Cybersecurity in Quantum Time

NIST PQC Standardization

Lily Chen
Computer Security Division, Information Technology Lab
National Institute of Standards and Technology (NIST)

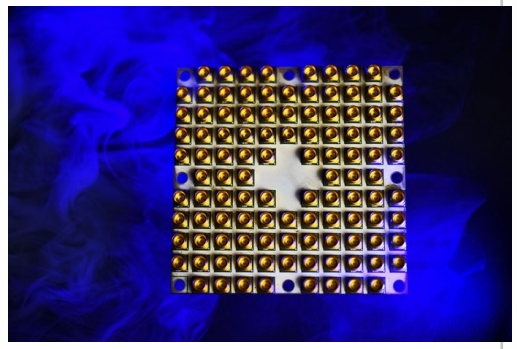
Outline

- Quantum Computers
- Impacts to Today's Cybersecurity
- NIST Initiatives
  - Post-Quantum Cryptography (PQC)
  - NIST Process in Standardizing PQC
  - Next Step Plan

Quantum Computers

- Exploit quantum mechanics to process information
- Use quantum bits = "qubits" instead of 0's and 1's
- Superposition – ability of quantum system to be in multiple states at the same time
- Potential to vastly increase computational power beyond classical computing limit
- Limitations:
  - When a measurement is made on quantum system, superposition collapses
  - Only good at certain problems
  - Quantum states are very fragile and must be extremely well isolated
  - Intersection of many developing fields: superconductors, nanotechnology, quantum electronics, etc…

