On Code-Based Public-Key Cryptography: Introduction to McNie

Jon-Lark Kim
Sogang University, S. Korea

The 5th Sino-Korea Int. conf. COding and Topics(SKICOT)
Shanghai, China, July 2-6, 2018

Outline

1. Introduction to Post-Quantum Cryptography
2. NIST Post-quantum cryptography competition
3. McNie: a new code-based cryptography
4. General algorithm specification
   - Key generation
   - Encryption
   - Decryption
5. Rank metric codes
   - Definition
   - Using 3-QC-LRPC codes
   - Using 4-QC-LRPC codes
6. Suggested parameters
7. Connection between Ouroboros-R and McNie

What is Post-Quantum Cryptography?

- According to Matteo Mariantoni(PQCrypto 2014), a quantum computer will be built in 15 years and can break RSA-2048.
- Post-quantum cryptography(PQC) is also known as “quantum-proof”, “quantum-safe” or “quantum-resistant”
- PQC meams cryptographic algorithms (usually public-key algorithms) that are thought to be secure against an attack by a classical computer *and* a quantum computer.

Question

Are RSA and ECDSA safer under quantum computers?

RSA and Elliptic Curve Cryptography are not safe.

- Peter Shore(1994) showed that there are polynomial time quantum algorithms to integer factorization problem and discrete log problem.
- This implies that RSA will be broken since it is based on the hardness of integer factorization.
- Elliptic curve cryptography(ECDSA), Digital Signature Algorithm(DSA) and Diffie-Hellman key exchange will be broken since they are based on the hardness of discrete log problem.
- Symmetric key cryptosystems such as AES, Triple DES needs larger keys.
- Hash functions such as SHA-1, SHA-2, and SHA-3 needs longer outputs (say double the outputs).

NIST competition

- NIST called for quantum-resistant cryptographic algorithms for new public-key crypto standards.
  - Digital signatures
  - Encryption/key-establishment
- The time line as follows.
  - Nov 30, 2017 Deadline for submissions
  - 3-5 years: Analysis phase
  - 2 years later - Draft standards ready
  - Workshop April 2018: submitter's presentations
  - One or two more workshops during the analysis phase

Submitted algorithms for NIST competition

	Signatures	KEM/Encryption	Overall
Lattice-based	5	21	26
Code-based	2	17	19
Multi-variate	7	2	9
Symmetric/Hash-based	3		3
Other	2	5	7
Total	**19**	**45**	**64**

- 82 submissions were received and 64 passed the first round.
- There are 5 submissions from Korea. They are McNie(Sogang U.), Lizard(Seoul Nat'l U.), pqsigRM(Seoul Nat'l U.), HiMQ-3(NIMS), EMBLEM and R.EMBLEM(Korea U.).
- There are 3 submissions from China. They are LAC(Chinese Academy of Science), Key Consensus from Lattice(Fudan University), Lepton(Shanghai Jiao Tong University).

25 Countries and 6 Continents

McEliece: the first code-based cryptography

- The McEliece cryptosystem and its variants are well known code-based public key cryptosystems:

$$\mathbf{c} = \mathbf{m}G + \mathbf{e}$$

$$\text{public key} G = AG'P,$$

  where $\mathbf{m}$ is a message, $\mathbf{c}$ is a ciphertext, G' is a secret generator matrix for a code which can correct errors $\mathbf{e}$, A is a secret invertible matrix, P is a secret permutation matrix.
- However, McEliece cryptosystems with many algebraic codes with good structures have been broken due to their structures except for Goppa codes.

McNie: a new code-based cryptography

- Our McNie is a new code-based public key cryptosystem which is less vulnerable against currently known structural attacks.
- There are five authors for McNie: Jon-Lark Kim, Lucky Galvez, Myung Jae Kim, Young Sik Kim, Nari Lee.
- McNie is one of the 64 algorithms which passed round 1 of 2017 NIST Competition for Post-Quantum Cryptography.
- McNie can use Hamming weight or rank weight in general.

