Post-Quantum Cryptosystems Based on Error-Correcting Codes

Marco Baldi[1] and Alessandro Barenghi[2]

[1]Università Politecnica delle Marche (m.baldi@univpm.it)
[2]Politecnico di Milano (alessandro.barenghi@polimi.it)

Second Italian Conference on Cyber Security (ITASEC18)

Milan, Italy
February 8th, 2018

Code-based crypto

- Code-based public-key cryptosystems were introduced by McEliece in 1978.
- Besides quantum resistance, they exhibit excellent algorithmic efficiency.

▶ R. McEliece, "Public-Key System Based on Algebraic Coding Theory," DSN Progress Report 44, pp. 114–116, 1978.

Code-based crypto

- Code-based public-key cryptosystems were introduced by McEliece in 1978.
- Besides quantum resistance, they exhibit excellent algorithmic efficiency.
- In 1986 Niederreiter introduced another code-based public-key cryptosystem in the syndrome domain, while McEliece works in the codeword domain.

- R. McEliece, "Public-Key System Based on Algebraic Coding Theory," DSN Progress Report 44, pp. 114–116, 1978.
- H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," Problems of Control and Information Theory, vol. 15, pp. 159–166, 1986.

Code-based crypto

- Code-based public-key cryptosystems were introduced by McEliece in 1978.
- Besides quantum resistance, they exhibit excellent algorithmic efficiency.
- In 1986 Niederreiter introduced another code-based public-key cryptosystem in the syndrome domain, while McEliece works in the codeword domain.
- McEliece and Niederreiter indeed are two formulations of the same code-based trapdoor.

- R. McEliece, "Public-Key System Based on Algebraic Coding Theory," DSN Progress Report 44, pp. 114–116, 1978.
- H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," Problems of Control and Information Theory, vol. 15, pp. 159–166, 1986.
- Y. X. Li, R. H. Deng and X. M. Wang, "On the equivalence of McEliece's and Niederreiter's public-key cryptosystems," IEEE Trans. Inf. Theory, vol. 40, no. 1, pp. 271–273, Jan 1994.

Trapdoors from decoding

First ingredient for a trapdoor

The problem of decoding a random linear code has no known solution in polynomial time.

- E. Berlekamp, R. McEliece and H. van Tilborg, "On the inherent intractability of certain coding problems," IEEE Trans. Inf. Theory, vol. 24, no. 3, pp. 384–386, May 1978.
- A. May, A. Meurer, E. Thomae, "Decoding Random Linear Codes in $\tilde{\mathcal{O}}(2^{0.054n})$," Advances in Cryptology ASIACRYPT 2011, Dec. 2011.

Trapdoors from decoding

First ingredient for a trapdoor

The problem of decoding a random linear code has no known solution in polynomial time.

Second ingredient for a trapdoor

Many families of non-random (Goppa, GRS, convolutional) and quasi-random (LDPC, MDPC) linear codes admit polynomial-time decoding algorithms.

- E. Berlekamp, R. McEliece and H. van Tilborg, "On the inherent intractability of certain coding problems," IEEE Trans. Inf. Theory, vol. 24, no. 3, pp. 384–386, May 1978.
- A. May, A. Meurer, E. Thomae, "Decoding Random Linear Codes in $\tilde{\mathcal{O}}(2^{0.054n})$," Advances in Cryptology ASIACRYPT 2011, Dec. 2011.

McEliece cryptosystem - key generation

Private key

- $k \times n$ generator matrix $\mathbf{G}$ of a secret Goppa code,
- random dense $k \times k$ non-singular "scrambling" matrix $\mathbf{S}$,
- random $n \times n$ permutation matrix $\mathbf{P}$.

McEliece cryptosystem - key generation

Private key

- $k \times n$ generator matrix $\mathbf{G}$ of a secret Goppa code,
- random dense $k \times k$ non-singular "scrambling" matrix $\mathbf{S}$,
- random $n \times n$ permutation matrix $\mathbf{P}$.

