New Reductions and Constructions for Module Learning With Errors

Auditions CNRS - Concours 06/02

Katharina Boudgoust

- Since Jan'22: Postdoc in Aarhus, hosted by P. Scholl (Denmark)
- Nov'21: PhD in Rennes, supervised by A. Roux-Langlois and P.-A. Fouque (France)
- May'18: MSc in Karlsruhe (Germany)

Research Context

The security in cryptography relies on presumably hard mathematical problems.

Currently used problems:

- Discrete Logarithm
- Factoring

Research Context

The security in cryptography relies on presumably hard mathematical problems.

Currently used problems:

- ~~Discrete Logarithm~~
- ~~Factoring~~

⚠ $\exists$ poly-time quantum algorithm [Sho97]

Quantum-resistant candidates:

- Euclidean Lattices
- Codes
- Isogenies
- Multivariate Systems
- **?**

Research Context

The security in cryptography relies on presumably hard mathematical problems.

Currently used problems:

- ~~Discrete Logarithm~~
- ~~Factoring~~

⚠ $\exists$ poly-time quantum algorithm [Sho97]

Quantum-resistant candidates:

- Euclidean Lattices
- Codes
- Isogenies
- Multivariate Systems
- **?**

Lattice problems in crypto:

- Short Integer Solution
- NTRU
- Learning With Errors

Research Context

The security in cryptography relies on presumably hard mathematical problems.

Currently used problems:

- ~~Discrete Logarithm~~
- ~~Factoring~~

⚠ $\exists$ poly-time quantum algorithm [Sho97]

Quantum-resistant candidates:

- Euclidean Lattices
- Codes
- Isogenies
- Multivariate Systems
- **?**

Lattice problems in crypto:

- Short Integer Solution
- NTRU
- Learning With Errors

$\underbrace{\qquad\qquad\qquad\qquad}_{\text{Lattice-Based Cryptography}}$

US National Institute of Standards and Technology (NIST) Competition

- 2016: start of NIST's post-quantum cryptography project
- 2022: selection of 4 schemes, 3 of them relying on lattice problems

Public Key Encryption

- Kyber: Learning With Errors

Digital Signature

- Dilithium: Learning With Errors
- Falcon: NTRU and Short Integer Solution

US National Institute of Standards and Technology (NIST) Competition

key role

- 2016: start of NIST's post-quantum cryptography project
- 2022: selection of 4 schemes, 3 of them relying on lattice problems

Public Key Encryption

- Kyber: Learning With Errors

Digital Signature

- Dilithium: Learning With Errors
- Falcon: NTRU and Short Integer Solution

My research:

- Hardness of Learning With Errors
- Construction of cryptographic schemes relying their security on it

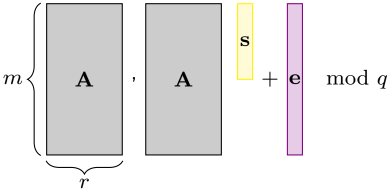
Binary Hardness of Module Learning With Errors

Joint work with C. Jeudy, A. Roux-Langlois and W. Wen

The Learning With Errors (LWE) Problem [Reg05]

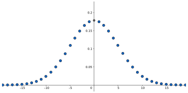
$\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$ for some integer q

$\mathbf{A} \sim \mathsf{Unif}(\mathbb{Z}_q^{m\times r})$, $\mathbf{s} \sim$ DistrS and $\mathbf{e} \sim$ DistrE



Given $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q)$, find $\mathbf{s}$

The Learning With Errors (LWE) Problem [Reg05]



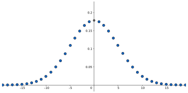
$\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$ for some integer q

$\mathbf{A} \sim$ Unif$(\mathbb{Z}_q^{m \times r})$, $\mathbf{s} \sim$ DistrS and $\mathbf{e} \sim$ DistrE

$$m\left\{ \underbrace{\mathbf{A}}_{r} \right. , \; \mathbf{A} \, \mathbf{s} + \mathbf{e} \mod q$$

Given $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q)$, find $\mathbf{s}$

