knowing!
   - Deployment via system updates
2. Migrating *all* applications to PQC is hard.
   - Requires careful inventory
   - Cryptographic bill of materials (CBOM)
   - Might require replacing appliances that are not updatable

What lessons can we learn from this?

1. Migrating some (many?) applications to PQC is easy.
   - You are already using PQC, possibly without knowing!
   - Deployment via system updates
2. Migrating *all* applications to PQC is hard.
   - Requires careful inventory
   - Cryptographic bill of materials (CBOM)
   - Might require replacing appliances that are not updatable

Creating a CBOM and “easy wins” can (and should!) be done in parallel

EU PQC migration guidelines

1. By the end of 2026:
   - First steps implemented
   - PQC transition planning and pilots for high- and medium-risk use cases initiated
2. By the end of 2030:
   - PQC transition for high-risk use cases completed
   - PQC transition planning and pilots for medium-risk use cases completed
   - Quantum-safe software and firmware upgrades enabled by default
3. By the end of 2035:
   - PQC transition for medium-risk use cases completed
   - PQC transition for low-risk use cases completed as much as feasible

https://digital-strategy.ec.europa.eu/en/library/coordinated-implementation-roadmap-transition-post-quantum-cryptography

Start “playing” with PQC

SSH

- OpenSSH 10.0 uses MLKEM768-X25519 as default key agreement
- Released in April 2025
- Already in Debian stable (trixie)

Start “playing” with PQC

HTTPS (nginx+OpenSSL)

- OpenSSL 3.5 has support for MLKEM768-X25519
- Released in April 2025
- Already in Debian stable (trixie)
- Instructions for setting up NGINX (can probably skip compilation from source): `https://www.linode.com/docs/guides/post-quantum-encryption-nginx-ubuntu2404/`
- Client-side supported by all major browsers

Start “playing” with PQC

Post-quantum VPN on top of WireGuard

https://rosenpass.eu