RQC, an IND-CCA2 PKE based on Rank Metric

Carlos Aguilar Melchor[2], Nicolas Aragon[1], Slim Bettaieb[5], Loïc Bidoux[5], Olivier Blazy[1], Alain Couvreur[6,7], Jean-Christophe Deneuville[1,4], Philippe Gaborit[1], Adrien Hauteville[1,7], Gilles Zémor[3]

[1] XLIM-DMI, University of Limoges [2] ISAE-SUPAERO, University of Toulouse, [3] IMB, University of Bordeaux, [4] ENAC, University of Toulouse, [5] Worldline, [6] INRIA, [7] LIX, École polytechnique

https://pqc-rqc.org

2019-08-24

Agenda

Agenda

Rank weight and support

Let $\beta_1, \ldots, \beta_m$ be a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. To each vector $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$, one can associate a matrix $\boldsymbol{M_x}$

$$\mathbf{x} = (x_0, \ldots, x_{n-1}) \in \mathbb{F}_{q^m}^n \leftrightarrow \boldsymbol{M_x} = \begin{pmatrix} x_{0,0} & \cdots & x_{0,n-1} \\ \vdots & \ddots & \ldots \\ x_{m-1,0} & \cdots & x_{m-1,n-1} \end{pmatrix} \in \mathbb{F}_q^{m \times n}$$

Definition (Rank weight)

Let $\mathbf{x} \in \mathbb{F}_{q^m}^n$, $|\mathbf{x}|_r = \text{Rank}(\boldsymbol{M_x})$ where $\boldsymbol{M_x} = (x_{i,j})$ with $x_j = \sum_{i=0}^{m-1} x_{i,j}\beta_i$.

Rank weight and support

Let $\beta_1, \ldots, \beta_m$ be a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. To each vector $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$, one can associate a matrix $\boldsymbol{M_x}$

$$\mathbf{x} = (x_0, \ldots, x_{n-1}) \in \mathbb{F}_{q^m}^n \leftrightarrow \boldsymbol{M_x} = \begin{pmatrix} x_{0,0} & \cdots & x_{0,n-1} \\ \vdots & \ddots & \ldots \\ x_{m-1,0} & \cdots & x_{m-1,n-1} \end{pmatrix} \in \mathbb{F}_q^{m \times n}$$

Definition (Rank weight)

Let $\mathbf{x} \in \mathbb{F}_{q^m}^n$, $|\mathbf{x}|_r = \text{Rank}(\boldsymbol{M_x})$ where $\boldsymbol{M_x} = (x_{i,j})$ with $x_j = \sum_{i=0}^{m-1} x_{i,j}\beta_i$.

Definition (Support)

The support of a word is the $\mathbb{F}_q$-subspace generated by its coordinates:

$$\text{Supp}(\boldsymbol{x}) = \langle x_0, \ldots, x_{n-1} \rangle_{\mathbb{F}_q}$$

Difficult problems in rank metric

Problem (Rank Syndrome Decoding problem)

Given $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$, $\boldsymbol{s} \in \mathbb{F}_{q^m}^{n-k}$, *an integer* r, *find* $\boldsymbol{e} \in \mathbb{F}_{q^m}^{n}$ *such that:*

- $\boldsymbol{He}^T = \boldsymbol{s}^T$
- $|\boldsymbol{e}|_r = r$

Difficult problems in rank metric

Problem (Rank Syndrome Decoding problem)

Given $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$, $\boldsymbol{s} \in \mathbb{F}_{q^m}^{n-k}$, *an integer* r, *find* $\boldsymbol{e} \in \mathbb{F}_{q^m}^{n}$ *such that:*

- $\boldsymbol{He}^T = \boldsymbol{s}^T$
- $|\boldsymbol{e}|_r = r$

Probabilistic reduction to the NP-Complete SD problem [GZ16]

Agenda

RQC description

Small weight vectors: $\mathcal{S}_w^n(\mathbb{F}_{q^m}) = \{ \mathbf{x} \in \mathbb{F}_{q^m}^n \text{ such that } |\mathbf{x}|_r = w \}$

RQC description

Small weight vectors: $\mathcal{S}_w^n(\mathbb{F}_{q^m}) = \{ \mathbf{x} \in \mathbb{F}_{q^m}^n \text{ such that } |\mathbf{x}|_r = w \}$

Public data: $\mathbf{G}$ is a generator matrix of some public code $\mathcal{C}$

Secret key: $\mathbf{sk} = (\mathbf{x}, \mathbf{y})$, Public key: $\mathbf{pk} = (\mathbf{h}, \mathbf{s})$, Ciphertext: $\mathbf{ct} = (\mathbf{u}, \mathbf{v})$