McNie- Key generation

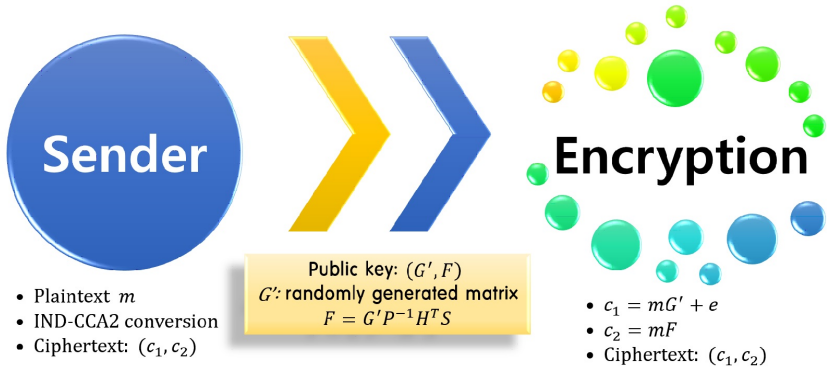
- Consider Hamming weight or rank weight.
- **Secret key:** (H, P, S, Φ_H)
  H: a parity check matrix for an $[n, k]$ code C over $\mathbb{F}_{q^m}$
  P: an $n \times n$ permutation matrix
  S: an $(n-k) \times (n-k)$ invertible matrix over $\mathbb{F}_{q^m}$
  Φ_H: an efficient decoding algorithm for C which corrects errors of weight up to r

- **Public key:** (G', F)
  G': Generator matrix for a random $[n, l]$ code over $\mathbb{F}_{q^m}$
  $F = G' P^{-1} H^T S$

McNie- Encryption

Message: $\mathbf{m} \in \mathbb{F}_{q^m}^{l}$

- Randomly generate $\mathbf{e} \in \mathbb{F}_{q^m}^{n}$ of weight r
- $Enc(\mathbf{m}) = (\mathbf{c_1}, \mathbf{c_2})$
  $\mathbf{c_1} = \mathbf{m}G' + \mathbf{e}$
  $\mathbf{c_2} = \mathbf{m}F = \mathbf{m}G'P^{-1}H^TS$



McNie- Decryption

Received vector: $\mathbf{c} = (\mathbf{c_1}, \mathbf{c_2})$

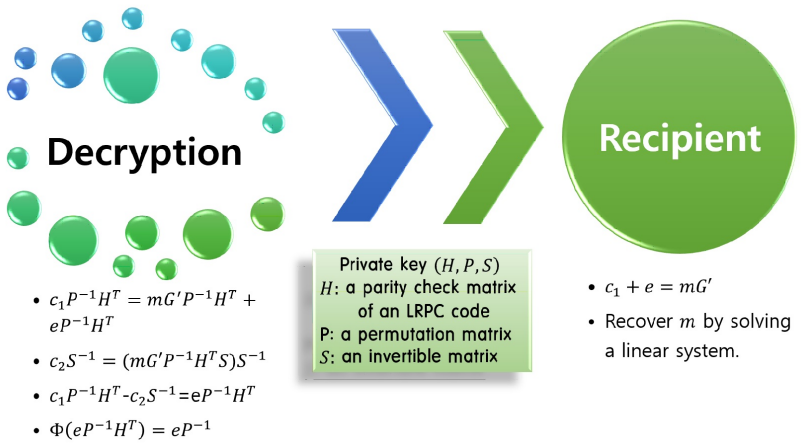
- Compute

$$\begin{aligned} \mathbf{s}' &= \mathbf{c_1} P^{-1} H^T - \mathbf{c_2} S^{-1} \\ &= (\mathbf{m} G' + \mathbf{e}) P^{-1} H^T \\ &\quad -(\mathbf{m} G' P^{-1} H^T S) S^{-1} \\ &= \mathbf{e} P^{-1} H^T \\ \mathbf{e}' &= \Phi_H(\mathbf{s}') = \mathbf{e} P^{-1} \\ \mathbf{e} &= \mathbf{e}' P \end{aligned}$$

- Solve the system

$$\mathbf{m} G' = \mathbf{c_1} - \mathbf{e}$$

to recover $\mathbf{m}$.



