Kyber and Dilithium

Alfred Menezes

cryptography101.ca

August 2024

Introduction

- In 2024, the US government's National Institute of Standards and Technology (NIST) published a suite of standards for quantum-safe key encapsulation mechanisms (KEM) and signature schemes.
- These schemes are intended to replace RSA and ECC, which succumb to quantum attacks.
- It's expected that Kyber (a KEM) and Dilithium (a signature scheme) will see the most deployment in the coming years.

Course objectives

1. Detailed description of Kyber
   - Module-Lattice-based Key Encapsulation Mechanism (ML-KEM)
   - FIPS 203
2. Detailed description of Dilithium
   - Module-Lattice-based Digital Signature Algorithm (ML-DSA)
   - FIPS 204
3. Appreciate the many optimizations introduced to facilitate fast implementations, and to decrease key, ciphertext and signature sizes.

Course outline

- V1: Introduction
- V2: The Kyber PKE and KEM
- V3: The Dilithium signature scheme
- V4: Number-Theoretic Transform (NTT)

V1: Introduction

Kyber and Dilithium

August 2024

V1 outline

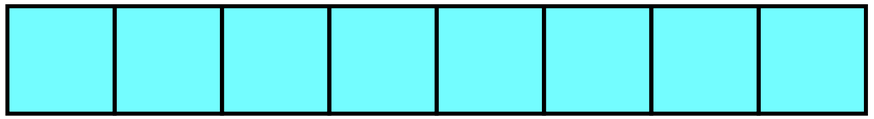
- V1a: Post-quantum cryptography
- V1b: Mathematical preliminaries

V1a: Post-quantum cryptography

1. Quantum computers
2. The threat of quantum computers (Shor and Grover)
3. PQC standardization

Quantum computers

- Conceived by Yuri Manin (1980) and Richard Feynman (1981), quantum computers are devices that use quantum-mechanical phenomena such as superposition, interference, and entanglement to perform operations on data.
- A qubit is the quantum analogue of a classical bit, and can be in two states at the same time, each with a certain probability.
- An *n*-qubit register can be in 2^n states at the same time, each with a certain probability.



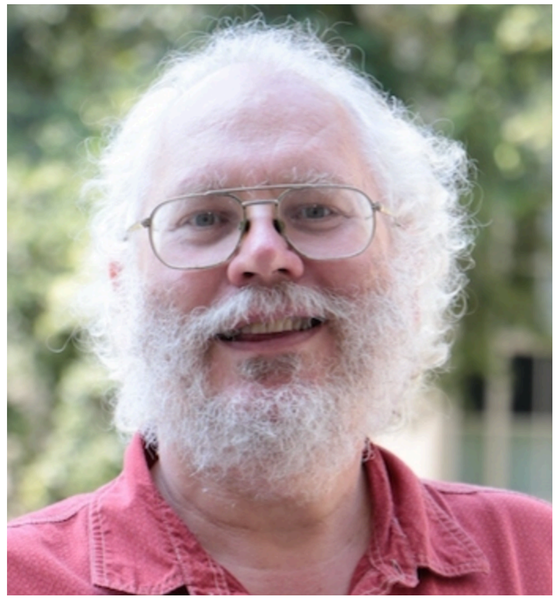
- When a function $f : \{0,1\}^n \rightarrow \{0,1\}^n$ is applied to an *n*-qubit register, it is simultaneously evaluated at *all* 2^n states.
- However, when the *n*-qubit register is measured, it reverts to being in one of the 2^n states according to its underlying probability distribution.
- *So, quantum computers are not "massively parallel machines."*

The threat of quantum computers: Shor

The public-key systems used in practice are:

- RSA: security is based on the hardness of integer factorization.
- DL: security is based on the hardness of the discrete logarithm problem.
- ECC: security is based on the hardness of the ECDLP.

Shor's algorithm: In 1994, Peter Shor discovered a very efficient (polytime) quantum algorithm for solving these problems. *So, all RSA, DL, and ECC implementations can be totally broken by quantum computers.*



The threat of quantum computers: Grover

Let $F : \{0,1\}^n \longrightarrow \{0,1\}$ be a function such that (i) F is efficiently computable; and (ii) $F(x) = 1$ for exactly one input $x \in \{0,1\}^n$.

Grover's algorithm: In 1996, Lov Grover discovered a quantum algorithm for finding the $x \in \{0,1\}^n$ with $F(x) = 1$ in $\sqrt{2^n}$ evaluations of F.

Exhaustive key search: Consider AES with an ℓ-bit key. Suppose that we have t known plaintext-ciphertext pairs (m_i, c_i), where t is such that the expected number of false keys is very close to 0.

Define $F : \{0,1\}^\ell \longrightarrow \{0,1\}$ by $F(k) = 1$ if $\text{AES}_k(m_i) = c_i$ for all $1 \leq i \leq t$; and $F(k) = 0$ otherwise.

Then Grover's algorithm can find the secret key in $2^{\ell/2}$ quantum operations.

