Post-Quantum Cryptography standards

Fernando Virdia
https://fundamental.domains

tl; dr

- NIST (US govt.) is standardising new crypto
- This will get into TLS, VPN, SSH libraries
- Needs to be scrutinised
- Needs to be benchmarked

Public-Key Cryptography

- Modern secure communications depend on public-key cryptography
- This allows to authenticate and securely communicate with other parties

Public-Key Cryptography

- Modern secure communications depend on public-key cryptography
- This allows to authenticate and securely communicate with other parties
- Security problems so far:
  - Modes of operation
  - Managing a PKI
  - Bad implementations
  - Bad parameters

Public-Key Cryptography

- Currently deployed public-key primitives (1976+) are based on three mathematical problems:

RSA

(Finite Field) DLOG

Elliptic Curves DLOG

- Hard to "break" crypto:
keys of length $n \implies O(2^{n^{1/3}})$ operations to break

Quantum computation

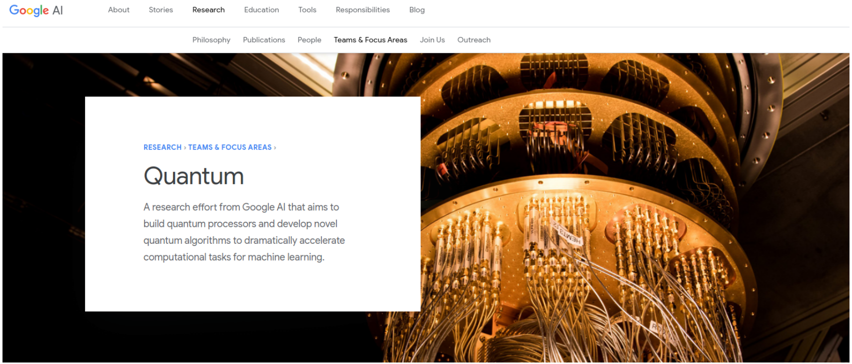
- [Sho97] introduces an algorithm that solves RSA/DLOG in $O(\text{poly}(n))$ ("easy")

Quantum computation

- [Sho97] introduces an algorithm that solves RSA/DLOG in $O(\text{poly}(n))$ ("easy")
- Yeah, sure, with a "quantum computer"

Quantum computation

- [Sho97] introduces an algorithm that solves RSA/DLOG in $O(\text{poly}(n))$ ("easy")
- Yeah, sure, with a "quantum computer"



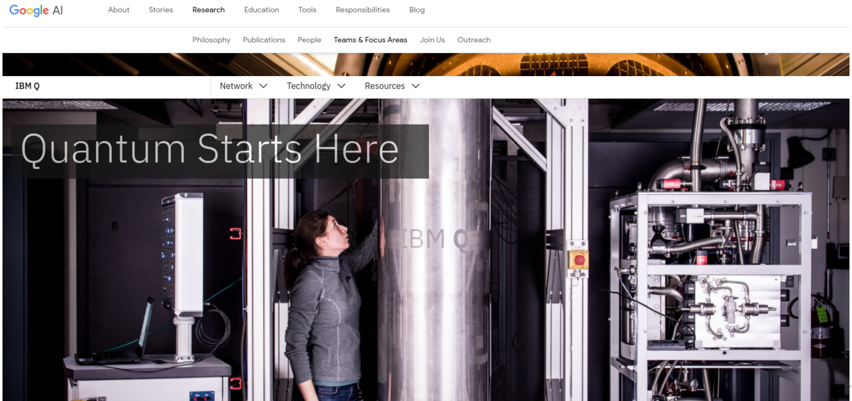
RESEARCH › TEAMS & FOCUS AREAS ›

Quantum

A research effort from Google AI that aims to build quantum processors and develop novel quantum algorithms to dramatically accelerate computational tasks for machine learning.

