Impact of Quantum Technologies to Cryptography
Tutorial – Part I

Ludovic Perret (ludovic.perret@lip6.fr)

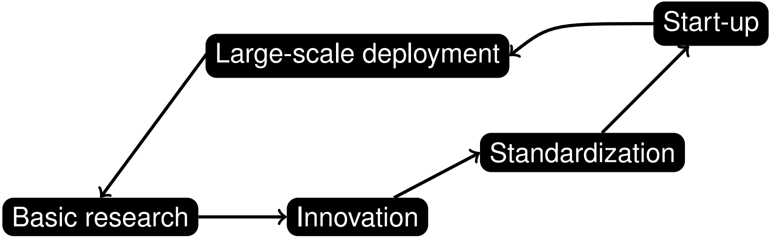
Sorbonne University/CNRS
Co-founder of CryptoNext Security

Computability in Europe 2023, 24th-28th July 2023, Batumi, Georgia

Introduction & Organization of the Tutorial

Post-Quantum Cryptography

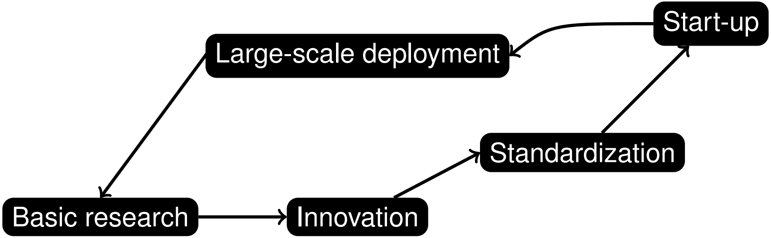
Cryptosystems secure both against classical and quantum adversaries



Introduction & Organization of the Tutorial

Post-Quantum Cryptography

Cryptosystems secure both against classical and quantum adversaries

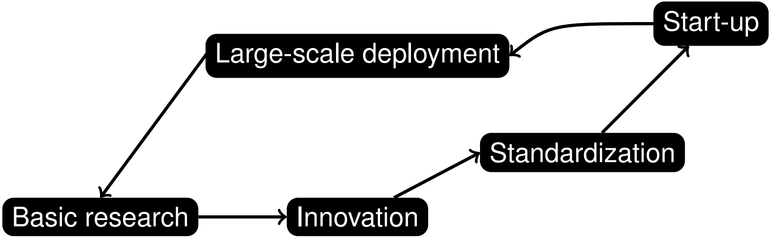


Part I. Cryptography in the era to quantum technologies

Introduction & Organization of the Tutorial

Post-Quantum Cryptography

Cryptosystems secure both against classical and quantum adversaries



Part I. Cryptography in the era to quantum technologies

Part II. On the use of quantum algorithms in cryptanalysis

Part III. A zoom on the design of post-quantum signature schemes

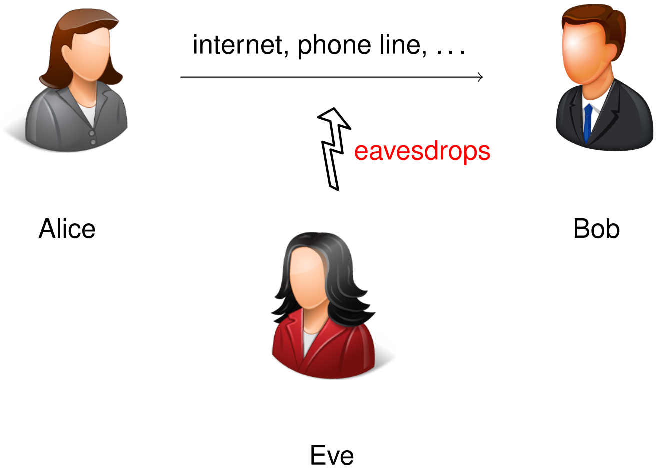
Outline

1. Cryptography Warm-Up
2. Quantum Impact
3. Transition toward quantum-resistant infrastructure

Outline

The basic goal of cryptography

Secure communication



Information security objectives

confidentiality	keeping information secret from all but those who are authorized to see it
integrity	ensuring information has not been altered by unauthorized or unknown means
authentication	corroborating the source of information

anonymity	concealing the identity of an entity involved in some process
non-repudiation	preventing the denial of previous commitments or actions
etc	. . .