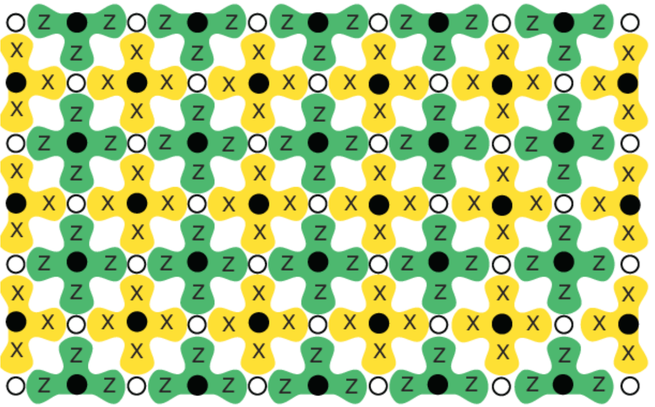
Thus, 256-bit AES keys should be used in order to achieve a 128-bit security level against quantum attacks.

When will quantum computers be built?

- 1998: (Jones & Mosca) 2-qubit quantum computer
- 2017: 50 qubits (IBM) tinyurl.com/IBMqc50
- 2019: 53 qubits (Google) tinyurl.com/GoogleQC
- 2021: 127 qubits (IBM) tinyurl.com/IBMqc127
- 2022: 433 qubits (IBM) tinyurl.com/IBMqc433
- Dec 2023: 1,121 qubits (IBM) tinyurl.com/IBMqc1121

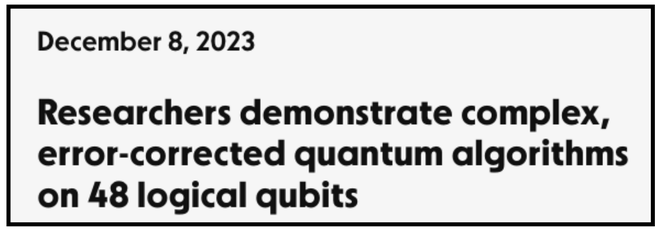
Fault-tolerant quantum computers?

- A quantum computer that can factor a 2048-bit RSA modulus using Shor's algorithm needs (at least) 2048-qubit registers.
- These qubits will have to be fault tolerant, i.e., error resistant.
- *The physical qubits that have been built so far are not (sufficiently) fault tolerant.*
  - The largest number factored using Shor's algorithm on a quantum computer is $21 = 3 \times 7$.
- So, the plan is to use quantum error correction to combine many (imperfect) physical qubits into one (almost perfect) logical qubit.
- Optimistic estimates are that thousands of physical qubits will be needed to build one logical qubit.
- *So, factoring 2048-bit RSA moduli might need millions of physical qubits*
  - Gidney & Ekers (2021): 6,000 logical qubits, 20,000,000 physical qubits, 8 hours.



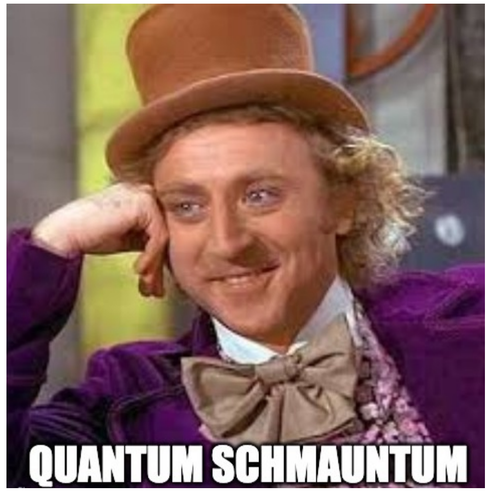
Fault-tolerant quantum computers? (2)

- It's important to note that the quantum computers built thus far are *not* (sufficiently) fault tolerant.
- So, while they are major scientific achievements, *they do not in any way threaten the security of presently-deployed cryptosystems*.
- It's still too early to be able to predict when scalable fault-tolerant quantum computers will be built.
- The next major milestone is to build a single logical qubit.
- December 2023: Major breakthrough in quantum error correction (Harvard/MIT/QuEra).
  - tinyurl.com/QC-ECC
  - See Scott Aaronson's blog post for an early analysis: scottaaronson.blog/?p=7651



- On the other hand, there is no fundamental reason why a large, fault-tolerant quantum computer *cannot* be built.

The threat of Shor and Grover



What does this mean for Internet security?

- Automatic software updates
- TLS, Signal, WhatsApp, Bluetooth, …..

Should we care?

- The NSA and other organizations are capturing and storing large amounts of internet traffic right now.
- Harvest Now Decrypt Later (HNDL) attacks.

What, if anything, should we do to mitigate the threat?

When should we take action? Now? In 5 years? In 10 years? In 20 years?

NSA's August 2015 announcement

"IAD will initiate a transition to quantum resistant algorithms in the not too distant future. Based on experience in deploying Suite B, we have determined to start planning and communicating early about the upcoming transition to quantum resistant algorithms. Our ultimate goal is to provide cost effective security against a potential quantum computer. We are working with partners across the USG, vendors, and standards bodies to ensure there is a clear plan for getting a new suite of algorithms that are developed in an open and transparent manner that will form the foundation of our next Suite of cryptographic algorithms."

"Until this new suite is developed and products are available implementing the quantum resistant suite, we will rely on current algorithms."

