Quantum Computing Doomsday Planning

JP Aumasson
Farida Aclimandos

TAURUS

BLACKALPS

About us

JP Aumasson

Taurus cofounder and CSO

Was here when it as called “AppSec”

Farida Aclimandos

EPFL masters, ex Taurus intern

First conference talk!

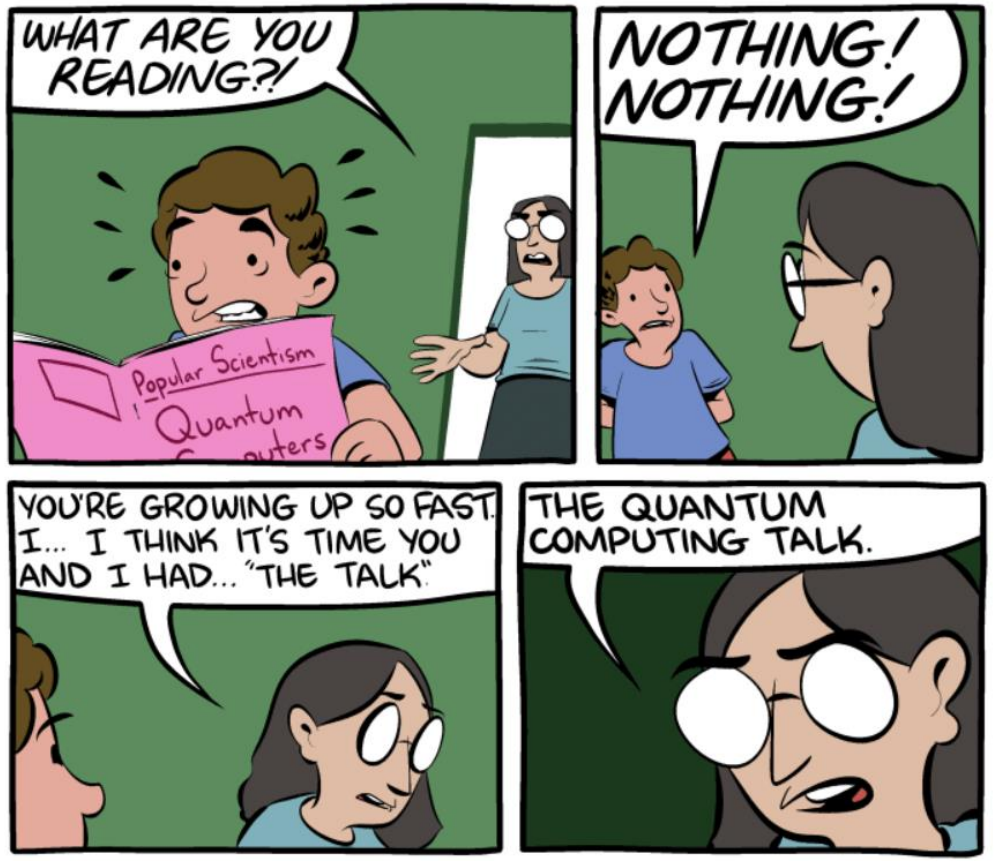
1. Introduction. *You have nothing to do but mention the quantum theory, and people will take your voice for the voice of science, and believe anything.*

—Bernard Shaw, *Geneva* (1938)

The Plan

1. Quantum Computing
2. Post-Quantum Cryptography
3. Attacking Real Protocols
4. Solutions Available

1. Quantum Computing



Why Quantum Computers?

Simulating Physics with Computers

Richard P. Feynman

Department of Physics, California Institute of Technology, Pasadena, California 91107

Received May 7, 1981

Why Quantum Computers?

Simulating Physics with Computers

Richard P. Feynman

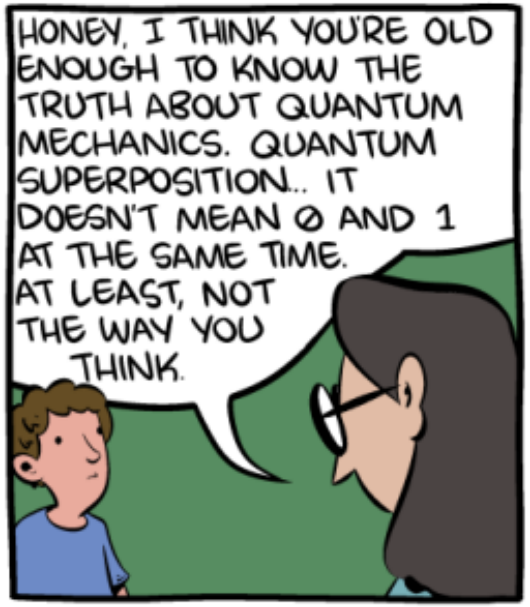
5. CAN QUANTUM SYSTEMS BE PROBABILISTICALLY SIMULATED BY A CLASSICAL COMPUTER?

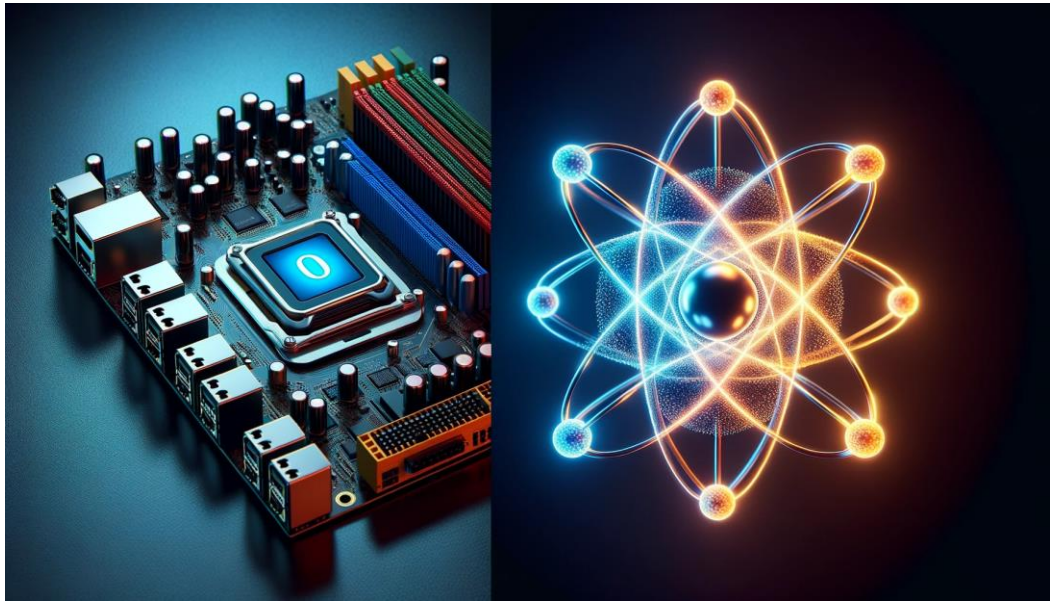
Now the next question that I would like to bring up is, of course, the interesting one, i.e., Can a quantum system be probabilistically simulated by a classical (probabilistic, I'd assume) universal computer? In other words, a computer which will give the same probabilities as the quantum system does. If you take the computer to be the classical kind I've described so far, (not the quantum kind described in the last section) and there're no changes in any laws, and there's no hocus-pocus, the answer is certainly, No! This is called the hidden-variable problem: it is impossible to represent the results of quantum mechanics with a classical universal device. To learn a little bit about it, I say let us try to put the quantum equations in a form as close as