Public key

$$\mathbf{G}' = \mathbf{S} \cdot \mathbf{G} \cdot \mathbf{P}$$

McEliece cryptosystem - key generation

Private key

- $k \times n$ generator matrix $\mathbf{G}$ of a secret Goppa code,
- random dense $k \times k$ non-singular "scrambling" matrix $\mathbf{S}$,
- random $n \times n$ permutation matrix $\mathbf{P}$.

Public key

$$\mathbf{G}' = \mathbf{S} \cdot \mathbf{G} \cdot \mathbf{P}$$

- The public code is permutation equivalent to the secret code.
- In some recent variants, $\mathbf{P}$ is replaced with a more general sparse matrix $\mathbf{Q}$ to avoid this permutation equivalence.

McEliece cryptosystem - encryption

1. Alice gets Bob's public key $\mathbf{G}'$.
2. She generates a random error vector $\mathbf{e}$ of length n and weight t.
3. She encrypts any k-bit block $\mathbf{u}$ as

$$\mathbf{x} = \mathbf{u} \cdot \mathbf{G}' + \mathbf{e} = \mathbf{c} + \mathbf{e}$$

McEliece cryptosystem - encryption

1. Alice gets Bob's public key $\mathbf{G}'$.
2. She generates a random error vector $\mathbf{e}$ of length n and weight t.
3. She encrypts any k-bit block $\mathbf{u}$ as

$$\mathbf{x} = \mathbf{u} \cdot \mathbf{G}' + \mathbf{e} = \mathbf{c} + \mathbf{e}$$

Alert

This only provides semantic security!

McEliece cryptosystem - encryption

1. Alice gets Bob's public key $\mathbf{G}'$.
2. She generates a random error vector $\mathbf{e}$ of length n and weight t.
3. She encrypts any k-bit block $\mathbf{u}$ as

$$\mathbf{x} = \mathbf{u} \cdot \mathbf{G}' + \mathbf{e} = \mathbf{c} + \mathbf{e}$$

Alert

This only provides semantic security!

CCA2 conversions

Suitable conversions exist to achieve security against adaptive chosen-ciphertext attacks

- K. Kobara, H. Imai, "Semantically secure McEliece public-key cryptosystems — conversions for McEliece PKC," PKC 2001, vol. 1992 of Springer LNCS, pp. 19–35, 2001.

McEliece cryptosystem - decryption

1. Bob computes

$$\begin{aligned}\mathbf{x}' &= \mathbf{x} \cdot \mathbf{P}^{-1} = \\ &= (\mathbf{u} \cdot \mathbf{S} \cdot \mathbf{G} \cdot \mathbf{P} + \mathbf{e}) \cdot \mathbf{P}^{-1} = \\ &= \mathbf{u} \cdot \mathbf{S} \cdot \mathbf{G} + \mathbf{e} \cdot \mathbf{P}^{-1}\end{aligned}$$

2. Bob decodes the secret code and obtains

$$\mathbf{u}' = \mathbf{u} \cdot \mathbf{S}$$

3. Bob computes $\mathbf{u} = \mathbf{u}' \cdot \mathbf{S}^{-1}$.

Attacks against McEliece/Niederreiter

General attacks

General attacks against McEliece/Niederreiter are those aimed at decoding the random-like public code.

Code-specific attacks

Specific attacks are those tailored to each code faimily (Goppa, GRS, convolutional, LDPC, MDPC, ...).

Attacks against McEliece/Niederreiter

Decoding attacks

Aimed at decrypting one or more ciphertexts without knowing the private key.

Key recovery attacks

Aimed at recovering the private key from the public key.

Decoding attacks

- The most dangerous decoding attacks (DAs) exploit information set decoding (ISD).
- The ISD principle was introduced by Prange in 1962.
- The first efficient algorithms were introduced by Lee-Brickell and Leon-Stern in 1988/89.
- These techniques have known great advances in recent years.