Security reduction

The McNie cryptosystem is based on the following hard problem:

(Rank) Syndrome Decoding Problem

Given an $n-k \times n$ matrix H, a vector $\mathbf{s}$ of length k and a positive integer r, find a vector $\mathbf{e}$ of (rank) weight r such that $\mathbf{s} = \mathbf{e}H^T$.

- $\mathbf{c_1} = \mathbf{m}G' + \mathbf{e}$ is an instance of the SD problem with parameters (n, l, r).
- $\mathbf{c_2} = \mathbf{m}F$ is an instance of the SD problem with parameters $(l, l-(n-k), r)$

$$\begin{aligned}(\mathbf{c_1}, \mathbf{c_2}) &= (\mathbf{m}G' + \mathbf{e}, \mathbf{m}F) \\ &= \mathbf{m}[G'|F] + (\mathbf{e}|\mathbf{0}) \\ &= \mathbf{m}G_1 + \mathbf{e_1}\end{aligned}$$

where $G_1 = [G'|F]$ and $\mathbf{e_1} = (\mathbf{e}|\mathbf{0})$. This is an instance of the SD problem with parameters $(2n-k, l, r)$. Since G' and F are public keys, the parity check matrix for the code generated by G_1 can be computed.

- This reduces to the (rank) syndrome decoding problem.

Apply McNie to rank metric codes

Let $\{\alpha_1, \alpha_2, \ldots, \alpha_m\}$ be a basis for $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$.

$$c = (c_1, \ldots, c_n) \in \mathbb{F}_{q^m}^n \Leftrightarrow \bar{c} = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{bmatrix}, \quad c_j = \sum_{i=1}^{m} c_{ij}\alpha_i$$

- **rank weight**: $w_R(c) = Rank(\bar{c})$
- **rank distance**: $d_R(c, c') = Rank(\bar{c} - \bar{c}')$

A *rank metric code* is an $[n, k]$ code over $\mathbb{F}_{q^m}$ equipped with the rank metric.

A family of rank metric codes used in McNie:

A *Low Rank Parity Check (LRPC)* code of rank d is an $[n, k]$ code over $\mathbb{F}_{q^m}$ that has for its parity check matrix an $(n-k) \times n$ matrix $H = (h_{ij})$ such that the sub-vector space of $\mathbb{F}_{q^m}$ generated by its coefficients h_{ij} has dimension at most d.

Using 3-quasi-cyclic LRPC codes

We use circulant matrices and construct quasi-cyclic LRPC codes over $\mathbb{F}_{q^m}$ in order to reduce key size.

Let n be a multiple of 3 and block size $blk = \frac{n}{3}$.

- Generate $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3 \in \mathbb{F}_{q^m}^{blk}$ s.t. dim $Supp(\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3) = d$
- Generate $\mathbf{g}_1, \mathbf{g}_2 \in \mathbb{F}_{q^m}^{blk}$.
- Let H_i, G_j be circulant matrices whose first row are $\mathbf{h}_i$ and $\mathbf{g}_j$, resp.
- Let $H = \begin{bmatrix} H_1 & H_2 & H_3 \end{bmatrix}$, $G' = \begin{bmatrix} I_{blk} & 0 & G_1 \\ 0 & I_{blk} & G_2 \end{bmatrix}$
- Take $P = I_n$ (to reduce a private key size) and $S = (H_1^T + G_1 H_3^T)^{-1}$ which is also a circulant matrix.
- $F = G' P^{-1} H^T S$ has the following form :

$$F = \begin{bmatrix} I_{\frac{n}{3}} \\ F' \end{bmatrix},$$

where $F' = (H_2^T + G_2 H_3^T)(H_1 + H_3 G_1^T)^{-1}$.

Using 4-quasi-cyclic LRPC codes

Let n be divisible by 4 and block size $blk = \frac{n}{4}$.

