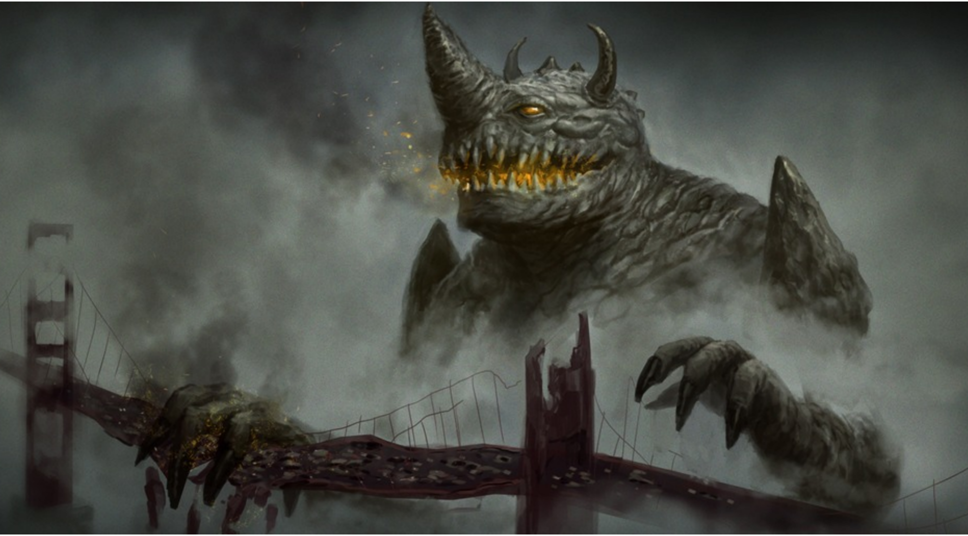
Quantum cryptanalysis – the catastrophe we know and don't know

Tanja Lange

PQCRYPTO
ICT-645622

29 Apr 2017

CataCrypt



Algorithms for Quantum Computation: Discrete Logarithms and Factoring

Peter W. Shor
AT&T Bell Labs
Room 2D-149
600 Mountain Ave.
Murray Hill, NJ 07974, USA

Abstract

A computer is generally considered to be a universal computational device; i.e., it is believed able to simulate any physical computational device with a cost in computation time of at most a polynomial factor. It is not clear whether this is still true when quantum mechanics is taken into consideration. Several researchers, starting with David Deutsch, have developed models for quantum mechanical computers and have investigated their computational properties. This paper gives Las Vegas algorithms for finding discrete logarithms and factoring integers on a quantum computer that take a number of steps which is polynomial in the input size, e.g., the number of digits of the integer to be factored. These two problems are generally considered hard on a classical computer and have been used as the basis of several proposed cryptosystems. (We

[1, 2]. Although he did not ask whether quantum mechanics conferred extra power to computation, he did show that a Turing machine could be simulated by the reversible unitary evolution of a quantum process, which is a necessary prerequisite for quantum computation. Deutsch [9, 10] was the first to give an explicit model of quantum computation. He defined both quantum Turing machines and quantum circuits and investigated some of their properties.

The next part of this paper discusses how quantum computation relates to classical complexity classes. We will thus first give a brief intuitive discussion of complexity classes for those readers who do not have this background. There are generally two resources which limit the ability of computers to solve large problems: time and space (i.e., memory). The field of analysis of algorithms considers the asymptotic demands that algorithms make for these resources as a function of the problem size. Theoretical

Universal quantum computers are coming, and are scary

- Shor's algorithm solves in polynomial time:
  - Integer factorization. RSA is dead.
  - The discrete-logarithm problem in finite fields. DSA is dead.
  - The discrete-logarithm problem on elliptic curves. ECDSA is dead.
- This breaks all current public-key cryptography on the Internet!
- Massive research effort. Tons of progress summarized in, e.g., https://en.wikipedia.org/wiki/Timeline_of_quantum_computing.

Universal quantum computers are coming, and are scary

- Shor's algorithm solves in polynomial time:
  - Integer factorization. RSA is dead.
  - The discrete-logarithm problem in finite fields. DSA is dead.
  - The discrete-logarithm problem on elliptic curves. ECDSA is dead.
- This breaks all current public-key cryptography on the Internet!
- Massive research effort. Tons of progress summarized in, e.g., https://en.wikipedia.org/wiki/Timeline_of_quantum_computing.
- Mark Ketchen, IBM Research, 2012, on quantum computing: "We re actually doing things that are making us think like, 'hey this isn't 50 years off, this is maybe just 10 years off, or 15 years off.' It's within reach."

