Protocols, cryptanalysis and mathematical cryptology (INFO-F514)

Post-quantum cryptography

Christophe Petit

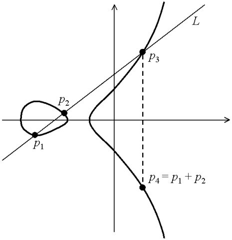
Université libre de Bruxelles

Are cryptographic protocols secure?

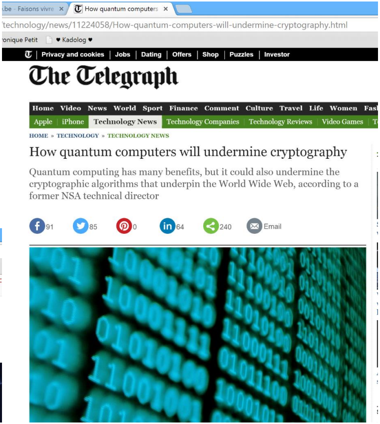
- Hard to anticipate all attacks, but we must try
- Best guarantees we have are by reduction:
  - Carefully define "secure" and attacker model (goal, resources, access to system)
  - Identify core problems that are plausibly hard to solve
  - Prove that breaking security implies solving the problems

Computational problems most used today

- **Integer factorization:** Given $n = pq$, where p and q are large prime integers, compute p and q
- **Computing discrete logarithms:** Given g generating a subgroup of $\mathbb{F}_q^*$, and given g^x, compute x
- **Computing discrete logarithms on elliptic curve groups:** Given some point P on the curve, and given another point $Q = [x]P$, compute x



The threat of quantum computers



Post-quantum cryptography

- Study of cryptographic algorithms that will (hopefully) remain secure when quantum computers are built
- Current approaches are based on lattice problems, error-correcting codes, multivariate polynomial equations, isogeny problems, non abelian groups, hash functions
- Move to post-quantum cryptography recommended by national security agencies including NSA, GCHQ, ...
- Algorithms exist but need thorough security analysis, implementations, adoption
- Ongoing standardization processes at NIST, ETSI, ...

Outline

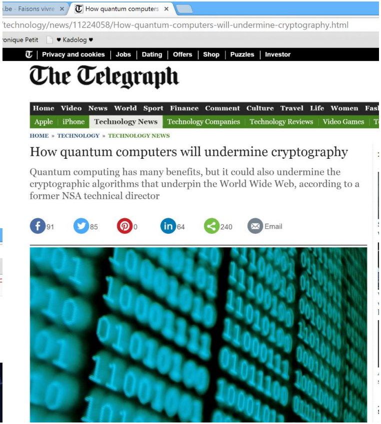
The threat of quantum computers

Post-quantum cryptography overview

Introduction to lattice-based cryptography

Outline

The threat of quantum computers



An algorithm for factorization

- Let $n = pq$ where $p, q, \frac{p-1}{2}, \frac{q-1}{2}$ prime
- Take a random in $\mathbb{Z}_n$
- Define a function $f : \mathbb{Z} \to \mathbb{Z}_n$

$$x \to a^x \bmod n$$

- Find T such that $f(x + T) = f(x)$ for all x
- Note T must be a multiple of $(p-1)(q-1)/4$

$$f(x + T) = a^{x+T} = a^x a^T = f(x)a^T$$

- Guess multiple, substitute $p = n/q$ in expression for T and solve quadratic equation to get factors

An algorithm for discrete logarithms

- Let g generating a cyclic group G of order n, and $h = g^x$
- Define a function $f : \mathbb{Z} \times \mathbb{Z} \to G$

$$(r, s) \to g^{-r} h^s$$

- Find $T = (t_1, t_2) \in \mathbb{Z}_n^2$ such that for all r, s we have

$$f(r + t_1, s + t_2) = f(r, s)$$

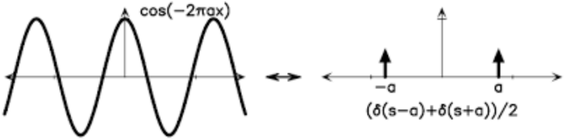
- Note (t_1, t_2) must be a multiple of $(x, 1)$

$$f(r + t_1, s + t_2) = g^{-r-t_1} h^{s+t_2} = f(r, s) g^{-t_1} h^{t_2}$$

- Compute $x = t_1 / t_2$

Quantum Fourier Transform

- In both algoritms for factoring and discrete logarithms, we need to compute the period of a periodic function
- Fourier transforms are good at finding periods:



- Core of Shor's algorithm is a quantum version of the Fast Fourier Transform algorithm

Impact of Shor's algorithm

- Factoring and (elliptic curve) discrete logarithm problems can be solved in polynomial time on a quantum computer
- Public key algorithms used today are mostly based on one of these problems

Grover's algorithm

- Given a function $C : \{1, \ldots, N\} \to \{0, 1\}$ such that $C(x) = 1$ for exactly one value x, compute this value
- Classically, given only black box access to C: $N/2$ random trials succeed with probability $1/2$
- Grover's quantum algorithm runs in $O(\sqrt{N})$
- When security depends on exhaustive key/message search we need to double bit lengths

Further impacts of quantum computers

- Collision finding algorithms are $O(\sqrt{N})$ classically but *might be* $O(\sqrt[3]{N})$ quantumly (hence hash functions *might* be easier to break)
- Impact of other quantum algorithms little studied
- Stronger attackers may exist with queries in superposition
- Standard security definitions need to be replaced to account for stronger quantum attacker
- Post-quantum security of Fiat-Shamir signatures and other standard protocol constructions unclear

Outline

The threat of quantum computers

Post-quantum cryptography overview

Introduction to lattice-based cryptography

Post-quantum cryptography

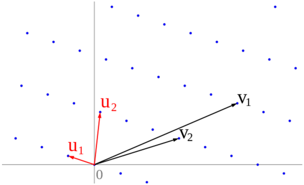
- Study of cryptographic algorithms that will (hopefully) remain secure when quantum computers are built
- Current approaches are based on lattice problems, error-correcting codes, multivariate polynomial equations, isogeny problems, non abelian groups, hash functions
- Move to post-quantum cryptography recommended by national security agencies including NSA, GCHQ, ...
- Algorithms exist but need thorough security analysis, implementations, adoption
- Ongoing standardization processes at NIST, ETSI, ...

NIST "competition"

- Post-quantum cryptography standardization effort by the American Institute for Standards and Technologies
- Focus on public key encryption, key encapsulation, signatures (sufficient for applications like TLS)
- Standard drafts expected by 2022-2024

Dec 2016	Call for new post-quantum algorithms to consider for standardization
Nov 2017	69 proposals accepted in Round 1
Dec 2018	27 candidates remaining in Round 2
Jul 2020	7 candidates remaining in Round 3 (+ 8 alternate algorithms)
2022-2024	Standard drafts expected

Main Lattice problems



- A lattice is a discrete subgroup of $\mathbb{R}^n$, typically given by a basis of vectors
- Shortest Vector Problem (SVP): Given a lattice basis, compute a shortest vector in the lattice
- Closest Vector Problem (CVP): Given a point and a lattice basis, compute closest lattice vector to the point

Lattice-based cryptography

- Leading approach for post-quantum cryptography
- Can be used for encryption, signatures, key exchange, and also advanced tools like fully homomorphic encryption
- Security typically rely on Learning-with-Error (LWE) and Small Integer Solution (SIS) rather than SVP and CVP
- Many parameters involved, so security evaluation difficult
- Very active research field, moving towards practice (cfr Google's experiments with NewHope)

Multivariate cryptography

- Let K be a finite field, let $R = K[x_1, \ldots, x_n]$ be a polynomial ring over K, and let $f_i \in R$
- Solving polynomial systems

$$\begin{cases} f_1(x_1, \ldots, x_n) = 0 \\ \ldots \\ f_m(x_1, \ldots, x_n) = 0 \end{cases}$$

  is a hard problem in general, in fact it is NP-hard
- Seems more suitable for signatures (e.g. Rainbow); many encryption schemes broken

Error-correcting codes

- Idea of error-correcting codes
  - Encode messages into larger codewords
  - Codewords transmitted over noisy channel
  - From an information-theoretic point of view, decoding back the original message possible for small noise
- Decoding linear codes is NP-hard in general, but efficient for particular families like Reed-Solomon, Goppa codes

McEliece cryptosystem (1978)