Cryptography in the old time

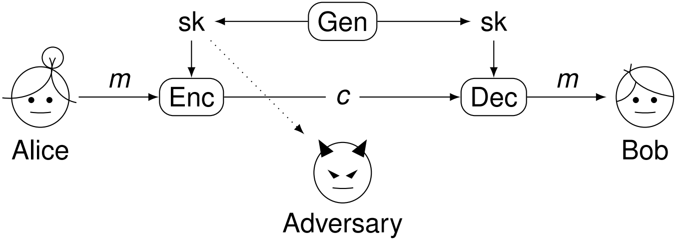


Figure: `Enigma` machine

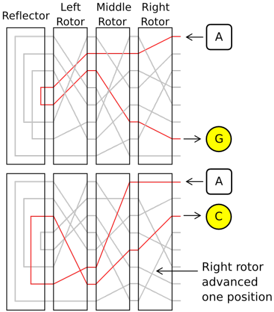


Figure: `Enigma` principle

Cryptography in the old time

Figure: `Enigma` machine

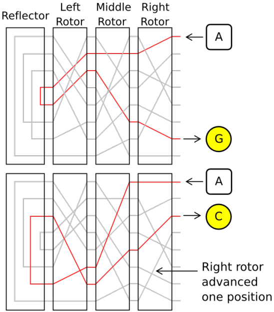


Figure: `Enigma` principle

Reputed unbreakable

The rise of computers

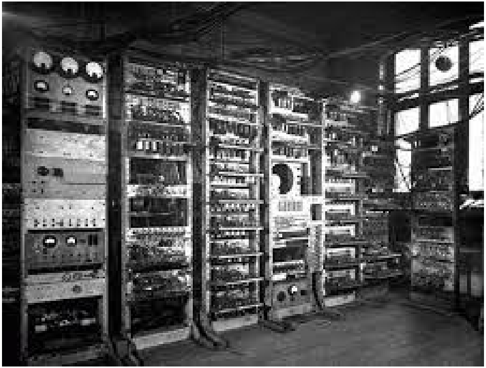


Figure: Turing's computer

Figure: Alan Turing

Full cryptanalysis of `Enigma` (and similar mechanical machines)

☛ Technology took cryptography down

How to formalize security ?

Figure: Claude Shannon

Intuition. Attacker should not be able to compute any information about m

Definition

An encryption scheme is **perfectly secret** (or Information Theoretically Secure, ITS) if for every random variable M, every message $m \in \mathcal{M}$ and every ciphertext $c \in \mathcal{C}$ with $\Pr(C = c) > 0$:

$$\Pr(M = m) = \Pr(M = m | C = c)$$

A perfectly secure scheme: one-time pad

Description

- Let $\ell \in \mathbb{N}$ be a parameter and $\oplus$ denotes component-wise XOR
  - Message space $\mathcal{M} = \{0,1\}^\ell$
  - Key space $\mathcal{K} = \{0,1\}^\ell$
- **Vernam's cipher:** $\mathrm{Enc}(K, m) = m \oplus K$ and $\mathrm{Dec}(K, c) = c \oplus K$

Figure: Red phone

- ☛ One-time pad is perfectly secret!
- ☛ Each key cannot be used more than once!
- ☛ Key is as long as the message
- ☛ One time-pad is optimal in the class of perfectly secret schemes

Block-ciphers

Problems

- the plaintexts and keys may be extremely long

Block-ciphers

Problems

- ❑ the plaintexts and keys may be extremely long

Idea

- ☛ Design ciphers that work on small blocks
- ☛ Expand the encryption key from a fixed-size secret-key

Block-ciphers

Idea

- ☛ Design ciphers that work on small blocks
- ☛ Expand the encryption key from a fixed-size secret-key

Description

$$\begin{aligned} \mathrm{Enc}_K(m) &:= \mathrm{Enc}(K, m) : \{0,1\}^\lambda \times \{0,1\}^n \to \{0,1\}^n \\ \mathrm{Enc}_K^{-1}(c) &:= \mathrm{Dec}_K(c) = \mathrm{Dec}(K, c) : \{0,1\}^\lambda \times \{0,1\}^n \to \{0,1\}^n \end{aligned}$$

$$\boxed{\forall K, \forall m : \mathrm{Dec}_K(\mathrm{Enc}_K(m)) = m}$$

Block-ciphers

Description

$$\begin{aligned} \mathrm{Enc}_K(m) &:= \mathrm{Enc}(K, m) : \{0,1\}^\lambda \times \{0,1\}^n \rightarrow \{0,1\}^n \\ \mathrm{Enc}_K^{-1}(c) &:= \mathrm{Dec}_K(c) = \mathrm{Dec}(K, c) : \{0,1\}^\lambda \times \{0,1\}^n \rightarrow \{0,1\}^n \end{aligned}$$

$$\forall K, \forall m : \mathrm{Dec}_K(\mathrm{Enc}_K(m)) = m$$

Data Encryption Standard (DES)