Universal quantum computers are coming, and are scary

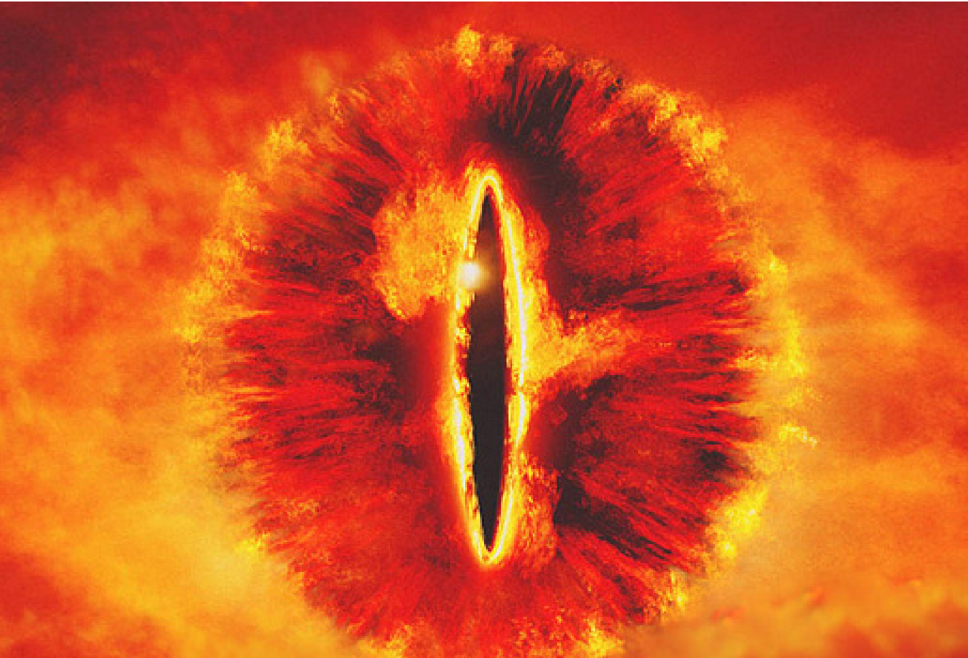
- Shor's algorithm solves in polynomial time:
  - Integer factorization. RSA is dead.
  - The discrete-logarithm problem in finite fields. DSA is dead.
  - The discrete-logarithm problem on elliptic curves. ECDSA is dead.
- This breaks all current public-key cryptography on the Internet!
- Massive research effort. Tons of progress summarized in, e.g., `https://en.wikipedia.org/wiki/Timeline_of_quantum_computing`.
- Mark Ketchen, IBM Research, 2012, on quantum computing: "We re actually doing things that are making us think like, 'hey this isn't 50 years off, this is maybe just 10 years off, or 15 years off.' It's within reach."
- Also, Grover's algorithm speeds up brute-force searches.
- Example: Only 2^{64} quantum operations to break AES-128;
  2^{128} quantum operations to break AES-256.

History of post-quantum cryptography

- 2003 Daniel J. Bernstein introduces term Post-quantum cryptography.
- PQCrypto 2006: International Workshop on Post-Quantum Cryptography.

History of post-quantum cryptography

- 2003 Daniel J. Bernstein introduces term Post-quantum cryptography.
- PQCrypto 2006: International Workshop on Post-Quantum Cryptography.
- PQCrypto 2008, PQCrypto 2010, PQCrypto 2011, PQCrypto 2013.
- 2014 EU publishes H2020 call including post-quantum crypto as topic.
- ETSI working group on "Quantum-safe" crypto.
- PQCrypto 2014.
- April 2015 NIST hosts first workshop on post-quantum cryptography
- August 2015 NSA wakes up



NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

Worse, now we get people saying “Don’t use post-quantum crypto, the NSA wants you to use it!”.

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

Worse, now we get people saying "Don't use post-quantum crypto, the NSA wants you to use it!". Or "NSA says NIST P-384 is post-quantum secure".

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

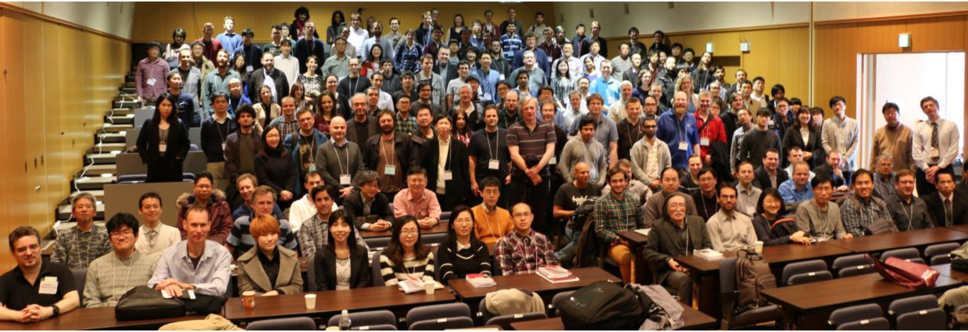
> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

Worse, now we get people saying “Don't use post-quantum crypto, the NSA wants you to use it!”. Or “NSA says NIST P-384 is post-quantum secure”. Or “NSA has abandoned ECC.”