- Idea:
  - Consider a code with efficient decoding
  - Secret key is invertible transformation on this code
  - Public key is resulting seemingly random code
  - Encryption is noisy encoding of messages with public key
  - Decryption uses secret transformation then decoding
- Using Goppa codes safe since 1978 but large keys (8Mb)
- Many shorter variants using other codes are broken
- Building code-based signatures seems more challenging

Isogeny problems

- An isogeny is a rational map from one curve to another; it is a group homomorphism
- Used in point-counting algorithms on elliptic curves, reducing discrete logarithm from one curve to another, early cryptographic protocols
- Recently suggested for post-quantum cryptography
- Isogeny computation problem: given two elliptic curves, compute an isogeny between them
- Related problem of endomorphism ring computation studied by David Kohel in his PhD thesis

Isogeny-based cryptography

- Key agreement protocols: SIDH, CSIDH

$$\begin{tikzcd}
 & E_A = E_0/G_A \arrow[dr, "\phi'_B"] & \\
E_0 \arrow[ur, "\phi_A"] \arrow[dr, "\phi_B"'] & & E_0/\langle G_A, G_B \rangle \\
 & E_B = E_0/G_B \arrow[ur, "\phi'_A"'] &
\end{tikzcd}$$

- Encryption protocols derived from key exchange
- Identification and signature protocols
- Many more recent protocols!

Isogeny-based cryptography: pros and challenges

- Shortest keys among all post-quantum algorithms
- Easy substitute for (Elliptic Curve) Diffie-Hellman key agreement, preserving forward secrecy
- Implementation for elliptic curve cryptography may be partly reused

- More security analysis needed
- Rather slow compared to other candidates
- Signatures not fully practical

- Ask me for references / thesis topics !

PQ approaches: state-of-the-art

- Lattices are the front-runner: efficient, versatile, well-studied, but hard to choose concrete parameters
- McEliece cryptosystem safe since 1978 but has large keys
- Very efficient multivariate signature schemes, but many encryption schemes broken
- Isogeny-based cryptography is hipe!
- Hash-based signatures are great, but no encryption
- Other approaches suggested and broken

- Ongoing work on security analysis and implementations

Outline

The threat of quantum computers

Post-quantum cryptography overview

Introduction to lattice-based cryptography

Why lattice-based cryptography?

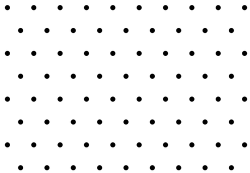
- Connection to NP-hard problems
- Worst-case vs average-case hardness
- No quantum attack
- Assumptions diversity: don't put all eggs in same basket
- Faster solutions to old problems (encryption, signatures)
- First solutions to other problems
  (fully homomorphic encryption, multilinear maps)

Outline

Outline

Lattices

- **Lattice** L: discrete subgroup of $\mathbb{R}^n$
  - Subgroup: L contains $av_1 + bv_2$ for all $a, b \in \mathbb{Z}$ and $v_1, v_2 \in L$
  - Discrete: non continuous ($\exists$ centered ball at 0 with no other lattice element)



Picture source: Wikipedia

- **Dimension** of L is n
- A lattice is **integer** if all lattice elements have integer coefficients

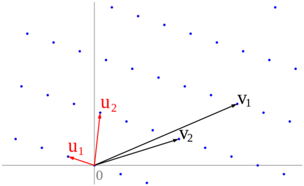
Lattices

- A **basis** of L is a minimal set of elements $\{v_i\}$ such that

$$L = \left\{ \sum_{i=1}^{r} a_i v_i | a_i \in \mathbb{Z} \right\}$$

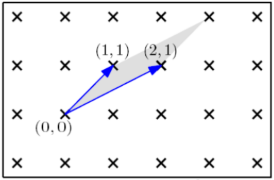
- **Rank** r of L is the size of a basis
- A lattice is **full-rank** if $r = n$
- We often represent a basis $\{v_i\}$ as a matrix $V \in \mathbb{R}^{n \times r}$, one column for all coefficients of one basis element
- In other words $L = \{Vx, x \in \mathbb{Z}^r\}$