Quantum Computers Principle

Compute by transforming a **quantum state**, composed of **quantum bits** (qubits)

This extends the Turing-Church model, thanks to quantum physical phenomena: superposition, entanglement, interference (TLDR: **magic**)





Qubits Superposition

$\alpha\ |0\rangle + \beta\ |1\rangle$ — *Measurement* → 0 with probability $|\alpha|^2$; 1 with probability $|\beta|^2$

Qubit state

α, β are complex, possibly negative "probabilities" called **amplitudes**

After a measurement the qubit stays 0 or 1 forever

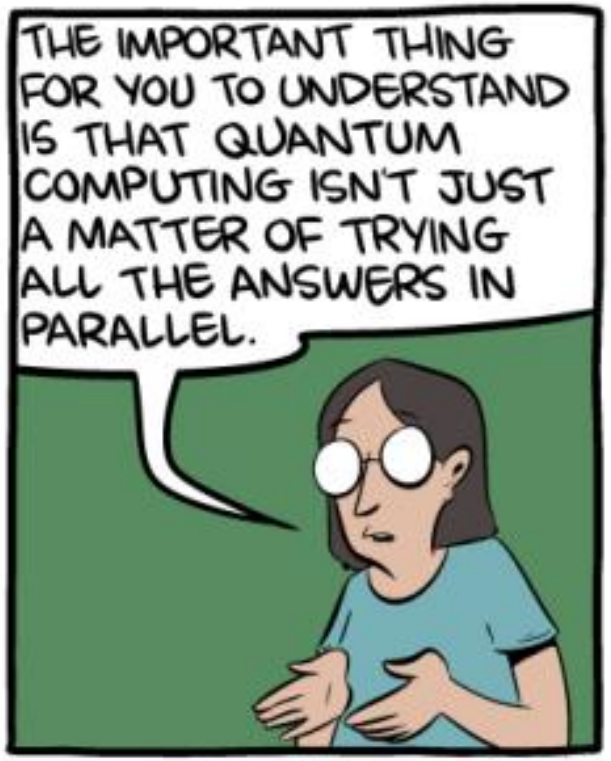
Real randomness!

Quantum Speedup

When a problem can be solved faster with quantum computer than classical computers

NOT about doing faster the same algorithms
NOT “trying all solutions at the same time”
NOT solving any hard problems (useless to crack passwords)

Exponential quantum speedup: why we’re here ☺
= from practically impossible to feasible



Shor's Algorithm

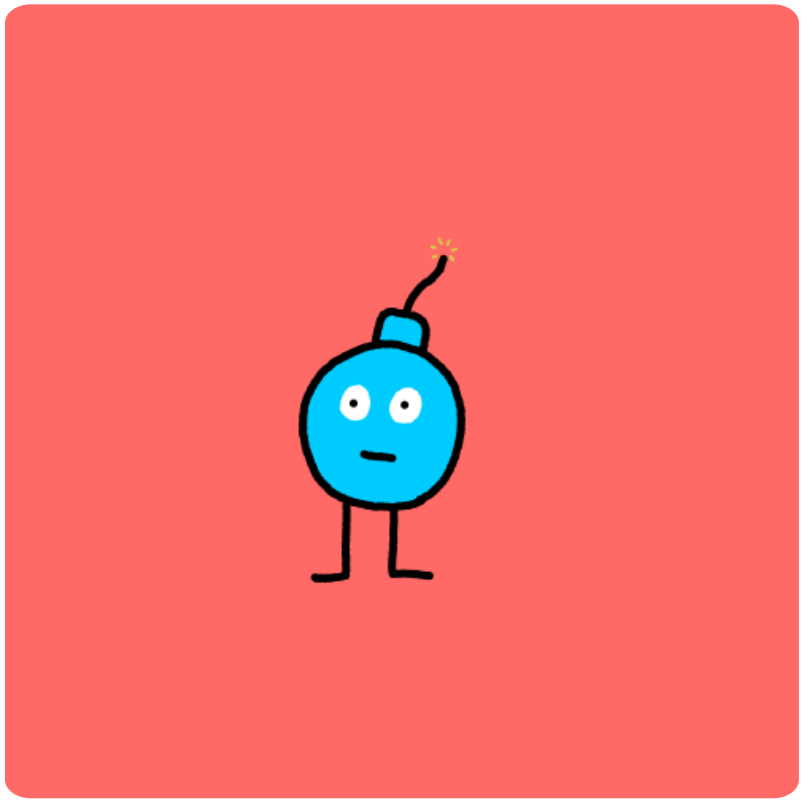
Efficient algorithm for the following problems:

Factoring: Find **p** given **n** = **pq**

→ RSA, Paillier: ***dead***

Discrete log: Find **d** given **y** = $\mathbf{x}^{\mathbf{d}}$ mod **p**

→ (EC)DSA, Diffie-Hellman: ***dead***



Practically impossible on a classical machine

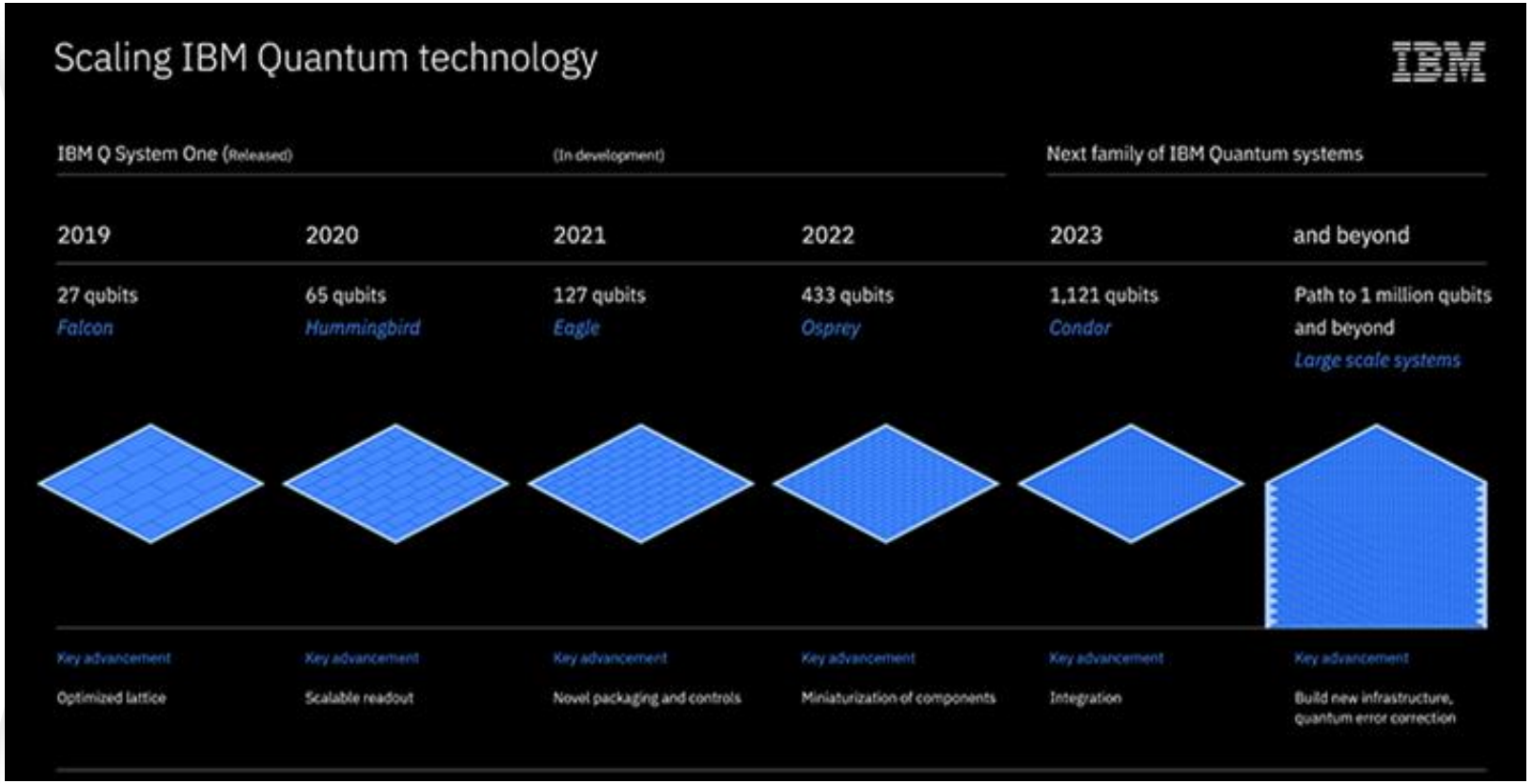
#ExponentialQuantumSpeedup

“Quantum Computers” That Exist Today

Not yet…

- Fault-tolerant
- Universal
- Scalable

Kinda useless ☺

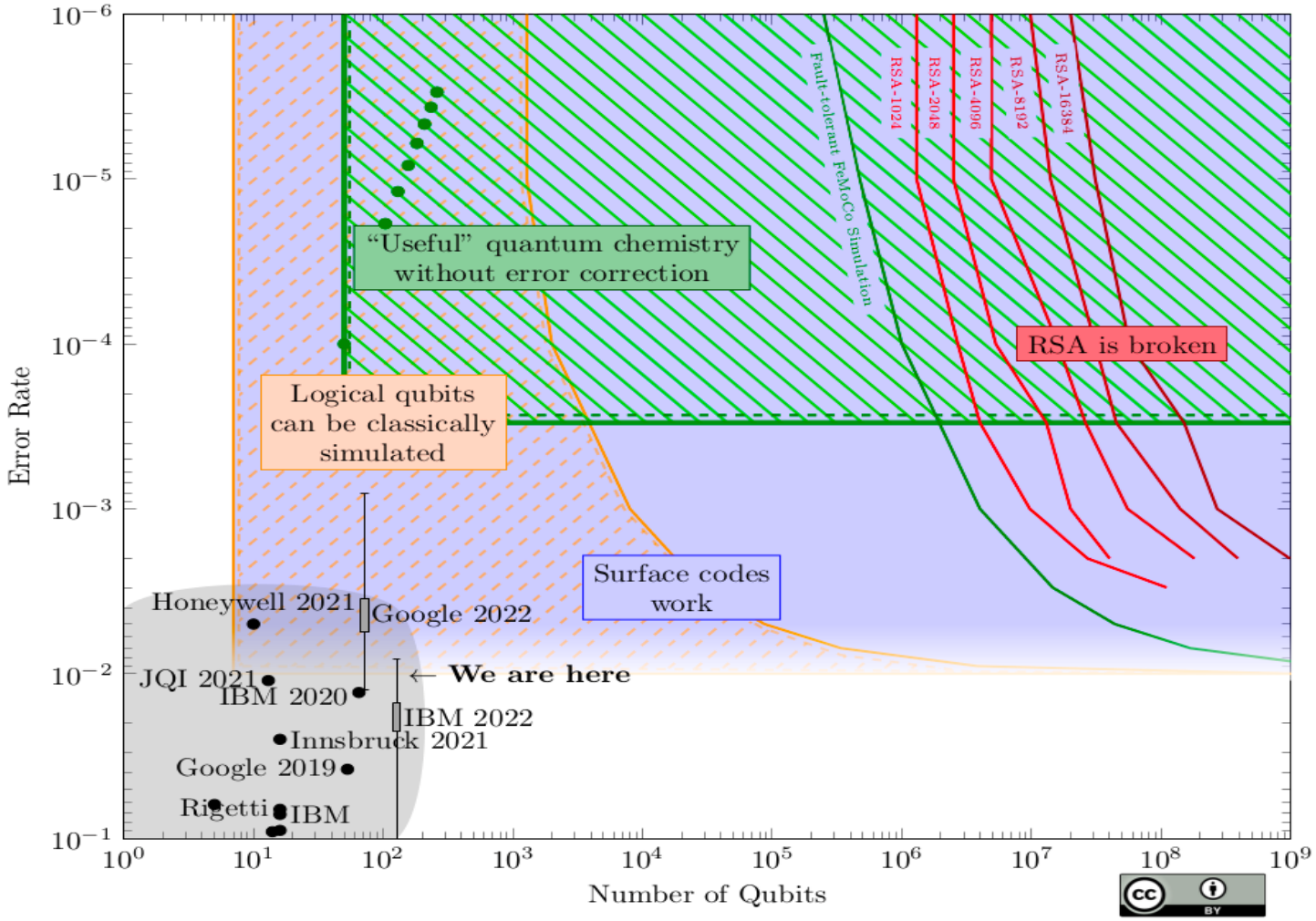


The QC Landscape

Graph showing possible QC applications for different QC capabilities:

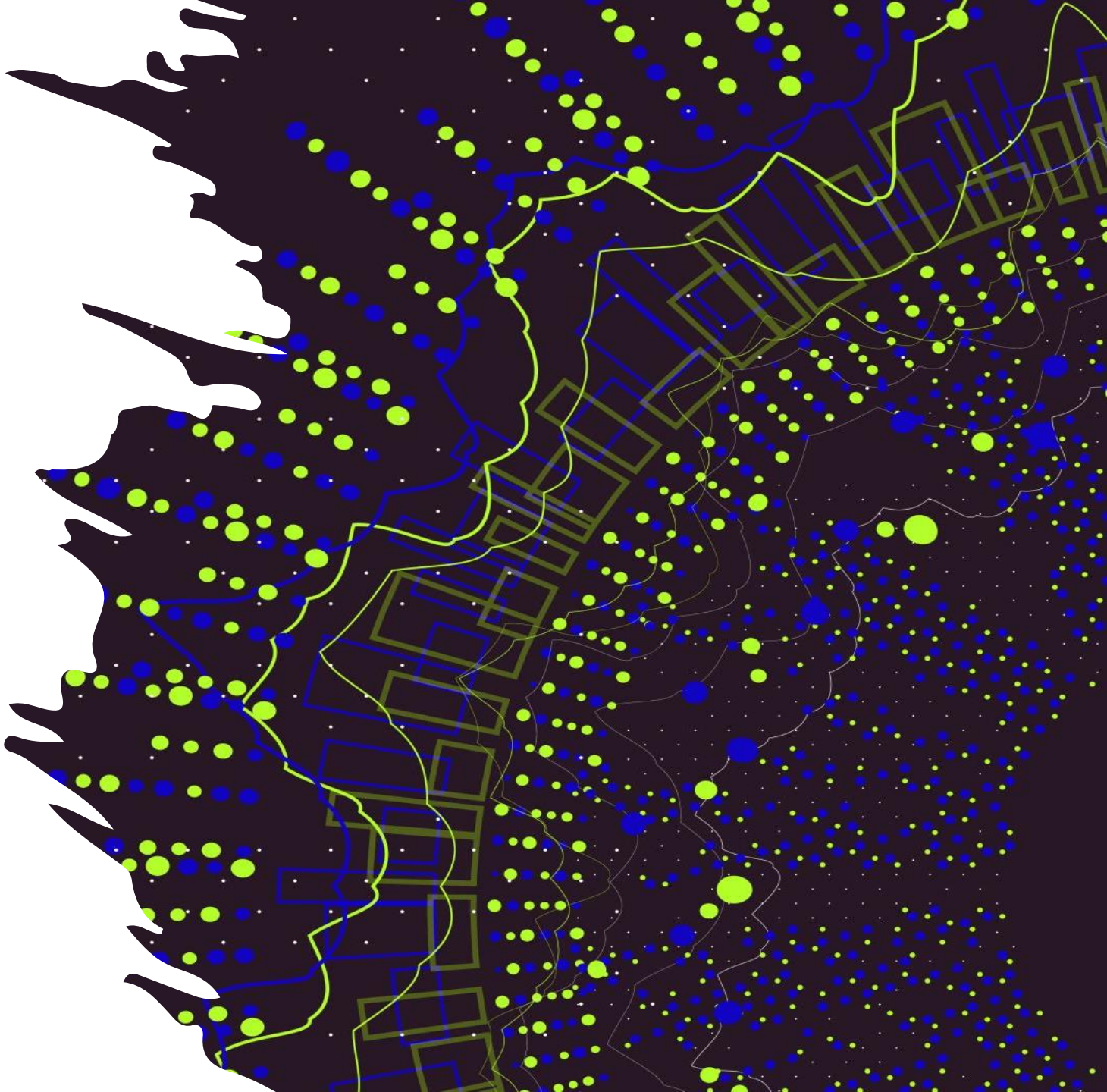
Vertical axis = error correction tech progress

Horizontal axis = size in terms of physical qubits (not logical)



By Samuel Jacques http://sam-jaques.appspot.com/quantum_landscape_2022

2. Post-Quantum Cryptography



“Post-Quantum” Crypto

Public-key cryptographic schemes designed to withstand attacks from both classical and quantum computers.

a.k.a. quantum-safe, quantum-resilient

2 types of algorithms:
- **Signatures** schemes
- **Key encapsulation mechanisms** (KEMs),

for encryption and key agreement

Different from “quantum cryptography”

Design Approaches

Post-quantum schemes can be based on…

Lattice problems ≈ solving equations with random errors
The best trade-off security assurance / speed / key size

Coding theory ≈ decoding with partial information
Large keys, both for signature and encryption

Hash trees ≈ breaking hash functions
Huge keys and signatures, signature only, least mathematical

Multivariate polynomials ≈ hardness of solving multivariate equations
Mostly for signatures, short signature values, but security less solid

Less mature approaches**: Elliptic curve isogenies, MPC-in-the-head**, **ZKP-based**

The NIST Standardization Project

NIST: The National Institute of Standards and technology, US gov agency

Open competition (anyone could submit) to select the **post-quantum crypto standards**

From 2017 to 2023, with an “encore” focusing only on signatures started in 2023

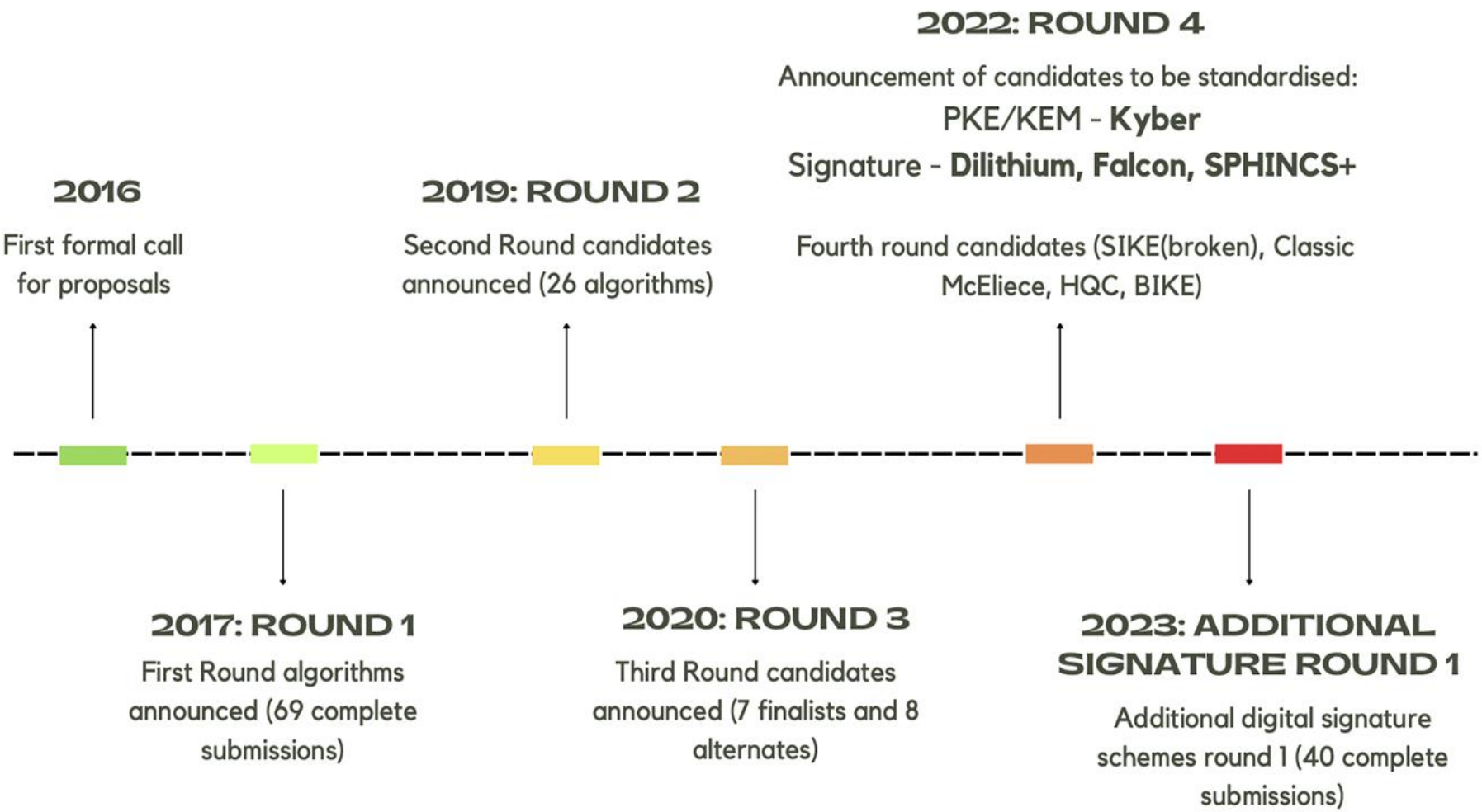
Selection criteria:

- Security
- Performance
- Simplicity & functionalities

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)

Security Levels

The NIST Standardization Project



NIST Post-Quantum Draft Standards

FIPS 203 (Draft)

Federal Information Processing Standards Publication

Module-Lattice-based Key-Encapsulation Mechanism Standard

Category: Computer Security **Subcategory: Cryptography**

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8900

This publication is available free of charge from:
https://doi.org/10.6028/NIST.FIPS.203.ipd

Published August 24, 2023

FIPS 204 (Draft)

Federal Information Processing Standards Publication

Module-Lattice-Based Digital Signature Standard

Category: Computer Security **Subcategory: Cryptography**

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8900

This publication is available free of charge from:
https://doi.org/10.6028/NIST.FIPS.204.ipd

Published August 24, 2023

FIPS 205 (Draft)

Federal Information Processing Standards Publication

Stateless Hash-Based Digital Signature Standard

Category: Computer Security **Subcategory: Cryptography**

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8900

This publication is available free of charge from:
https://doi.org/10.6028/NIST.FIPS.205.ipd

Published: August 24, 2023

https://csrc.nist.gov/pubs/fips/203/ipd
https://csrc.nist.gov/pubs/fips/204/ipd
https://csrc.nist.gov/pubs/fips/205/ipd

Post-Quantum Crypto Performance

Algorithm	Public key (bytes)	Ciphertext (bytes)	Key gen. (ms)	Encaps. (ms)	Decaps. (ms)	
ECDH NIST P-256	64	64	0.072	0.072	0.072	Elliptic curves (not post-quantum)
SIKE p434	330	346	13.763	22.120	23.734	Isogeny-based
Kyber512-90s	800	736	0.007	0.009	0.006	Lattice-based
FrodoKEM-640-AES	9,616	9,720	1.929	1.048	1.064	

Table 1: Key exchange algorithm communication size and runtime

Algorithm	Public key (bytes)	Signature (bytes)	Sign (ms)	Verify (ms)	
ECDSA NIST P-256	64	64	0.031	0.096	
Dilithium2	1,184	2,044	0.050	0.036	Lattice-based
qTESLA-P-I	14,880	2,592	1.055	0.312	
Picnic-L1-FS	33	34,036	3.429	2.584	Zero-knowledge proof-based

Table 2: Signature scheme communication size and runtime

From "Benchmarking Post-Quantum Cryptography in TLS" https://eprint.iacr.org/2019/1447

3. Attacking Real Protocols

Risk Levels

Not terrible: Signatures (ECDSA, Ed25519, etc.)
Can be reissued with a post-quantum algorithm
Use cases: Bitcoin, firmware signing, application signing

Bad: Key agreement (Diffie-Hellman, ECDH, etc.)
Partially Mitigated by secret internal states and reseeding
Use cases: TLS, WireGuard, end-to-end messaging

Very annoying: Encryption (RSA encryption, ECIES, etc.)
Encrypted messages compromised forever
Use cases: PGP email, encrypted backups

TLS

The most important internet security protocol:

HTTPS, M2M, mobile apps, VPNs, etc.

TLS is 2 protocols

- **Handshake**: key establishment (asymmetric)

- **Record**: encrypt data (symmetric)

(PSK version obviously quantum-safe)

TLS Quantum Attack

“Store now, decrypt later”

The attacker must store

- The Handshake (to recover the session keys)
- The encrypted data to decrypt

Easier if the attacker knows how much traffic has been transmitted (to find the nonce), but not mandatory

End-to-End Encryption (E2EE)

E2EE as used in Signal and WhatsApp consists in two sub protocols to determine message keys:

- Extended triple Diffie-Hellman key agreement (X3DH)
- Double Ratchet protocol (hashing based)

Designed to provide forward secrecy

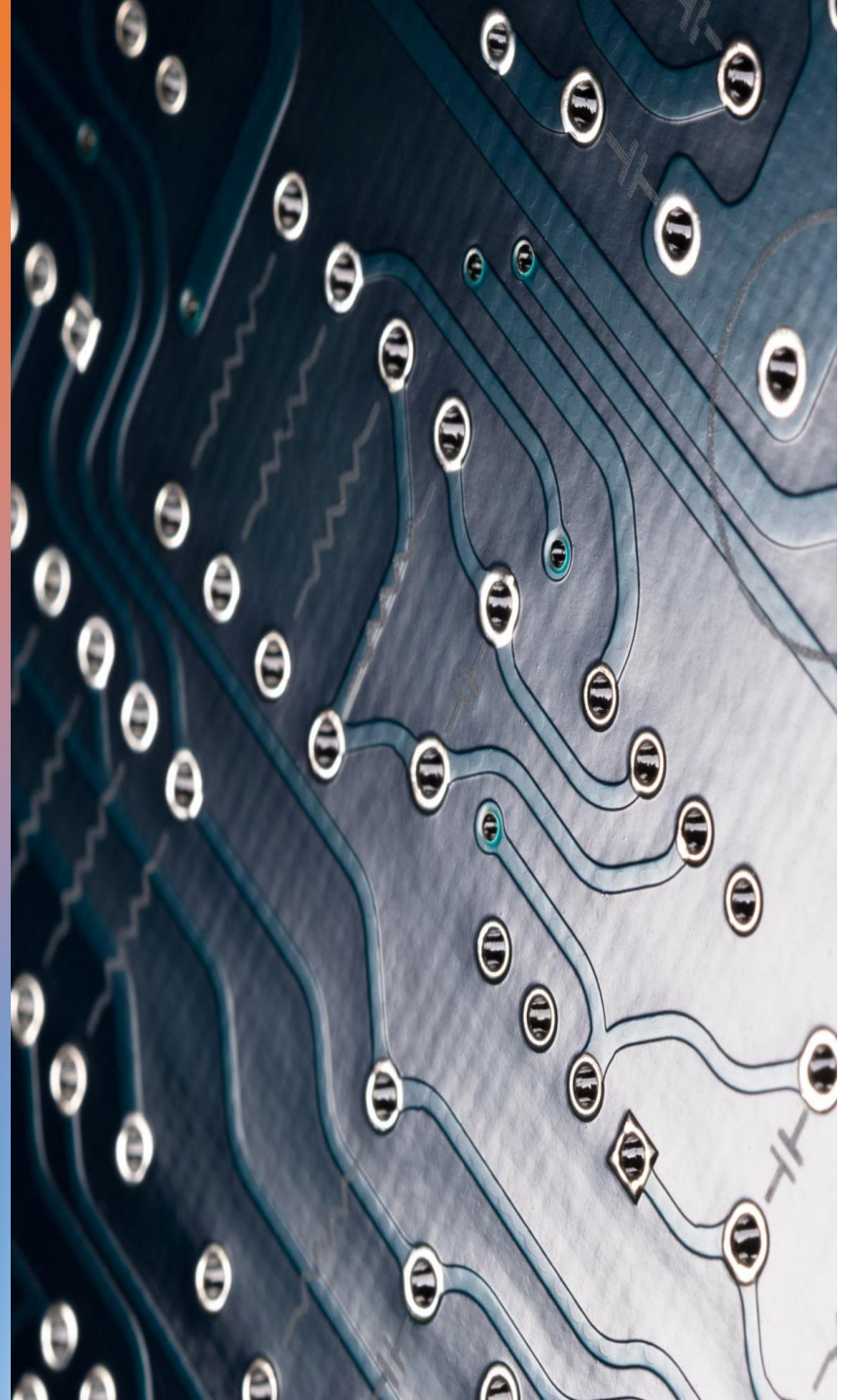
End-to-End Encryption (E2EE) Quantum Attack

“Store now, decrypt later”

The attacker must capture all communications from the handshake up to the encrypted content to be decrypted

Required to recover the message keys, must:

- Break all ephemeral DH
- Recompute the key chains



4G and 5G Communications

Authenticated key agreement (AKA) protocol relies on a symmetric key of 128 bits shared between the user (stored in the SIM) and its home network provider.

