

# Quantum Computers vs. Computers Security

JP Aumasson / @veorq — Kudelski Security

**DEF CON**  
LAS VEGAS **23**

Schrodinger equation

Uncertainty principle

Entanglement

Hilbert spaces

**Nobody understands this stuff, and you don't  
need it to understand quantum computing**

Wave functions

EPR pairs

Unitary matrices

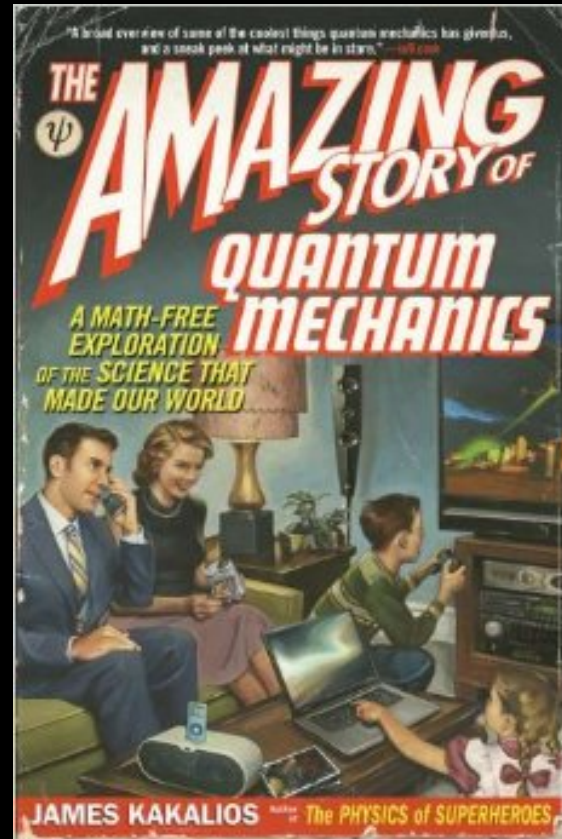
Tensor products

Bell states

# Agenda

1. QC 101
2. In practice
3. Breaking crypto
4. Post-quantum crypto
5. Quantum key distribution
6. Quantum copy protection
7. Quantum machine learning
8. Conclusions

# 1. QC 101



# Quantum mechanics

Nature's OS

Applications

Gravity

Electromagnetism

Nuclear forces

OS

Quantum mechanics

Hardware

Mathematics

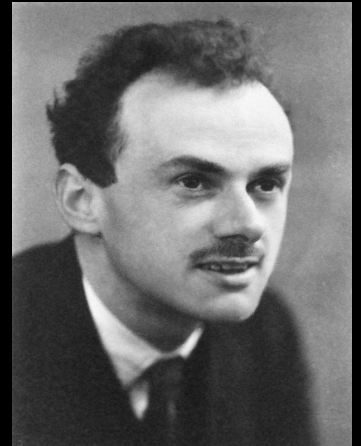
# Quantum mechanics — cont.