- Defined by US National Bureau of Standards, 1976
- Key length : 56 bits
- Block-size : 64 bits
- Complete deprecation, National Institute of Standards (NIST), 2017

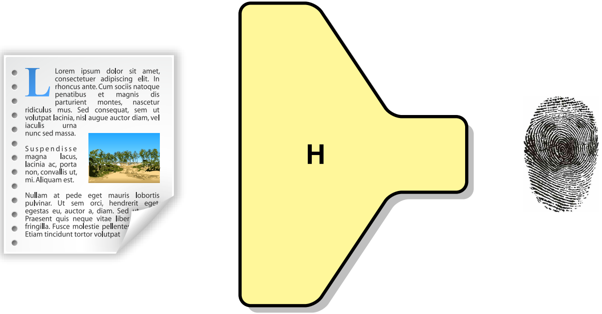
Advanced Encryption Standard (AES)

- Defined by NIST, 2001
- open call for proposals, competitive process
- Key length : 128/192/256 bits
- block-size : 128 bits
- Widely deployed

Hash functions

Hash functions compute fingerints

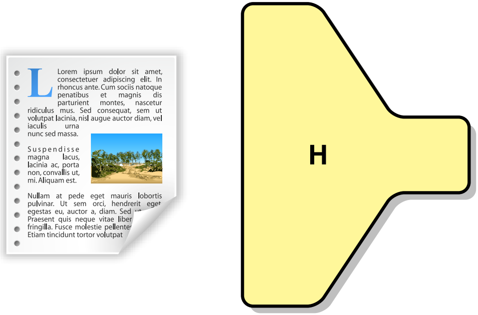
Various uses



Hash functions

Hash functions compute fingerints

Various uses

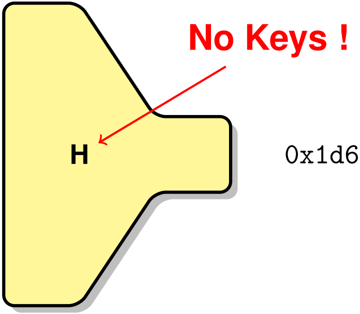


0x1d66ca77ab361c6f

Hash functions

Hash functions compute fingerints

Various uses



0x1d66ca77ab361c6f

Public-key cryptography

Limitations of symmetric cryptography

- ☛ Key-distribution needs physical meeting
- ☛ The number of keys for k users is $\Theta(k^2)$

Public-key cryptography

anyone can lock it →

← the key is needed to unlock

Diffie and Hellman, 1976

- ☛ The concept, no implementation
- ☛ A protocol for key-exchange

Diffie–Hellman (DH) key-exchange

$(\mathbb{G}, \cdot)$ a finite cyclic group; $\langle g \rangle = \mathbb{G}$

$$y_a = g^a$$

$$y_b = g^b$$

Alice

Bob

$\downarrow$ $\downarrow$

$K_a = {y_b}^a$

$K_b = {y_a}^b$

Eve

$$K_a = {y_b}^a = (g^b)^a = g^{ab} = (g^a)^b = {y_a}^b = K_b$$

Computational security

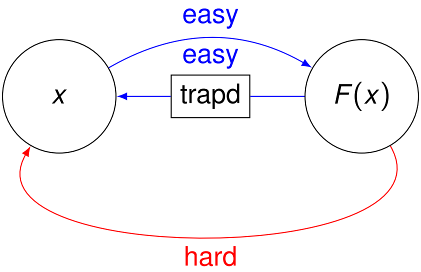
Discrete Logarithm problem

- Given a cyclic group $(\mathbb{G}, g)$ and $y \in \mathbb{G}$
- Find integer s such that $y = g^s$

- **Assumption.** It should be **computationally difficult** to find s from y
- How to choose $\mathbb{G}$: $\mathbb{G} = (\mathbb{Z}/n\mathbb{Z}^{\times}, \cdot)$ for some integer p or elliptic curves
- **Security level.** Base-2 logarithm of the complexity of the **best** algorithm
  - Symmetric cryptography : security level given by the bit-size of the secret-key, typically $128/192/256$
  - Public-key cryptography : same, more tricky analysis

