Post-Quantum Cryptography

Sebastian Schmittner

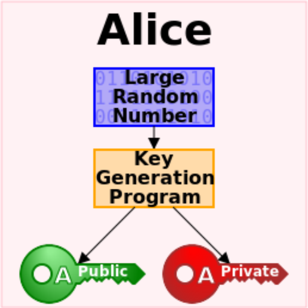
Institute for Theoretical Physics
University of Cologne

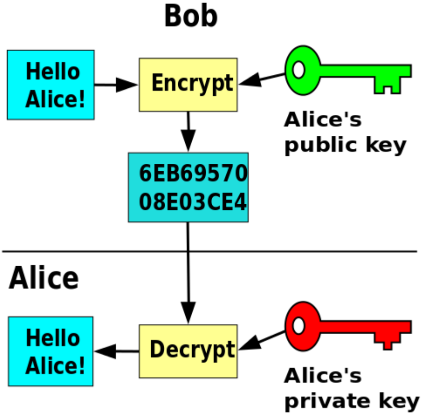
2015-10-26 Talk @ U23 @ CCC Cologne

Introduction
Review: Asymmetric Cryptography
Quantum Computer: Shor's Algorithm
Complexity

Post-Quantum Cryptography
Overview
Lattice-based cryptography
Learning with errors