Particles in the universe behave **randomly**

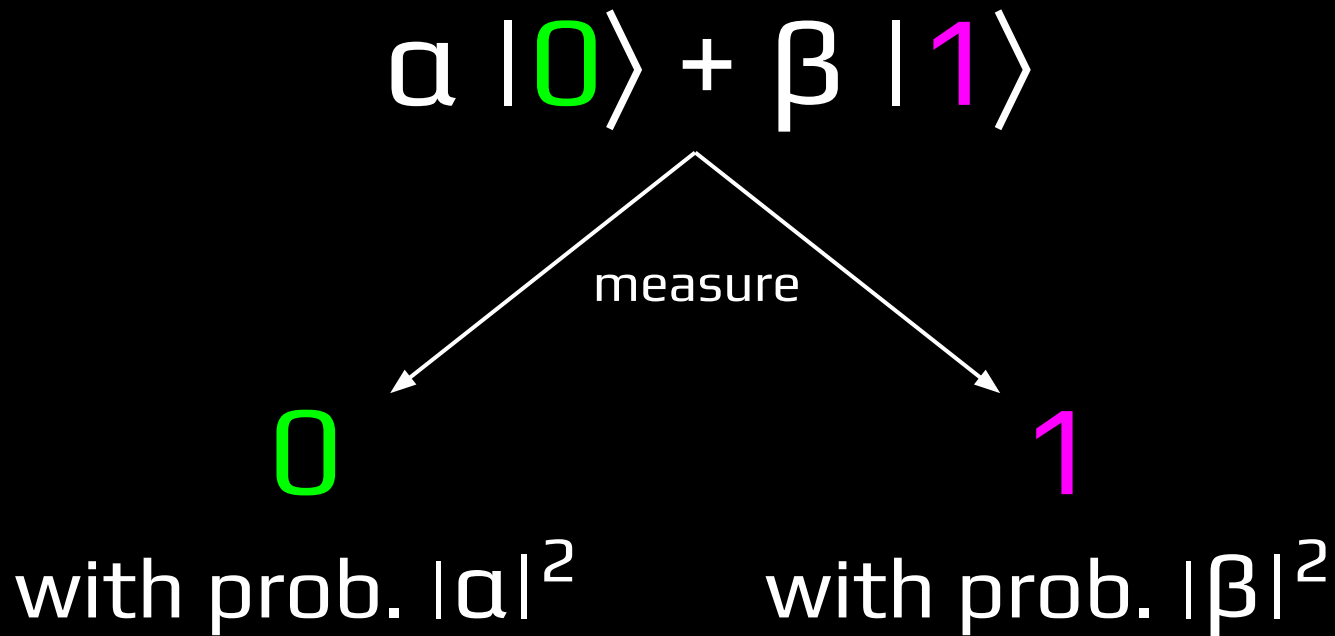
Their probabilities can be **negative**

"Negative energies and probabilities should not be considered as nonsense. They are well-defined concepts mathematically, like a negative of money."

—Paul Dirac, 1942



# Quantum bit (qubit)



Stays 0 or 1 forever!

# Quantum byte

$$\alpha_{0x00} |0x00\rangle + \dots + \alpha_{0xfe} |0xfe\rangle + \alpha_{0xff} |0xff\rangle$$

The  $\alpha$ 's are called **amplitudes**

Generalizes to 32- or 64-bit quantum words

# Quantum computer

Set of **quantum registers**

Qubits/qubytes/quwords

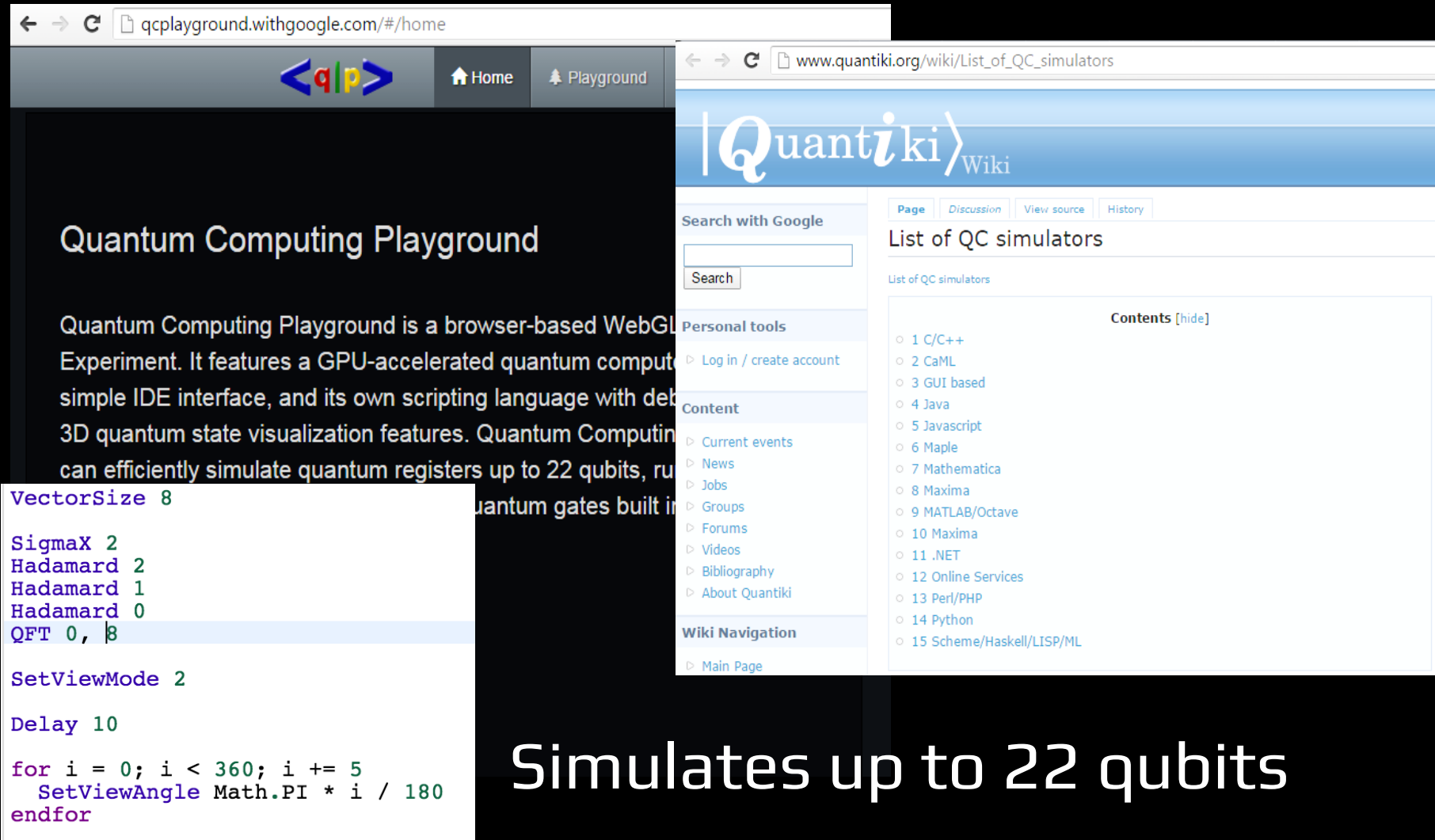
**Quantum assembly** instructions

Modify probabilities with matrix multiplications

A program usually ends with a **measurement**

Can't be simulate classically!