- E. Prange, "The use of information sets in decoding cyclic codes, Information Theory," IRE Transactions on, vol. 8, no. 5, pp. 5–9, 1962.
- P. Lee, E. Brickell, "An observation on the security of McElieces public-key cryptosystem," Advances in Cryptology - EUROCRYPT 88, pp 275–280, 1988.
- J. Leon, "A probabilistic algorithm for computing minimum weights of large error-correcting codes," IEEE Trans. Inform. Theory, vol. 34, no. 5, pp. 1354–1359, 1988.
- J. Stern, "A method for finding codewords of small weight," Coding Theory and Applications, vol. 388 of Springer LNCS, pp. 106-113, 1989.

Modern information set decoding

- The general decoding problem can be reduced to that of searching low weight codewords.
- Modern approaches exploit the birthday paradox to search for low weight codewords.
- Lower bounds on complexity have been found recently by Niebuhr et al.

- C. Peters, "Information-set decoding for linear codes over F_q," Post-Quantum Cryptography, vol. 6061 of Springer LNCS, pp. 81–94, 2010.
- D. J. Bernstein, T. Lange, C. Peters, "Smaller decoding exponents: ball-collision decoding," CRYPTO 2011, vol. 6841 of Springer LNCS, pp 743–760, 2011.
- A. May, A. Meurer, E. Thomae, "Decoding random linear codes in $O(2^{0.054n})$," ASIACRYPT 2011, vol. 7073 of Springer LNCS, pp. 107124, 2011.
- A. Becker, A. Joux, A. May, and A. Meurer, "Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding," Advances in Cryptology - EUROCRYPT 2012, vol. 7237 of Springer LNCS, pp. 520–536, 2012.
- R. Niebuhr, E. Persichetti, P.-L. Cayrel, S. Bulygin, J. Buchmann, "On lower bounds for information set decoding over F_q and on the effect of partial knowledge," Int. J. Inf. Coding Theory, vol. 4, no. 1, pp. 47–78, 2017.

Pre-quantum VS post-quantum decoding attacks

- Grover's algorithm is a quantum algorithm introduced for performing efficient database searches.

- D. J. Bernstein, "Grover vs. McEliece," in Post-Quantum Cryptography, vol. 6061 of Springer LNCS, pp. 73–80, 2010.
- S.H.S. de Vries, "Achieving 128-bit Security against Quantum Attacks in OpenVPN," Master Thesis, University of Twente, 2016.

Pre-quantum VS post-quantum decoding attacks

- Grover's algorithm is a quantum algorithm introduced for performing efficient database searches.
- For searching one entry of an unsorted list of n entries,
  - The best classical algorithm requires $n/2$ steps on average.
  - Grover's algorithm requires $\pi/4\sqrt{n}$ steps using $\log_2(n)$ qubits.

- D. J. Bernstein, "Grover vs. McEliece," in Post-Quantum Cryptography, vol. 6061 of Springer LNCS, pp. 73–80, 2010.
- S.H.S. de Vries, "Achieving 128-bit Security against Quantum Attacks in OpenVPN," Master Thesis, University of Twente, 2016.

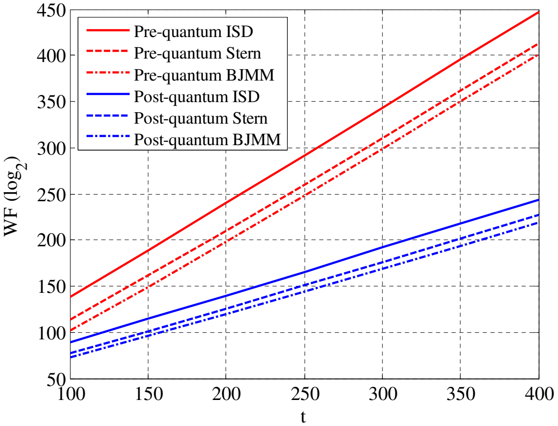
Pre-quantum VS post-quantum decoding attacks

- Grover's algorithm is a quantum algorithm introduced for performing efficient database searches.
- For searching one entry of an unsorted list of n entries,
  - The best classical algorithm requires $n/2$ steps on average.
  - Grover's algorithm requires $\pi/4\sqrt{n}$ steps using $\log_2(n)$ qubits.
- Grover's algorithm reduces the number of iterations but does not reduce the cost per iteration.
- However, it somehow impacts the work factor of ISD.