Equivalent bases



- The red an black bases generate the same lattice: $v_1 = 2u_2 - 5u_1$, $v_2 = u_2 - 3u_1$, and $u_1 = v_1 - 2v_2$, $u_2 = 3v_1 - 5v_2$
- The sets $\{u_i\}$, $\{v_i\}$ generate the same lattice iff there exists $S \in \mathbb{Z}^{r\times r}$ such that $U = VS$ and $\det S = \pm 1$

Fundamental parallelepiped and Determinant



Picture credit: Oded Regev

- Let B be a lattice basis
- We can associate to it a **fundamental parallelepiped** $\mathcal{P}(B)$ consisting of all points modulo B
- The **determinant** of lattice L is $\det(L) = \sqrt{|\det(B \cdot B^t)|}$ (does not depend on basis B) $(= |\det B|$ if $n = r)$
- Determinant is the **volume** of fundamental parallelepiped

Scalar product and Euclidean norm

- Given $u = (u_1, \ldots, u_n), v = (v_1, \ldots, v_n) \in \mathbb{R}^n$, their **scalar product** is $\langle u, v \rangle := \sum_{i=1}^n u_i v_i$
- Scalar product is **bilinear**: $\forall \alpha \in \mathbb{R}$, $\langle \alpha u, v \rangle = \langle u, \alpha v \rangle = \alpha \langle u, v \rangle$
- $u, v \in \mathbb{R}^n$ are **orthogonal** if $\langle u, v \rangle = 0$
- **Euclidean norm** of $v \in \mathbb{R}^n$ is $||v|| = \sqrt{\sum_i v_i^2} = \sqrt{\langle v, v \rangle}$
- Basis $\{b_1, \ldots, b_n\}$ is **orthogonal** if $\langle b_i, b_j \rangle = 0 \quad \forall i \neq j$, in other words iff $B^t \cdot B$ is a diagonal matrix
- $u, v \in \mathbb{R}^n$ are **parallel** if $\langle u, v \rangle = ||u|| \cdot ||v||$

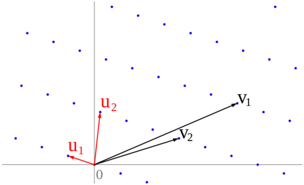
The shortest vector problem (SVP)

- We call λ_1 the shortest norm in the lattice

$$\lambda_1(L) = \min_{v \in L,\ v \neq 0} ||v||$$

- **Shortest vector problem** (SVP):
  given a basis $\{v_1, \dots, v_n\}$ for L,
  find $v \in L$ with $||v|| = \lambda_1(L)$

Good and bad bases



- Some bases make SVP easier
- A "good" basis has shorter vector norms
- A "good" basis has nearly orthogonal vectors
  (as nearly parallel vectors can lead to shorter vectors)

Upper bounding shortest vectors (1)

- Convex body theorem: For any lattice L of rank n, any convex set $S \subset span(L)$ symmetric about the origin, if $vol(S) > 2^n \det L$ then S contains nonzero lattice point

  Proof:

  - Consider a fundamental parallelipiped $\mathcal{P}(B)$ consisting of all points modulo a basis B of L
  - Consider the set $S' = \{x \mid 2x \in S\}$
  - By volume condition there exist $z_1, z_2 \in S'$ reducing to same point in $\mathcal{P}(B)$, i.e. $z_1 - z_2 \in L$
  - By definition $2z_1, 2z_2 \in S$ and since S symmetric and convex we have $z_1 - z_2 \in S$

Upper bounding shortest vectors (2)

- Minkowski's first theorem: we have

$$\lambda_1 < \sqrt{n}(\det L)^{1/n}$$

Proof: remark that volume of ball $\mathcal{B}(0, r)$ is bigger than $(2r/\sqrt{n})^n$ and apply previous theorem on $S = \mathcal{B}(0, \sqrt{n}(\det L)^{1/n})$

- Minkowski's second theorem: we have

$$\left(\prod_{i=1}^{n} \lambda_i\right)^{1/n} < \sqrt{n}(\det L)^{1/n}$$

where the **successive minima** $\lambda_k(L)$ are the smallest λ such that there are at least k linearly independent vectors with norms at most λ (proof: see Goldwasser-Micciancio)