# Quantum computer simulators



The image shows two overlapping browser windows. The background window is the Quantum Computing Playground website, which features a dark theme and a navigation bar with 'Home' and 'Playground' links. The foreground window is the Quantiki Wiki page titled 'List of QC simulators', which has a light blue header and a list of 15 simulation languages. A code editor window is overlaid on the bottom left, displaying a quantum circuit script. The text 'Simulates up to 22 qubits' is written in large white font at the bottom right of the image.

**Quantum Computing Playground**

Quantum Computing Playground is a browser-based WebGL Experiment. It features a GPU-accelerated quantum computer, a simple IDE interface, and its own scripting language with default 3D quantum state visualization features. Quantum Computing Playground can efficiently simulate quantum registers up to 22 qubits, run quantum gates built in

```
VectorSize 8

SigmaX 2
Hadamard 2
Hadamard 1
Hadamard 0
QFT 0, 8

SetViewMode 2

Delay 10

for i = 0; i < 360; i += 5
    SetViewAngle Math.PI * i / 180
endfor
```

**Quantiki Wiki**

Search with Google

Personal tools

- Log in / create account

Content

- Current events
- News
- Jobs
- Groups
- Forums
- Videos
- Bibliography
- About Quantiki

Wiki Navigation

- Main Page

List of QC simulators

Contents [hide]

- 1 C/C++
- 2 CaML
- 3 GUI based
- 4 Java
- 5 Javascript
- 6 Maple
- 7 Mathematica
- 8 Maxima
- 9 MATLAB/Octave
- 10 Maxima
- 11 .NET
- 12 Online Services
- 13 Perl/PHP
- 14 Python
- 15 Scheme/Haskell/LISP/ML

Simulates up to 22 qubits

# The killer app

## **Simulating Physics with Computers**

**Richard P. Feynman**

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

*Received May 7, 1981*

Impossible with a classical computer

Possible with a quantum computer!

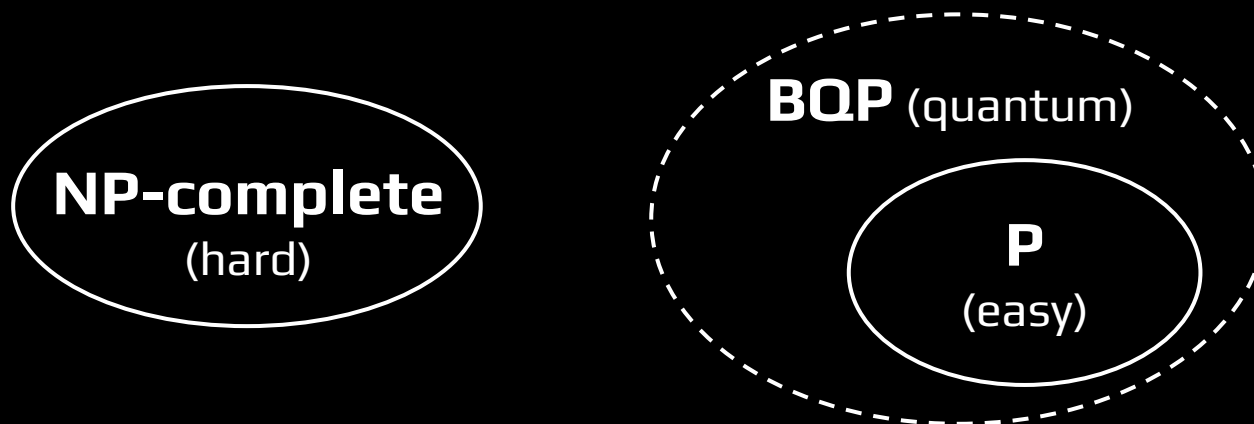
# QC vs. hard problems

Ever heard about **NP-complete** problems?

Solution hard to find, but easy to verify

SAT, scheduling, Candy Crush, etc.