- D. J. Bernstein, "Grover vs. McEliece," in Post-Quantum Cryptography, vol. 6061 of Springer LNCS, pp. 73–80, 2010.
- S.H.S. de Vries, "Achieving 128-bit Security against Quantum Attacks in OpenVPN," Master Thesis, University of Twente, 2016.

Pre-quantum VS post-quantum decoding attacks

Pre- and post-quantum WF of some ISD algorithms versus t, for codes with $n = 12000$, $k = 6000$.



Alternatives to Goppa codes

Goppa codes
[McEliece78]

GRS codes
[Niederreiter86]

QD codes
[MisBar09]

Conv. codes
[LönJoh12]

LDPC codes
[MonRosSho00]

QC codes
[Gaborit05]

GRS subcodes
[BerLoi05]

QC-LDPC codes
[BalBodChi08]

Transformed GRS codes
[BalBiaChiRosSch16]

QC-MDPC codes
[MisTilSenBar12]

Alternatives to Goppa codes

Goppa codes
[McEliece78]

GRS codes
[Niederreiter86]

QD codes
[MisBar09]

Conv. codes
[LönJoh12]

LDPC codes
[MonRosSho00]

QC codes
[Gaborit05]

GRS subcodes
[BerLoi05]

QC-LDPC codes
[BalBodChi08]

Transformed GRS codes
[BalBiaChiRosSch16]

QC-MDPC codes
[MisTilSenBar12]

Alternatives to Goppa codes

Goppa codes
[McEliece78]

GRS codes
[Niederreiter86]

QD codes
[MisBar09]

Conv. codes
[LönJoh12]

LDPC codes
[MonRosSho00]

QC codes
[Gaborit05]

GRS subcodes
[BerLoi05]

QC-LDPC codes
[BalBodChi08]

Transformed GRS codes
[BalBiaChiRosSch16]

QC-MDPC codes
[MisTilSenBar12]

Alternatives to Goppa codes

Goppa codes
[McEliece78]

GRS codes
[Niederreiter86]

QD codes
[MisBar09]

Conv. codes
[LönJoh12]

LDPC codes
[MonRosSho00]

QC codes
[Gaborit05]

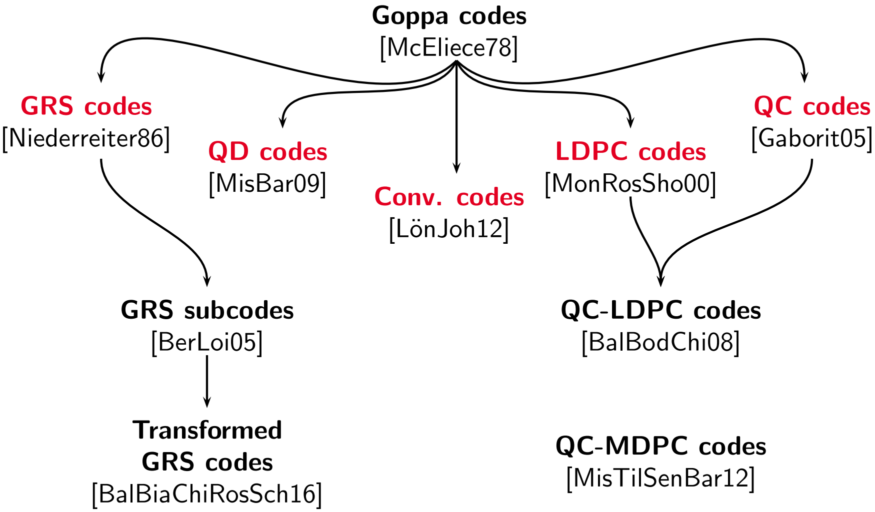
GRS subcodes
[BerLoi05]