IBM's 50-qubit quantum computer
November 2017

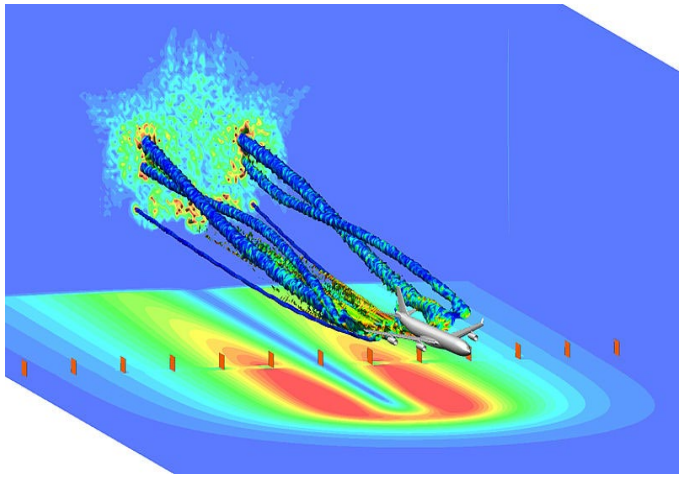


Intel's 49-qubit chip "Tangle-Lake"
January 2018

Google's 72-qubit chip "Bristlecone"
March 2018

Quantum Computers – New Paradigm

Design new materials and drugs

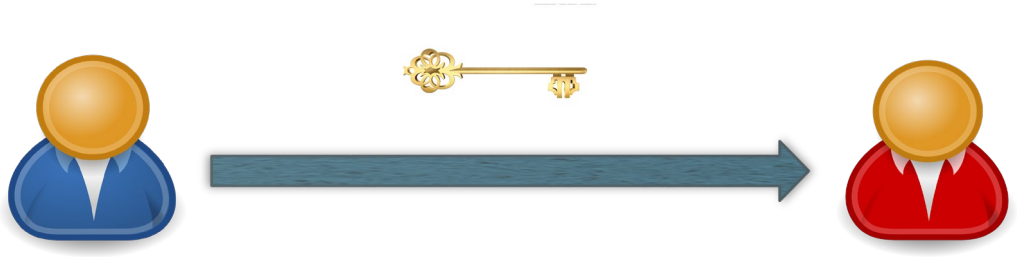


Simulation and data processing

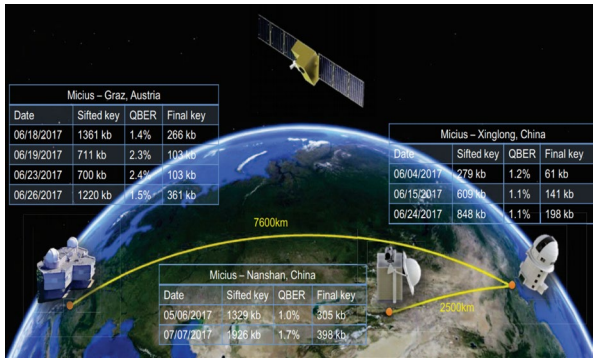
Sensing and measuring

- Known to solve many problems previously thought to be intractable

Quantum Key Distribution



- Using quantum mechanics to enable two parties to share a random secret key
- It can solve key distribution problem when quantum interface is available in a pairwise manner
- Today's many-to-many network such as Internet uses public key cryptography to establish keys for data protection



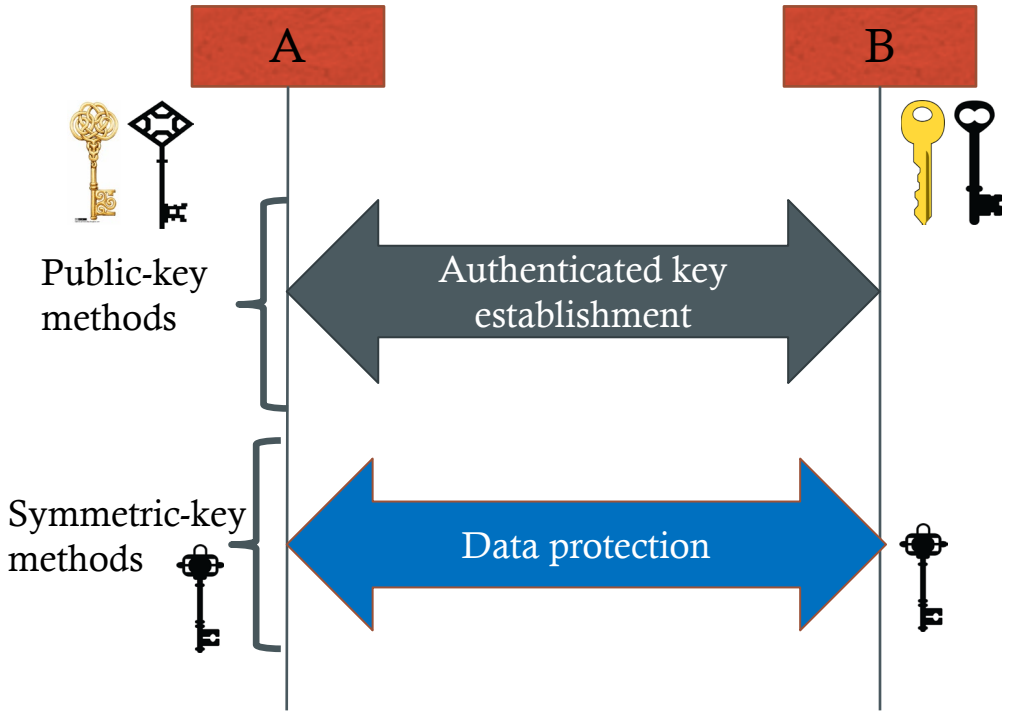
Courtesy of Qiang Zhang, USTC

Beijing – Shanghai QKD Backbone

Tokyo QKD network
http://www.uqcc.org/QKDnetwork/

Today's Usage of Public-Key and Symmetric-Key Crypto

- To set up communication protection, public key and symmetric key cryptography schemes are used together, e.g. TLS, IPsec, etc.
  - Use public key cryptography to establish keys and authenticate users through signatures
  - Use symmetric key cryptography to encrypt and authenticate bulk data
- For trusted platform, signature is used for software authentication and authorization, while symmetric key algorithms are used for secure storage



Why Public-Key Cryptography is Secure?