Alice		Bob
$\text{seed}_{\mathbf{h}} \xleftarrow{\$} \{0,1\}^{\lambda}$, $\mathbf{h} \xleftarrow{\text{seed}_{\mathbf{h}}} \mathbb{F}_{q^m}^n$, $(\mathbf{x}, \mathbf{y}) \xleftarrow{\$} \mathcal{S}_{\omega}^{2n}(\mathbb{F}_{q^m})$, $\mathbf{s} \leftarrow \mathbf{x} + \mathbf{h} \cdot \mathbf{y}$	$\xrightarrow{\text{seed}_{\mathbf{h}}, \mathbf{s}}$	
	$\xleftarrow{\mathbf{u}, \mathbf{v}}$	$(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3) \xleftarrow{\$} \mathcal{S}_{\omega_r}^{3n}(\mathbb{F}_{q^m})$, $\mathbf{u} \leftarrow \mathbf{r}_1 + \mathbf{h} \cdot \mathbf{r}_2$, $\mathbf{v} \leftarrow \mathbf{m}\mathbf{G} + \mathbf{s} \cdot \mathbf{r}_2 + \mathbf{r}_3$
$\mathbf{m} \leftarrow \mathcal{C}.\text{Decode}\left(\mathbf{v} - \mathbf{u} \cdot \mathbf{y}\right)$		

Figure: Informal description of RQC.PKE

RQC description

Correctness

$$\mathbf{v} - \mathbf{u} \cdot \mathbf{y} = \mathbf{m}\mathbf{G} + (\mathbf{x} + \mathbf{h} \cdot \mathbf{y}) \cdot \mathbf{r}_2 + \mathbf{r}_3 - (\mathbf{r}_1 + \mathbf{h} \cdot \mathbf{r}_2) \cdot \mathbf{y}$$

$$= \mathbf{m}\mathbf{G} + \mathbf{x} \cdot \mathbf{r}_2 - \mathbf{y} \cdot \mathbf{r}_1 + \mathbf{r}_3$$

RQC description

Correctness

$$\mathbf{v} - \mathbf{u} \cdot \mathbf{y} = \mathbf{mG} + (\mathbf{x} + \mathbf{h} \cdot \mathbf{y}) \cdot \mathbf{r}_2 + \mathbf{r}_3 - (\mathbf{r}_1 + \mathbf{h} \cdot \mathbf{r}_2) \cdot \mathbf{y}$$
$$= \mathbf{mG} + \mathbf{x} \cdot \mathbf{r}_2 - \mathbf{y} \cdot \mathbf{r}_1 + \mathbf{r}_3$$

Decrypts whenever the public code $\mathcal{C}$ decodes the small weight error $\mathbf{x} \cdot \mathbf{r}_2 - \mathbf{y} \cdot \mathbf{r}_1 + \mathbf{r}_3$ for $(\mathbf{x}, \mathbf{y})$ and $(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3)$ small rank weight vectors

RQC description

Correctness

$$\mathbf{v} - \mathbf{u} \cdot \mathbf{y} = \mathbf{mG} + (\mathbf{x} + \mathbf{h} \cdot \mathbf{y}) \cdot \mathbf{r}_2 + \mathbf{r}_3 - (\mathbf{r}_1 + \mathbf{h} \cdot \mathbf{r}_2) \cdot \mathbf{y}$$
$$= \mathbf{mG} + \mathbf{x} \cdot \mathbf{r}_2 - \mathbf{y} \cdot \mathbf{r}_1 + \mathbf{r}_3$$

Decrypts whenever the public code $\mathcal{C}$ decodes the small weight error $\mathbf{x} \cdot \mathbf{r}_2 - \mathbf{y} \cdot \mathbf{r}_1 + \mathbf{r}_3$ for $(\mathbf{x}, \mathbf{y})$ and $(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3)$ small rank weight vectors

- Choice for $\mathcal{C}$: **Gabidulin codes** hence no decryption failure

Semantic security

Theorem

Under the assumption of the hardness of the 2-DIRSD and 3-DIRSD problems, RQC is IND-CPA in the Random Oracle Model.

Semantic security

Theorem

Under the assumption of the hardness of the 2-DIRSD and 3-DIRSD problems, RQC is IND-CPA in the Random Oracle Model.