**QC does not** solve NP-complete problems!



# Quantum speedup

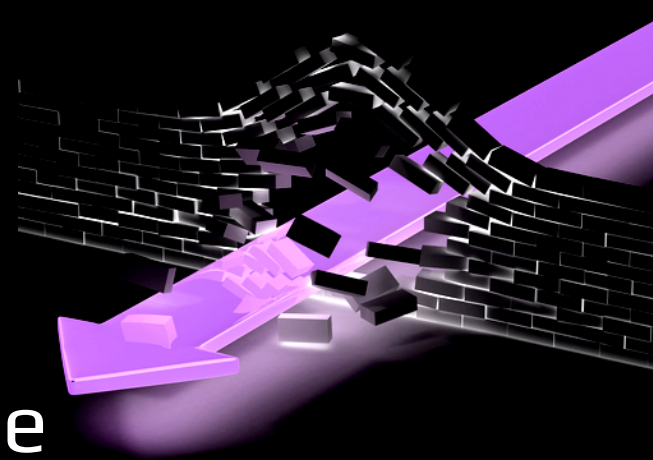
Making the impossible possible

Example: **factoring integers**

Hard classically (exponential-ish)

Easy with a quantum computer!

Obvious application: **breaking RSA!**



# Quantum parallelism

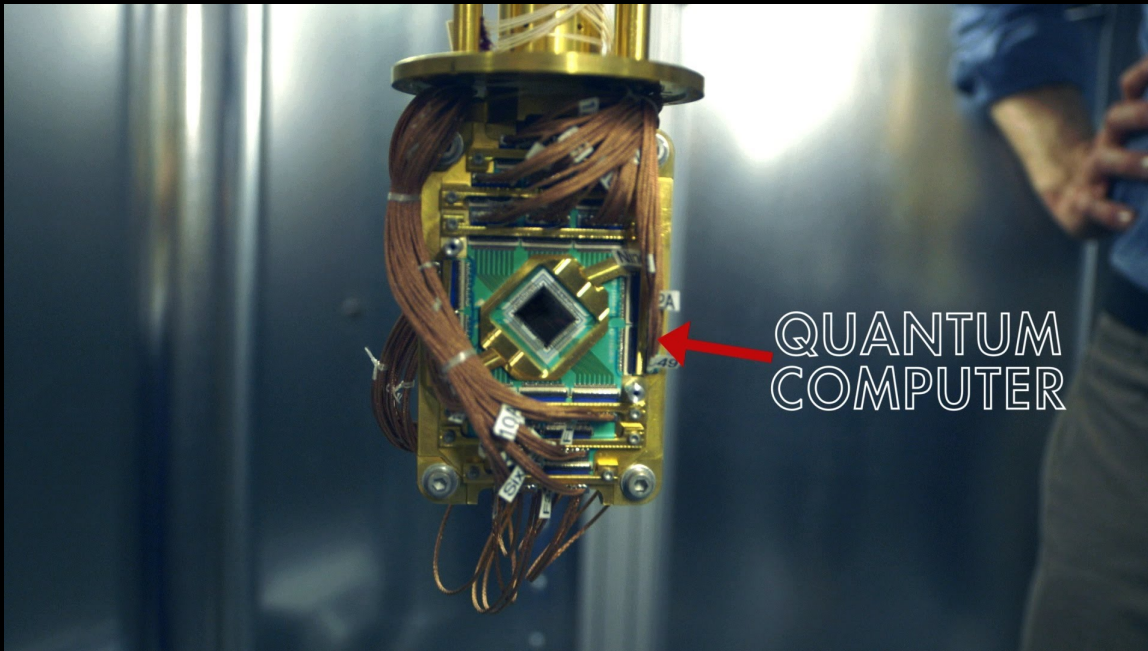
QC kind of encode all values simultaneously

But they **do not** “try every answer in parallel”

You can only **observe one** result, not all



## 2. In practice



In practice

# Factoring experiments

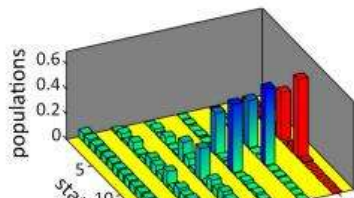
The quantum speed-up poster child

**QUANTUM PROCESSOR CALCULATES THAT  $15 = 3 \times 5$  (WITH ALMOST 50% ACCURACY!)**

By Rebecca Boyle Posted August 20, 2012

**143 is largest number yet to be factored by a quantum algorithm**

April 11, 2012 by Lisa Zyga feature



**Quantum factorization of 56153 with only 4 qubits**

**Nikesh S. Dattani<sup>1,2,\*</sup> Nathaniel Bryans<sup>3,†</sup>**

<sup>1</sup> Quantum Chemistry Laboratory, Kyoto University, 606-8502, Kyoto, Japan, <sup>2</sup> Physical & Theoretical Chemistry Laboratory, Oxford University, OX1 3QZ, Oxford, UK, <sup>3</sup> University of Calgary, T2N 4N1, Calgary, Canada. \*[dattani.nike@gmail.com](mailto:dattani.nike@gmail.com),