Expected size of shortest vector

- **Gaussian heuristic**: let $V = \det(L)$.
  If L is a reasonably random lattice we expect that

  $$\lambda_1 \approx \text{ radius of a ball with volume } V$$

  (only a factor 2 smaller than Minkowski's bound)
- For Euclidean norm we have $V(\mathcal{B}(0,R)) = \frac{\pi^{n/2}}{(n/2)!} R^n$
- This heuristic works well for many cryptographic lattices
- Some crypto lattice distributions have very small λ_1
  by construction; then use similar heuristic for other λ_i

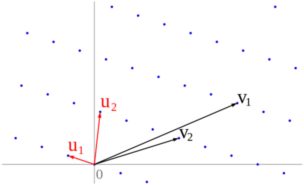
The closest vector problem (CVP)

- For a lattice L and a point $t \in \mathbb{R}^n$, define distance

$$d(t, L) := \min_{v \in L} ||v - t||$$

- **Closest vector problem**:
  Given a basis $\{v_1, \ldots, v_n\}$ for L and given $t \in \mathbb{R}^n$,
  find $v \in L$ with $||v|| = d(t, L)$

Good and bad bases



- Good bases also make CVP easier: all points in the fundamental parallelepiped are close to basis vectors
- See Babai's *nearest plane algorithm*

Decisional SVP and CVP

- **Decision-SVP:** Given a basis $\{v_1, \ldots, v_n\}$ for L and a rational $r \in \mathbb{Q}$, determine whether $\lambda_1(L) \leq r$ or not
- **Decision-CVP:** Given a basis $\{v_1, \ldots, v_n\}$ for L, a point $t \in \mathbb{Z}^n$ and a rational $r \in \mathbb{Q}$, determine whether $d(t, L) \leq r$ or not
- Can solve decision problems if can solve search problems
- Converse also true, but needs some work

Are SVP and CVP hard?

- Decisional CVP is NP-hard
- Search and Decisional CVP are equivalent
- Search and Decisional SVP are equivalent
- Can solve SVP if can solve CVP
- Heuristically the converse if also true

Approximate SVP and CVP

- γ-**approximate shortest vector problem**:
  Given a basis $\{v_1, \ldots, v_n\}$ for L,
  find $v \in L$ with $||v|| \leq \gamma\lambda_1(L)$
- γ-**approximate closest vector problem**:
  Given a basis $\{v_1, \ldots, v_n\}$ for L and given $t \in \mathbb{R}^n$,
  find $v \in L$ with $||v|| \leq \gamma d(t, L)$
- Standard SVP and CVP if $\gamma = 1$

Are approximate SVP and CVP hard?

- Still NP-hard for $\gamma < n^{1/\log\log n}$
- Becomes easier for larger γ
- Unlikely to be NP-hard for $\gamma > \sqrt{n/\log n}$
- LLL achieves $\gamma = 2^{(n-1)/2}$ in polynomial time
- In cryptography we need $\gamma = n^c$ hard with $c \geq 1$
- Note that NP-hardness is not known for these parameters, so we need to **assume** that these problems are hard

Worst case vs Average case hardness

- NP-hardness refers to worst-case hardness
- In cryptography we want average case hardness since we need some entropy on the keys
- Average case hard $\Rightarrow$ worst case hard, but not other way around in general
- Interesting property of lattice-based cryptography: worst-case to average-case reductions!

Other lattice problems

- **Gap SVP**: for approximation factor $\gamma > 1$ and radius r, returns YES if $\lambda_1 \leq r$, return NO if $\lambda_1 \geq \gamma r$, and may return YES or NO otherwise
- **ISVP**: find vectors with norms equal to **successive minima**: $\lambda_k(L)$ is the smallest λ such that there are at least k linearly independent vectors with norms at most λ
- And many others...

Modular lattices

- A lattice is **modular** if $\exists q < \det(L)$ with $L \supset q\mathbb{Z}^n$
- In cryptography we often use

$$L_{A,q} = \{x \in \mathbb{R}^n | Ax = 0 \bmod q\}$$

  for some matrix $A \in \mathbb{Z}^{m\times n}$ with entries reduced modulo q
- Typically $n \approx m \log m$

  (Caution: here columns of A are not lattice vectors !)