- Generate $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_8 \in \mathbb{F}_{q^m}^{blk}$ s.t. $\dim Supp(\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_8) = d$.
- Generate vectors $\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3 \in \mathbb{F}_{q^m}^{blk}$.
- Let $H = \begin{bmatrix} H_1 & H_2 & H_3 & H_4 \\ H_5 & H_6 & H_7 & H_8 \end{bmatrix}$, $G' = \begin{bmatrix} I_{blk} & 0 & 0 & G_1 \\ 0 & I_{blk} & 0 & G_2 \\ 0 & 0 & I_{blk} & G_3 \end{bmatrix}$
- Take $P = I_n$ and $\bar{S} = \begin{bmatrix} S_1 & S_2 \\ S_3 & S_4 \end{bmatrix}$, where S_1, S_2, S_3, S_4 are $blk \times blk$ circulant matrices.
- $\bar{F} = G' P^{-1} H^T S = \begin{bmatrix} F_1 & F_2 \\ F_3 & F_4 \\ F_5 & F_6 \end{bmatrix}$
- Reduce $\bar{F}$ in column echelon form $F = \bar{F}E = \begin{bmatrix} I_{blk} & 0 \\ 0 & I_{blk} \\ F' & F'' \end{bmatrix}$, where

$$E = \begin{bmatrix} E_1 & E_2 \\ E_3 & E_4 \end{bmatrix} = \begin{bmatrix} (F_2^{-1}F_1 - F_4^{-1}F_3)^{-1}F_2^{-1} & (F_4^{-1}F_3 - F_2^{-1}F_1)^{-1}F_4^{-1} \\ -F_4^{-1}F_3E_1 & -F_2^{-1}F_1E_2 \end{bmatrix}.$$

Suggested parameters

Parameter	n	k	l	blk	d	r	m	q	Category
`encrypt/3Q_128_1`	93	62	62	31	3	5	37	2	1
`encrypt/3Q_128_2`	105	70	70	35	3	5	37	2	1
`encrypt/3Q_192_1`	111	74	74	37	3	7	41	2	3
`encrypt/3Q_192_2`	123	82	82	41	3	7	41	2	3
`encrypt/3Q_256_1`	111	74	74	37	3	7	59	2	5
`encrypt/3Q_256_2`	141	94	94	47	3	9	47	2	5

Table: Suggested parameters using 3-quasi-cyclic LRPC codes

Parameter	n	k	l	blk	d	r	m	q	Category
`encrypt/4Q_128_1`	60	30	45	15	3	5	37	2	1
`encrypt/4Q_128_2`	72	36	54	18	3	5	37	2	1
`encrypt/4Q_192_1`	76	38	57	19	3	7	41	2	3
`encrypt/4Q_192_2`	84	42	63	21	3	7	41	2	3
`encrypt/4Q_256_1`	76	38	57	19	3	7	53	2	5
`encrypt/4Q_256_2`	88	44	66	22	3	8	47	2	5

Table: Suggested parameters using 4-quasi-cyclic LRPC codes

Key sizes for suggested parameters

Parameter	Decryption		Public Key Size (bytes)	Private Key Size (bytes)	Message Size (bytes)	Ciphertext Size (bytes)
	failure 1	failure 2				
`encrypt/3Q_128_1`	-17	-34	431	194	314	579
`encrypt/3Q_128_2`	-20	-34	486	218	358	653
`encrypt/3Q_192_1`	-17	-26	569	247	454	764
`encrypt/3Q_192_2`	-20	-26	631	274	505	846
`encrypt/3Q_256_1`	-17	-62	819	337	636	1097
`encrypt/3Q_256_2`	-20	-22	829	348	699	1110

Table: Key sizes for the suggested parameters for McNie using 3-QC-LRPC codes

Parameter	Decryption		Public Key Size (bytes)	Private Key Size (bytes)	Message Size (bytes)	Ciphertext Size (bytes)
	failure 1	failure 2				
`encrypt/4Q_128_1`	-16	-34	347	340	215	422
`encrypt/4Q_128_2`	-21	-34	417	401	264	505
`encrypt/4Q_192_1`	-18	-26	487	465	336	590
`encrypt/4Q_192_2`	-21	-26	539	512	373	651
`encrypt/4Q_256_1`	-18	-50	630	584	432	761
`encrypt/4Q_256_2`	-20	-30	647	601	461	781