Optional reading: "A riddle wrapped in an enigma" eprint.iacr.org/2015/1018

PQC standardization

- tinyurl.com/pqc-nist

- NIST solicited proposals for quantum-resistant signature and key encapsulation algorithms.
- November 30, 2017: 69 submissions in Round 1.
- January 30, 2019: 26 submissions selected for Round 2.
- July 22, 2020: 7+8 submissions selected for Round 3.
- July 5, 2022: Key encapsulation scheme Kyber, and signature schemes Dilithium, Falcon, SPHINCS+ chosen for standardization.
  (The signature schemes LMS and XMSS had already been standardized in SP 800-28.)
- On August 13 2024, NIST published the standards FIPS 203 (Kyber), FIPS 204 (Dilithium), and FIPS 205 (SPHINCS+). (FIPS 206 for Falcon is expected to be completed in a year or two.)

NSA's Commercial National Security Algorithm Suite 2.0

September 7, 2022

Public-key

CRYSTALS-Dilithium

CRYSTALS-Kyber

Lattice-based signature

Lattice-based key encapsulation scheme

Symmetric-key

Advanced Encryption Standard (AES)

Secure Hash Algorithm (SHA)

Hash-based signature schemes

Software and Firmware Updates

Xtended Merkle Signature Scheme (XMSS)

Leighton-Micali Signature (LMS)

CNSA 2.0 Timeline

Google and PQC

February 23, 2023

Sundar Pichai
CEO of Google and Alphabet

We are also taking steps to develop quantum computing responsibly, given its powerful potential. Our partnerships with governments and the security community are helping to create systems that can protect internet traffic from future quantum computer attacks. And we're making sure services like Google Cloud, Android and Chrome remain safe and secure in a quantum future.

- Quantum-safe cryptography is used internally in Google for data in transit.
- Actively involved in enabling quantum-safe cryptography in TLS (this is very complex, in part because of the large signature sizes).
- Coming soon: Quantum-safe cryptographic primitives in Tink.

Messaging

SIGNAL

- September 2023: Added quantum resistance for HNDL protection: $\text{root}_0 = \text{KDF}(aV, zB, zV, zT_1, SS)$ where SS is a shared secret key obtained using Kyber.
- See: PQXDH protocol.
- Lots of remaining work to make Signal fully post-quantum secure.

Apple

- February 2024: iMessage with post-quantum security (PQ3).
- Quantum-safe root key establishment + quantum-safe rekeying.
- See: Douglas Stebila, "Security analysis of the iMessage PQ3 protocol", eprint.iacr.org/2024/357

Amazon and PQC

Quantum computing at Amazon

Amazon Braket

Accelerate quantum computing research

1 free hour of simulation time per month

for a year with AWS Free Tier

Provides access to quantum hardware:

OQC IONQ

IQuEra rigetti

Hybrid post-quantum TLS with AWS KMS

s2n-tls: ECDH + Kyber

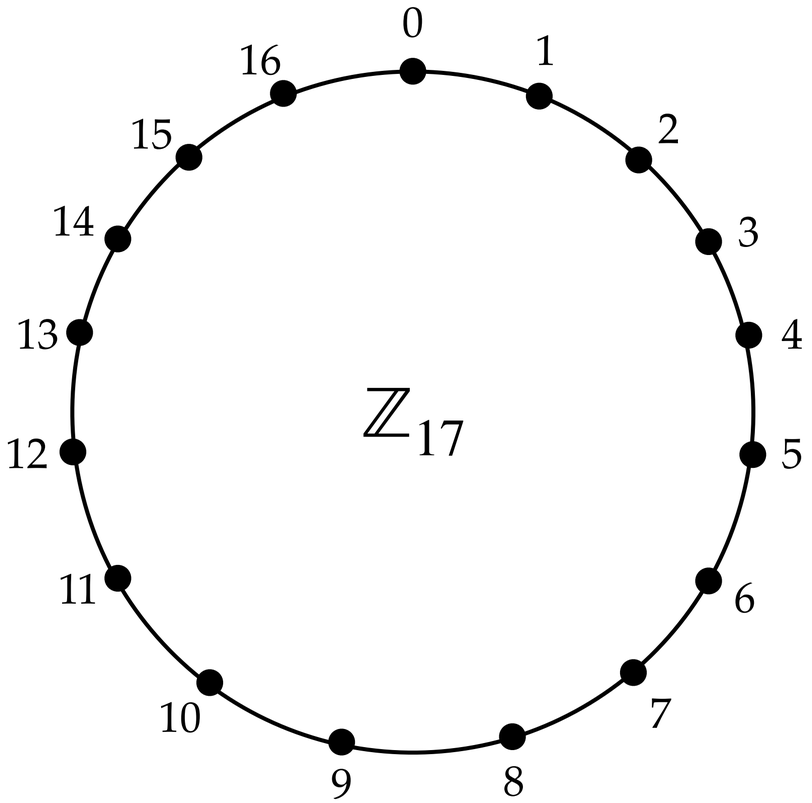
AWS Center for Quantum Computing

V1b: Mathematical prerequisites

1. Modular arithmetic
2. The polynomial ring
   $R_q = \mathbb{Z}_q[x]/(x^n + 1)$
3. The module R_q^k
4. "Small" polynomials
5. Lattice problems:
   MLWE, D-MLWE and MSIS
6. Why lattices?