Beyond DH key-exchange

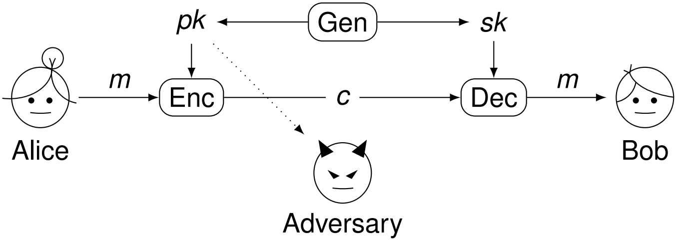
Trapdoor function: is easy to compute, difficult to inverse without special information, the "*trapdoor*".



Beyond DH key-exchange

Trapdoor function: is easy to compute, difficult to inverse without special information, the "*trapdoor*".

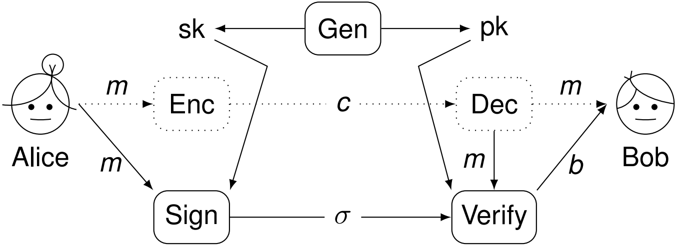
- A **Public-Key Encryption** (PKE) scheme can be constructed from any trapdoor permutation
- **Key-Encapsulation Mechanism** (KEM) : key-exchange using a PKE



Beyond DH key-exchange

Trapdoor function: is easy to compute, difficult to inverse without special information, the “*trapdoor*”.

☛ A **Digital Signature Scheme** (DSS) can be constructed from any trapdoor permutation.



Beyond DH key-exchange

Trapdoor function: is easy to compute, difficult to inverse without special information, the "*trapdoor*".

Factorization

Given two primes p and q.

- **easy** to compute $N = p \times q$
- **hard** to get p and q from N (**factorization**)

Key Size (Bits) Comparison

AES	RSA (N)/DH(p)	ECC (order q)
56	512	112
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	512

- Factorization Record, RSA829 [Boudot, Thomé, Gaudry, Heniniger, Zimmermann, 2020].

Cryptography in practice

Limitation of public-key cryptography

- It is order of magnitude slower than secret-key cryptography

Cryptography in practice

Limitation of public-key cryptography

- It is order of magnitude slower than secret-key cryptography

Hybrid encryption (KEM/DEM paradigm)

- Use public-key cryptography to exchange keys
- then secret-key cryptography for protecting large traffic

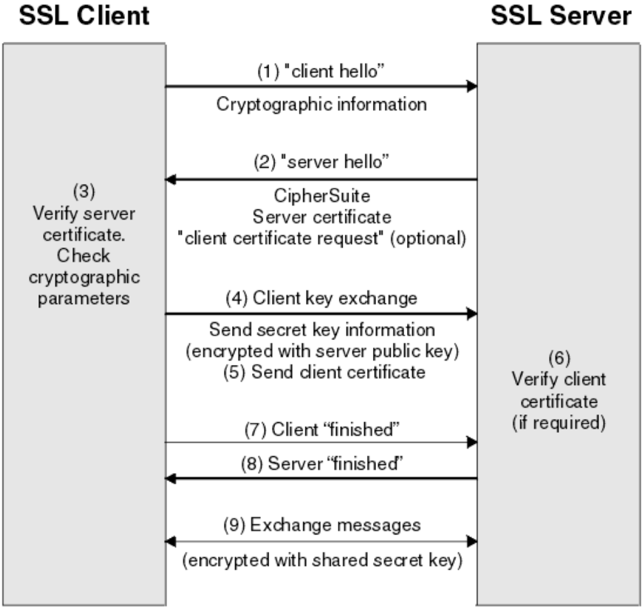
Cryptography in practice

Hybrid encryption (KEM/DEM paradigm)

- Use public-key cryptography to exchange keys
- then secret-key cryptography for protecting large traffic

confidentiality	block cipher (AES128)
integrity	Hash functions (SHA2/SHA3)
authentication	Message Authentication Code (MAC) symmetric-key primitive can be constructed from a hash function
authentication	Certificate public-key primitive roughly public-key +signature by a TTP