- A problem is hard if no polynomial time algorithm is known to solve it
- The hardness is categorized by computing complexity - generally expressed as a function $n \rightarrow f(n)$, where n is the size of the input, e.g.
  - If $f(n)$ is a polynomial, then the problem is not hard
  - If $f(n) = c \cdot e^{h(n)}$ i.e. exponential, then the problem is hard
- Practically, it means that it is infeasible to solve it with the currently available computing resource
- The hardness on certain problems is used as the basic assumptions for some cryptographic schemes, e.g.
  - RSA is based on the hardness of integer factorization, given integer n $(= p \cdot q)$ find p and q
  - Diffie-Hellman key agreement is based on the hardness of discrete logarithm problem, given $y \in$ GF(p)* and generator g , find x , such that $y = gx \, mod \, p$

Quantum Impact

- Quantum computing changed what we have believed about the hardness of discrete log and factorization problems
  - Using quantum computers, an integer n can be factored in polynomial time using Shor's algorithm
  - The discrete logarithm problem can also be solved by Shor's algorithm in polynomial time
- As a result, the public key cryptosystems deployed since the 1980s will need to be replaced
  - RSA signatures, DSA and ECDSA (FIPS 186-4)
  - Diffie-Hellman Key Agreement over finite fields and elliptic curves (NIST SP 800-56A)
  - RSA encryption (NIST SP 800-56B)
- We have to look for quantum-resistant counterparts for these cryptosystems
- Quantum computing also impacted security strength of symmetric key based cryptography algorithms
  - Grover's algorithm can find AES key with approximately $\sqrt{2^n}$ operations where n is the key length
  - Intuitively, we should double the key length, if 2^{64} quantum operations cost about the same as 2^{64} classical operations

Security Strength – Classical vs. Quantum

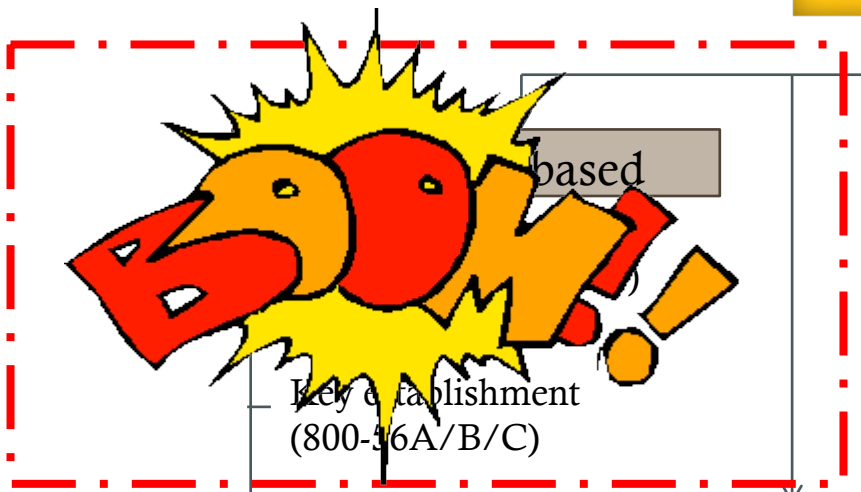
Algorithm/key length	Classical Security	Quantum security
RSA (|n|=2048)	112 bits	≈ 0 bits
Diffie-Hellman (|p| = 2048)	112 bits	≈ 0 bits
ECDSA* with group size |q| = 256	128 bits	≈ 0 bits
AES-128	128 bits	64 bits
AES-192	192 bits	96 bits
AES-256	256 bits	128 bits

*ECDSA stands for Elliptic Curve Digital Signature Algorithm, signature over a subgroup of curve $E(F_p)$ with size q.