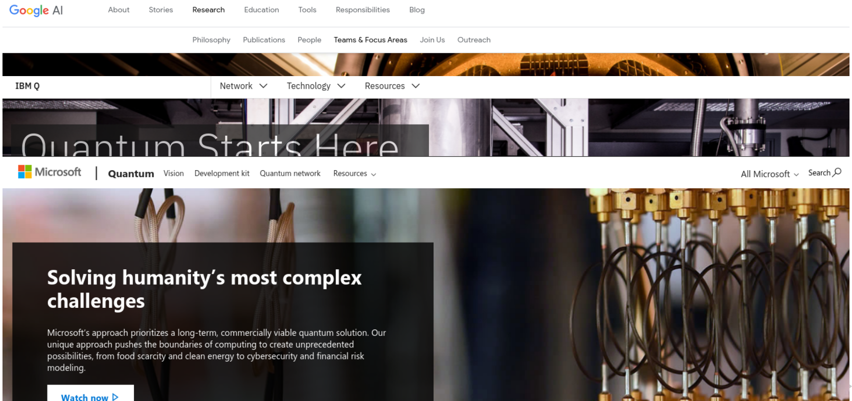
Quantum computation

- [Sho97] introduces an algorithm that solves RSA/DLOG in $O(\text{poly}(n))$ ("easy")
- Yeah, sure, with a "quantum computer"



Quantum computation

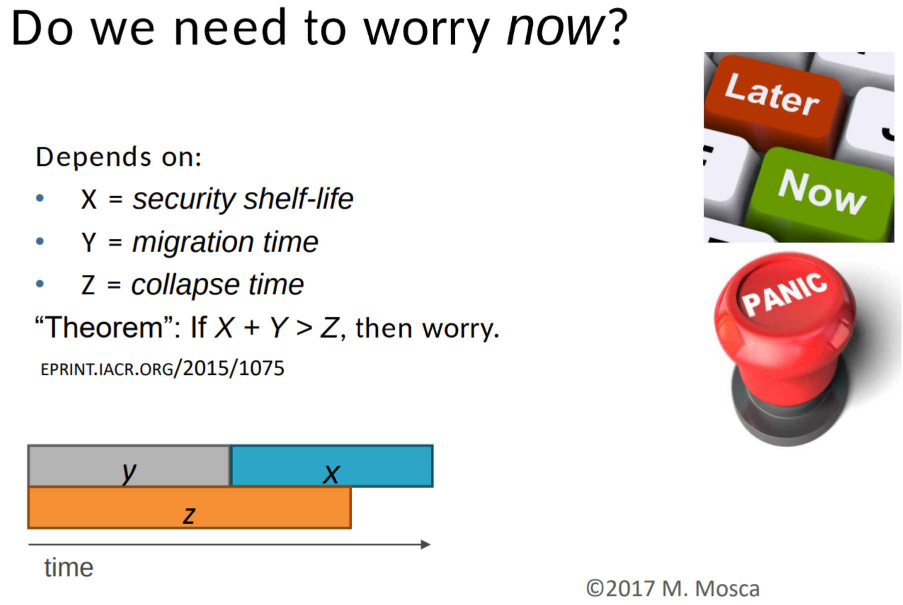
- [Sho97] introduces an algorithm that solves RSA/DLOG in $O(\text{poly}(n))$ ("easy")
- Yeah, sure, with a "quantum computer"



Quantum Starts Here

Solving humanity's most complex challenges

Microsoft's approach prioritizes a long-term, commercially viable quantum solution. Our unique approach pushes the boundaries of computing to create unprecedented possibilities, from food scarcity and clean energy to cybersecurity and financial risk modeling.



Do we need to worry *now*?

Depends on:

- X = *security shelf-life*
- Y = *migration time*
- Z = *collapse time*

“Theorem”: If $X + Y > Z$, then worry.