The Learning With Errors (LWE) Problem [Reg05]



$\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$ for some integer q

$\mathbf{A} \sim \mathsf{Unif}(\mathbb{Z}_q^{m \times r})$, $\mathbf{s} \sim$ DistrS and $\mathbf{e} \sim$ DistrE

$$1000 \approx m \left\{ \underbrace{\mathbf{A}}_{r \,\approx\, 500} \right. , \quad \mathbf{A}\,\mathbf{s} + \mathbf{e} \mod q \approx 2^{15}$$

Given $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q)$, find $\mathbf{s}$

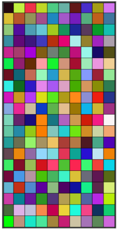
⚠ Storage	$m(r+1)\log_2 q$ bits
⚠ Computation	$O(mr)$ operations over $\mathbb{Z}_q$

Idea

Improve efficiency by adding **structure**!

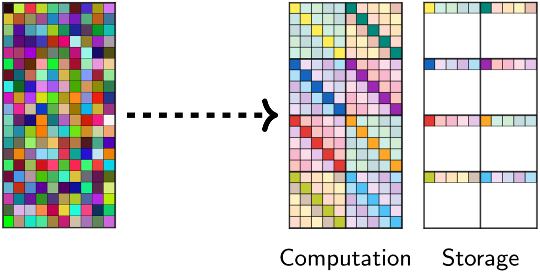
Idea

Improve efficiency by adding **structure**!



Idea

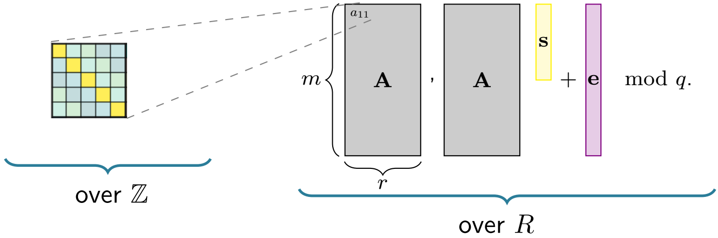
Improve efficiency by adding **structure**!



Module Learning With Errors (Module-LWE) [BGV12, LS15]

Idea: replace $\mathbb{Z}$ by the ring of integers R of some number field K
sample $\mathbf{A}$ random over $R \Rightarrow$ structured over $\mathbb{Z}$

$\mathbf{A} \sim \mathsf{Unif}(R_q^{m\times r})$, $\mathbf{s} \sim$ DistrS and $\mathbf{e} \sim$ DistrE



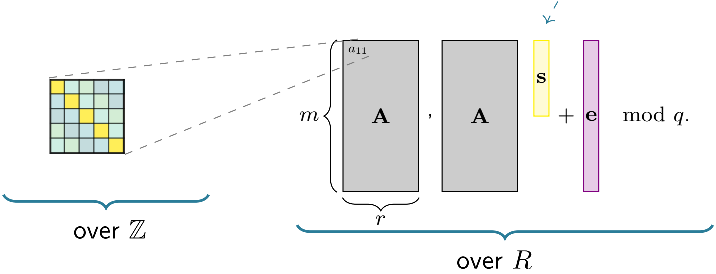
Given $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q)$, find $\mathbf{s}$

Module Learning With Errors (Module-LWE) [BGV12, LS15]

Idea: replace $\mathbb{Z}$ by the ring of integers R of some number field K
sample $\mathbf{A}$ random over $R \Rightarrow$ structured over $\mathbb{Z}$

$\mathbf{A} \sim \text{Unif}(R_q^{m\times r})$, $\mathbf{s} \sim$ DistrS and $\mathbf{e} \sim$ DistrE

secret key

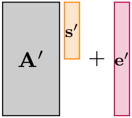


Given $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q)$, find $\mathbf{s}$

The security of many lattice-based schemes relies on the assumed hardness of Module-LWE.