Table: Key sizes for the suggested parameters for McNie using 4-QC-LRPC codes

McNie vs other cryptosystems

Security Level	McNie		DC-LRPC [3]	DC-MDPC [6]	QD-Goppa [7]	Goppa [2]
	3-quasi	4-quasi				
128	**3441**	**2775**	2809	9857	32768	1537536
192	**4551**	**3895**	-	-	45056	4185415
256	**6549**	**5035**	-	32771	65536	7667855

Table: Key-size (bits) comparison with other code-based cryptosystems

Security Level	McNie		NTRU	RSA	ECC	ECC AWC
	3-quasi	4-quasi				
128	**3441**	**2775**	4939	3072	256	277280
192	**4551**	**3895**	6523	7680	384	936618
256	**6549**	**5035**	8173	15360	512	1595434

Table: Comparison of key sizes (bits)

Recent attack on McNie based on 3,4-QC LRPC codes by P. Gaborit

- Let $\mathbf{m} = (m_1, m_2, \ldots, m_l)$
- From $\mathbf{c_2} = \mathbf{m}F$, we obtain $n-k$ linear relations of the m_i's. Hence, all the m_i's can be expressed in terms of some fixed $l-(n-k)$ coordinates.
- We can rewrite c_1 as

$$c_1 = \mathbf{m}'G'' + \mathbf{e}$$

  where G'' is of dimension $l-(n-k)$.
- So one can attack a code of dimension $l-(n-k)$ instead of a code of dimension l which reduces the security level.

Improvement on generic attacks on RSD(Rank Syndrome Decoding) by Aragon, Gaborit, Hauteville, Tillich [1]

The attack is an adaptation of the ISD attack to RSD.

This improvement uses the $\mathbb{F}_{q^m}$-linearity of the code.

The main idea is to consider the code $C' = C + \mathbb{F}_{q^m}\mathbf{e}$. The problem is then reduced to finding a weight r codeword in C'.

Instead of looking for the support E of the error $\mathbf{e}$, we can look for a multiple αE, $\alpha \in \mathbb{F}_{q^m}^*$, of the support. This attack has complexity

$$\mathcal{O}(n-k)^3 m^3 q^{r\frac{(k+1)m}{n}-m}.$$

Updated parameters for 3,4-QC LRPC codes

n	l	k	d	r	m	q	failure	Key Size (bytes)	security
120	80	80	3	8	53	2	-23	795	128
138	92	92	3	10	67	2	-25	1156	192
156	104	104	3	12	71	2	-27	1385	256

Table: New suggested parameters for McNie using 3-quasi-cyclic LRPC code

n	l	k	d	r	m	q	failure	Key Size (bytes)	security
92	46	69	3	10	59	2	-36	849	128
112	56	84	3	13	67	2	-38	1173	192
128	64	96	3	16	73	2	-36	1460	256

Table: New suggested parameters for McNie using 4-quasi-cyclic LRPC code

Second attack on McNie

After the presentation at NIST conference on PQC, another attack on McNie was reported.

- Recently Terry Shue Chien Lau and Chik How Tan [5] gave an attack on McNie based on 3 or 4 quasi cyclic LRPC codes to reduce the security level.

Revised Parameters in [11]								
3-Quasi-Cyclic LRPC								
n	k	l	d	r	m	q	Claimed Security	KRA Complexity
120	80	80	3	8	53	2	128	87
138	92	92	3	10	67	2	192	103
156	104	104	3	12	71	2	256	107
4-Quasi-Cyclic LRPC								
n	k	l	d	r	m	q	Claimed Security	KRA Complexity
92	46	69	3	10	59	2	128	107
112	56	84	3	13	67	2	192	119
128	64	96	3	16	73	2	256	127

- In our algorithm we assumed that $P = I_n$. Hence if the permutation matrix $P \neq I_n$, this attack can be avoided.