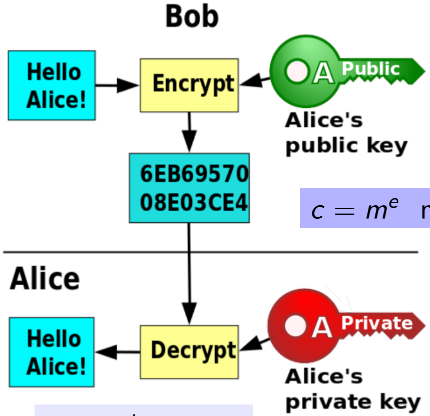
Asymmetric Cryptography





RSA

1977 by Ron Rivest, Adi Shamir, and Leonard Adleman at MIT



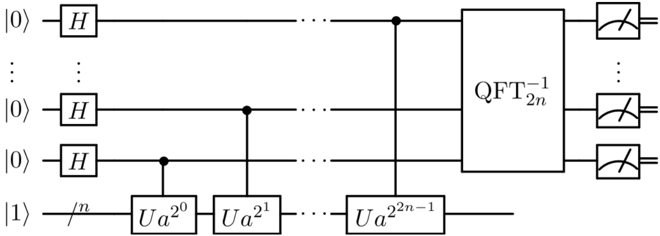
$$N = pq$$
$$e < \varphi$$

$$c = m^e \mod N$$

$$\varphi = (p-1)(q-1)$$
$$d = e^{-1} \mod \varphi$$

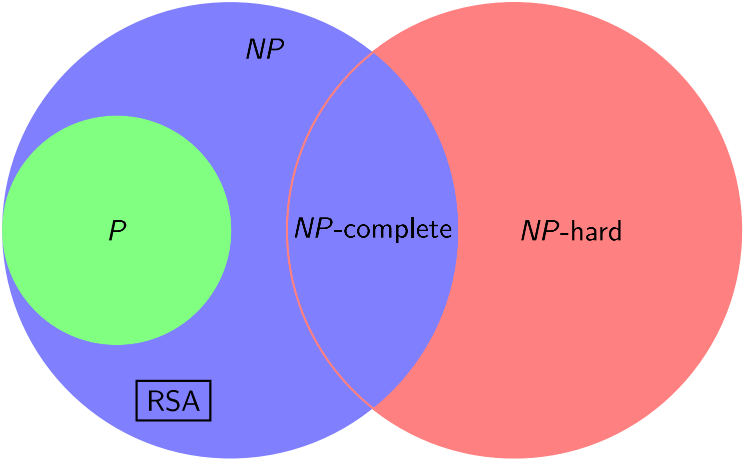
$$m = c^d \mod N$$

Shor's Algorithm

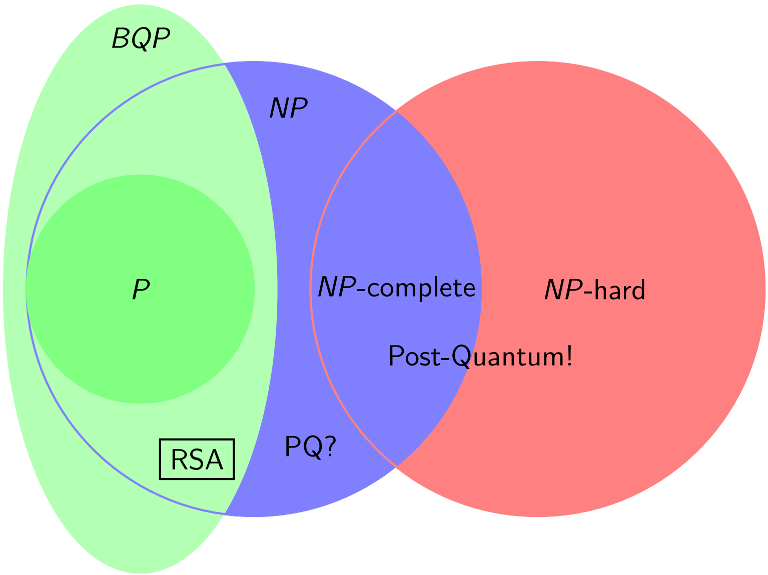


- Combined classical/quantum probabilistic algorithm
- Essential step: find period of $x \mapsto a^x \mod N$ via superposition, quantum Fourier transform and measurement
- **Quantum computer breaks: RSA, DSA, (hyper-)elliptic curve cryptography,...**
- Need for "post-quantum" cryptography

Complexity



Complexity



Post-Quantum Cryptography[1]

Existing PQ-cryptography schemes:

- Secret-key (Symmetric encryption, AES, 1998)
- Hash-based (Signature, Hash trees, 1979)
- Code-based (McEliece, 1978)
- Lattice-based (NTRU, 1998)
- Multivariate-quadratic-equations (Signature, HFE^{v-}, 1996)

Why RSA?

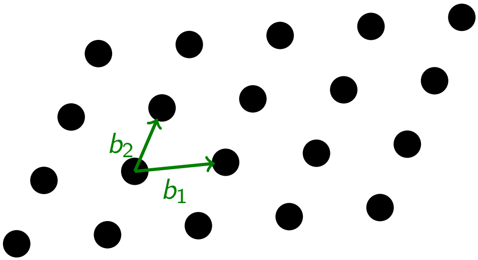
- Security level: attack needs 2^b operations
- RSA: key length $n_{RSA} \propto b^3/(\log b)^2$
- McEliece: key length $n_{McEliece} \propto b^2/(\log b)^2$

But: $n_{McEliece}/n_{RSA}(b = 128) \approx 10^2 \sim 10^3$ due to pre-factors

[1] Bernstein, Buchmann, Dahmen: Post-quantum cryptography. Springer '09.

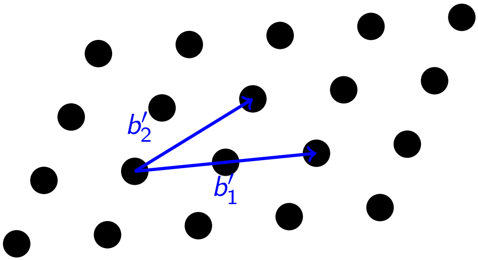
Lattice-based cryptography

- Choose a basis $B = \{b_1, \ldots, b_n\}$ of $\mathbb{R}^n$
- The finite set $L = \mathrm{Spann}_{\mathbb{Z}_q}(B)$ is called a (periodic) lattice



Lattice-based cryptography

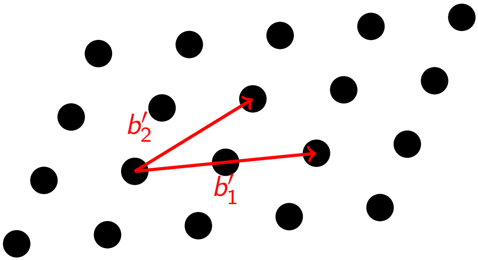
- Choose a basis $B = \{b_1, \ldots, b_n\}$ of $\mathbb{R}^n$
- The finite set $L = \text{Spann}_{\mathbb{Z}_q}(B)$ is called a (periodic) lattice



- B' Basis $\Leftrightarrow$ linear independent *and* $\text{Spann}_{\mathbb{Z}_q}(B') \stackrel{?}{=} L$