Module-LWE with Binary Secrets

Standard Module-LWE



$s'_i \in \{0, \ldots, q-1\}$

Module-LWE with Binary Secrets

Standard Module-LWE	Binary Secret Module-LWE
$\mathbf{A}' \mathbf{s}' + \mathbf{e}'$	$\mathbf{A} \mathbf{s} + \mathbf{e}$
$s'_i \in \{0, \ldots, q-1\}$	$s_i \in \{0, 1\}$

Why binary secrets?

- Efficiency
- Functionality (e.g., Fully Homomorphic Encryption)

Module-LWE with Binary Secrets

Standard Module-LWE $\leq$ Binary Secret Module-LWE

$s'_i \in \{0, \ldots, q-1\}$

$s_i \in \{0, 1\}$

Why binary secrets?

- Efficiency
- Functionality (e.g., Fully Homomorphic Encryption)

Contribution:

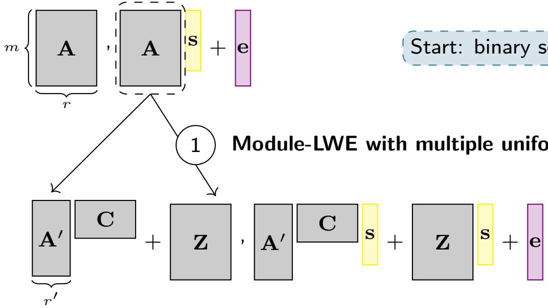
- Proving hardness of Module-LWE with a binary secret
- $\dim(\mathbf{s}) > \dim(\mathbf{s}')$ and $\|\mathbf{e}\| > \|\mathbf{e}'\|$

Proof of Hardness of Module-LWE with Binary Secrets

$$m\left\{\underbrace{\boxed{\mathbf{A}}}_{r}\right. , \quad \boxed{\mathbf{A}}\,\boxed{\mathbf{s}} + \boxed{\mathbf{e}}$$

Start: binary secret $\mathbf{s}$

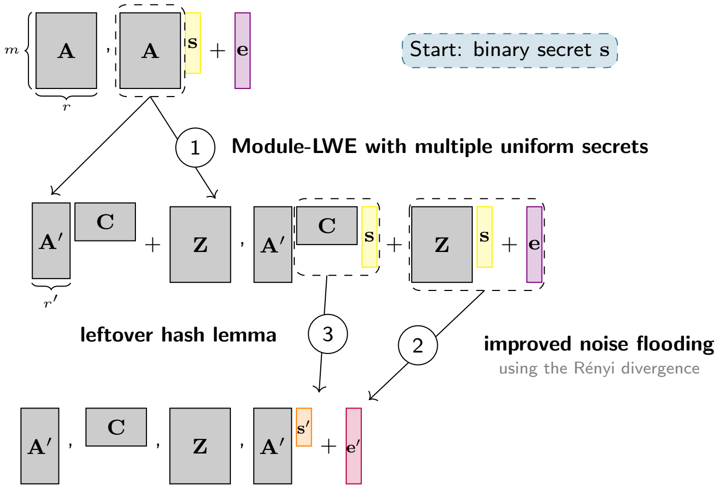
Proof of Hardness of Module-LWE with Binary Secrets



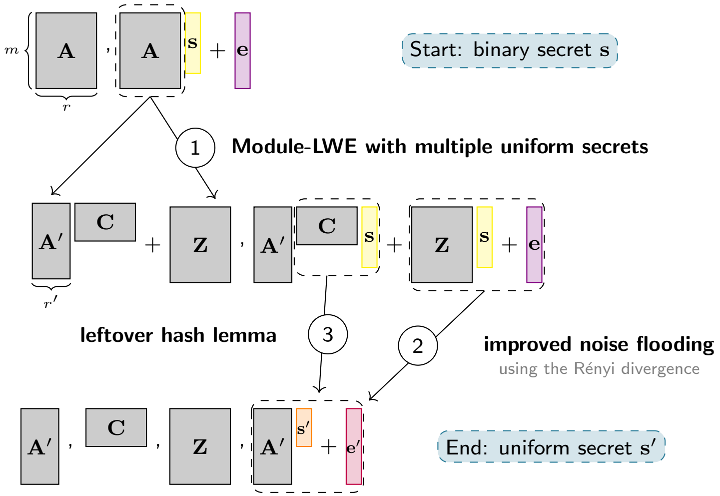
Start: binary secret $\mathbf{s}$

Module-LWE with multiple uniform secrets

Proof of Hardness of Module-LWE with Binary Secrets



Proof of Hardness of Module-LWE with Binary Secrets



Contributions:

- Proving hardness of Module-LWE
  - with a binary secret [BJRW20] [BJRW21]

Contributions:

- Proving hardness of Module-LWE
  - with a binary secret [BJRW20] [BJRW21]
  - with secret of high enough entropy [BJRW22]
  - with η-bounded secrets and noise [BJRW23]

Impact:

- NIST: Kyber & Dilithium use Module-LWE with $\eta \leq 4$ for secret and noise

Contributions:

- Proving hardness of Module-LWE
  - with a binary secret [BJRW20] [BJRW21]
  - with secret of high enough entropy [BJRW22]
  - with η-bounded secrets and noise [BJRW23]
- (Dis)prove hardness of new lattice problems
  - middle-product learning with rounding [BBD+19]
  - partial Vandermonde LWE [BSS22]
  - easy instances of partial Vandermonde LWE [BGP22]

Best Early Career Researcher Award at Crypto'22

- Construct cryptographic schemes on Module-LWE which allow
  - to aggregate signatures [BT23]
  - to threshold decryption [BS23]

Impact:

- NIST: Kyber & Dilithium use Module-LWE with $\eta \leq 4$ for secret and noise

Contributions:

- Proving hardness of Module-LWE
  - with a binary secret [BJRW20] [BJRW21]
  - with secret of high enough entropy [BJRW22]
  - with η-bounded secrets and noise [BJRW23]
- (Dis)prove hardness of new lattice problems
  - middle-product learning with rounding [BBD+19]
  - partial Vandermonde LWE [BSS22]
  - easy instances of partial Vandermonde LWE [BGP22]

Best Early Career Researcher Award at Crypto'22

- Construct cryptographic schemes on Module-LWE which allow
  - to aggregate signatures [BT23]
  - to threshold decryption [BS23]

Impact:

- NIST: Kyber & Dilithium use Module-LWE with $\eta \leq 4$ for secret and noise

My Research:

- security foundations of lattice-based cryptography
- construction of advanced cryptographic schemes

Research Project:

New Reductions and Constructions for Module Learning With Errors

Research Project

I. Hardness of Module Learning With Errors

- Entropic noise distribution
- Relation between different rings and metrics
- Relation to Partial Vandermonde problems

$$\left(\mathbf{A} \,,\ \mathbf{A}\mathbf{s} + \mathbf{e} \right)$$

II. Advanced Lattice-Based Encryption

- Threshold decryption
- Key-updatable encryption

III. Advanced Lattice-Based Signatures

- Aggregate signatures
- Threshold signatures

Research Project

I. Hardness of Module Learning With Errors

- Entropic noise distribution
- Relation between different rings and metrics
- Relation to Partial Vandermonde problems

$$\mathbf{A},\ \mathbf{A}\mathbf{s}+\mathbf{e}$$

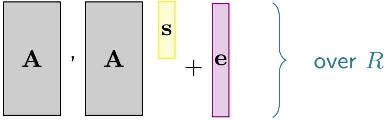
II. Advanced Lattice-Based Encryption

- Threshold decryption
- Key-updatable encryption

III. Advanced Lattice-Based Signatures

- Aggregate signatures
- Threshold signatures

I. Relation Between Different Rings



State of the art:

- Almost all practical schemes: $R = \mathbb{Z}[x]/(x^d + 1)$
- Theoretical results: any ring of integers of a number field

Goal:

- Study relation between M-LWE over different rings

$\rightarrow$ Impacts the security of standardized schemes

II. Threshold Decryption

Motivation:

- Distribute secret key among several parties $\rightarrow$ higher security
- For instance: storing sensitive data, multi-party computations, ...
- 2023: NIST called for standardization