McNie based on Gabidulin codes

- Terry Shue Chien Lau and Chik How Tan [5] also suggested McNie based on Gabidulin codes, which public key sizes are about 1.8 KB at the 128 security level with no error failure probability.

m	n	k	l	t_1	t_2	q	Sec.	PK	SK	CT
43	38	14	37	9	3	2	128	1.88KB	0.98KB	0.33KB
44	40	14	38	10	3	2	129	1.94KB	1.07KB	0.36KB
50	45	19	44	10	3	2	192	3.21KB	1.36KB	0.44KB
52	47	19	45	11	3	2	198	3.40KB	1.48KB	0.49KB
57	52	20	51	13	3	2	257	4.70KB	1.80KB	0.60KB
59	54	22	51	13	3	2	257	4.88KB	1.94KB	0.63KB

Security of the GPT cryptosystems

Table III. Comparison of the three public-key cryptosystems.

PKC	Parameters	Public Key size (bits)	Complexity	Security
McEliece	Binary, $n = 1024$, $K = 534$, $t = 50$.	2^{19}	$> 2^{64}$	Insecure Bernstein Attack
McEliece Modified 1	Binary, $n = 1632$, $K = 1269$, $t = 33$.	2^{19}	$>2^{64}$	Secure
McEliece Modified 2	Binary, $n = 2960$, $K = 2288$, $t = 56$.	2^{21}	$>2^{64}$	Secure
McEliece Modified 3	Binary, $n = 6624$, $K = 5129$, $t = 115$.	2^{23}	$>2^{64}$	Secure
Niederreiter	q-array, $n = 128$, $d = 64$, $r = 63$.	32,000	$>2^{21}$	Insecure
Niederreiter Modified 1	q-array, $n = 128$, $d = 64$, $r = 63$.	32,000	$>2^{75}$	Secure
GPT Modified	$q = 2^{20}$, $n = 20$, $d = 9$, $k = 12$, $t_1 = 1$.	4800	$>2^{46}$	Insecure Gibson Attack
GPT Modified 1	$q = 2^{20}$, $n = 20$, $d = 9$, $k = 12$, $t_1 = 2$.	4800	$>2^{66}$	Insecure Gibson Attack
GPT Modified 2	$q = 2^{20}$, $n = 20$, $d = 11$, $k = 10$, $t_1 = 3$.	4800	$>2^{86}$	Insecure Gibson Attack
GPT Modified 3	$q = 2^{29}$, $n = 29$, $d = 15$, $k = 15$, $t = 7$, $t_1 = 4$, $s = 1$.	19 Kbits	$>2^{160}$	Insecure Overbeck's Attack
GPT Modified 4	$q = 2^{32}$, $n = 32$, $d = 17$, $k = 16$, $t = 8$, $t_1 = 4$, $p = 3$.	13 Kbits	$>2^{164}$	Insecure Overbeck's Attack
GPT Reducible Rank codes	$q = 2^{20}$, $N = 20$, $n = 40$, $k = 24$, $t = 4$, $t_1 = 3$, $m = 10$, $r = 2$.	10 Kbits	$>2^{70}$	Insecure Overbeck's Attack
Instrumental approach	$q = 2^{20}$, $n = 20$, $d = 11$, $k = 10$, $t = 5$, $t_1 = 4$,	4 Kbits	$>2^{80}$	Secure against all known Attack

Note that the Instrumental approach at the bottom of the table is not secure now.

A modified McNie with Gabidulin

- We have modified McNie to make an enhanced McNie with Gabidulin whose public key size is about 1.4 KB at the 128 security level. We will submit our result soon.

Table 1: Suggested parameters for McNie2-Gabidulin

n	k	l	q	m	r	Sec	PK	SK	CT
24	12	22	2	41	6	128	1.476	0.308	0.185
32	16	24	2	53	8	192	2.756	0.530	0.318
36	18	29	2	59	9	256	4.116	0.664	0.399

Connection between Ouroboros-R and McNie

- Gaborit posted a message recovery attack on the McNie cryptosystem that significantly reduced the security of the original suggested parameters.
- To avoid the this attack, we modify the encryption algorithm by introducing an error $\mathbf{e_2}$ on $\mathbf{c_2}$.
- We submitted a below joint paper.
  P. Gaborit, L. Galvez, A. Hauteville, J.-L. Kim, M. J. Kim, Y.-S. Kim, "Dual-Ouroboros: An improvement of the McNie Scheme", submitted to Advances in Mathematics of Communication.