Modular arithmetic

- Modulus: $q \geq 2$.
- $a \equiv b \pmod{q}$ means that $a - b$ is an integer multiple of q.
- $r = a \bmod q$ means that r is the remainder upon dividing the integer a by q (so $0 \leq r \leq q-1$).
- Integers modulo q: $\mathbb{Z}_q = \{0,1,2,\ldots,q-1\}$, where addition, subtraction and multiplication are performed modulo q.
- Example: $\mathbb{Z}_{17} = \{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16\}$.
  - In $\mathbb{Z}_{17}$, $9+15=7$, $9-15=11$, and $9 \times 15 = 16$.
  - More precisely, $9+15 = 24 \equiv 7 \pmod{17}$,
    $9-15 = -6 \equiv 11 \pmod{17}$,
    and $9 \times 15 = 135 \equiv 16 \pmod{17}$.



Polynomial rings

- Let q be a prime modulus.
- $\mathbb{Z}_q[x]$ is the set of all polynomials in x with coefficients in $\mathbb{Z}_q$.
- When adding, subtracting, multiplying and dividing polynomials in $\mathbb{Z}_q[x]$, all coefficient arithmetic is performed in $\mathbb{Z}_q$.
- Example: Let $q = 7$, and consider $f(x) = 5 + 4x^2 + 3x^3 \in \mathbb{Z}_7[x]$ and $g(x) = 6 + 3x + 2x^2 \in \mathbb{Z}_7[x]$. Then:
  - $f(x) + g(x) = 4 + 3x + 6x^2 + 3x^3$.
  - $f(x) - g(x) = 6 + 4x + 2x^2 + 3x^3$.
  - $f(x) \times g(x) = 2 + x + 6x^2 + 2x^3 + 3x^4 + 6x^5$.

The polynomial ring $R_q = \mathbb{Z}_q[x]/(x^n + 1)$

- Let q be a prime modulus, and let n be a positive integer.
- The polynomial ring $R_q = \mathbb{Z}_q[x]/(x^n + 1)$ is comprised of the polynomials in $\mathbb{Z}_q[x]$ of degree less than n, with multiplication of polynomials performed modulo the reduction polynomial $x^n + 1$.
- So, to multiply polynomials $f(x), g(x) \in R_q$:

  i. Multiply $f(x)$ and $g(x)$ in $\mathbb{Z}_q[x]$, obtaining a polynomial $h(x)$ of degree at most $2n - 2$.

  ii. Divide $h(x)$ by $x^n + 1$ to get a remainder polynomial $r(x)$ of degree at most $n - 1$.

  iii. Then $f(x) \times g(x) = r(x)$ in R_q.
- Note: The size of R_q is q^n.

Example: The polynomial ring $R_q = \mathbb{Z}_{41}[x]/(x^4 + 1)$

Let $q = 41$ and $n = 4$.

- Then R_q is comprised of the polynomials in $\mathbb{Z}_{41}[x]$ of degree at most 3.
- Let $f(x) = 32 + 17x^2 + 22x^3 \in R_q$ and $g(x) = 11 + 7x + 19x^2 + x^3 \in R_q$.
- In $\mathbb{Z}_q[x]$,
  $h(x) = f(x) \times g(x) = 24 + 19x + 16x^2 + 24x^3 + 26x^4 + 25x^5 + 22x^6$.
- *The division of $h(x)$ by $x^4 + 1$ can be accomplished by replacing x^4 by -1, x^5 by $-x$, and x^6 by $-x^2$, and then simplifying.*
- We obtain $r(x) = 24 + 19x + 16x^2 + 24x^3 - 26 - 25x - 22x^2$
  $= 39 + 35x + 35x^2 + 24x^3$.
- So, $f(x) \times g(x) = 39 + 35x + 35x^2 + 24x^3$ in R_q.

Representing polynomials as vectors

- A polynomial $f(x) = a_0 + a_1 x + \cdots + a_{n-1}x^{n-1}$ in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$ can be represented by its vector of coefficients $f = (a_0, a_1, \ldots, a_{n-1})$.
  The vector has length exactly n.
- Example: Consider $R_q = \mathbb{Z}_{41}[x]/(x^4 + 1)$.
  - The polynomials $f(x) = 23 + 11x^2 + 7x^3 \in R_q$ and $g(x) = 40 + 5x + 16x^2 \in R_q$ can be represented by the vectors $f = (23, 0, 11, 7)$ and $g = (40, 5, 16, 0)$.
  - In R_q, we have $f + g = (22, 5, 27, 7)$, $f - g = (24, 36, 36, 7)$, and $f \times g = (12, 3, 29, 7)$.