EPRINT.IACR.ORG/2015/1075

Later
Now
PANIC
y
x
z
time

©2017 M. Mosca

Post-Quantum Cryptography

- Priorities:
  - PKE: public-key encryption/key exchange
  - SIG: digital signatures/certificates

Post-Quantum Cryptography

- Priorities:
  - PKE: public-key encryption/key exchange
  - SIG: digital signatures/certificates
- BTW, what about symmetric crypto (AES/Chacha)?
  - Those should be fine

Post-Quantum Cryptography

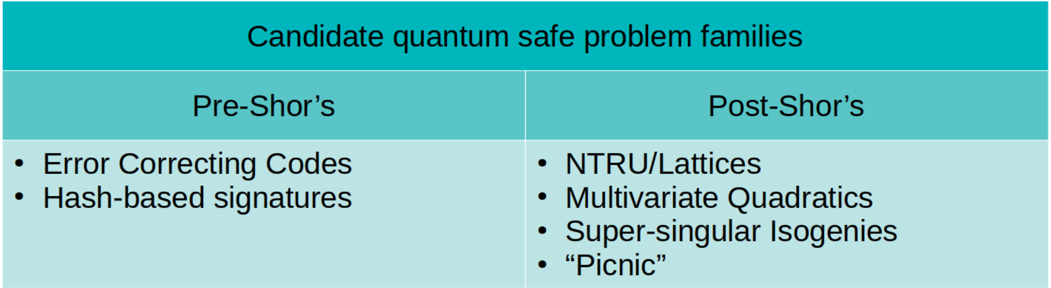
- Is all previous crypto really broken?
- No:
    - PKE: McEliece/Niederreiter, NTRU
    - SIG: hash-based signatures

Post-Quantum Cryptography

- Is all previous crypto really broken?
- No:
  - PKE: McEliece/Niederreiter, NTRU
  - SIG: hash-based signatures
- But slower and harder to manage, so they were never deployed
- Some RFCs and standards exist:
  `https://tools.ietf.org/id/draft-mcgrew-hash-sigs-11.html`
  `https://tools.ietf.org/html/rfc8391`
  `https://webstore.ansi.org/standards/ascx9/ansix9982010r2017`
  `https://standards.ieee.org/standard/1363_1-2008.html`

PQC jargon

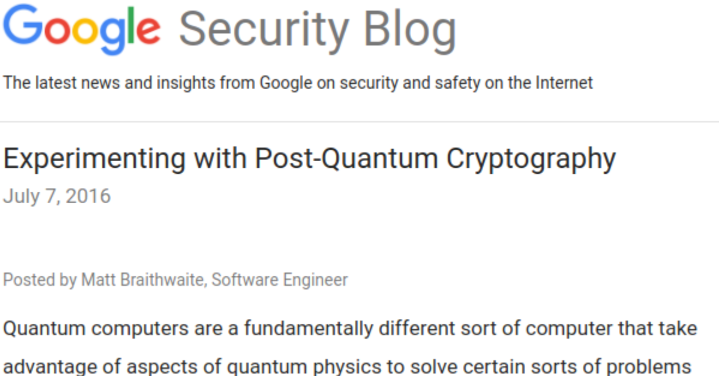
- Since [Sho97], new (maybe) “quantum safe” problems for PKE/SIG



Candidate quantum safe problem families	
Pre-Shor’s	Post-Shor’s
• Error Correcting Codes • Hash-based signatures	• NTRU/Lattices • Multivariate Quadratics • Super-singular Isogenies • “Picnic”

PQC timeline: 2016

- Google deploys NewHope key exchange (lattice-based) on Chrome Canary as an experiment
- Hybrid mode of operation guarantees pre-quantum security even if NewHope were broken



Google Security Blog

The latest news and insights from Google on security and safety on the Internet

Experimenting with Post-Quantum Cryptography

July 7, 2016

Posted by Matt Braithwaite, Software Engineer

Quantum computers are a fundamentally different sort of computer that take advantage of aspects of quantum physics to solve certain sorts of problems

PQC timeline: 2016

- The NSA (indirectly) warns the US Govt. about the need for post-quantum cryptographic standards