Connection between Ouroboros-R and McNie: Key generation

Key Generation

H : a parity check matrix for an $[n, k]$ an LRPC code over $\mathbb{F}_{q^m} \Rightarrow H' = \begin{bmatrix} H & I \end{bmatrix}$ is still a parity-check for an LRPC code.

$\Phi_{H'}$: an efficient decoding algorithm using H', which can correct errors of weight up to r.

G : a random generator matrix for an $[n, l]$ linear code

P : a random isometric matrix

$F = GP^{-1}H^T$

- Public Key: (G, F)
- Secret Key: $(H, \Phi_{H'})$

Definition

An $n \times n$ invertible matrix P over $\mathbb{F}_{q^m}$ is called an *isometric matrix* if P preserves the weight of any vector in $\mathbb{F}_{q^m}^n$, i.e., for every $\mathbf{x} \in \mathbb{F}_{q^m}^n$, $wt(\mathbf{x}P) = wt(\mathbf{x})$.

Connection between Ouroboros-R and McNie

Encryption

Randomly generate $\mathbf{e_1} \in \mathbb{F}_{q^m}^{n}$ and $\mathbf{e_2} = \mathbb{F}_{q^m}^{n-k}$ such that $rk(\mathbf{e}) = rk(\mathbf{e_1}, \mathbf{e_2}) = r$

$$\begin{aligned} \mathbf{c_1} &= \mathbf{m}G + \mathbf{e_1} \\ \mathbf{c_2} &= \mathbf{m}F + \mathbf{e_2}. \end{aligned}$$

The message $\mathbf{m} \in \mathbb{F}_{q^m}^{l}$ is encrypted as $Enc(\mathbf{m}) = (\mathbf{c_1}, \mathbf{c_2})$.

Connection between Ouroboros-R and McNie

Encryption

Randomly generate $\mathbf{e_1} \in \mathbb{F}_{q^m}^n$ and $\mathbf{e_2} = \mathbb{F}_{q^m}^{n-k}$ such that $rk(\mathbf{e}) = rk(\mathbf{e_1}, \mathbf{e_2}) = r$

$$\begin{aligned} \mathbf{c_1} &= \mathbf{m}G + \mathbf{e_1} \\ \mathbf{c_2} &= \mathbf{m}F + \mathbf{e_2}. \end{aligned}$$

The message $\mathbf{m} \in \mathbb{F}_{q^m}^l$ is encrypted as $Enc(\mathbf{m}) = (\mathbf{c_1}, \mathbf{c_2})$.

Decryption

When $\mathbf{y} = (\mathbf{c_1}, \mathbf{c_2})$ is received, compute

$$\begin{aligned} \mathbf{c_1}P^{-1}H^T - \mathbf{c_2} &= \mathbf{m}GP^{-1}H^T + \mathbf{e_1}P^{-1}H^T - \mathbf{m}GP^{-1}H^T - \mathbf{e_2} \\ &= \mathbf{e_1}P^{-1}H^T - \mathbf{e_2} \\ &= (\mathbf{e_1}P^{-1}, -\mathbf{e_2})H'^T \\ &= \mathbf{e}'H'^T \end{aligned}$$

Since $rk(\mathbf{e}') = rk(\mathbf{e_1}P^{-1}, -\mathbf{e_2}) = r$ apply $\Phi_{H'}$ to obtain $(\mathbf{e}_1', -\mathbf{e_2})$ and then apply the isometric matrix P to $\mathbf{e}_1' = \mathbf{e_1}P^{-1}$ to obtain $\mathbf{e_1}$.
Finally, solve the system $\mathbf{m}G' = \mathbf{c_1} - \mathbf{e_1}$ to recover $\mathbf{m}$.

Dual-Ouroboros KEM

This modification when adapted into a KEM leads to a noncyclic dual version of the Ouroboros-R scheme.