- IND-CPA RQC PKE $\rightarrow$ IND-CCA2 RQC KEM using [HHK17]
- IND-CCA RQC KEM $\rightarrow$ IND-CCA2 RQC PKE using Hybrid Encryption

Parameters

	Public Key	Secret Key	Ciphertext	Shared Secret	DFR
RQC 128	853	40	1,690	64	0
RQC 192	1,391	40	2,766	64	0
RQC 256	2,284	40	4,552	64	0

Figure: RQC sizes expressed in bytes

Agenda

NIST comments on RQC

From NIST report [AAA+19]

- Rank metric adds significant diversity to the standardization process
- RQC has the most conservative approach to IND-CCA2 security in rank metric (no DFR, no code indistinguishability assumption)

NIST comments on RQC

From NIST report [AAA+19]

- ◇ Rank metric adds significant diversity to the standardization process
- ◇ RQC has the most conservative approach to IND-CCA2 security in rank metric (no DFR, no code indistinguishability assumption)

- ◇ Additional analysis on algebraic attacks is required
- ◇ RQC suffers in decryption speed

Security-related changes

- **Improved analysis on algebraic attacks using Groebner basis**

Security-related changes

- **Improved analysis on algebraic attacks using Groebner basis**
- RQC relies on ideal codes (generalization from quasi-cyclic codes)

Security-related changes

- **Improved analysis on algebraic attacks using Groebner basis**
- RQC relies on ideal codes (generalization from quasi-cyclic codes)
- Parameters updated so that error weight increases regularly with each level of security (small increase in parameter size)

Reference implementation

- New reference implementation available (2019/08/24)
- **No longer depends on external librairies**

Reference implementation

- New reference implementation available (2019/08/24)
- **No longer depends on external librairies**

- Outperforms MPFQ-based and NTL-based implementations
- Rank-Based Cryptography Library (rbc-lib.org) will be released to promote community development on rank based cryptography

Reference implementation

- New reference implementation available (2019/08/24)
- **No longer depends on external librairies**

- Outperforms MPFQ-based and NTL-based implementations
- Rank-Based Cryptography Library (rbc-lib.org) will be released to promote community development on rank based cryptography

- Recently submitted to SUPERCOP (reference and optimized)

Optimized implementation

- New AVX2 implementation available (2019/08/24)
- **Significant improvement on decapsulation time**

Optimized implementation

- New AVX2 implementation available (2019/08/24)
- **Significant improvement on decapsulation time**

	AVX2 Implementation			Improvement vs 2019/04/10		
	Keygen	Encaps	Decaps	Keygen	Encaps	Decaps
RQC 128	0.20	0.28	1.02	x3.5	x4.6	**x6.5**
RQC 192	0.38	0.55	2.22	x3.0	x4.0	**x6.6**
RQC 256	0.62	0.89	3.74	x2.9	x4.0	**x6.2**

Figure: Performances in millions of CPU cycles and comparison to reference implementation from 2019/04/10 package using an i7-7820 @3.6GHz CPU

Towards constant time implementation

- New Gabidulin code's decoding algorithm without branching related to the weigth of the error to be decoded [BBGM19]
- **Small performance overhead** (less than 10%)

Towards constant time implementation

- New Gabidulin code's decoding algorithm without branching related to the weigth of the error to be decoded [BBGM19]
- **Small performance overhead** (less than 10%)

- Implementation of the new algorithm available. Additional effort required to get a constant-time implementation (ongoing work)

Conclusion

Take away

- RQC is a code-based **IND-CCA2 PKE** using the **rank metric**
- Rank metric **adds diversity** to the standardization process

Conclusion

Take away

- ◇ RQC is a code-based **IND-CCA2 PKE** using the **rank metric**
- ◇ Rank metric **adds diversity** to the standardization process

- ◇ RQC features **attractive key sizes** w.r.t. to code-based schemes
- ◇ RQC features a **conservative approach**
  - **No decryption failure**
  - No code indistinguishability assumption

Conclusion

Take away

- RQC is a code-based **IND-CCA2 PKE** using the **rank metric**
- Rank metric **adds diversity** to the standardization process

- RQC features **attractive key sizes** w.r.t. to code-based schemes
- RQC features a **conservative approach**
  - **No decryption failure**
  - No code indistinguishability assumption

- RQC features **good performances** w.r.t. to code-based schemes
  - Constant time achievable with small overhead

Conclusion

[AAA+19] Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, Ray Perlner, Angela Robinson, and Daniel Smith-Tone. *Status report on the first round of the NIST post-quantum cryptography standardization process*. NIST, 2019.

[AAB+17] Carlos Aguilar-Melchor, Nicolas Aragon, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Alain Couvreur, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, and Gilles Zémor. Rank Quasi-Cyclic (RQC). 2017.

[BBGM19] Slim Bettaieb, Loïc Bidoux, Philippe Gaborit, and Etienne Marcatel. Preventing timing attacks against RQC using constant time decoding of Gabidulin codes. In *International Conference on Post-Quantum Cryptography*, pages 371–386. Springer, 2019.

[GZ16] Philippe Gaborit and Gilles Zémor. On the hardness of the decoding and the minimum distance problems for rank codes. *IEEE Transactions on Information Theory*, 62(12):7245–7252, 2016.

[HHK17] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In *Theory of Cryptography Conference*, pages 341–371. Springer, 2017.

Thank you for your attention. Questions ?