The module R_q^k

- Let k be a positive integer.
- The elements of the module R_q^k are the length-k vectors of polynomials in R_q.
- Addition and subtraction of elements in R_q^k is component-wise (so the result is also an element in R_q^k).
- The inner product (multiplication) of two vectors in R_q^k results in a polynomial in R_q.
- *All vectors in R_q^k will be written as column vectors.*

Example: R_q^k

- Let $q = 137,\ n = 4,\ R_q = \mathbb{Z}_{137}[x]/(x^4+1),\ \ k = 3.$

- Let $a = \begin{bmatrix} 93 + 51x + 34x^2 + 54x^3 \\ 27 + 87x + 81x^2 + 6x^3 \\ 112 + 15x + 46x^2 + 122x^3 \end{bmatrix}$ and $b = \begin{bmatrix} 40 + 78x + x^2 + 119x^3 \\ 11 + 31x + 57x^2 + 90x^3 \\ 108 + 72x + 47x^2 + 14x^3 \end{bmatrix} \in R_q^k.$

- Then $a + b = \begin{bmatrix} 133 + 129x + 35x^2 + 36x^3 \\ 38 + 118x + x^2 + 96x^3 \\ 83 + 87x + 93x^2 + 136x^3 \end{bmatrix},\ a - b = \begin{bmatrix} 53 + 110x + 33x^2 + 72x^3 \\ 16 + 56x + 24x^2 + 53x^3 \\ 4 + 80x + 136x^2 + 108x^3 \end{bmatrix},$

and $a^T \cdot b = a[1]b[1] + a[2]b[2] + a[3]b[3] = 93 + 59x + 44x^2 + 132x^3.$

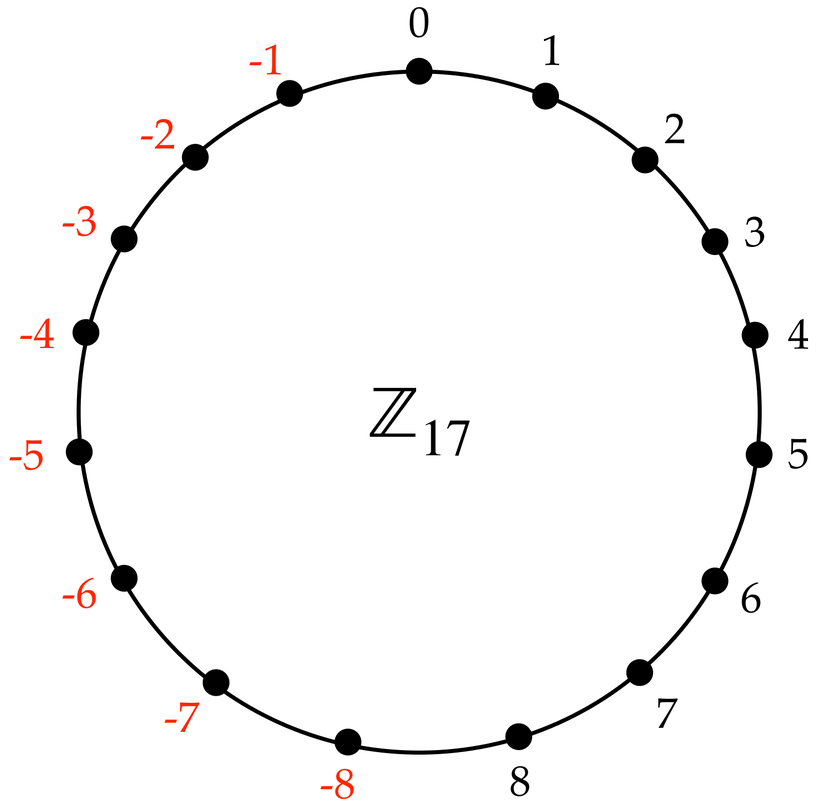
Size

- We introduce a notion of "size" for:
  - integers in $\mathbb{Z}_q$,
  - polynomials in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$, and
  - vectors of polynomials in R_q^k.
- This size is the "infinity norm", denoted by $\| \cdot \|_\infty$.
- For this, we'll need the notion of "symmetric mod".

Symmetric mod: q odd

- Let q be odd, and $r \in \mathbb{Z}_q$.
- Then r mods $q = \begin{cases} r & \text{if } r \le (q-1)/2, \\ r-q & \text{if } r > (q-1)/2, \end{cases}$