Less than 128-bit security against quantum attacks (Grover's algorithm quadratic speedup)

5G defines the authentication protocol EAP-TLS that uses public-key cryptography: **not post-quantum**.

VPN / Secure Channel

TLS-based: cf. TLS

IPsec-based: Similar case as TLS: need to break IKEv2's Diffie-Hellman

WireGuard: DH-based too, but public keys less exposed. Tweaks documented to be post-quantum. Also supports PSKs.

Blockchain Applications

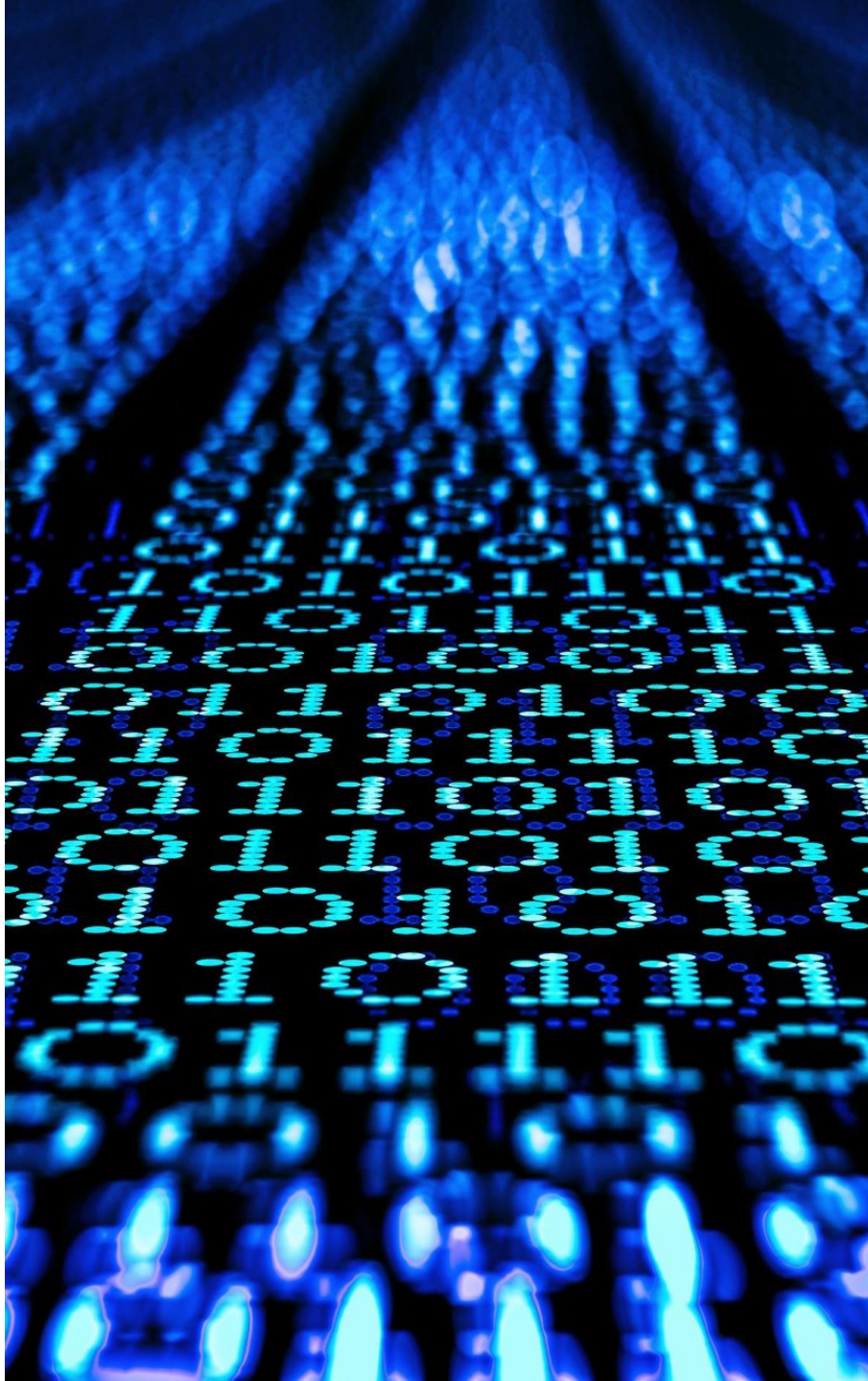
Signatures: private/public key pair for each account

- Generally, ECDSA or Ed25519
- BLS signatures (Eth validators)

These would be broken by QC, to steal funds

But the signatures can be **upgraded** before it's too late

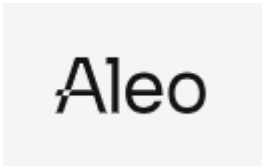
Consensus protocols also use public-key crypto for key agreement (for example libp2p), but less critical



Zero Knowledge Proof Systems

Complex protocols using various building blocks, often quantum-**unsafe**

As used for **private transfers and private programs** (zkEVMs, etc.)



Privacy leak (ZKness): impacted, but limited by the proof size

Proof cheat (soundness): like signatures, not a “store now break it later” case

4. Solutions Available

Quantum-Safe Tunnels

Many VPN providers have a post-quantum option (hybrid key agreement)

Examples: PQ'd TLS by **Cloudflare** with X25519/Kyber and **AWS** with ECDH/Kyber

Hybrid key exchange in practice

We have added ECDHE-with-Kyber ciphersuite to TLS 1.3 in s2n (our open-source TLS library)

These are deployed (but inactive) everywhere s2n is deployed

Active AWS Key Management Service, AWS Secrets Manager, and AWS Certificate Manager

	Bandwidth (bytes)	Total handshakes	Average (ms)	p0 (ms)	p50 (ms)	p90 (ms)	p99 (ms)
ECDHE (classic)	3,574	2,000	3.08	2.07	3.02	3.95	4.71
ECDHE + Kyber 512	5,898	2,000	3.36	2.38	3.17	4.28	5.35