Lattice-based cryptography

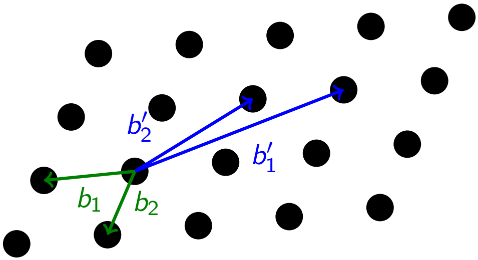
- Choose a basis $B = \{b_1, \ldots, b_n\}$ of $\mathbb{R}^n$
- The finite set $L = \mathrm{Spann}_{\mathbb{Z}_q}(B)$ is called a (periodic) lattice



- B' Basis $\Leftrightarrow$ linear independent *and* $\mathrm{Spann}_{\mathbb{Z}_q}(B') \neq L$

Lattice-based cryptography

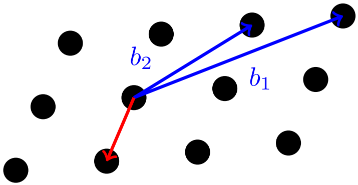
- Choose a basis $B = \{b_1, \ldots, b_n\}$ of $\mathbb{R}^n$
- The finite set $L = \mathrm{Spann}_{\mathbb{Z}_q}(B)$ is called a (periodic) lattice



- B' Basis $\Leftrightarrow$ linear independent *and* $\mathrm{Spann}_{\mathbb{Z}_q}(B') = L$
- $\Leftrightarrow B' = UB$ for unimodular $U \in \mathrm{Gl}_n(\mathbb{Z})$.
- Lenstra–Lenstra–Lovász lattice (LLL) basis reduction

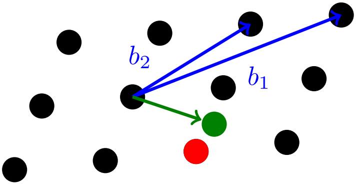
Lattice problems

Given a basis B



Shortest Vector Prob. (SVP)

- Find shortest $v \in L$
- NP-hard for max-Norm
- Used to secure NTRUEncrypt public key cryptosystem

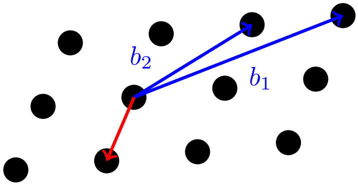


Closest Vector Problem (CVP)

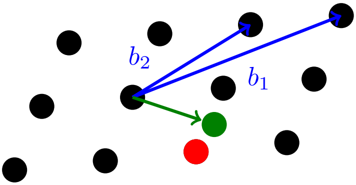
- Find closest $v \in L$ to given $\tilde{v} \in \mathbb{R}^n \setminus L$
- Goldreich-Goldwasser-Halevi (GGH) cryptosystem

Lattice problems

Given a basis B



Shortest Vector Prob. (SVP)

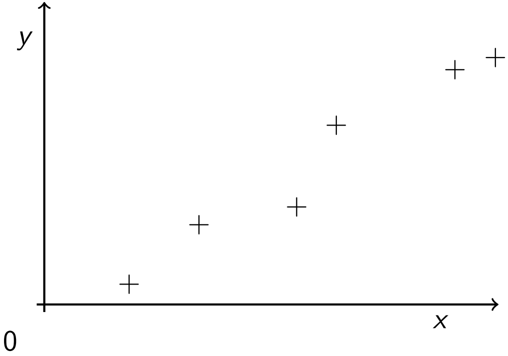


Closest Vector Problem (CVP)

- Decision Problems: $GapSVP_\beta$ and $GapCVP_\beta$
  - $||v - \tilde{v}|| < 1$ or $||v - \tilde{v}|| > \beta$?
- Polynomialtime-equivalent and both in NP
- Easy for large β
- NP-hard for e.g. $\beta \in o\left(n^{1/\log\log n}\right)$, in particular for $\beta \in O(1)$

Learning with errors (LWE)

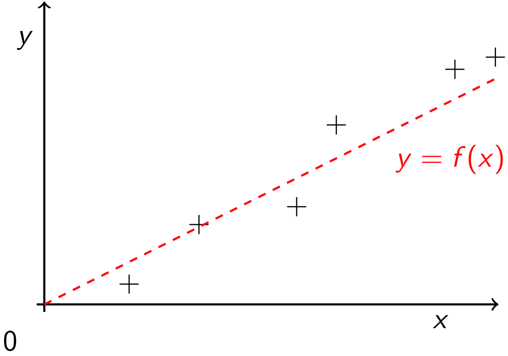
Rough idea



- $f : \mathbb{Z}_q^n \to \mathbb{Z}_q$ linear, i.e. $f(x) = v \cdot x$ for some vector v
- Error: $y = f(x) + \eta$ with random variable η (e.g. gaussian)
- Can we "learn" the function f from samples $\{(x, y)\}$?