Post-quantum becoming mainstream

- PQCrypto 2016: 22–26 Feb in Fukuoka, Japan, $>$ 200 people



- NIST is calling for post-quantum proposals; submissions due Nov 2017.
- https://2017.pqcrypto.org/ events in NL
  - Jun 19 – 23 PQCRYPTO school (Eindhoven)
  - Jun 22 – 23 ECRYPT-CSA Executive school (Eindhoven)
  - Jun 26 – 28 PQCrypto (Utrecht)

Upgrade now? Upgrade later?

- Upgrade now!
  - Rolling out crypto takes long time.
  - Every message encrypted with pre-quantum crypto is lost.
  - Need to be up & running when quantum computers come.

Upgrade now? Upgrade later?

- Upgrade now!
  - Rolling out crypto takes long time.
  - Every message encrypted with pre-quantum crypto is lost.
  - Need to be up & running when quantum computers come.
- Upgrade later!
  - Current options are not satisfactory.
  - Once rolled out, it's hard to change systems.
  - Please wait for the research results, will be much better!

Upgrade now? Upgrade later?

- Upgrade now!
  - Rolling out crypto takes long time.
  - Every message encrypted with pre-quantum crypto is lost.
  - Need to be up & running when quantum computers come.
- Upgrade later!
  - Current options are not satisfactory.
  - Once rolled out, it's hard to change systems.
  - Please wait for the research results, will be much better!
- But what about users who rely on long-term secrecy of today's communication?
- Recommend very conservative systems now; users who care will accept performance issues and gladly update to faster/smaller options later.
- Recommend now, standardize later. General roll out later.
- But: Find out now where you rely on crypto; make an inventory.
- Important to raise awareness.

Initial recommendations of long-term secure post-quantum systems

Daniel Augot, Lejla Batina, Daniel J. Bernstein, Joppe Bos,
Johannes Buchmann, Wouter Castryck, Orr Dunkelman,
Tim Güneysu, Shay Gueron, Andreas Hülsing,
Tanja Lange, Mohamed Saied Emam Mohamed,
Christian Rechberger, Peter Schwabe, Nicolas Sendrier,
Frederik Vercauteren, Bo-Yin Yang

Initial recommendations

- **Symmetric encryption** Thoroughly analyzed, 256-bit keys:
  - AES-256
  - Salsa20 with a 256-bit key

  Evaluating: Serpent-256, ...
- **Symmetric authentication** Information-theoretic MACs:
  - GCM using a 96-bit nonce and a 128-bit authenticator
  - Poly1305
- **Public-key encryption** McEliece with binary Goppa codes:
  - length $n = 6960$, dimension $k = 5413$, $t = 119$ errors

  Evaluating: QC-MDPC, Stehlé-Steinfeld NTRU, ...
- **Public-key signatures** Hash-based (minimal assumptions):
  - XMSS with any of the parameters specified in CFRG draft
  - SPHINCS-256

  Evaluating: HFEv-, ...

So, what do we know?

- Grover's algorithm:
  Assume unique $s \in \{0,1\}^n$ has $f(s) = 0$.
- Grover's algorithm finds s in $2^{n/2}$ steps.

So, what do we know?

- Grover's algorithm:
  Assume unique $s \in \{0,1\}^n$ has $f(s) = 0$.
- Grover's algorithm finds s in $2^{n/2}$ steps. Whatever a "step" is.

So, what do we know?

- Grover's algorithm:
  Assume unique $s \in \{0,1\}^n$ has $f(s) = 0$.
- Grover's algorithm finds s in $2^{n/2}$ reversible computations.

So, what do we know?

- Grover's algorithm:
  Assume unique $s \in \{0,1\}^n$ has $f(s) = 0$.
- Grover's algorithm finds s in $2^{n/2}$ reversible computations.
- Each step computes f on n qubits in superposition.
- Applying Grover's algorithm to AES: quantum resource estimates (PQCrypto 2017)
  Markus Grassl, Brandon Langenberg, Martin Roetteler, Rainer Steinwandt.
  Give very detailed analysis of costs of breaking AES-$\{128, 192, 256\}$.