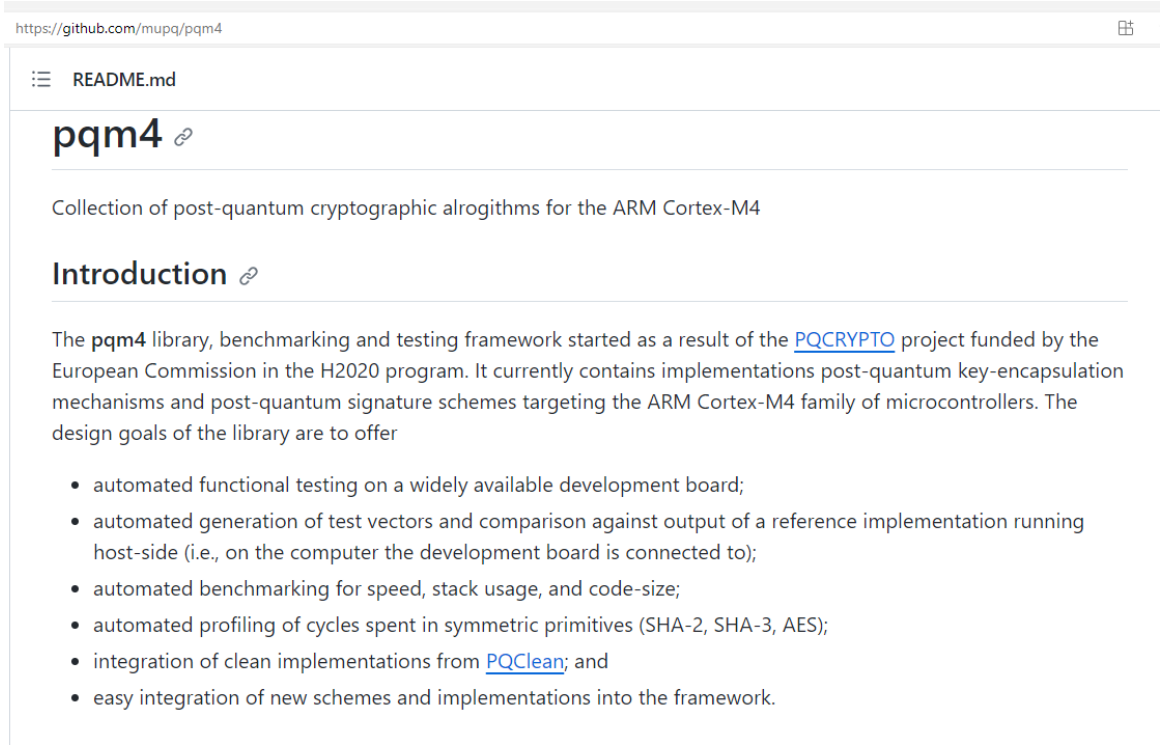
aws

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Quantum-Safe Software

Several companies offer software libraries for PQC

Also many good open-source projects:


https://github.com/mupq/pqm4

README.md

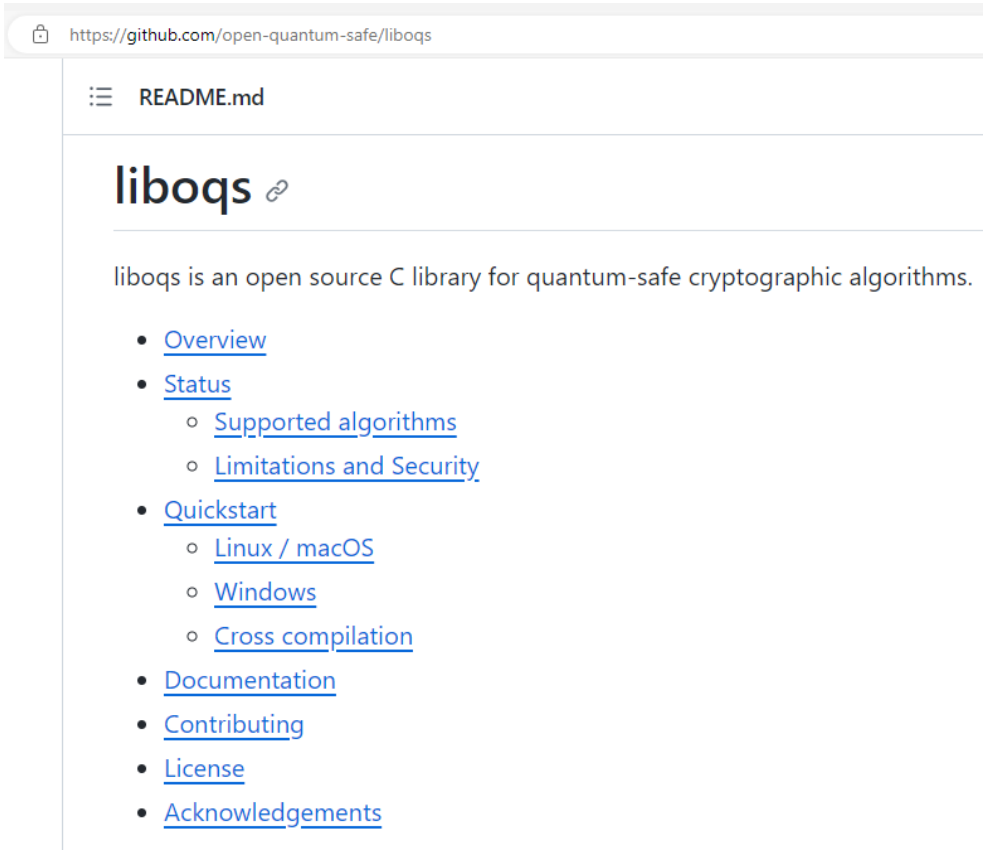
pqm4

Collection of post-quantum cryptographic alrogithms for the ARM Cortex-M4

Introduction

The **pqm4** library, benchmarking and testing framework started as a result of the PQCRYPTO project funded by the European Commission in the H2020 program. It currently contains implementations post-quantum key-encapsulation mechanisms and post-quantum signature schemes targeting the ARM Cortex-M4 family of microcontrollers. The design goals of the library are to offer

- automated functional testing on a widely available development board;
- automated generation of test vectors and comparison against output of a reference implementation running host-side (i.e., on the computer the development board is connected to);
- automated benchmarking for speed, stack usage, and code-size;
- automated profiling of cycles spent in symmetric primitives (SHA-2, SHA-3, AES);
- integration of clean implementations from PQClean; and
- easy integration of new schemes and implementations into the framework.


https://github.com/open-quantum-safe/liboqs

README.md

liboqs

liboqs is an open source C library for quantum-safe cryptographic algorithms.

- Overview
- Status
  - Supported algorithms
  - Limitations and Security
- Quickstart
  - Linux / macOS
  - Windows
  - Cross compilation
- Documentation
- Contributing
- License
- Acknowledgements

Quantum-Safe Hardware

Hardware blocks, co-processors, and hardware countermeasures

For example from PQShield

PQPlatform - CoPro

Post-Quantum Cryptography Processor (PQ-HW-COP)

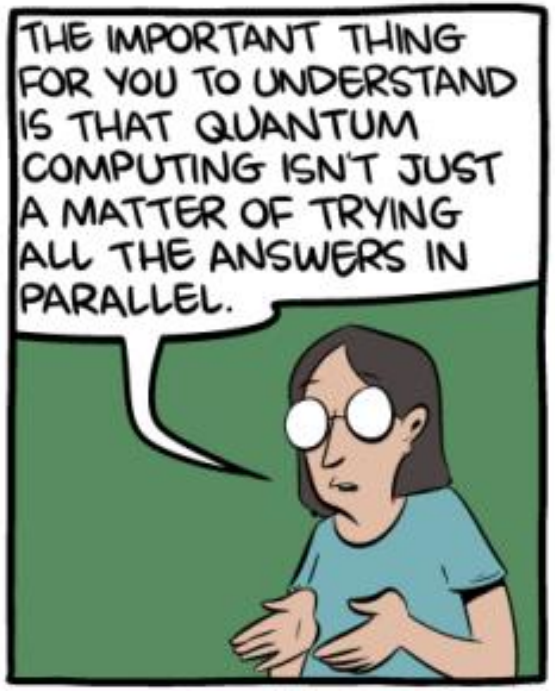
PQPlatform - CoPro adds PQShield's state-of-the-art post-quantum cryptography (PQC) to your security sub-system, with optional side-channel countermeasures (SCA). CoPro can be optimized for minimum area as part of an existing security sub-system.