Learning with errors (LWE)

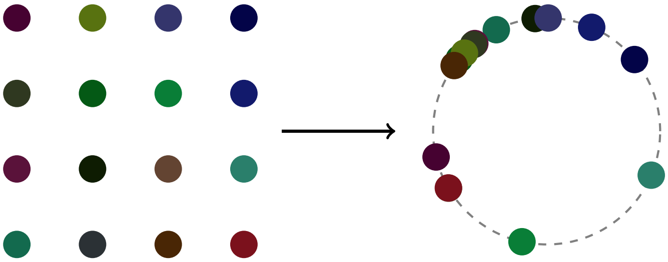
Rough idea



- $f : \mathbb{Z}_q^n \to \mathbb{Z}_q$ linear, i.e. $f(x) = v \cdot x$ for some vector v
- Error: $y = f(x) + \eta$ with random variable η (e.g. gaussian)
- Can we "learn" the function f from samples $\{(x, y)\}$?

Learning with errors (LWE)

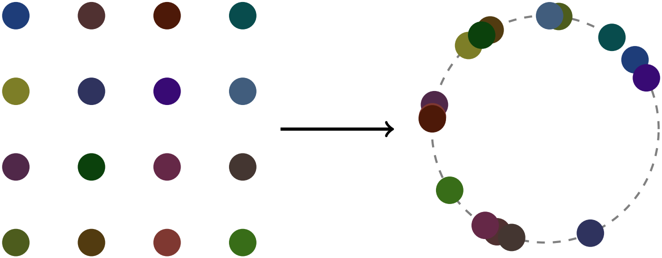
More precise idea



- Replace target space by $\mathbb{T} = \mathbb{R}/\mathbb{Z} \simeq U(1) \simeq S^1$
  - Group homomorphism $\mathbb{Z}_q \to \mathbb{T}$, i.e. $y \mapsto y/q$
- Distribution ϕ of random variable η on $\mathbb{T}$
- Find $v \in \mathbb{Z}_q^n$ from polynomially many $(x,\ v \cdot x/q + \eta)$

Learning with errors (LWE)

More precise idea



- Decision version: ϕ uniform or gaussian?
- Equivalent to search for not to large prime q
- No easy instances
- GapSVP can be reduced to LWE
- LWE translates into Regev's public key cryptosystem

Key exchange

General idea + example: Diffie-Hellman

- Public: Set of commuting functions $\{f_a\}$, e.g. $f_a(x) = e^a \mod N$, and starting value x
- Private: every participant chooses random a_i
- Exchange: everybody publishes $f_{a_i}(x)$
  - Computing a from x and $f_a(x)$ needs to be hard
- Compute and publish $f_{a_i}\left(f_{a_j}(x)\right)$
- ... (actually do this more cleverly with many participants ;)
- Finally everybody possesses a common key $F(x)$ with $F = f_{a_1} \circ f_{a_2} \circ \ldots = f_{a_2} \circ f_{a_1} \circ \ldots$
- E.g. $(e^a)^b = e^{ab} = \left(e^b\right)^a$ (also true $\mod N$)

Ring learning with errors key exchange (RLWE-KEX)

Rough idea

- Public: polynomial $a(x) = \sum_{i=1}^{n} a_i x^i$
- Private: *small* (max norm of coefficients) polynomials s and e
- (Almost) commuting operations:

$$(as_A + e_A)s_B + e_B = as_A s_B + e_a s_B + e_B \tag{1}$$
$$\approx (as_B + e_B)s_A + e_A = as_A s_B + e_B s_A + e_A \tag{2}$$

- Treating $e_B s_A + e_A$ and $e_A s_B + e_B$ as errors
- Detailed description of the algorithm: `https://en.wikipedia.org/wiki/Ring_learning_with_errors_key_exchange`

Many Thanks!