Quantum Computing 101: How to Crack RSA

Walter C. Daugherity
Department of Computer Science
Texas A&M University
BH2002@security.mailshell.com

Biography

Walter C. Daugherity is a Senior Lecturer in Computer Science and Electrical Engineering at Texas A&M University. He received a bachelor's degree from Oklahoma Christian University, and master's and doctor's degrees from Harvard University. His research interests include fuzzy logic, object-oriented programming, and quantum computing.

Biography (Continued)

With David A. Church he created the first course in quantum computing at Texas A&M University, which will be offered for the third time in the Fall 2002 semester.

Abstract

- What is quantum computing?
- How does it work?
- Why is it exponentially faster than “classical” computing?
- How can a quantum computer crack RSA?

Quantum Computing

- Quantum state = vector in a Hilbert space
  - Eigenstates |0> and |1> (*e.g.*, spin-up and spin-down of a spin-1/2 particle)
- Superposition
  - Combination ω_0 |0> + ω_1 |1>
  - ω = amplitude, ω^* ω = probability of eigenvalue
- Interference
  - Produced by phase angle differences
  - Constructive or destructive

go to main article

How Spin States Can Make Qubits

The spin of a particle in a dc magnetic field is analogous to a spinning top that is precessing around the axis of the field. In such a field, the particle assumes one of two states, spin up or spin down, which can represent 0 and 1 in digital logic. A particle in one spin state can be pushed toward another by a radio frequency pulse perpendicular to the magnetic field. A pulse of the right frequency and duration will flip the spin completely [top]. A shorter RF pulse will tip the spin into a superposition of the up and down state [bottom], allowing simultaneous calculations on both states. ---*IEEE Spectrum Online*

Quantum Computing

- Entanglement
  - Two mutually dependent qubits have a joint state
  - *E.g.*, the 2-qubit system (|00> + |11>)/$\sqrt{2}$ has a quantum state which cannot be “factored” into two 1-qubit states
- Teleportation
  - Reproduce a quantum state at another location
  - Initial state is destroyed in the process

Quantum Teleportation

Entire quantum particles can be "sent" from one place to another over any distance. The process starts with a sender and a receiver, Alice and Bob. The pair are on opposite sides of the universe but are in possession of photons A and B, respectively, which are entangled. Alice also holds photon C, which is in a state that she wants to teleport to Bob. Entangled particles have the property that a measurement on one immediately determines the state of the other. If Alice performs a procedure that entangles photons A and C, photon B, held by Bob, is forced to adopt the original state, a particular polarization, say, of photon C. Bob can only measure this state if Alice sends him details of the type of experiment he must do to get the message, and this can only be done at or below the speed of light. Although only the quantum state of photon C is teleported, when photon B adopts this state, it cannot be distinguished from photon C. To all intents and purposes, it has become photon C. This is what physicists mean when they say photon C has been teleported from Alice to Bob.

Teleportation was first demonstrated by a group of researchers at the University of Innsbruck using the experimental setup shown here. Pairs of entangled photons, with polarization orthogonal to each other, are generated by splitting an ultraviolet laser pulse using a crystal called a parametric down-coverter. One of the pair (photon A) is sent to Alice while the other (photon B) is sent to Bob. Meanwhile, a message photon (C) is prepared in a state that is to be teleported to Bob-- in this case, 45-degree polarization. This is sent to Alice and arrives coincidentally with photon A at a beam-splitter. If the photons leave the splitter and strike both detectors, they have become entangled, and Alice sends notice of the entanglement to Bob. Bob can then carry out a measurement on photon B to confirm that it is in the 45-degree polarization state that the message photon C started off in. *---IEEE Spectrum Online*

--J.M.

Quantum Computing

- Quantum Cryptography
  - Relies on Heisenberg's uncertainty principle: Can't measure rectilinear and diagonal polarization simultaneously, so eavesdropping is detected
  - *I.e.*, provably secure

Exponential Speedup

- *N* qubits can hold 2^N values in superposition, *i.e.*, simultaneously
- A single operator (function evaluation) on such a register evaluates the function for all 2^N values in the time it would take to do one evaluation

Application to Cryptography

- Conventional (private key) cryptography
- Public key cryptography
- RSA
- Cracking RSA
- Shor’s quantum algorithm

Conventional Encryption

- M = one block of the message, typically 64 bits, *i.e.*, 8 characters, of plaintext
- K = secret key
- Ciphertext C = E(M,K)

Conventional Decryption

- C = one block of ciphertext
- K = secret key
- M = D(C,K), the original plaintext

Security of Conventional Encryption

- Need a strong encryption algorithm: even with many plaintext/ciphertext pairs an opponent cannot decrypt other ciphertext or discover the key.
- Sender and receiver need to obtain copies of the secret key securely and keep it secure.
- Note: Key is secret, algorithm is not.

Guessing the Secret Key

Key Size (bits)	Number Of Keys	Time at 1 us each	Time at 1 ps each
32	4.3×10^{9}	35.8 min	2.15 ms
56	7.2×10^{16}	1142 yr	10 hr
128	3.4×10^{38}	5.4×10^{24} yr	5.4×10^{18} yr
168	3.7×10^{50}	5.9×10^{36} yr	5.9×10^{30} yr

Why Public-Key Cryptography?