Transformed GRS codes
[BalBiaChiRosSch16]

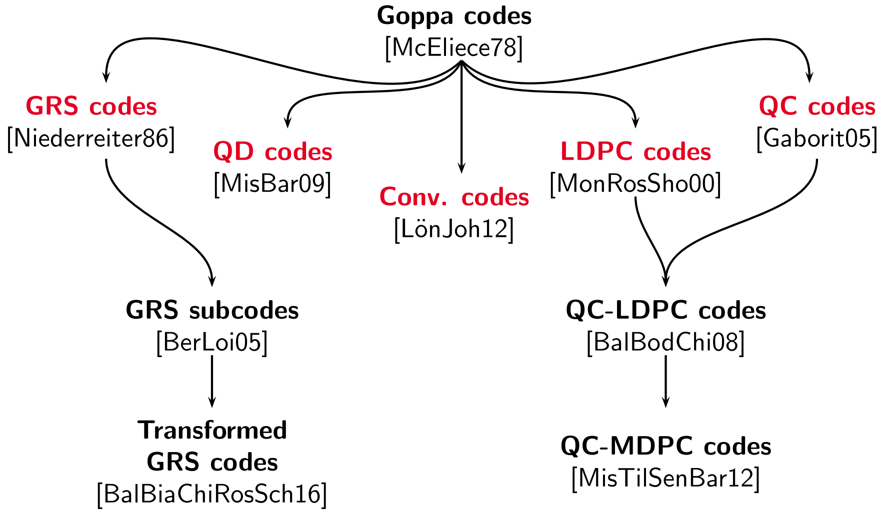
QC-LDPC codes
[BalBodChi08]

QC-MDPC codes
[MisTilSenBar12]

Alternatives to Goppa codes



Alternatives to Goppa codes



LDPC codes in the McEliece cryptosystem

- Low-density parity-check (LDPC) codes are capacity-achieving codes under belief propagation (BP) decoding.
- They allow a random-based design, which results in large key spaces.
- The low density of their matrices could be attractive to achieve compact representations.
- All this makes them interesting for the use in McEliece/Niederreiter.

- C. Monico, J. Rosenthal, and A. Shokrollahi, "Using low density parity check codes in the McEliece cryptosystem," Proc. IEEE ISIT 2000, Sorrento, Italy, Jun. 2000, p. 215.
- M. Baldi, F. Chiaraluce, "Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes," Proc. IEEE ISIT 2007, Nice, France, Jun. 2007, pp. 2591–2595.

LDPC codes in the McEliece cryptosystem

- LDPC codes are capacity-achieving codes under BP decoding.
- They allow a random-based design, which results in large key spaces.
- The low density of their matrices could be attractive to achieve compact representations.
- All this makes them interesting for the use in McEliece/Niederreiter.

Alert

Public codes cannot be LDPC codes as well, otherwise secret codes are likely to be exposed.

- C. Monico, J. Rosenthal, and A. Shokrollahi, "Using low density parity check codes in the McEliece cryptosystem," Proc. IEEE ISIT 2000, Sorrento, Italy, Jun. 2000, p. 215.
- M. Baldi, F. Chiaraluce, "Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes," Proc. IEEE ISIT 2007, Nice, France, Jun. 2007, pp. 2591–2595.
- A. Otmani, J.P. Tillich, L. Dallot, "Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes," Proc. SCC 2008, Beijing, China, Apr. 2008.

Key recovery attacks based on decoding errors

- Recently, it has been shown that QC-MDPC and QC-LDPC code-based McEliece cryptosystem may suffer from attacks exploiting decoding errors.

- Q. Guo, T. Johansson, P. Stankovski, "A Key Recovery Attack on MDPC with CCA Security Using Decoding Errors," Advances in Cryptology ASIACRYPT 2016, vol. 10031 of Springer LNCS, pp. 789–815.
- T. Fabšič, V. Hromada, P. Stankovski, P. Zajac, Q. Guo, T. Johansson, "A Reaction Attack on the QC-LDPC McEliece Cryptosystem," PQCrypto 2017, vol. 10346 of Springer LNCS, pp. 51–68.