Only for numbers with special patterns

In practice

# Building quantum computers

Qubits obtained from **physical phenomena**

Photons

Molecules

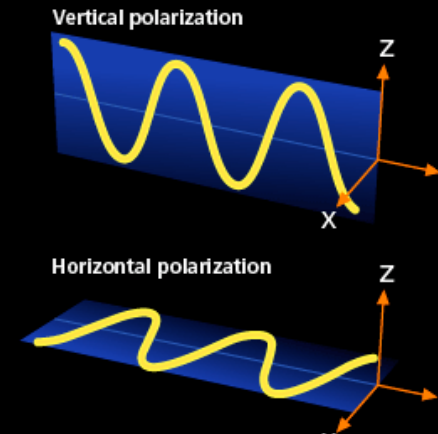
Superconducting

Many challenges:

Qubits mixed up with the environment

Cooling systems to a low temperature

Scaling to a useful number of qubits



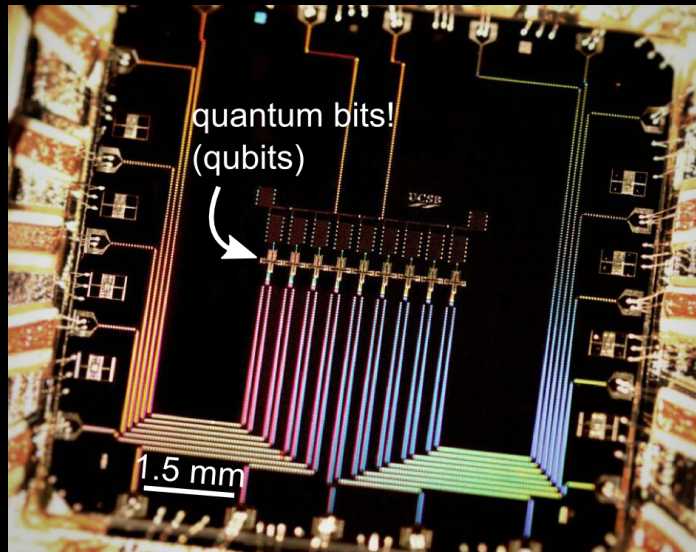
In practice

# Recent result (2015)

## Stable 9-qubit system

“suppression of environment-induced errors”

“quantum non-demolition parity measurements”



### State preservation by repetitive error detection in a superconducting quantum circuit

J. Kelly, R. Barends, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus, B. Campbell, Yu Chen, Z. Chen, B. Chiaro, A. Dunsworth, I.-C. Hoi, C. Neill, P. J. J. O'Malley, C. Quintana, P. Roushan, A. Vainsencher, J. Wenner, A. N. Cleland & John M. Martinis

[Affiliations](#) | [Contributions](#) | [Corresponding authors](#)

*Nature* **519**, 66–69 (05 March 2015) | doi:10.1038/nature14270

# 3. Breaking crypto



Breaking crypto

**TL;DR: We're doomed**

**RSA: broken**

**Diffie-Hellman: broken**

**Elliptic curves: broken**

**El Gamal: broken**



# RSA

Based on the hardness of **factoring**

Knowing  $N = pq$ , look for  $p$  and  $q$

Hard on a classical computer (probably)

**BUT easy on a quantum computer!**

# Discrete logarithms

Problem behind **Diffie-Hellman, ECC**

Knowing  $g$  and  $g^y$ , look for  $y$

Hard on a classical computer (probably)

**BUT easy on a quantum computer!**

# What about symmetric ciphers?

Grover algorithm FTW!

AES-128 security

Classical: 128-bit

Quantum: **64-bit**



Upgrade to 256-bit keys for 128-bit security

## 4. Post-quantum crypto

hope

# Post-quantum crypto

Alternatives to RSA, Diffie-Hellman, ECC  
Seem resistant to QC

<http://pqcrypto.org/>



The image shows a screenshot of the NIST Information Technology Laboratory (ITL) website. The NIST logo is in the top left corner. Below it, the text 'Information Technology Laboratory' is visible. A navigation bar contains links for 'About ITL', 'Publications', and 'Topic/Subject'. Below the navigation bar, a breadcrumb trail reads 'NIST Home > ITL > Computer Security'. To the right of the website screenshot is a banner for 'The Seventh International Conference on Post-Quantum Cryptography' held in Fukuoka, Japan, from February 24-26, 2016. Below the banner, the text 'Workshop on Cybersecurity in a Post-Quantum World' is displayed.

**NIST**  
Information Technology Laboratory  
About ITL ▼ Publications Topic/Subject  
NIST Home > ITL > Computer Security

**The Seventh International Conference  
on Post-Quantum Cryptography**  
Fukuoka, Japan, February 24-26, 2016

Workshop on Cybersecurity in a Post-Quantum World

# Hash-based signatures

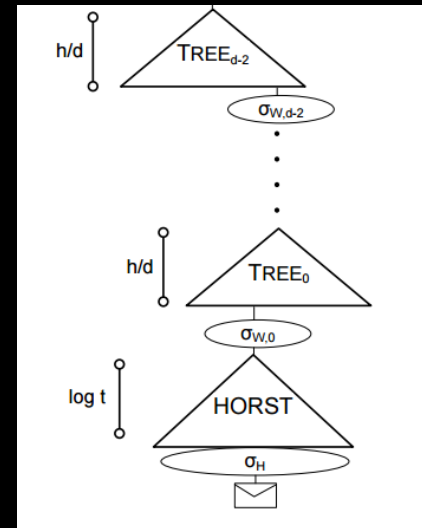
Problem: inverting **hash functions**

SPHINCS signatures <http://sphincs.cr.yp.to/>

41 KB signatures

1 KB public and private keys

Slow (100s signatures/sec)