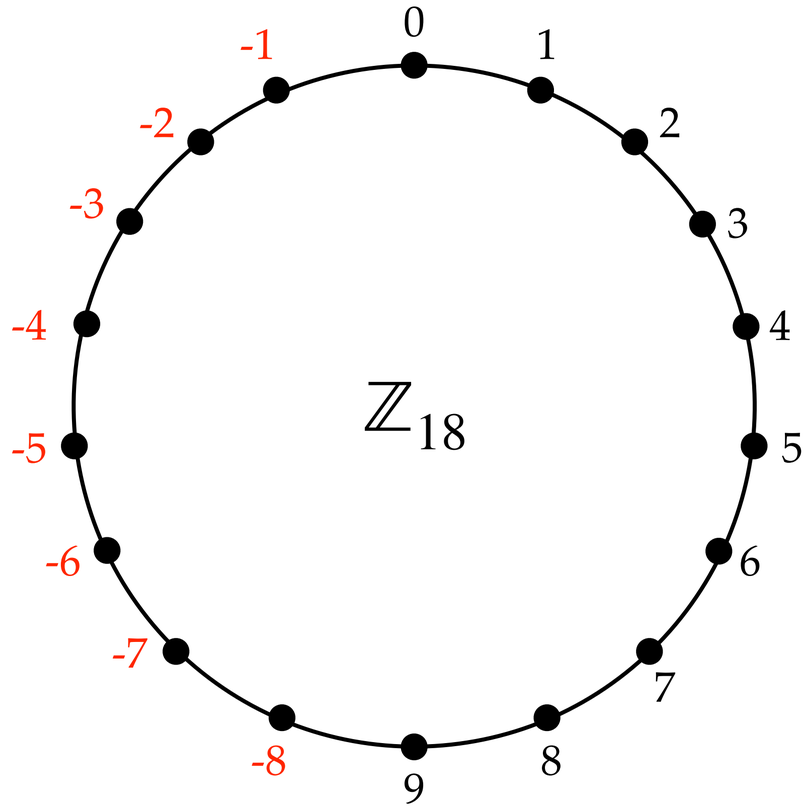
  so $-(q-1)/2 \le r \text{ mods } q \le (q-1)/2$.
- Example: Let $q = 17$.
  - $-8 \le r \text{ mods } 17 \le 8$.
  - $6 \text{ mods } 17 = 6$, and $13 \text{ mods } 17 = -4$.
  - $9 + 15 \text{ mods } 17 = 7$.
  - $9 - 15 \text{ mods } 17 = -6$.
  - $9 \times 15 \text{ mods } 17 = -1$.
- Note: mods q is also written as $\text{mod}^{+} q$.



Symmetric mod: q even

- Let q be even, and $r \in \mathbb{Z}_q$.
- Then r mods $q = \begin{cases} r & \text{if } r \le q/2, \\ r - q & \text{if } r > q/2, \end{cases}$

  so $-q/2 < r \text{ mods } q \le q/2$.
- Example: Let $q = 18$.
  - $-8 \le r \text{ mods } 18 \le 9$.
  - $6 \text{ mods } 18 = 6$, and $13 \text{ mods } 18 = -5$.
  - $9 + 15 \text{ mods } 18 = 6$.
  - $9 - 15 \text{ mods } 18 = -6$.
  - $9 \times 15 \text{ mods } 18 = 9$.
- Note: mods q is also written as $\text{mod}^{+} q$.



Size of polynomials

- Integers modulo q. Let $r \in \mathbb{Z}_q$. Then $\|r\|_\infty = |\, r \bmod q \,|$.
  - Example: Let $q = 19$. Then $\|7\|_\infty = 7$ and $\|18\|_\infty = 1$.
  - Note that $0 \le \|r\|_\infty \le (q-1)/2$ if q is odd, and $0 \le \|r\|_\infty \le q/2$ if q is even.

- Ring elements. Let $f(x) = f_0 + f_1 x + \cdots + f_{n-1}x^{n-1} \in R_q$.
  Then $\|f\|_\infty = \max \|f_i\|_\infty$.
  - Example: Let $f(x) = 1 + 12x + 3x^3 + 18x^5 \in R_{19}$. Then $\|f\|_\infty = 7$, since the mods q representation of f is $1 - 7x + 3x^3 - x^5$.
- Module elements. Let $a = [a_1, a_2, \ldots, a_k]^T \in R_q^k$.
  Then $\|a\|_\infty = \max \|a_i\|_\infty$.

"Small" polynomials

- A polynomial $f \in R_q$ is small if $\|f\|_\infty$ is "small".
- Let η be a positive integer that is small compared to $q/2$.
- Define $S_\eta = \{f \in R_q \mid \|f\|_\infty \leq \eta\}$ to be the set of polynomials in R_q all of whose coefficients have size at most η.
  S_η is the set of "small" polynomials.
- Example: Let $q = 31$. Then $1 + 30x + 29x^2 + x^4 + 2x^5 \in S_2$.
- Example: S_1 is the set of polynomials in R_q all of whose coefficients (when reduced mods q) are -1, 0, or 1.