PQC timeline: 2016

- The NSA (indirectly) warns the US Govt. about the need for post-quantum cryptographic standards
- The National Institute of Standards and Technologies (NIST) publishes a call for proposals for PKE and SIG [Nat16]
- Plan: run a standardisation process like for AES and SHA3

PQC timeline: 2016

- Anyone can submit proposals
- Need to provide:
  - A written specification with a security analysis
  - A reference implementation in C99
  - (Optional) An optimised implementation (C + ASM)
    - Main target: x64 CPUs
  - Test vectors
- Mailing list for the process @ `https://groups.google.com/a/list.nist.gov/forum/#!forum/pqc-forum`

PQC timeline: 2017

- NIST announces the received candidates:
  - 82 submitted
  - 69 accepted
  - 49 PKE, 20 SIG
- The first round of the process starts
- Some candidates fall during the first weeks

PQC timeline: 2017

- Suggested timeline by NIST:
  - December 2017: First round
  - January 2019: Second round
  - 2020/2021: Third round
  - 2022/2024: Draft Standards Available

PQC timeline: 2018

- First NIST PQC workshop (April)
- The authors of the standing schemes present their work

PQC timeline: 2019

- Candidates accepted to the second round announced:
  17 PKE, 9 SIG

- Request for FPGA/hardware implementations

PQC timeline: 2019

- Candidates accepted to the second round announced: 17 PKE, 9 SIG
- Request for FPGA/hardware implementations
- Cloudflare+Google run large scale TLS/PQC experiments: `https://csrc.nist.gov/Presentations/2019/measuring-tls-key-exchange-with-post-quantum-kem`

PQC timeline: 2019

- Candidates accepted to the second round announced:
  17 PKE, 9 SIG
- Request for FPGA/hardware implementations
- Cloudflare+Google run large scale TLS/PQC experiments:
  `https://csrc.nist.gov/Presentations/2019/measuring-tls-key-exchange-with-post-quantum-kem`
- Second NIST PQC workshop (August)
- Suggestion (by NIST):
  - Let's have a third round
  - But maybe let's already standardise some scheme at the end of the second

Can we play with this, already?

- NIST website contains links to all specs and implementations
- This requires compiling one by one each scheme. . .

Can we play with this, already?

- NIST website contains links to all specs and implementations
- This requires compiling one by one each scheme. . .
- LibOQS project: `https://openquantumsafe.org/`
  - Collects many of the submitted schemes
  - Provides unified API + wrappers for C#, C++, Python, Go
  - Provides test/benchmarking capabilities
  - Integrates with Open{SSH, SSL} + integration by Microsoft into OpenVPN fork

Can we help?

Can we help?

- Yes!
- PQC schemes are slower and larger than current crypto
- Not clear how they will interact with existing protocols and infrastructure
- How much slower? How much energy consuming? How much heavier?
- How flexible are current libraries? Hard-coded buffer lengths anybody?

Where to publish results?

- Pre-prints (IACR) @ `https://ia.cr`
- Workshops/conferences:
  - PQCrypto @ `https://pqcrypto.org/conferences.html`
  - NIST Workshops @ `https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/workshops-and-timeline`
  - Many IACR conferences @ `https://www.iacr.org/`
- Issues/questions/experiment results:
  - NIST PQC mailing list @ `https://groups.google.com/a/list.nist.gov/forum/#!forum/pqc-forum`

Thank you

You can find:

- NIST PQC @ `https://www.nist.gov/pqcrypto`
- Crypto papers @ `https://ia.cr`
- me @ `https://fundamental.domains`

National Institute of Standards and Technology.
Submission requirements and evaluation criteria for the Post-Quantum Cryptography standardization process.
http://csrc.nist.gov/groups/ST/post-quantum-crypto/documents/call-for-proposals-final-dec-2016.pdf, December 2016.

Peter W. Shor.
Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer.
SIAM J. Comput., 26(5):1484–1509, October 1997.