# Multivariate signatures

Problem: solve complex systems of equations

$$0 = X_1 X_2 X_3 + X_1 X_3 + X_2 X_4$$

$$1 = X_1 X_3 X_4 + X_2 X_3 X_4$$

$$0 = X_1 X_3 + X_2 X_3$$

Many schemes have been broken :-/

# QC vs signatures and encryption

Minor impact on **signatures**

Just issue new post-quantum signatures

**Encryption** compromised anyway

Old ciphertexts could be decrypted

# Code-based crypto

Problem: decoding **error-correcting codes**

Schemes: McEliece (1979), Niederreiter (1986)

Limitations:

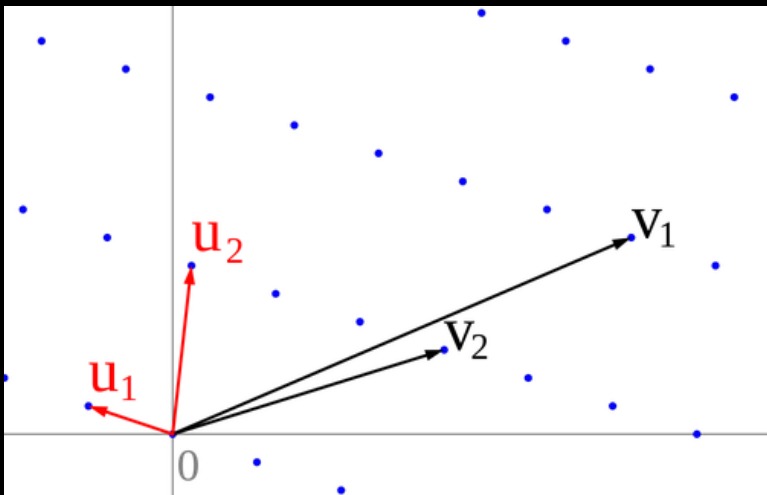
- Large keys (a few KB+)

- Fewer optimized implementations

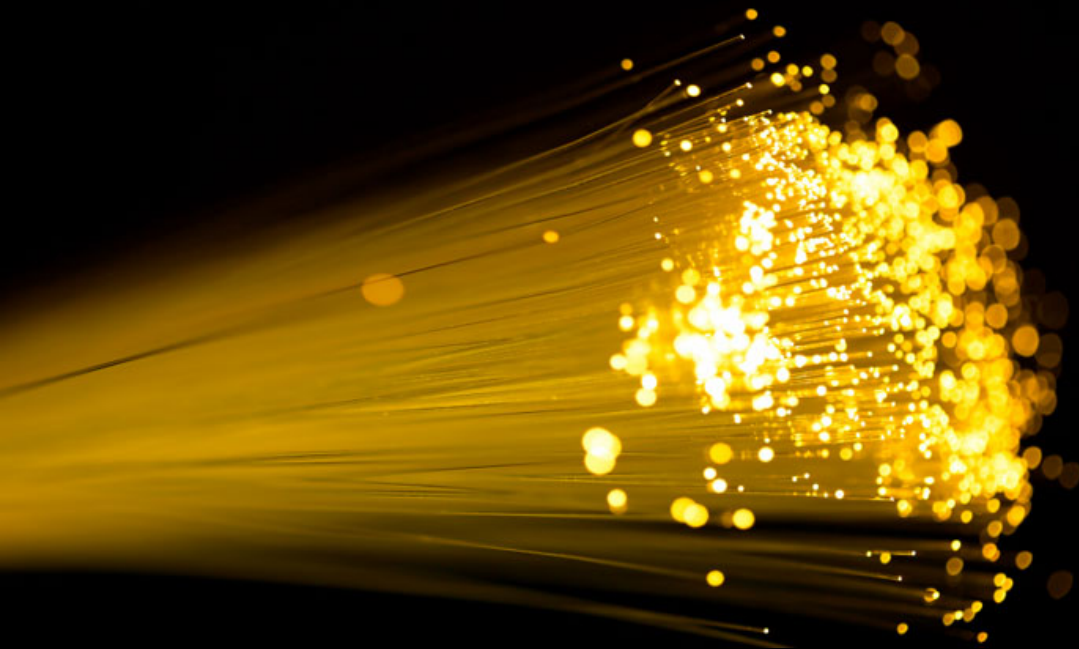
# Lattice-based crypto

Encryption and signature schemes

**Learning-with-errors:** learn a simple function given results with random noise



# 5. Quantum key distribution



# Quantum key distribution (QKD)

Establish a **shared key** between 2 parties

“Quantum Diffie-Hellman”