Conclusion

	#gates		depth		#qubits
k	T	Clifford	T	overall	
128	$1.19 \cdot 2^{86}$	$1.55 \cdot 2^{86}$	$1.06 \cdot 2^{80}$	$1.16 \cdot 2^{81}$	$2,953$
192	$1.81 \cdot 2^{118}$	$1.17 \cdot 2^{119}$	$1.21 \cdot 2^{112}$	$1.33 \cdot 2^{113}$	$4,449$
256	$1.41 \cdot 2^{151}$	$1.83 \cdot 2^{151}$	$1.44 \cdot 2^{144}$	$1.57 \cdot 2^{145}$	6,681

Table: Resource estimates for Grover to attack AES-k, $k \in \{128, 192, 256\}$.

Conclusion:
Only `SubBytes` involves T-gates and called a minimum of 296 times (AES-128) and up to 420 (AES-256). Results in quantum circuits of quite moderate complexity. Seems prudent to move away from 128-bit keys.

Summary for Grover

- Definitely fewer than 2^{128} operations to break AES-128.
- Few qubits needed.

PQCRYPTO
ICT-645622

Summary for Grover

- Definitely fewer than 2^{128} operations to break AES-128.
- Few qubits needed.
- Pretty deep circuits, will need a very stable quantum computer to run.
- Significantly more than 2^{64} operations to break AES-128.

Summary for Grover

- Definitely fewer than 2^{128} operations to break AES-128.
- Few qubits needed.
- Pretty deep circuits, will need a very stable quantum computer to run.
- Significantly more than 2^{64} operations to break AES-128.
- Good risk management: move to AES-256; no noticeable impact on performance.

How about Shor?

- For systems that can be broken using Shor, time changes from $O(2^n)$ to $O(\mathrm{poly}(n))$.
- Far fewer operations than Grover; size pretty unclear:
  - Breaking RSA-key N with $\log_2 N = n$ needs computations on $2n$ qubits.
  - Breaking DLP needs computations on group elements and a pair of scalars.
- Shor's discrete logarithm quantum algorithm for elliptic curves
  John Proos and Christof Zalka (2003 onward)

  > *"A 160 bit elliptic curve cryptographic key could be broken on a quantum computer using around 1000 qubits while factoring the security-wise equivalent 1024 bit RSA modulus would require about 2000 qubits.*

How about Shor?

- For systems that can be broken using Shor, time changes from $O(2^n)$ to $O(\mathrm{poly}(n))$.
- Far fewer operations than Grover; size pretty unclear:
  - Breaking RSA-key N with $\log_2 N = n$ needs computations on $2n$ qubits.
  - Breaking DLP needs computations on group elements and a pair of scalars.
- Shor's discrete logarithm quantum algorithm for elliptic curves
  John Proos and Christof Zalka (2003 onward)

  > *"A 160 bit elliptic curve cryptographic key could be broken on a quantum computer using around 1000 qubits while factoring the security-wise equivalent 1024 bit RSA modulus would require about 2000 qubits.*

- Often voiced opinion: ECC will fall first.

How about Shor?

- For systems that can be broken using Shor, time changes from $O(2^n)$ to $O(\mathrm{poly}(n))$.
- Far fewer operations than Grover; size pretty unclear:
  - Breaking RSA-key N with $\log_2 N = n$ needs computations on $2n$ qubits.
  - Breaking DLP needs computations on group elements and a pair of scalars.
- Shor's discrete logarithm quantum algorithm for elliptic curves
  John Proos and Christof Zalka (2003 onward)

  > *"A 160 bit elliptic curve cryptographic key could be broken on a quantum computer using around 1000 qubits while factoring the security-wise equivalent 1024 bit RSA modulus would require about 2000 qubits.*

- Often voiced opinion: ECC will fall first.

The answer is more nuanced

- ECC group operations have special cases; no fun in superposition.
- Inversions or projective coordinates? Either way more blow up than predicted by [PZ].
- Martin Roetteler, Rainer Steinwandt + co-authors:
  - Detailed studies of quantum circuits for binary field inversion;
  - detailed studies of ECC operations in superposition;
  - so far done only for binary fields and requiring much more than 1000 qubits.
- Need same work for prime fields (there were some initiatives).
- For giant security levels RSA sure will require more qubits than same-security ECC.
- Jury is still out for crossover point of ECC and RSA (even for binary curves vs. RSA).

Further resources

Summer school on post-quantum crypto

Eindhoven, 19–23 June 2017

https://2017.pqcrypto.org/school/index.html

Executive school on post-quantum crypto

Eindhoven, 22–23 June 2017

https://2017.pqcrypto.org/exec/index.html

PQCrypto 2017

Utrecht, 26–28 June 2017

https://2017.pqcrypto.org/conference/index.html

https://pqcrypto.org: Our survey site.

https://pqcrypto.eu.org: PQCRYPTO EU project.