SIS

- **Small integer solution** (SIS): given q, A and ν, find x with $Ax = 0 \bmod q$ and $||x|| \leq \nu$
- A short vector in $L_{A,q}$ gives a solution to SIS
- SIS harder when A has less columns and more rows
- SIS has solutions when ν and n large enough

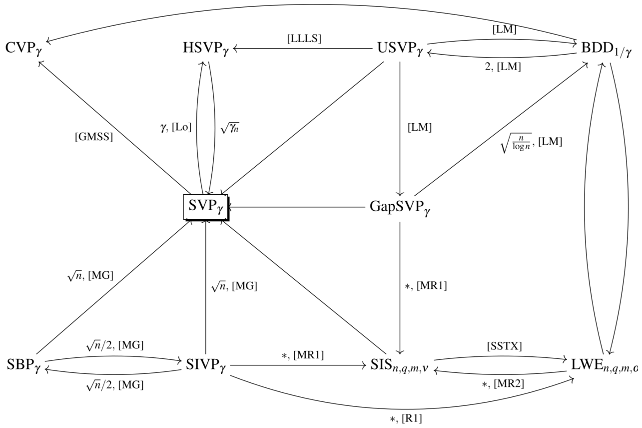
Learning with errors (LWE)

- Let q a modulus and let $s \in \mathbb{Z}_q^n$
- Let $B << q$ some noise bound
- LWE sample is (a, t) with a uniformly chosen in $\mathbb{Z}_q^n$, e uniformly chosen in $[-B, B]$, and $t = \langle a, s \rangle + e$
- **LWE problem**: given m samples (a_i, t_i), recover s
- Could use linear algebra if $B = 0$
- Other distributions for e can be used
  (in fact, we usually use Gaussian distributions)

Learning with errors (2)

- CVP-type problem for the matrix A generated by a_i:
  Given A and t, find $As \in L$ such that $e = t - As$ is small
  (in fact *bounded distance decoding*: such solution exists)
- Extension of **Learning Parity with Noise**,
  a NP-hard problem from coding theory
- **Decision LWE:** given samples (a_i, t_i) that are either
  LWE samples or random samples, guess distribution

Some relationships between lattice problems



Laarhoven, van de Pol, de Weger, Solving Hard Lattice Problems and the Security of Lattice-Based Cryptosystems

Arrow from Problem A to Problem B means "Problem A can be solved using an algorithm for Problem B"

Ideal lattices

- Lattice-based schemes need to include a basis of the lattice in the public key, typically n^2 coefficients
- Ideal lattices:
  - Choose a polynomial ring $R = \mathbb{Z}[x]/f(x)$ (typically $f(x) = x^n + 1$ and $n = 2^e$)
  - See a vector $v = (v_0, \ldots, v_{n-1})$ as a polynomial $v(x) = v_0 + v_1 x + v_2 x^2 + \ldots + v_{n-1} x^{n-1}$ in that ring
  - Ideal lattice is generated by $x^i v(x) \bmod f(x)$
  - Only store the n coefficients of v

Ideal lattices are modular

- Taking Hermite normal form, we get $q \in \mathbb{Z} \cap \langle v(x) \rangle$
- Deduce $qx^i \in \langle v(x) \rangle$ hence $L \supset q\mathbb{Z}^n$

Are ideal lattices secure?

- Crypto rule of thumb: any structure that improves efficiency, also decreases security
- Recent results: approximate SVP for ideal lattices not as hard as for general lattices
- No impact on ring LWE so far
- NIST candidate NTRU, Falcon based on ideal lattices
- NIST Round 3 candidates Kyber, Saber, Dilithium based on module lattices

Outline

Remember: hash functions

$$H : \{0,1\}^* \times K \to \{0,1\}^n$$

- A hash function satisfies
  - **Collision resistance**
    if hard to find m, m' such that $H_k(m) = H_k(m')$
  - **Preimage resistance**
    if given h, hard to find m such that $H_k(m) = h$
  - **Second preimage resistance**
    if given m, hard to find m' such that $H_k(m') = h$

  for a uniformly generated key $k \in K$
- We often build a fixed-length hash function and then use Merkle-Damgaard transform

Ajtai's hash functions

- Key generation: choose a random modular lattice

$$L_{q,A} = \{x \in \mathbb{R}^n | Ax = 0 \bmod q\}$$

- Define $H : \{0,1\}^n \to \mathbb{Z}_q^m : x \to Ax \bmod q$
- Collisions $Ax = Ax'$ implies solving SIS **on average** $A(x - x') = 0 \bmod q$ with $(x - x') \in \{-1, 0, 1\}^n$ small