Cryptography in practice



Cryptography is a commodity

€150bn in 2023

Cybersecurity market

Outline

1. Cryptography Warm-Up
2. Quantum Impact
3. Transition toward quantum-resistant infrastructure

Quantum threat to secret-key cryptography (1/2)

Grover's algorithm

- $F : \{0,1\}^n \to \{0,1\}$
- Find $\mathbf{x}^* \in \{0,1\}^n$ such that $F(\mathbf{x}^*) = 1$
- $\left\lceil \frac{\pi}{4}\sqrt{\frac{2^n}{|F^{-1}(1)|}} \right\rceil$ evaluations of F as a quantum circuit

Quantum threat to secret-key cryptography (1/2)

Grover's algorithm

- $F : \{0,1\}^n \to \{0,1\}$
- Find $\mathbf{x}^* \in \{0,1\}^n$ such that $F(\mathbf{x}^*) = 1$
- $\left\lceil \frac{\pi}{4} \sqrt{\frac{2^n}{|F^{-1}(1)|}} \right\rceil$ evaluations of F as a quantum circuit

- Given $\big(m, c = \mathrm{Enc}(K, m)\big) \in \{0,1\}^n \times \{0,1\}^n$
- $F : \{0,1\}^\lambda \to \{0,1\}$ is the function that returns 1 if $c = \mathrm{Enc}(K^*, m)$.

Quantum threat to secret-key cryptography (1/2)

- Given $\left(m, c = \text{Enc}(K, m)\right) \in \{0,1\}^n \times \{0,1\}^n$
- $F : \{0,1\}^\lambda \to \{0,1\}$ is the function that returns 1 if $c = \text{Enc}(K^*, m)$.

Impact

Quantum exhaustive search in $O(\sqrt{2^\lambda})$ calls to F

- Exponential speedup toward classical approaches
- $\approx$ double the key-length

Resource estimates

V. Gheorghiu, M. Mosca.
"*Benchmarking the Quantum Cryptanalysis of Symmetric, Public-Key and Hash-Based Cryptographic Schemes.*"
arXiv.org 2019.

Quantum threat to secret-key cryptography (1/2)

- Given $\left(m, c = \mathrm{Enc}(K, m)\right) \in \{0,1\}^n \times \{0,1\}^n$
- $F : \{0,1\}^\lambda \to \{0,1\}$ is the function that returns 1 if $c = \mathrm{Enc}(K^*, m)$.

Impact

Quantum exhaustive search in $O(\sqrt{2^\lambda})$ calls to F

- Exponential speedup toward classical approaches
- $\approx$ double the key-length

Resource estimates

V. Gheorghiu, M. Mosca.
"*Benchmarking the Quantum Cryptanalysis of Symmetric, Public-Key and Hash-Based Cryptographic Schemes.*"
arXiv.org 2019.

Quantum threat to secret-key cryptography (2/2)

Beyond Grover

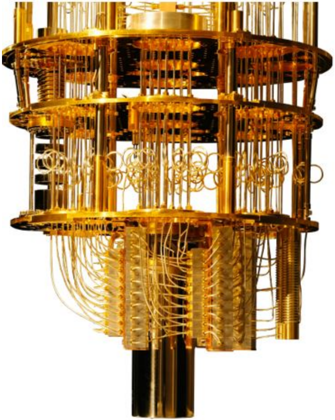
M. Kaplan, G. Leurent, A. Leverrier, M. Naya-Plasencia.
"*Breaking Symmetric Cryptosystems Using Quantum Period Finding.*"
CRYPTO 2016.

Simon's problem

- $F : \{0,1\}^n \to \{0,1\}$
- Find $\mathbf{s} \in \{0,1\}^n$ such that $F(\mathbf{x} \oplus \mathbf{s}) = F(\mathbf{x})$
- quantum polynomial-time

Quantum threat to public-key cryptography

(**Large)** Quantum computers will be able **break current public-key cryptography**