State of the art:

- Low security and good efficiency [BS23]
- High security and poor efficiency [DLN+21]

Goals:

- Propose solution with high security and good efficiency

II. Threshold Decryption

⚠ Single Point of Failure

Motivation:

- Distribute secret key among several parties → higher security
- For instance: storing sensitive data, multi-party computations, ...
- 2023: NIST called for standardization

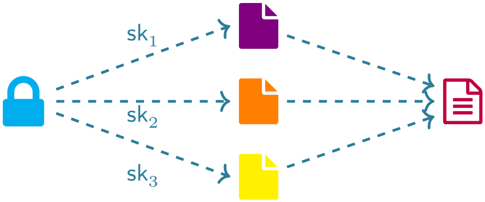
State of the art:

- Low security and good efficiency [BS23]
- High security and poor efficiency [DLN+21]

Goals:

- Propose solution with high security and good efficiency

II. Threshold Decryption



Motivation:

- Distribute secret key among several parties $\rightarrow$ higher security
- For instance: storing sensitive data, multi-party computations, ...
- 2023: NIST called for standardization

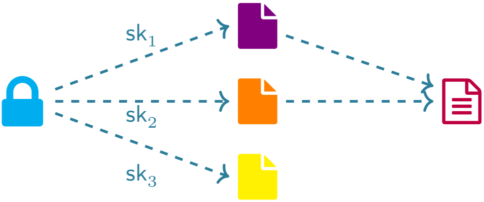
State of the art:

- Low security and good efficiency [BS23]
- High security and poor efficiency [DLN+21]

Goals:

- Propose solution with high security and good efficiency

II. Threshold Decryption



Motivation:

- Distribute secret key among several parties $\rightarrow$ higher security
- For instance: storing sensitive data, multi-party computations, ...
- 2023: NIST called for standardization

State of the art:

- Low security and good efficiency [BS23]
- High security and poor efficiency [DLN$^+$21]

Goals:

- Propose solution with high security and good efficiency

III. Threshold Signatures

Motivation:

- Distribute secret key among several parties → higher security
- For instance: decentralized currencies, blockchains, ...

State of the art:

- Only few solutions, all use powerful cryptographic tools [BGG$^+$18, ASY22]
- NIST's signatures Falcon and Dilithium seem not well-suited

Goals:

- Study if less popular schemes are better suited, with simple tools

III. Threshold Signatures

Single Point of Failure

Motivation:

- Distribute secret key among several parties $\rightarrow$ higher security
- For instance: decentralized currencies, blockchains, ...

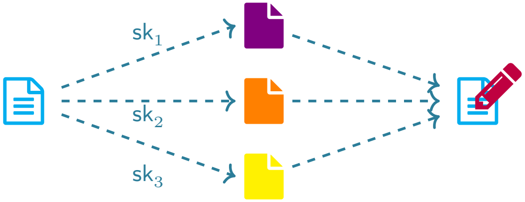
State of the art:

- Only few solutions, all use powerful cryptographic tools [BGG+18, ASY22]
- NIST's signatures Falcon and Dilithium seem not well-suited

Goals:

- Study if less popular schemes are better suited, with simple tools

III. Threshold Signatures



Motivation:

- Distribute secret key among several parties $\rightarrow$ higher security
- For instance: decentralized currencies, blockchains, ...

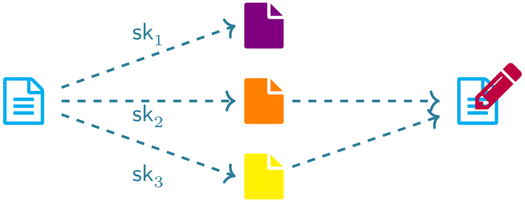
State of the art:

- Only few solutions, all use powerful cryptographic tools [BGG+18, ASY22]
- NIST's signatures Falcon and Dilithium seem not well-suited

Goals:

- Study if less popular schemes are better suited, with simple tools

III. Threshold Signatures



Motivation:

- Distribute secret key among several parties $\rightarrow$ higher security
- For instance: decentralized currencies, blockchains, ...