Worst case to average case reduction

- Goal: solve **any** instance of $\tilde{O}(n)$-SIVP given an algorithm that solves **random** instances of SIS
  (γ-SIVP $=$ finding n linearly independent lattice vectors, the largest one being as small as possible, up to factor γ)
- Let B a lattice basis, defining an SIVP problem
- Consider parallelepiped $\mathcal{P}(B)$ consisting of all points of $\mathbb{R}^n$ modulo B
- Divide $\mathcal{P}(B)$ into q^n regularly spaced cells
- Associate cells to $\mathbb{Z}_q^n$ elements (use map $z \to f(z) = \lfloor qB^{-1}z \rceil$)

Worst case to average case reduction (2)

- Informal lemma: large enough random vectors modulo B lead to uniformly distributed points on $\mathcal{P}(B)$
  (usually take normal distributions with $\sigma = c\lambda_n$)

- Choose large enough $r_i \in \mathbb{R}^n$ with additional requirement that $r_i \bmod B$ is the corner of a cell
- Provide q and $a_i = f(r_i)$ to the SIS solver and receive solution $z_i \in \{-1, 0, 1\}$ with $\sum a_i z_i = 0 \bmod q$
- Deduce lattice point $z = \sum_i r_i z_i$ with $||z||_2 \leq cn\lambda_n$

- Note that λ_n can be guessed with binary search, or take the current best approximation and repeat

Using ideal lattices

- Improve efficiency using A with special structure
- Taking circulant matrices is a bad idea
  - Lattice points correspond to elements in a principal ideal
    $$\langle a(X) \rangle \subset R = \mathbb{Z}[X]/(X^n - 1)$$
  - If $\gcd(a(X), X^n - 1) \neq 1$ then there exists $z_0 \neq 0$ with
    $$a(X) z_0(X) = 0 \bmod (X^n - 1)$$
  - Deduce collision $(z, z + z_0)$ for every z

Using ideal lattices (2)

- Solution: replace $X^n - 1$ by an irreducible polynomial
- Taking $f(X) = X^n + 1$ and $n = 2^k$ has some efficiency advantages (use Fast Fourier Transform, etc)
- Security still based on worst case hardness assumptions but for **ideal** lattice problems

GGH cryptosystem: basic idea

- Private key is well-chosen good basis of a lattice
  (basis with short, nearly orthogonal vectors)
- Public key is well-chosen bad basis A for the same lattice
  (for example, the Hermite normal form of the lattice)
- Encryption of "small" m is $As + m$, for well-chosen s
  (so that result is reduced modulo Hermite basis)
- Decryption is LWE / CVP like problem
  (in fact bounded distance decoding),
  easy given the private key but hard otherwise

GGH cryptosystem: remarks

- Similar to McEliece's code-based cryptosystem (1978)
- Probabilistic by padding the message with random noise (for example $m \to m + 2r$)
- No formal reduction to a hard problem and original parameters broken, but eventually led to LWE schemes
- Not CCA secure (given a ciphertext, can re-randomize it and ask the decryption oracle for plaintext)
- Can use hash functions / random oracles to transform CPA encryption into CCA encryption (Fujisaki-Okamoto)

NTRU cryptosystem (sketch)

- Let p, q coprime integers with $p << q$
- Let $R = \mathbb{Z}[X]/(X^n - 1)$
- Private key : polynomials $f, g \in R$ with small coefficients such that f invertible modulo p and q
- Public key: $h = pf^{-1}g \bmod q$
- Encryption of small $m \in R$: take random small $r \in R$ and return $c = m + hr \bmod q$
- Decryption of c is $m' = (cf \bmod q)\, f^{-1} \bmod p$
- Correctness: modulo q we have $cf = mf + pgr$ and right-hand term is small so no reduction modulo q

NTRU : link with lattices

- Public key is

$$A = \begin{pmatrix} I & 0 \\ H & qI \end{pmatrix}$$

where H is cyclic matrix corresponding to h

- Private key is short vector corresponding to f, g. Equivalently a matrix

$$B = \begin{pmatrix} F & \tilde{F} \\ G & \tilde{G} \end{pmatrix}$$

where F, G are cyclic matrices corresponding to f, g and $\tilde{F}, \tilde{G}$ are well-chosen matrices so that $\mathcal{L}(A) = \mathcal{L}(B)$