Key recovery attacks based on decoding errors

- Recently, it has been shown that QC-MDPC and QC-LDPC code-based McEliece cryptosystem may suffer from attacks exploiting decoding errors.
- The attack is built upon two facts:
  1. The decryption failure probability is not zero and depends on the structure of the secret key.
  2. Eve can estimate such a probability by observing Bob's reactions during decryption of some special ciphertexts.

- Q. Guo, T. Johansson, P. Stankovski, "A Key Recovery Attack on MDPC with CCA Security Using Decoding Errors," Advances in Cryptology ASIACRYPT 2016, vol. 10031 of Springer LNCS, pp. 789–815.
- T. Fabšič, V. Hromada, P. Stankovski, P. Zajac, Q. Guo, T. Johansson, "A Reaction Attack on the QC-LDPC McEliece Cryptosystem," PQCrypto 2017, vol. 10346 of Springer LNCS, pp. 51–68.

Key recovery attacks based on decoding errors

- Recently, it has been shown that QC-MDPC and QC-LDPC code-based McEliece cryptosystem may suffer from attacks exploiting decoding errors.
- The attack is built upon two facts:
  1. The decryption failure probability is not zero and depends on the structure of the secret key.
  2. Eve can estimate such a probability by observing Bob's reactions during decryption of some special ciphertexts.
- This limits the life of the keys, which must be renewed often (or the systems can be used with one-time ephemeral keys).

- Q. Guo, T. Johansson, P. Stankovski, "A Key Recovery Attack on MDPC with CCA Security Using Decoding Errors," Advances in Cryptology ASIACRYPT 2016, vol. 10031 of Springer LNCS, pp. 789–815.
- T. Fabšič, V. Hromada, P. Stankovski, P. Zajac, Q. Guo, T. Johansson, "A Reaction Attack on the QC-LDPC McEliece Cryptosystem," PQCrypto 2017, vol. 10346 of Springer LNCS, pp. 51–68.

Two Proposals

- We proposed two QC-LDPC based primitives to the NIST Post-Quantum Cryptography Standardization Process
  - LEDAkem (Low dEnsity parity-check coDe-bAsed key encapsulation mechanism)
  - LEDApkc (Low-dEnsity parity-check coDe-bAsed public-key cryptosystem)
- Both proposals built employing QC-LDPC codes as the core building block
- Proposed parameters require $\geq 2^{\lambda}, \lambda \in \{128, 192, 256\}$ operations to run the best attack on a quantum computer

Choice of the QC-LDPC codes

- The chosen QC-LDPC $(n,\ k)$ codes are described by block circulant generator/parity matrices
- The parameter values n and k are s.t. $n = n_0 p, k = k_0 p$ with p prime and $k_0 = n_0 - 1$
  - We proposed parameter sets for $n_0 \in \{2, 3, 4\}$
- The values for p were chosen so that $ord_p(2) = p - 1$ to allow efficient sampling of invertible circulant blocks
  - Any odd-weight polynomial $\in \mathbb{F}_2[x]/\langle x^p + 1\rangle$ is invertible if $ord_p(2) = p - 1$

LEDAkem

- Relies on Niederreiter's variant of the McEliece cryptosystem
  - Given a random-looking parity matrix $\mathbf{H}$ and a syndrome vector $\mathbf{s}$, find an error vector $\mathbf{e}$ corresponding to it, with weight $\leq t$
  - Problem proven to be NP-complete for a random matrix $\mathbf{H}$
  - Smaller amount of information encrypted w.r.t. corresponding McEliece cryptosystem (encoded in $\mathbf{e}$) is enough for key encap
- Obtains the symmetric key employing the error vector with weight t as the input of a KDF