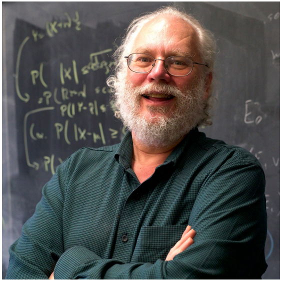


Shor's algorithm

Polynomial-time quantum algorithms for RSA/Diffie-Hellman

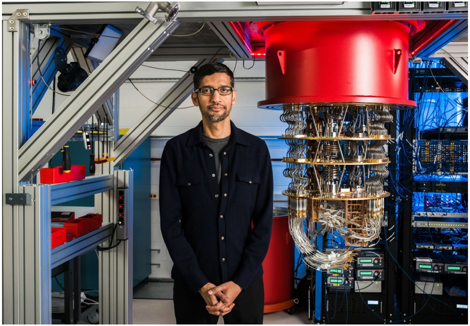
RSA1024 – classic $\approx$ 400 years

RSA1024 – quantum $\approx$ hours



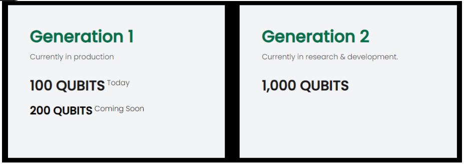
Quantum computing limits

2019 : "*Quantum supremacy*" by Google

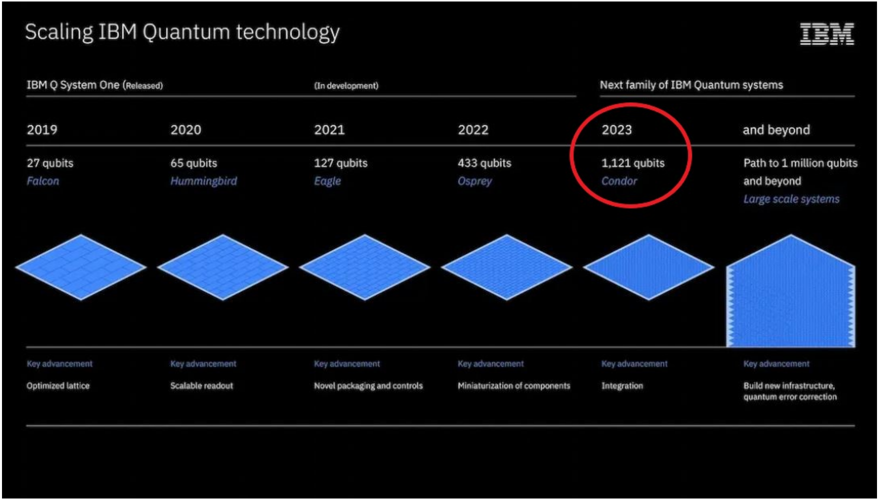


Sycamore : 53/70 qubits

2022 : $\geqslant$ 100 qubits by Pasqal



Quantum computing limits



Quantum computing limits

C. Gidney, M. Ekera.
"*How to factor* 2048 *bit* RSA *integers in* 8 *hours using* 20 *million noisy qubits*."
Quantum, 2021.

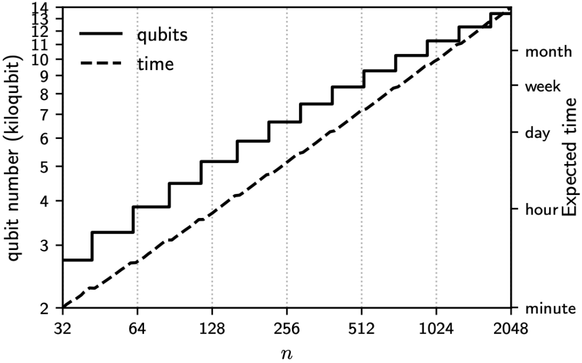
		Parameters						Retry	Volume (megaqubitdays)		Qubits (megaqubits)	Runtime (hours)
n	n_e	d_1	d_2	δ_{off}	c_{mul}	c_{exp}	c_{sep}	Risk	per run	expected	per run	per run
1024	$3(n/2-1)-40$	15	27	5	5	5	1024	6%	0.5	0.5	9.7	1.3
2048		15	27	4	5	5	1024	31%	4.1	5.9	20	5.1
3072		17	29	6	4	5	1024	9%	19	21	38	12
4096		17	31	9	4	5	1024	5%	48	51	55	22
8192		19	33	4	4	5	1024	5%	480	510	140	86
12288		19	33	3	4	5	1024	12%	1700	1900	200	200
16384		19	33	4	4	5	1024	24%	3900	5100	270	350

Extrapolating (paranoid)

- 9 years for RSA2048
- 8 years for RSA1024
- Time for a cryptographic transition $5/10$ years

Quantum computing limits

E. Gouzien, N. Sangouard
"*Factoring 2048-bit* RSA *integers in* 177 *days with* 13436 *qubits and a multimode memory*."
Physical Review Letters, 2021.