State of the art:

- Only few solutions, all use powerful cryptographic tools [BGG+18, ASY22]
- NIST's signatures Falcon and Dilithium seem not well-suited

Goals:

- Study if less popular schemes are better suited, with simple tools

Integration and Updates

Updates:

- RFP-013 Cryptonet Network Grant from Protocol Labs (25.000 USD)
- "Overfull: Too Large Aggregate Signatures Based on Lattices" with Adeline Roux-Langlois accepted at The Computer Journal

Integration:

- Research Group ECO at the LIRMM in Montpellier (UMR 5506)
  - computer algebra, cryptography and algorithmic number theory

  $\rightarrow$ contribution: lattice-based cryptography
- Research Group AriC at ENS Lyon (UMR 5668)
  - lattice-based cryptography, advanced cryptographic constructions

  $\rightarrow$ contribution: reductions and constructions on Module-LWE
- Research Group CARAMBA at Inria Nancy (UMR 7503)
  - factorization, algebraic curves and algebraic methods for cryptanalysis

  $\rightarrow$ contribution: lattices, lattice-based cryptography

Updates:

- RFP-013 Cryptonet Network Grant from Protocol Labs (25.000 USD)
- "Overfull: Too Large Aggregate Signatures Based on Lattices" with Adeline Roux-Langlois accepted at The Computer Journal

Integration:

- Research Group ECO at the LIRMM in Montpellier (UMR 5506)
  - computer algebra, cryptography and algorithmic number theory

  → contribution: lattice-based cryptography
- Research Group AriC at ENS Lyon (UMR 5668)
  - lattice-based cryptography, advanced cryptographic constructions

  → contribution: reductions and constructions on Module-LWE
- Research Group CARAMBA at Inria Nancy (UMR 7503)
  - factorization, algebraic curves and algebraic methods for cryptanalysis

  → contribution: lattices, lattice-based cryptography

Merci.

Shweta Agrawal, Damien Stehlé, and Anshu Yadav.
Round-optimal lattice-based threshold signatures, revisited.
In *ICALP*, volume 229 of *LIPIcs*, pages 8:1–8:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

Shi Bai, Katharina Boudgoust, Dipayan Das, Adeline Roux-Langlois, Weiqiang Wen, and Zhenfei Zhang.
Middle-product learning with rounding problem and its applications.
In *ASIACRYPT (1)*, volume 11921 of *Lecture Notes in Computer Science*, pages 55–81. Springer, 2019.

Dan Boneh, Rosario Gennaro, Steven Goldfeder, Aayush Jain, Sam Kim, Peter M. R. Rasmussen, and Amit Sahai.
Threshold cryptosystems from threshold fully homomorphic encryption.
In *CRYPTO (1)*, volume 10991 of *Lecture Notes in Computer Science*, pages 565–596. Springer, 2018.

Katharina Boudgoust, Erell Gachon, and Alice Pellet-Mary.
Some easy instances of ideal-svp and implications on the partial vandermonde knapsack problem.
In *CRYPTO (2)*, volume 13508 of *Lecture Notes in Computer Science*, pages 480–509. Springer, 2022.

Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan.
(leveled) fully homomorphic encryption without bootstrapping.

In *ITCS*, pages 309–325. ACM, 2012.

Katharina Boudgoust, Corentin Jeudy, Adeline Roux-Langlois, and Weiqiang Wen.
Towards classical hardness of module-lwe: The linear rank case.
In *ASIACRYPT (2)*, volume 12492 of *Lecture Notes in Computer Science*, pages 289–317. Springer, 2020.

Katharina Boudgoust, Corentin Jeudy, Adeline Roux-Langlois, and Weiqiang Wen.
On the hardness of module-lwe with binary secret.
In *CT-RSA*, volume 12704 of *Lecture Notes in Computer Science*, pages 503–526. Springer, 2021.

Katharina Boudgoust, Corentin Jeudy, Adeline Roux-Langlois, and Weiqiang Wen.
Entropic hardness of module-lwe from module-ntru.
In *INDOCRYPT*, volume 13774 of *Lecture Notes in Computer Science*, pages 78–99. Springer, 2022.

