NIST's Post-Quantum Cryptography Project

Rene Peralta
NIST PQC team

NUTMIC 2017 Warsaw

Quantum Computers

- Potentially much more powerful than classical computers
  - Conjecture: A classical computer needs **exponential time** to simulate a quantum computer (in the general case)
  - Conjecture: quantum computers cannot solve NP-hard problems in polynomial time.
- Exponential speedups
  - Simulating the dynamics of physical processes
  - Factoring large integers (Shor's algorithm)
  - Discrete logarithms in any abelian group (Shor's algorithm)
- And some polynomial speedups
  - Unstructured search (Grover's alg.), collision finding

Implications for Crypto

- “Large” quantum computers would break most of our public-key crypto
  - RSA, Diffie-Hellman key exchange, elliptic curve crypto
- Symmetric crypto would be affected, but not broken
  - Keys will have to be longer.

Long-term privacy and security implications

- Full transition to alternatives takes a long time (> 10 years).
- Today’s data needs to remain secure 5-10 years (longer in some cases, such as medical data).

NIST's PQC project

- To monitor progress in quantum computers and quantum algorithms.
- To find and standardize quantum-resistant alternatives for PKE, key-agreement, and digital signatures.
- To ensure transparency of the process and legitimacy of the outcome.

Not a Competition

- We hope at the end of the day there will be significant community consensus.

- We may standardize several algorithms.

- The evaluation criteria is not set in stone, it may evolve during the next few years.

The Call For Proposals

- Candidate algorithms may now be submitted http://csrc.nist.gov/groups/ST/post-quantum-crypto/cfp-announce-dec2016.html

- Deadline is November 30, 2017

The PQC Forum

- The wording of the CFP followed public discussion on the pqc-forum (pqc-forum@nist.gov).
- This is also where submissions and germane issues - such as evaluation criteria - will be discussed.
- To join send mail to pqc-forum-request@nist.gov with subject=subscribe

Proposals sought

- Public-key encryption
- Key-encapsulation
- Digital signature

Out of scope for this CFP but still of interest to the PQC project

- Stateful hash-based signatures
- Hybrid modes

Post-Quantum Cryptography

Cryptosystems	Hard problem	Trapdoor
Lattice-based	Finding short vectors in a high-dimensional lattice	Nice basis for the lattice (short, almost-orthogonal vectors)
Code-based	Decoding a random binary linear code	Linear trans-formations that reveal structure of the code
Multivariate	Solving a random system of multivariate quadratic equations over a finite field	Linear trans-formations that reveal structure of the equations

More …

- **Stateless hash-based signatures**
  - May be too big …
- **Isogenies of supersingular elliptic curves**
  - Useful for key exchange?
- **Quantum key distribution**
  - Information-theoretic security
  - Requires optical fiber, distance limited to ~200 km
  - Chinese model …

Security Evaluation

- Cryptanalysis: what are the best known attacks?

- Foundations: do we believe an underlying primitive is hard for quantum computers? (in practice we are likely to see two assertions:
  - problem is hard for classical computers;
  - No clear quantum speedup beyond Grover's.
- Security proofs can reduce hardness to that of an underlying primitive.

How well do these cryptosystems work in practice?

- Size of keys, time/circuit complexity
- Size of messages, size of signatures
- Ease of implementation, how to set the parameters
- Does it fit nicely with TLS, other higher-level protocols?
- Vulnerabilities to side channel attacks?

LWE Problem (“learning with errors”)

- Secret s in $(Z_q)^n$
  - $q = \text{poly}(n)$
- Given (enough) samples (a,b) in $(Z_q)^n \times Z_q$
  - a is uniformly random
  - $b = a^T s + e$, where e is Gaussian distributed, w/ std dev q/poly(n)
- Can we determine s?
  - “Decoding a random linear code over Z_q”

- **Claim: samples (a,b) look pseudorandom!**

How Things Look Like Now

- Signatures: hash-based , code-based, lattice-based, multivariate…
- PKE : lattice-based, code-based, multivariate, …
- Key agreement: PKE, lattice-based, isogeny-based, …

How Things Look Like Now

- Speed looks good.
- Key sizes may increase significantly.
- Some signature sizes look big.
- Possibly significant increase in ciphertext size for short plaintexts.
- **We need industry to do an impact assessment.**

Public Discussion

- Ongoing discussion regarding “security-levels” and derived parametrization.
- Suspicion that NIST is just doing NSA’s bidding.
- Demands that future standards make bad implementations harder.

TIMELINE

Dec 20, 2016	Formal Call for Proposals
Nov 30, 2017	Deadline for submissions
Early 2018	Workshop - Submitter's Presentations
3-5 years	Analysis Phase - NIST will report findings *1-2 workshops during this phase*
2 years later	Draft Standards ready

THANKS