Quantum computing limits

Have Chinese scientists really cracked RSA encryption with a quantum computer?

The researchers say they could crack 2048-bit RSA using a quantum computer with a few hundred qubits. Not everyone is convinced.

Bao Yan et al.
"*Factoring integers with sublinear resources on a superconducting quantum processor.*"
ArXiv 2022.

Quantum computing limits

Adi Shamir predictions – 2016

"There will be no full size quantum computers capable of factoring RSA keys".

Time bomb effect

Harvest now, decrypt later

Time bomb effect

Connected objects with long life cycle

Outline

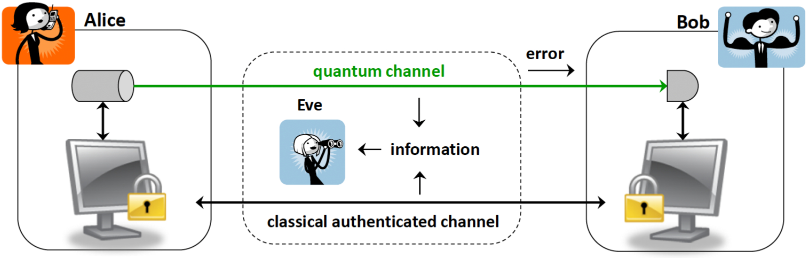
1. Cryptography Warm-Up
2. Quantum Impact
3. **Transition toward quantum-resistant infrastructure**

A risk perceived as major

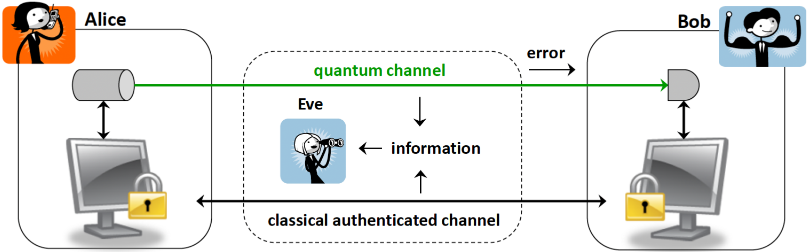
Solutions

Quantum-Key Distribution (`QKD`)

- two channels : **authenticated classical** and quantum
- **Unconditional security** based on quantum physics
  - Practical limitations : distance, cost, . . .



Solutions



National Security Agencies (French ANSSI, UK GCHQ, US NSA,...) usually argue **against** current deployment of QKD

Solutions

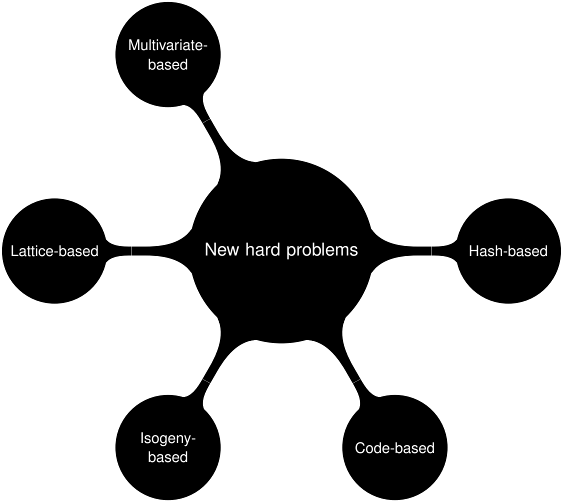
National Security Agencies (French `ANSSI`, UK `GCHQ`, US `NSA`,...) usually argue **against** current deployment of `QKD`

- Out-of-band distribution of a pre-shared key for `ITS` `MAC` authentication
- Key expansion with `QKD`
- Encryption of traffic with a block-cipher (**computational** assumption)

Solutions

Post-Quantum Cryptography (PQC)

- ❑ **Computational security** based on new hard algorithmic problems
- ❑ Natural integration into security protocols



Polynomial System Solving over Finite Fields (PoSSo_q)

q, size of field	n, nb. of variables	m, nb. of equations

PoSSo_q

Input. non-linear polynomials $p_1, \ldots, p_m \in \mathbb{F}_q[x_1, \ldots, x_n]$
Question. Find – if any – $(z_1, \ldots, z_n) \in \mathbb{F}_q^n$ such that:

$$\begin{cases} p_1(z_1, \ldots, z_n) = 0 \\ \vdots \\ p_m(z_1, \ldots, z_n) = 0 \end{cases}$$