- Encryption of m is $(-r, m)^T$ modulo $\mathcal{L}(A)$

NTRU: security

- Recommended parameters (Wikipedia, citing NTRU website)

	N	q	p
Moderate Security	167	128	3
Standard Security	251	128	3
High Security	347	128	3
Highest Security	503	256	3

- No security proof for original scheme
- If secret polynomials are generated in a proper way then becomes CPA-secure under ideal lattice assumptions (see Stehlé-Steinfeld 2011)

LWE-based cryptosystem

- Parameters: integers n, m, ℓ, t, r, q and real $\alpha > 0$
- Let $f : \mathbb{Z}_t^\ell \to \mathbb{Z}_q^\ell$ defined by

$$z \to f(z) = [(q/t)z]$$

  "rounded scaling" (here $q > t$)
- Let $f_{-1} : \mathbb{Z}_q^\ell \to \mathbb{Z}_t^\ell$ defined by

$$z \to f_{-1}(z) = [(t/q)z]$$

  "inverse" of f

LWE-based cryptosystem (2)

- Private key is $S \in \mathbb{Z}_q^{n\times\ell}$ uniformly random
- Public key is $(A, P) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^{m\times\ell}$ with
  - $P = AS + E$
  - $E \in \mathbb{Z}_q^{m\times n}$ normal distribution with $\sigma = \alpha q/\sqrt{2\pi}$
  - $A \in \mathbb{Z}_q^{m\times n}$ uniformly random
- Encryption of $v \in \mathbb{Z}_t^\ell$ is

$$(u, c) = (A^T a, P^T a + f(v))$$

  with a uniformly random in $\{-r, \ldots, r\}^m$
- Decryption of (u, c) is

$$v' = f_{-1}(c - S^T u)$$

LWE-based cryptosystem (3)

- Kind of lattice version of ElGamal
- Correctness: we have

$$\begin{aligned} c - S^T u &= P^t a + f(v) - S^T A^T a \\ &= (AS + E)^T a + f(v) - S^T A^T a) \\ &= E^T a + f(v) \end{aligned}$$

hence $f_{-1}(c - S^T u) = v$ as long as

$$||E^T a||_\infty < q/2t$$

Security

- Distinguishing (A, P) from uniformly random pairs implies solving Decisional LWE
- Encryptions with random pairs leak no information on messages (when $\#\text{inputs} = (2r+1)^m >> \#\text{outputs} = q^{n+\ell}$)
- Together these two observations imply CPA security (if you distinguish two ciphertexts then the keys are not random)
- Concrete hardness of LWE: see Albrecht-Player-Scott
- CCA encryption scheme follows from generic reductions such as Fujisaki-Okamoto (more direct constructions now exist)

Digital signatures: basic idea

- Private key is a good basis B of a lattice
- Public key is a bad basis for the same lattice
- Let H a collision resistant hash function with image in $\mathbb{R}^n$
- To sign, compute $H(m)$, use Babai's nearest plane algorithm with good basis to obtain close lattice point s, and return it
- To verify, check that s and $H(m)$ are close
- Examples: GGH signatures, NTRU signatures

Digital signatures: improvements

- Basic idea broken [Nguyen-Regev]
  - Signature (m, s) leaks $s - H(m)$ a uniformly distributed point in (a translation of) the fundamental parallelipiped

  - Attacker obtains several (m_i, s_i) then recovers B by solving an optimization problem

- Solution: signature a quite close vector (distance $\approx c\lambda_n$), making sure distribution of $s - H(m)$ is independent of B

Outline

The threat of quantum computers

Post-quantum cryptography overview

Introduction to lattice-based cryptography

Conclusion

- Large scale quantum computers will break currently used public key cryptography protocols
- Ongoing large scale effort to replace them, based on a wide variety of mathematical tools
- Lattice-based cryptography is the front-runner: fast, versatile, better studied
- Many research challenges !

References

- Micciancio-Goldwasser, *Complexity of Lattice Problems*
- Micciancio-Regev, *Lattice-based cryptography*
- Peikert, *A decade of lattice cryptography*