PQPlatform - CoPro is designed to be run by an existing CPU in your security system, using PQShield's supplied firmware.

- PQC algorithm execution time
  - SCA disabled (unmasked): between 440k-5,100k cycles for each supported PQC algorithm at NIST security level 5
  - SCA enabled (masked): between 440k-18,000k cycles for each supported PQC algorithm at NIST security level 5

    Note: Execution time is an *average*, as Dilithium signing contains a probabilistic step known as 'rejection sampling'.
- Size: ~125KGe

The End


THE IMPORTANT THING FOR YOU TO UNDERSTAND IS THAT QUANTUM COMPUTING ISN'T JUST A MATTER OF TRYING ALL THE ANSWERS IN PARALLEL.

Conclusion

Quantum computers should not be on top of your worries

But more and more discussed with clients/auditors

Standards not finalized yet, expect more commercial support once they are

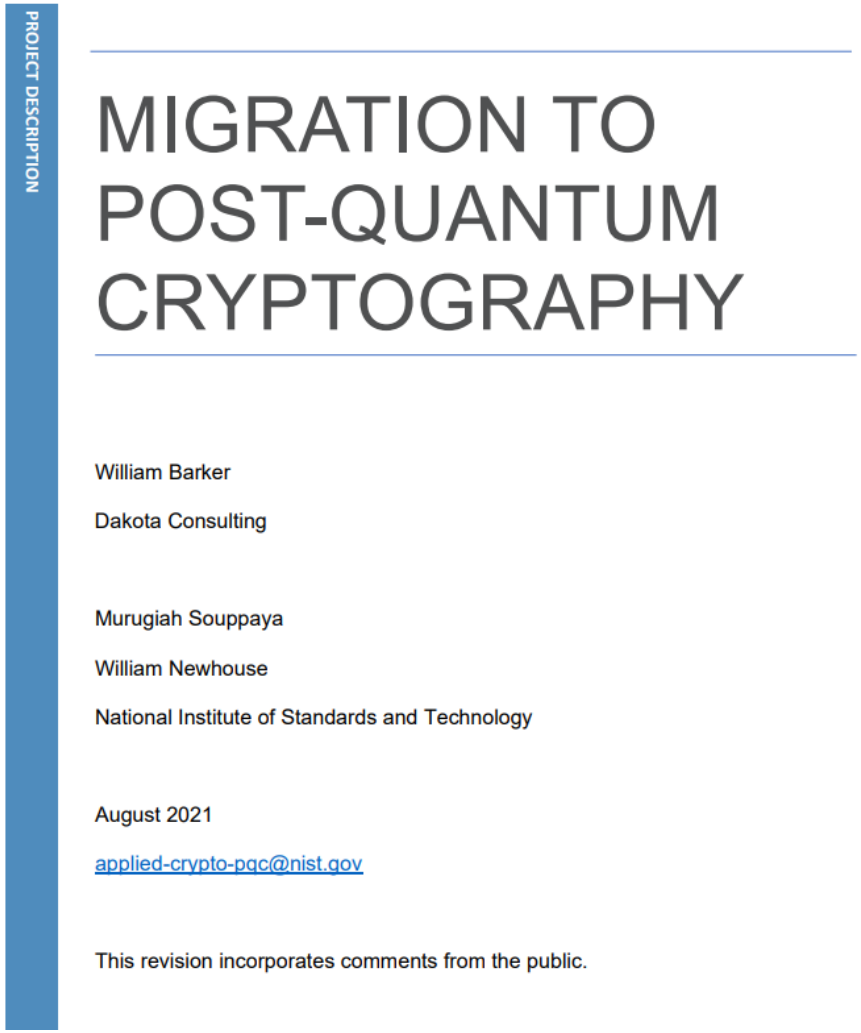
Prioritize key exchange and encryption over signatures

Doing a **risk assessment** in your organization is not too complex

- Inventory of all cryptography usage and protocols
- Identify the riskiest cases (in terms of business value vs. quantum risk)

See our page https://github.com/veorq/awesome-post-quantum

References


PROJECT DESCRIPTION

MIGRATION TO POST-QUANTUM CRYPTOGRAPHY

William Barker

Dakota Consulting

Murugiah Souppaya

William Newhouse

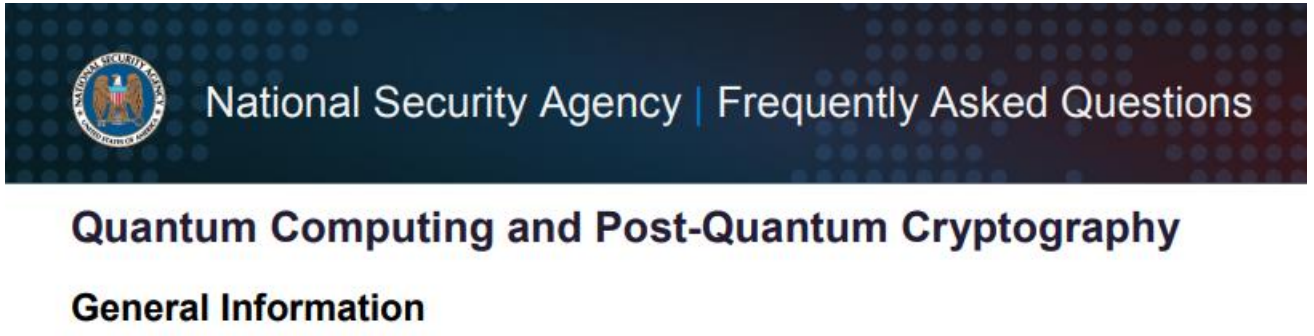
National Institute of Standards and Technology

August 2021

applied-crypto-pqc@nist.gov

This revision incorporates comments from the public.

https://www.nccoe.nist.gov/sites/default/files/2022-07/pqc-migration-project-description-final.pdf


National Security Agency | Frequently Asked Questions

Quantum Computing and Post-Quantum Cryptography

General Information

https://media.defense.gov/2021/Aug/04/2002821837/-1/-1/1/Quantum_FAQs_20210804.PDF


RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

Search...

About ANSSI | ANSSI's Organisation | What we do | Regulation | Scientific standing | Digital Risk Management

Accueil > Liste Des Publications > Follow up position paper on Post-Quantum Cryptography

Follow up position paper on Post-Quantum Cryptography

This document is an update of ANSSI's position on the post-quantum cryptography transition in view of the recent advances in the topic. It should be read as an addendum to 2022's publication [1].

https://cyber.gouv.fr/en/publications/follow-position-paper-post-quantum-cryptography

https://www.smbc-comics.com/comic/the-talk-3

http://math.nist.gov/quantum/zoo/

Thank you!

jp@taurushq.com
faclimandos@gmail.com

TAURUS