Not quantum computing, strictly speaking

“Security based on the laws of physics”

Eavesdropping will cause errors

Keys are truly random

## Quantum key distribution

# BB84

First QKD protocol, not really quantum

Alice's random bit	0	1	1	0	1	0	0	1
Alice's random sending basis	+	+	X	+	X	X	X	+
Photon polarization Alice sends	↑	→	↘	↑	↘	↗	↗	→
Bob's random measuring basis	+	X	X	X	+	X	+	+
Photon polarization Bob measures	↑	↗	↘	↗	→	↗	→	→

# Caveats

Like any security system, it's complicated



# Security

Quantum cryptography is secure... except when it's not

Researchers close one security hole in quantum key distribution, but seem to ...

Eventually relies on **classical crypto**

Typically with frequent key changes

**QKD implementations** have been attacked

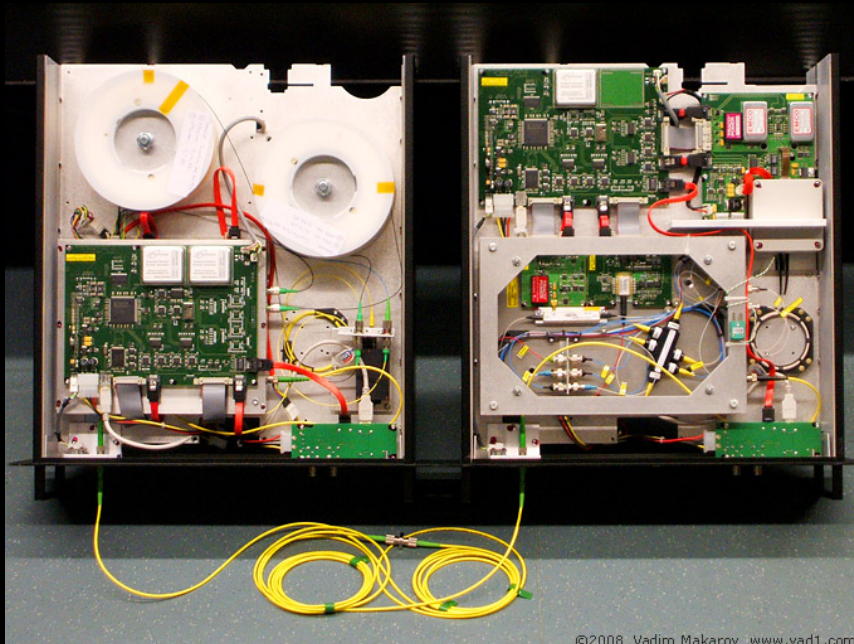
"Quantum hacking"



# Deployment

Dedicated optical fiber links

Point-to-point, limited distance ( $< 100$  km)

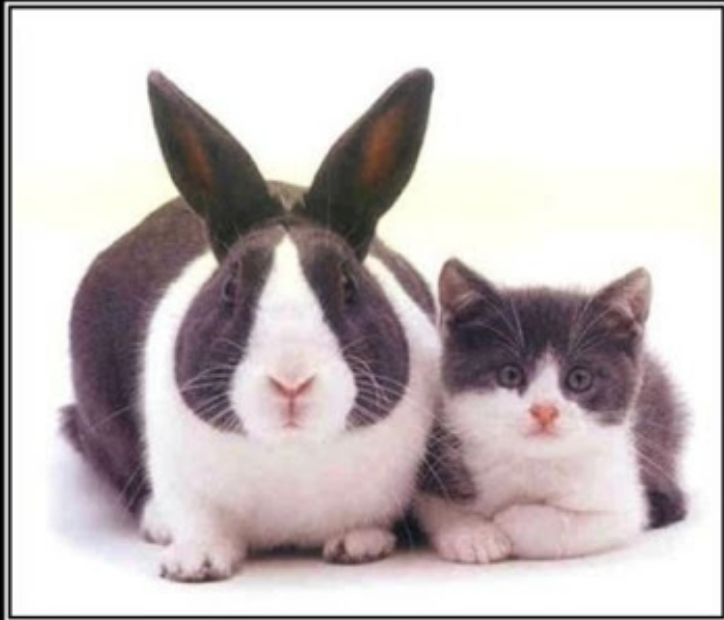


# 6. Quantum copy protection



# Quantum copy protection

Idea: leverage the **no-cloning principle**  
'cos you can't know everything about a qubit



CLONING

Results may vary

Quantum copy protection

# Quantum cash

Impossible to counterfeit, **cos' physics** (1969)