- Key distribution
  - Secret keys for conventional cryptography
  - Unforgeable public keys (digital certificate)
- Message authentication

Public-Key Encryption

- M = one block of the message, typically 64 bits, *i.e.*, 8 characters, of plaintext
- KU = receiver's public key
- Ciphertext C = E(M,KU)

Public-Key Decryption

- C = one block of ciphertext
- KR = receiver’s private (secret) key
- M = D(C,KR), the original plaintext

Public History of Public-Key Encryption

- 1976 - Proposed by Diffie and Hellman
  - Relies on difficulty of computing discrete logarithms (solve $a^x = b \bmod n$ for x)
- 1977 - RSA algorithm developed by Rivest, Shamir, and Adleman
  - Relies on difficulty of factoring large numbers
  - RSA129 (129 digits) published as a challenge

Public History of Public-Key Encryption (continued)

- 1994 - RSA129 cracked by 1600 networked computers
- 1999 - RSA140 cracked by 185 networked computers in 8.9 CPU-years
- 1999 – RSA155 (512-bit key) cracked by 300 networked computers
- 2002 – RSA recommends 1024-bit keys

The RSA Algorithm

- Select two primes p and q
- Calculate n = p q
- Calculate $\phi(n) = (p-1)(q-1)$
- Select e such that $1 < e < \phi(n)$ and $\gcd(\phi(n),e) = 1$
- Calculate $d = e^{-1} \bmod \phi(n)$
- Public key KU = {e,n}
- Private key KR = {d,n}

Example

- Select two primes p=7 and q=17
- Calculate n = p q = 119
- Calculate $\phi(n) = (p-1)(q-1) = 96$
- Select e such that $1 < e < \phi(n)$ and $\gcd(\phi(n),e) = 1$, *e.g.*, e = 5
- Calculate $d = e^{-1} \bmod \phi(n)$, *e.g.*, d = 77
- Public key KU = {e,n} = {5,119}
- Private key KR = {d,n} = {77,119}

Example (continued)

- Plaintext M = 19
- Ciphertext C = M^e mod n = 19^5 mod 119 = 66
- Plaintext M = C^d mod n = 66^{77} mod 119 = 19

Cracking RSA

- Factor n, which is public, yielding p and q
- Calculate $\phi(n) = (p-1)(q-1)$
- Calculate $d = e^{-1} \bmod \phi(n)$ (e is public)
- Private key KR = {d,n}

Cracking RSA (Example)

- Factor 119, which is public, yielding 7 and 17
- Calculate $\phi(119) = (7-1)(17-1) = 96$
- Calculate $5^{-1} = 77 \bmod 96$
- Private key KR = {77,119}

So How Hard is Factoring?

Year	Decimal Digits	MIP-Years
1964	20	0.000009
1974	45	0.001
1984	71	0.1
1994	129	5000
?	2000	2.9×10^9

Shor's Algorithm to Factor n

- Choose q (with small prime factors) such that $2n^2 <= q <= 3n^2$
- Choose x at random such that gcd(x,n)=1
- Calculate the discrete Fourier transform of a table of x^a mod n, order log(q) times, each time yielding some multiple of q/r, where r=period

Shor's Algorithm (continued)

- Use a continued fraction technique to determine r
- Two factors of n are then gcd($x^{r/2}$ - 1,n) and gcd($x^{r/2}$ + 1,n)
- If the factors are 1 and n, try again.

Key Features

- The discrete Fourier transform maps equal amplitudes into unequal amplitudes, so measuring the quantum state is more likely to yield a result close to some multiple of 1/r.
- The period can be “quantum-computed”efficiently.

Implementation

- “By 2000, it is expected that a quantum computer will factor 15 = 3 * 5.”
- Scaling up for larger numbers is theoretically unlimited
- If you can build a big enough quantum computer, you can crack RSA-1024 (about 300 decimal digits) in your lifetime.

The Future

- Quantum-effect memory
- Special-purpose experimental computers
- Commercial availability
- Impact on public-key cryptography

For Further Information

http://www.qubit.org

http://feynman.media.mit.edu/quanta/nmrqc-darpa/index.html

http://www.theory.caltech.edu/~quic/index.html

http://qso.lanl.gov/qc/

http://www.research.ibm.com/quantuminfo/

Reference Sites

http://www.theory.caltech.edu/people/preskill/ph229/references.html

http://www.duke.edu/~msm7/phy100/References.html

http://www.magiqtech.com/QIref.html

http://www.cs.caltech.edu/~westside/quantum-intro.html

http://www.cs.umbc.edu/~lomonaco/qcomp/Qcomp.html

http://gagarin.eecs.umich.edu/Quantum/papers/

http://astarte.csustan.edu/~tom/booklists/qc-refs-2001.pdf

http://www.stanford.edu/~zimmej/T361/Final%20Project/references.htm

Fall 2002 Course
Intro to Quantum Computing
ELEN 689-607 / PHYS 689-601
Texas A&M University

Instructors: Dr. Walter C. Daugherity

Dr. David A. Church

Recommended prerequisites are a knowledge of linear algebra (e.g., MATH 304) and one course in physics.

Enrollment is limited.