NIST Cryptographic Standards

- NIST developed the first encryption standards in 1970s, Data Encryption Standards (DES), and published as Federal Information Processing Standard (FIPS) 47
- Over 40 years, NIST continues to evolve its cryptographic standards to keep pace with new cryptographic technologies and advanced analysis methods
  - 1997-2000 NIST held a block cipher competition and selected a new block cipher algorithm Advanced Encryption Standard (AES), specified in FIPS 197
  - 2007-2012 NIST held a hash function competition and selected a hash function names SHA-3 specified in FIPS 202
- In late 1980s and beginning of 1990s, NIST started to standardize public key cryptography for Internet and e-commerce need
  - SP 800-56A (key agreement, e.g. Diffie-Hellman, elliptic curve Diffie-Hellman)
  - SP 800-56B (RSA based encryption/key transport)
  - FIPS 186 (RSA signatures, DSA. ECDSA)

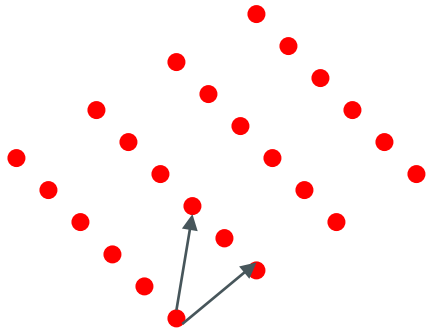
Quantum Impact to NIST Cryptography Standards

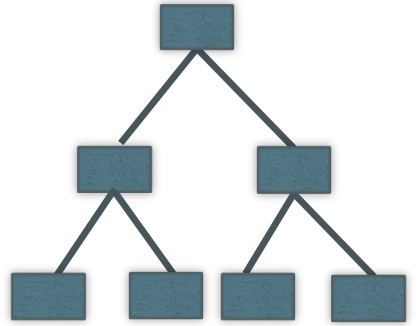


- Crypto standards
  - based
    - Key establishment (800-56A/B/C)
  - Tools
    - RNG (800-90A/B/C)
    - KDF (800-108, 800-135)
  - Symmetric key based
    - AES (FIPS 197)
      TDEA (800-67)
    - Modes of operations (800 38A-38G)
    - SHA-1/2 (FIPS 180) and SHA-3 (FIPS 202)
    - Randomized hash (800-106)
    - HMAC (FIPS 198)
    - SHA3 derived functions (parallel hashing, KMAC, etc. (800-185)
  - Guidelines
    - Hash usage/security (800-107)
    - Transition (800-131A)
    - Key generation (800-133)
    - Key management (800-57)

Post-Quantum Cryptography (PQC)

- Post-quantum cryptography algorithms are classical cryptographic algorithms which are considered to be able to resist quantum attacks
  - They must be based on hard problems which are still hard even when large scale quantum computers become available
- Some actively researched PQC categories
  - Lattice-based
  - Code-based
  - Multivariate
  - Hash based signatures
  - Isogeny-based schemes



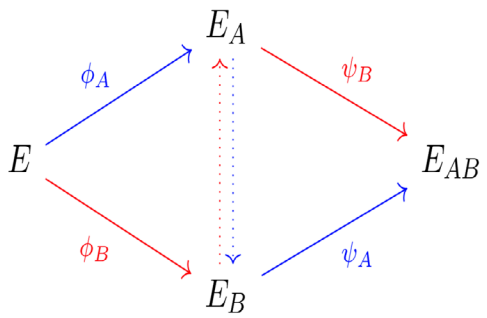


$$p^{(1)}(x_1,\ldots,x_n) = \sum_{i=1}^{n}\sum_{j=i}^{n} p_{ij}^{(1)} \cdot x_i x_j \quad + \quad \sum_{i=1}^{n} p_i^{(1)} \cdot x_i + p_0^{(1)}$$

$$p^{(2)}(x_1,\ldots,x_n) = \sum_{i=1}^{n}\sum_{j=i}^{n} p_{ij}^{(2)} \cdot x_i x_j \quad + \quad \sum_{i=1}^{n} p_i^{(2)} \cdot x_i + p_0^{(2)}$$