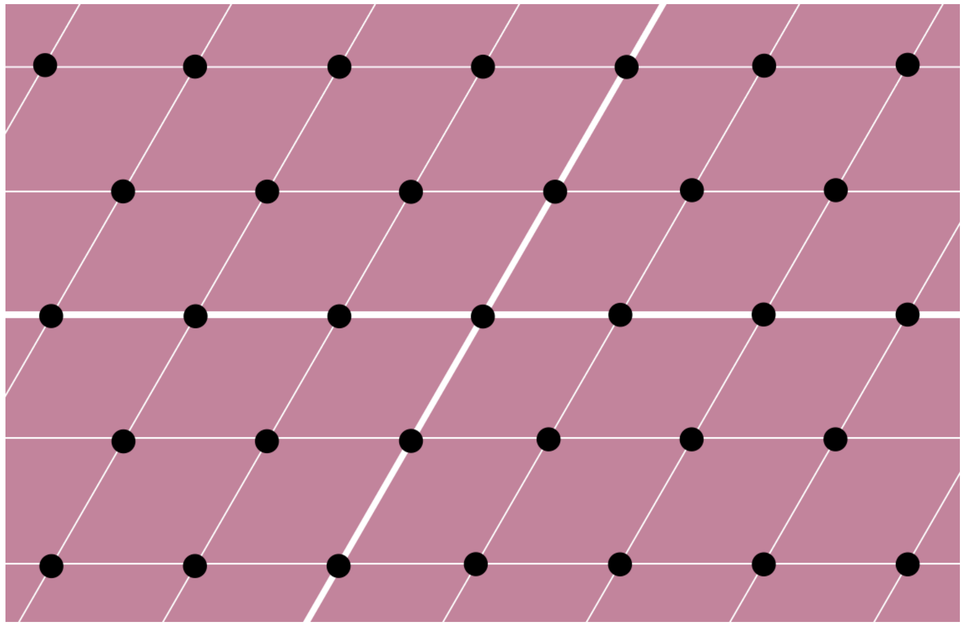
Product of small polynomials

- Claim: The product of two small polynomials in R_q is also (relatively) small.
- Example: Consider $R_q = \mathbb{Z}_{137}[x]/(x^4 + 1)$ (so $q = 137$ and $n = 4$).
  - Let $\eta = 2$, and consider $f(x) = 1 + x - 2x^2 + 2x^3 \in S_2$ and $g(x) = -2 + 2x^2 - x^3 \in S_2$.
  - Then $f(x) \cdot g(x) = 3 + 129x + 8x^2 + 134x^3 \equiv 3 - 8x + 8x^2 - 3x^3$, so $f \cdot g \in S_8$.

Product of small polynomials (2)

- Claim: If $f \in S_{\eta_1}$ and $g \in S_{\eta_2}$, then $fg \in S_{n\eta_1\eta_2}$.
- Justification:
  Consider $f(x) = f_0 + f_1 x + \cdots + f_{n-1}x^{n-1} \in S_{\eta_1}$ and
  $g(x) = g_0 + g_1 x + \cdots + g_{n-1}x^{n-1} \in S_{\eta_2}$, and let
  $h(x) = f(x)g(x) = h_0 + h_1 x + \cdots + h_{n-1}x^{n-1}$.
- For each $i \in [0, n-1]$, we have
  $h_i = f_0 g_i + f_1 g_{i-1} + \cdots + f_i g_0 - f_{i+1} g_{n-1} - f_{i+2} g_{n-2} - \cdots - f_{n-2} g_{i+2} - f_{n-1} g_{i+1}$,
  so $\|h_i\|_\infty \le n\eta_1\eta_2$.
- Thus, $h \in S_{n\eta_1\eta_2}$. □
- Similarly, if $a \in S^k_{\eta_1}$ and $b \in S^k_{\eta_2}$, then $a^T b \in S_{kn\eta_1\eta_2}$.

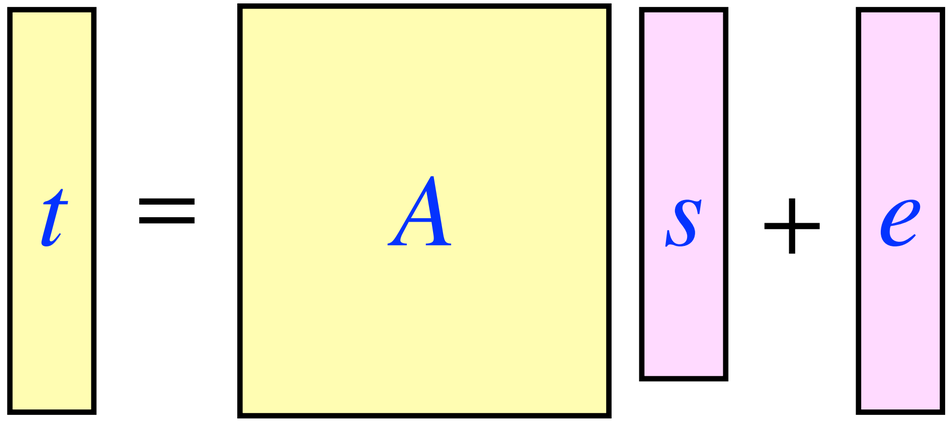
Lattice problems: MLWE, D-MLWE and MSIS



- The security of Kyber is based on the hardness of the Decisional-Module Learning With Errors (D-MLWE) problem, which in turn is related to the hardness of the MLWE problem.
- The security of Dilithium is based on the hardness of the D-MLWE and Module Short Integer Solutions (MSIS) problems.
- I'll introduce the MLWE and D-MLWE problems next.
- A description of MSIS will be deferred to the Dilithium chapter (V3).

Lattice problem: MLWE

- Parameters:
  - prime q,
  - integers n, k, ℓ with $k \geq \ell$, and
  - integers $\eta_1, \eta_2 \ll q/2$.