PoSSo_q is NP-hard [Garey-Johnson, 1979]

Polynomial System Solving over Finite Fields (PoSSo_q)

q, size of field	n, nb. of variables	m, nb. of equations

PoSSo_q

Input. non-linear polynomials $p_1, \ldots, p_m \in \mathbb{F}_q[x_1, \ldots, x_n]$
Question. Find – if any – $(z_1, \ldots, z_n) \in \mathbb{F}_q^n$ such that:

$$\begin{cases} p_1(z_1, \ldots, z_n) = 0 \\ \vdots \\ p_m(z_1, \ldots, z_n) = 0 \end{cases}$$

PoSSo_q is NP-hard [Garey-Johnson, 1979]

Foundation

NP problem **cannot** be solved in poly-time by a quantum Turing machine.

C. H. Bennett, E. Bernstein, G. Brassard and U. V. Vazirani.
"*Strengths and Weaknesses of Quantum Computing*".
SIAM J. Comput., 1997.

NIST post-quantum standardization process

- **Round 1**: 2016 – 2018 : 82 submissions → 69 round-1 candidates
- **Round 2**: 2019 – 2020 → 26 algorithms
- **Round 3**: 2020 – 2022 → 7 finalists and 8 alternates
- **Selection**: 2022 : first set of algorithms selected (1 KEM and 2 DSS)

NIST post-quantum standardization process

- **Round 1**: 2016 – 2018 : 82 submissions → 69 round-1 candidates
- **Round 2**: 2019 – 2020 → 26 algorithms
- **Round 3**: 2020 – 2022 → 7 finalists and 8 alternates
- **Selection**: 2022 first set of algorithms selected (1 KEM and 2 DSS)
- **Round 4**: 2023 : selected round-3 and new signature algorithms

Post-Quantum Cryptography (PQC) standardization process

First PQC standards

2017 : NIST started a standardization process for PQC

2022 : **first** set of post-quantum standards

- 1 lattice-based KEM (Kyber)
- 3 signature schemes : 2 lattice-based (Dilithium/Falcon) and 1 hash-based (Sphincs+)

2023/2024 : Official standards

National Institute of Standards and Technology

Performances

AES	RSA (N)/DH(p)	ECC (order q)
80	1024	160
112	2048	224
128	3072	256

Figure: Key-sizes (bits)

Name	Size (bytes)		Performance (cycles)		
	#pk	#ct	KEYGEN	ENCAPSULATE	DECAPSULATE
Kyber512	800	768	33 856	45 200	34 572

Performances

AES	RSA (N)/DH(p)	ECC (order q)
80	1024	160
112	2048	224
128	3072	256

Figure: Key-sizes (bits)

Name	Size (bytes)		Performance (cycles)		
	#pk	#ct	KEYGEN	ENCAPSULATE	DECAPSULATE
Kyber512	800	768	33 856	45 200	34 572

Name	Size (bytes)		Performance (cycles)		
	#pk	#sig	KEYGEN	SIGN	VERIFY
Dilithium2	1 312	2 430	124 031	333 013	118 412
Falcon512	897	666	18 722 000	386 678	82 340
SPHINCS+s	32	7 856	144 000 000	1 100 000 000	1 190 000

A boom in PQC standardization – cryptography

Standardization for basic PQC primitives

- NIST Round-4 for additional KEM (since 2022)
- NIST call for additional signature schemes (since 2023)
- ISO JTC 1/SC 27/WG 2
  - Larger portfolio of PQC algorithms than NIST standards

New NIST call for digital signature schemes

NIST. "*Call for Additional Digital Signature Schemes for the Post-Quantum Cryptography Standardization Process.*" October 2022.

- More diversities in the computational assumptions
- Short signature sizes
- Deadline, June 1st, 2023
  - 50 submissions (23 submissions for round-1)

NIST National Institute of Standards and Technology

A boom in PQC standardization – cryptography

Standardization for basic PQC primitives

- NIST Round-4 for additional KEM (since 2022)
- NIST call for additional signature schemes (since 2023)
- ISO JTC 1/SC 27/WG 2
  - Larger portfolio of PQC algorithms than NIST standards

Standardization of advanced PQC

Upcoming NIST call for Multi-Party Threshold Schemes

- Building blocks for Privacy-Enhancing Technologies
- Homomorphic encryp., threshold signature schemes, . . .

ISO

NIST
National Institute of Standards and Technology