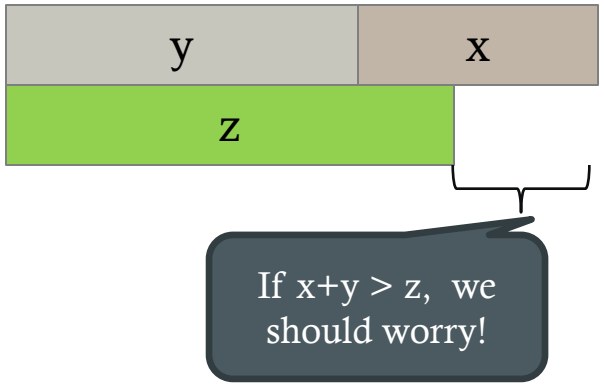
$$\vdots$$

$$p^{(m)}(x_1,\ldots,x_n) = \sum_{i=1}^{n}\sum_{j=i}^{n} p_{ij}^{(m)} \cdot x_i x_j \quad + \quad \sum_{i=1}^{n} p_i^{(m)} \cdot x_i + p_0^{(m)}$$



PQC Standardization – Is it too early?

- It has been a long debate among researchers and practitioners on whether it is too early to look into PQC standardization
- "A one-in-seven chance that some fundamental public-key crypto will be broken by quantum by 2026, and a one-in-two chance of the same by 2031" – Michele Mosca, U. of Waterloo)
- The experience tells that we need at least several years to develop and deploy PQC standards
- If we require 5-year backward secrecy, we certainly need to start standardization



- y is the time taken for developing and deploying PQC standards
- x is the time for "backward secrecy" (maintain secrecy for the information encrypted x years ago)
- z is the time before quantum computers are available

NIST PQC Standardization – Preparation

- 2012 – NIST begin PQC project
  - Research and build NIST team
- April 2015 – 1st NIST PQC workshop
- Feb 2016 – NIST Report on PQC (NISTIR 8105)
- Feb 2016 – NIST preliminary announcement of standardization plan

NIST PQC Standardization – Call for Proposals

- Dec 2016 – Announcement of call for proposals with requirements and criteria (Federal Register Notice)
- The scope of submission
  - Public key encryption /Key establishment
  - Digital signature
- Security Notions
  - Signature - Existentially unforgeable with respect to adaptive chosen message attack (EUF-CMA)
    - Assume the attacker has access to no more than 2^{64} signatures for chosen messages
  - Encryption - Semantically secure with respect to adaptive chosen ciphertext attack (IND-CCA2)
    - For KEM with ephemeral keys, use IND-CPA security notion
    - Assume the attacker has access to no more than 2^{64} decryptions for chosen ciphertexts

Security Categories for submissions

Level	Security Description
I	At least as hard to break as AES128 (exhaustive key search)
II	At least as hard to break as SHA256 (collision search)
III	At least as hard to break as AES192 (exhaustive key search)
IV	At least as hard to break as SHA384 (collision search)
V	At least as hard to break as AES256 (exhaustive key search)

- Computational resources should be measured using a variety of metrics
- NIST asked submitters to focus on levels 1,2, and 3
  - Levels 4 and 5 for high security
- These are understood to be preliminary estimates

Other properties

- Drop-in replacements - Compatibility with existing protocols and networks such as TLS, IKE, etc.
- Perfect forward secrecy, like ephemeral Diffie-Hellman
  - Using one-time key requires fast key generation
- Resistance to side-channel attacks
  - Constant time implementation is ideal, because countermeasures add implementation burden
- Misuse resistance, and
- More