Qubits with some secret encoding

Only the bank can authenticate bills

Decentralized using (classical) pubkey crypto



# Quantum software protection

Using quantum techniques

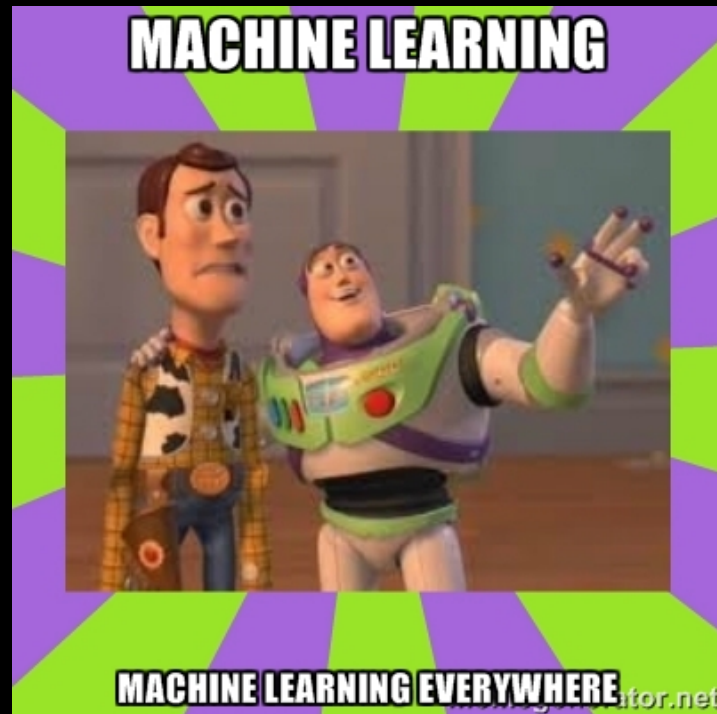
"Obfuscate" the functionality

Make copies impossible

```
verify(pwd) {  
    return pwd == "p4s5w0rD"  
} # we want to hide the password (or anything related: hash...)
```

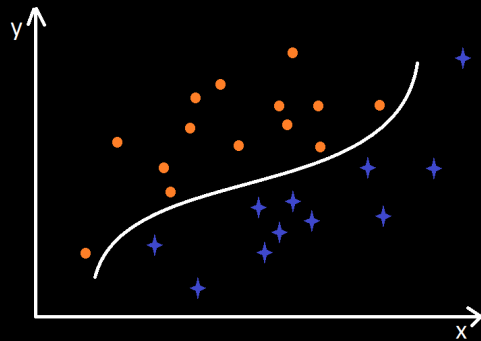
1. Turn `verify()` into a list of qubits
2. Verification: apply a transform that depends on `pwd`, then measure the qubits

# 7. Quantum machine learning

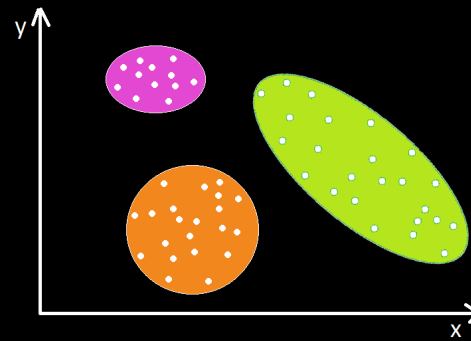


# Machine learning

“Science of getting computers to act without being explicitly programmed” —Andrew Ng



Supervised



Unsupervised

Successful for spam filtering, fraud detection, OCR, recommendation systems

# ML and security: no silver bullet

## Intrusion detection (network, endpoint)

Problem of false positives' cost

Many abnormal patterns that aren't attacks

## Vendors give neither

Details on the techniques used, nor

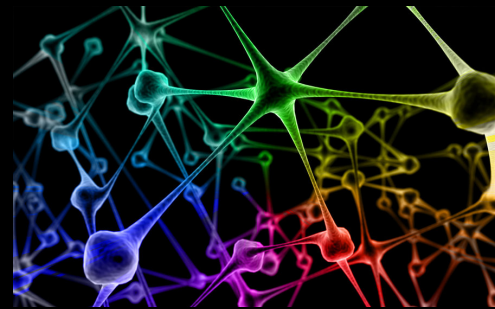
Effectiveness figures or measurements

# Quantum machine learning

“Port” of basic ML techniques to QC, like

k-means clustering

Neural networks



Many use **Grover** for a **square-root speedup**

Potential exponential speedup, but...

# Quantum RAM (QRAM)

Awesome concept

Addresses given in superposition

Read values retrieved in superposition

Many QML algorithms need QRAM

But it'd be extremely **complicated to build**



## **8. Conclusions**

# Quantum computers su\*\*

ARE NOT superfaster computers

WOULD NOT solve NP-hard problems

MAY NEVER BE BUILT anyway

**MAXIMUMPC** ▾

[Best of the Best](#) [Build a PC](#) [Features](#) [Reviews](#) [How-Tos](#)

## MIT Scientist Offers \$100k Prize To Anyone Able To Prove Quantum Computing Is Useless

Brad Chacos Feb 7, 2012

# Quantum computers are awesome

Would BREAK ALL CRYPTO deployed (pubkey)

Give new meaning and power to COMPUTING

May teach us a lot about NATURE



**Thank you!**