LEDAkem

Key Generation

1. Generate a random $r \times n$ binary block circulant matrix $\mathbf{H} = [\mathbf{H}_0, \ldots, \mathbf{H}_{n_0-1}]$ with column weight $d_v \ll n$
2. Generate a random, non-singular, $n \times n$ binary block circulant matrix $\mathbf{Q}$ with column weight $m \ll n$
3. Compute $\mathbf{L} = \mathbf{H} \times \mathbf{Q} = [\mathbf{L}_0, \ldots, \mathbf{L}_{n_0-1}]$
4. Private key: $\mathbf{H}, \mathbf{Q}$; Public Key $\mathbf{M} = (\mathbf{L}_{n_0-1})^{-1} \times \mathbf{L}$

LEDAkem

Key Encapsulation

1. Generate a random n-bit error vector $\mathbf{e}$ with weight t
2. Compute the ciphertext (syndrome) $\mathbf{s} = \mathbf{M}\mathbf{e}^T$
3. Derive the shared secret $\mathbf{x} = \mathrm{KDF}(\mathbf{e})$

Key Decapsulation

1. Obtain $\mathbf{e}$ as $\textsc{Decode}(\mathbf{s}, \mathbf{H}, \mathbf{Q})$
2. Derive the shared secret $\mathbf{x} = \mathrm{KDF}(\mathbf{e})$

LEDApkc

- Built on the original McEliece cryptosystem
- The McEliece scheme does not provide semantic security if a systematic generator matrix $\mathbf{G}$ is employed
- We employ the conversion proposed by Kobara and Imai to achieve IND-CCA2 guarantees, and employ a systematic $\mathbf{G}$
  - Reduces the size of the keypair
  - Speeds up the encryption process overall (the conversion is less expensive than large polynomial multiplications in sw)
- Encryption and decryption reuse primitives from KEM
  - Smaller binary code size/silicon area in implementations

Security Evaluation

- Current best passive attacks have exponential complexity on a quantum computer
    - Parameters designed to withstand Stern's ISD implemented on a quantum computer
    - Classical security margins are $\approx 2\times$ w.r.t. quantum ones
    - Estimates computed with exact formulas (not asymptotic bounds)
- Parameters designed for a DFR in the 10^{-9} range or lower
    - Experimentally validated via Monte Carlo simulations
    - Reaction attacks are expected to take 2yr+ of continuous decryption queries under extremely favourable conditions

Proposed parameters

λ	n_0	p	d_v	m	t	DFR
128	2	27,779	17	7	224	$\approx 8.3 \cdot 10^{-9}$
	3	18,701	19	7	141	$\lesssim 10^{-9}$
	4	17,027	21	7	112	$\lesssim 10^{-9}$
2–3	2	57,557	17	11	349	$\lesssim 8 \cdot 10^{-8}$
	3	41,507	19	11	220	$\lesssim 8 \cdot 10^{-8}$
	4	35,027	17	13	175	$\lesssim 8 \cdot 10^{-8}$
4–5	2	99,053	19	13	474	$\lesssim 10^{-8}$
	3	72,019	19	15	301	$\lesssim 10^{-8}$
	4	60,509	23	13	239	$\lesssim 10^{-8}$

Efficient Implementation Strategies

Circulant matrix representation/arithmetics

- Represent circulant blocks as elements of $\mathbb{F}_2[x]/\langle x^p + 1\rangle$
  - Reduces both time and space complexity for arithmetics

Remove invertibility check for Q

- $\mathrm{Perm}(\mathbf{Q})$ is odd and $< p \Rightarrow \mathbf{Q}$ is invertible

Efficient decoding

- Specialized bit-flipping decoder taking into account $\mathbf{Q}$

Efficient Implementation results

- A reference implementation in ISO-C99 without platform specific optimization achieves running times in the tens of ms range
  - NIST ref. platform: Base Intel x86-64 ISA (early 2006 CPUs)
- Further optimizations:
  - Sub-quadratic poly multiplication arithmetics
  - Use x86-64 ISA extensions (e.g. CLMUL) and vector units
  - Devise a dedicated HW implementation

Thanks for the attention

Questions?

`https://www.ledacrypt.org`