Complexities of PQC Standards

- Scope with three main cryptographic primitives (encryption, key establishment, signature)
- Both classical attacks and quantum attacks
- Both theoretical and practical aspects
- Multiple factor tradeoffs (security, key sizes, signature sizes, ciphertext expansion, etc.)
- Migrations, and
- Anything which we have never handled in the previous standards

Submissions to NIST Call for Proposals

- 82 total submissions received from 26 Countries, 6 Continents
- 69 accepted as "complete and proper" (5 since withdrawn) in December 2017

	Signatures	KEM/Encryption	Overall
Lattice-based	5	21	26
Code-based	2	17	19
Multi-variate	7	2	9
Stateless Hash-based/Symmetric based	3		3
Other	2	5	7
Total	**19**	**45**	**64**

Evaluation of the 1st Round

- NIST team had seminars to present each candidate by team members to understand how it works, look into security analysis provided by the submitters, raise questions, discuss pros and cons, etc.
- Security analysis
  - Research publications at conferences and journals (e.g. PQCrypto)
  - Official comments - Over 300 official comments
  - E-mail discussions at pqc-forum – 926 posts
- Performance
  - Evaluation resources include
    - NIST's internal testing with submitters' code
    - Preliminary benchmarks – SUPERCOP, OpenQuantumSafe, etc.

Selection of second round candidates

- Security
  - Candidates which were broken, significantly attacked, or difficult to establish confidence in their security were left out
  - Candidates which provided clear design rationale and reasonable security proofs to established reasonable confidence in security are advanced
- Performance
  - Candidates with obvious performance or key/signature/ciphertext size issues for existing applications were not advanced - even though they might have been well prepared with good ideas

The 2nd round candidates

KEM/Enc

Lattice –based (9):
Crystals-Kyber; FrodoKEM; LAC; NewHope; NTRU; NTRU Prime; Round 5; Saber; Three Bears

Code –based (7):
Classic McEliece; NTS-KEM; BIKE; HQC; Rollo; LEDAcrypt; RQC

Isogeny –based (1):
SIKE

Signature

Lattice –based (3):
Crystals-Dilithium; Falcon; qTESLA

Symmetric –based (2) :
Sphincs+; Picnic

Multivariate (4):
GeMSS; LUOV; MQDSS; Rainbow

* See NISTIR 8240 for a summary of each of the 2nd round candidates

Second round evaluation

- NIST will hold the 2nd PQC Standardization Conference August 22-24, 2019 in Santa Barbara (right after crypto 2019)
- Security is very critical and we have a lot to understand, e.g.
  - Generic vs. structured (e.g. LWE vs. R-LWE) – Structured have smaller key sizes and/or are more efficient
  - Security impact on optimized versions – how far an optimization can go to maintain security
  - Newer security assumptions
- Performance evaluation is important to make the future standards useable
  - Performance (hardware + software) will play much more of a role in the second round
  - More benchmarks through different platforms and implementations
  - Evaluate how candidates fit into applications/protocols and identify show stoppers

Preparation for Migration

- Enable crypto agility for each function (public key encryption/key encapsulation, signature) when it is possible
- Understand implementation costs and required bandwidth/space for transmitting and storing keys, signatures and ciphertext
- Discuss tradeoff preferences in each application – identify special restrictions, limitations, and show stoppers
- Gain first-hand experience through trial implementations e.g. hybrid mode or dual signatures as a temporary solution
- Do not commit to a specific candidate for long-term products until NIST makes its selection for standardization

Future plans

- The 2nd PQC Standardization Conference will be held in August 2019
- Spend 12-18 months to analyze and evaluate the 2nd round candidates
- Start a 3rd round and/or select algorithms to standardize 2020-2021
- Release draft standards in 2022-2023 for public comments

Information on NIST PQC Standardization

- For NIST PQC project, please follow us at
  https://www.nist.gov/pqcrypto
- To submit a comment, send e-mail to pqc-comments@nist.gov
- Join discussion mailing list pqc-forum@nist.gov