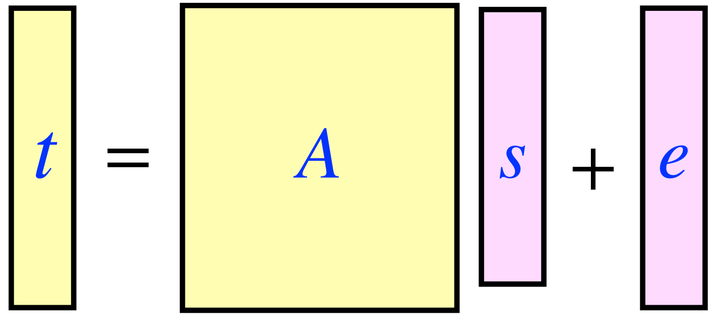
- Module Learning With Errors (MLWE) instance:
  - $A \in_R R_q^{k \times \ell}$ (where $R_q = \mathbb{Z}_q[x]/(x^n + 1)$).
  - $t = As + e$, where $s \in_R S_{\eta_1}^{\ell}$ and $e \in_R S_{\eta_2}^{k}$ (so $t \in R_q^k$).
- Required: s.

Example: MLWE

- Parameters: $q = 541$, $n = 4$, $k = 3$, $\ell = 2$, $\eta_1 = 3$, $\eta_2 = 2$, so $R_q = \mathbb{Z}_{541}[x]/(x^4 + 1)$.
- MLWE instance generation: Randomly select

$$A = \begin{bmatrix} 442 + 502x + 513x^2 + 15x^3 & 368 + 166x + 37x^2 + 135x^3 \\ 479 + 532x + 116x^2 + 41x^3 & 12 + 139x + 385x^2 + 409x^3 \\ 29 + 394x + 503x^2 + 389x^3 & 9 + 499x + 92x^2 + 254x^3 \end{bmatrix} \in R_q^{3\times 2},$$

$$s = \begin{bmatrix} 2 - 2x + x^3 \\ 3 - 2x - 2x^2 - 2x^3 \end{bmatrix} \in S_3^2, \text{ and } e = \begin{bmatrix} 2 - 2x - x^2 \\ 1 + 2x + 2x^2 + x^3 \\ -2 - x^2 - 2x^3 \end{bmatrix} \in S_2^3,$$

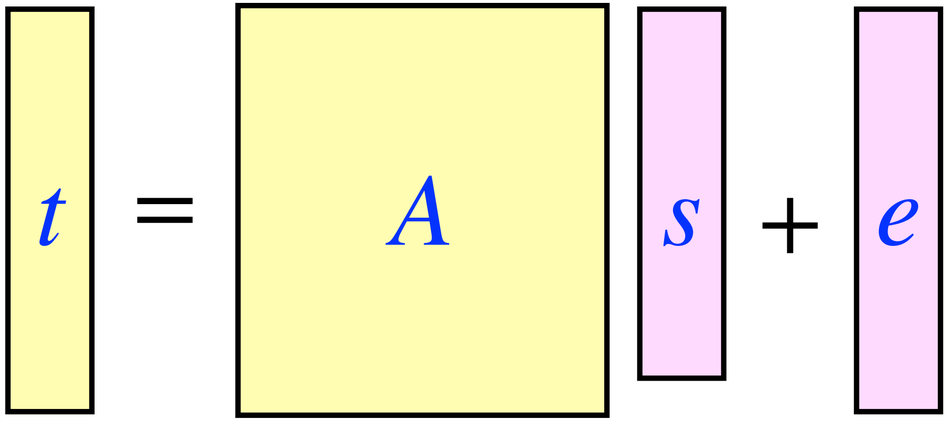


and compute $t = As + e = \begin{bmatrix} 30 + 252x + 401x^2 + 332x^3 \\ 247 + 350x + 259x^2 + 485x^3 \\ 534 + 234x + 137x^2 + 443x^3 \end{bmatrix}$. Note that $\|t\|_\infty = 259$.

- MLWE problem: Given (A, t), determine s.

Lattice problem: D-MLWE

- Parameters:
  - prime q,
  - integers n, k, ℓ with $k \geq \ell$, and
  - integers $\eta_1, \eta_2 \ll q/2$.



  - Decisional-Module Learning With Errors (MLWE) instance:
    - $A \in_R R_q^{k \times \ell}$ (where $R_q = \mathbb{Z}_q[x]/(x^n + 1)$).
    - $z \in R_q^k$, where $z = t$ with probability $\frac{1}{2}$ (and $t = As + e$ with $s \in_R S_{\eta_1}^{\ell}$ and $e \in_R S_{\eta_2}^{k}$) and $z \in_R R_q^k$ with probability $\frac{1}{2}$.
- Required: Determine whether (A, z) is an MLWE instance (so $z = t$) or not.

Why lattices?

Lattices play two roles in assessing the hardness of MLWE, D-MLWE and MSIS.

1. In 2012, Langlois and Stehlé proved that the *average-case* hardness of MLWE, D-MLWE and MSIS is at least that of the *worst-case quantum* hardness of certain natural lattice problems in certain structured lattices. However, this provable guarantee is highly asymptotic in nature, and so it isn't clear what assurances, if any, it offers about the hardness of MLWE, D-MLWE and MSIS in practice.

https://www.latticechallenge.org/

2. The MLWE, D-MLWE and MSIS problems can be rephrased as lattice problems. These lattice problems are being intensively studied, and the fastest attacks known are used to assess the hardness of MLWE, D-MLWE and MSIS, and thereby justify concrete parameters for Kyber and Dilithium.