Encapsulation: pick vectors $\mathbf{r} \in \mathbb{F}_{q^m}^{l}$, $\mathbf{e_1} \in \mathbb{F}_{q^m}^{n}$ and $\mathbf{e_2} \in \mathbb{F}_{q^m}^{n-k}$ such that $\mathbf{e} = (\mathbf{e_1}, \mathbf{e_2})$ has weight r, $E = Supp(\mathbf{e})$

$$\mathbf{c_1} = \mathbf{r}G + \mathbf{e_1}, \quad \mathbf{c_2} = \mathbf{r}F + \mathbf{e_2}.$$

The encapsulation is $\mathbf{c} = (\mathbf{c_1}, \mathbf{c_2})$ and the shared key is $K = \text{Hash}(E)$.

Dual-Ouroboros KEM

This modification when adapted into a KEM leads to a noncyclic dual version of the Ouroboros-R scheme.

Encapsulation: pick vectors $\mathbf{r} \in \mathbb{F}_{q^m}^{l}$, $\mathbf{e_1} \in \mathbb{F}_{q^m}^{n}$ and $\mathbf{e_2} \in \mathbb{F}_{q^m}^{n-k}$ such that $\mathbf{e} = (\mathbf{e_1}, \mathbf{e_2})$ has weight r, $E = Supp(\mathbf{e})$

$$\mathbf{c_1} = \mathbf{r}G + \mathbf{e_1}, \quad \mathbf{c_2} = \mathbf{r}F + \mathbf{e_2}.$$

The encapsulation is $\mathbf{c} = (\mathbf{c_1}, \mathbf{c_2})$ and the shared key is $K = \text{Hash}(E)$.

Decapsulation: same except that only the support E is needed to recover the shared key K.

	Ouroboros-R	Dual-Ouroboros
public key	$\mathbf{h} \in \mathbb{F}_{q^m}^{N \times N}$ $\mathbf{s} = \mathbf{x} + \mathbf{hy} \in \mathbb{F}_{q^m}^{N \times N}$ $= \begin{pmatrix} I_N & \mathbf{h} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}$	$G \in \mathbb{F}_{q^m}^{l \times n}$ $F = GP^{-1}H \in \mathbb{F}_{q^m}^{l \times (n-k)}$
private key	$\mathbf{x}, \mathbf{y} \in \mathbb{F}_{q^m}^{N \times N}$	$H \in \mathbb{F}_{q^m}^{(n-k) \times n}$ $n \times n$ isometric matrix P
encryption	$\mathbf{s_r} = \mathbf{r_2 h} + \mathbf{r_1}$ $\mathbf{s_e} = \mathbf{r_2 s} + \mathbf{e_r}$	$\mathbf{c_1} = \mathbf{r}G + \mathbf{e_1}$ $\mathbf{c_2} = \mathbf{r}F + \mathbf{e_2}$

References


Aragon, N., Gaborit, P., Hauteville, H., Tillich, J.-P.: Improvement of generic attacks on the rank-syndrome decoding problem. 2017. <hal-01618464>


Bernstein, D.J., Lange, T., and Peters, C.: Attacking and defending the McEliece cryptosystem. In Proceedings of the 2nd International Workshop on Post-Quantum Cryptography, PQCrypto '08, pp. 31–46, Springer-Verlag, Berlin, Heidelberg (2008).


Gaborit, P., Ruatta, O., Schrek, J., Tillich, J. P., Zémor, G.: Rank based Cryptography: a credible post-quantum alternative to classical crypto. In NIST 2015: Workshop on Cybersecurity in a Post-Quantum World 2015 (2015).


P. Gaborit, L. Galvez, A. Hauteville, J.-L. Kim, M. J. Kim, Y.-S. Kim, "Dual-Ouroboros: An improvement of the McNie Scheme", submitted to Advances in Mathematics of Communication.


Lau, T. S. C., Tan, C. H.: Key Recovery Attack on McNie based on Low Rank Parity Check Codes and its Reparation, IWSEC 2018, Sep. 3-5, 2018


Misoczki, R., Tillich, J. P., Sendrier, N. and Barreto, P. S.: MDPC-McEliece: New McEliece variants from moderate density parity-check codes. IEEE International Symposium on Information Theory - ISIT 2013, pp. 2069-2073 (2013).


Misoczki, R., and Barreto, P. S.: Compact McEliece keys from Goppa codes. In Selected Areas in Cryptography, pp. 376–392 (2009)

THANK YOU
VERY MUCH!