Katharina Boudgoust, Corentin Jeudy, Adeline Roux-Langlois, and Weiqiang Wen.
On the hardness of module learning with errors with short distributions.
J. Cryptol., 36(1):1, 2023.

Katharina Boudgoust and Peter Scholl.
Simple threshold (fully homomorphic) encryption from LWE with polynomial modulus.
IACR Cryptol. ePrint Arch., page 16, 2023.

Katharina Boudgoust, Amin Sakzad, and Ron Steinfeld.
Vandermonde meets regev: public key encryption schemes based on partial vandermonde problems.
Des. Codes Cryptogr., 90(8):1899–1936, 2022.

Katharina Boudgoust and Akira Takahashi.
Sequential half-aggregation of lattice-based signatures.
IACR Cryptol. ePrint Arch., page 159, 2023.

Julien Devevey, Benoît Libert, Khoa Nguyen, Thomas Peters, and Moti Yung.
Non-interactive cca2-secure threshold cryptosystems: Achieving adaptive security in the standard model without pairings.
In *Public Key Cryptography (1)*, volume 12710 of *Lecture Notes in Computer Science*, pages 659–690. Springer, 2021.

Craig Gidney and Martin Ekerå.
How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits.
Quantum, 5:433, 2021.

Elie Gouzien and Nicolas Sangouard.
Factoring 2048 rsa integers in 177 days with 13436 qubits and a multimode memory, 2021.

Adeline Langlois and Damien Stehlé.
Worst-case to average-case reductions for module lattices.
Des. Codes Cryptogr., 75(3):565–599, 2015.

Oded Regev.
On lattices, learning with errors, random linear codes, and cryptography.
In *STOC*, pages 84–93. ACM, 2005.

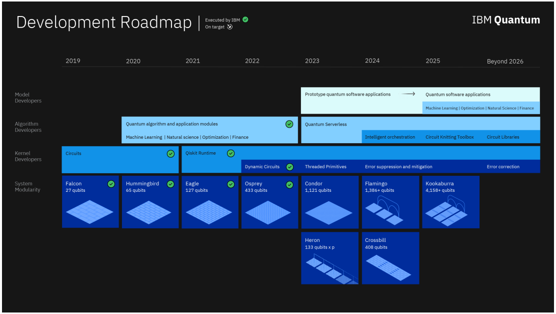
Peter W. Shor.
Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer.
SIAM J. Comput., 26(5):1484–1509, 1997.

Backup

Quantum Computing

We need: ≥ 1 million qbits (run-time/memory) to factor a 2048 bits RSA integer [GE21, GS21].

We have: ca. 433 qbits - IBM Quantum Roadmap*



But:

- Store now, encrypt later
- Safely switching takes time
- Optimization still necessary

*research.ibm.com/blog/ibm-quantum-roadmap

NIST Competition (Continued)

Goal: Standardize digital signatures (DS) and key exchange mechanisms (KEM), that are secure against quantum computers.

12/2016 Call for proposals

11/2017 82 candidates submitted (21 for the AES competition in 1998 and 64 for the SHA3 competition in 2008)

12/2017 69 submissions accepted

- 5 out of 20 DS based on lattices
- 21 out of 49 KEMs based on lattices

04/2018 1st NIST PQC Standardization Conference

01/2019 End of 1st round $\rightarrow$ 2nd round

- 3 out of 9 DS based on lattices
- 9 out of 17 KEMs based on lattices

08/2019 2nd NIST PQC Standardization Conference

07/2020 End of second round $\rightarrow$ 3rd round

- 5 out of 7 finalists are based on lattices (Kyber, NTRU, Saber, Dilithium, Falcon)
- 2 out of 8 alternate candidates are based on lattices (NTRU Prime, FrodoKEM)

07/2022 End of third round

- 3 out of 4 standardized schemes are based on lattices (Kyber, Dilithium, Falcon)

NIST organizes regular workshops and moderates a discussion forum.