



An effective structural attack on the post-quantum signature scheme pqsigRM

4th June 2025 r.



$|0\rangle \rightarrow |1\rangle$

$|0\rangle$

$-H-$

Andrei Rubenkov and Ivan Chizhov

Outline

1. Basic definitions
2. Reed-Muller codes
3. Codes in pqsigRM signature schemes
4. Hadamard Square of $\mathcal{PQR}$ -code
5. Attack steps
6. Further separation of blocks
7. Attack step after block separation
8. Results of the work
9. Attack step after block separation
10. Cleaning the matrix subblocks

The Relevance of Post-Quantum Cryptography

- ▶ In 1994, Shor's Algorithm was published, solving the factorization problem on a quantum computer in $O(\log^3(n))$;
- ▶ Finding new one-way functions for their implementation in asymmetric cryptography;
- ▶ Syndromic Decoding Problem from Error Correcting Code Theory;
- ▶ In 1978, R.J. McEliece proposed a code-based public-key cryptosystem.

[1/8] Basic definitions

Basic definitions

- ▶ Let V_n be the linear space of binary vectors of length n ;
- ▶ A **linear block** $[n, k]$ -**code** or simply a code is a linear subspace of dimension k of the space V_n ;
- ▶ The matrix $G_{\mathcal{C}}$ that defines the basis of the code $\mathcal{C}$ is called the **generating matrix** of this code. That is, it is a matrix such that:

$$\mathcal{C} = \{m \cdot G_{\mathcal{C}} | m \in V_k\}$$

[2/8] Reed-Muller codes

Reed-Muller codes

$$\mathcal{RM}(r, m) = \{ \Omega_f = (x_1, \dots, x_n), n = 2^m \mid f(y_1, \dots, y_m) = \\ a_0 \oplus \bigoplus_{s=1}^t \bigoplus_{1 \leq i_1 < \dots < i_s \leq m} a_{i_1, \dots, i_s} y_{i_1} \dots y_{i_s}, t \leq r \}$$

Cryptosystems based on Reed-Muller codes

- ▶ Proposed by V.M. Sidelnikov in 1994 as an alternative to the McEliece cryptosystem.
- ▶ The original cryptosystem is built on the basis of Reed-Muller codes $\mathcal{RM}(r, m)$.
- ▶ The Minder-Shokrollahi[2] and Chizhov-Borodin[3] structural attacks were proposed

[3/8] Codes in pqsigRM signature schemes

Modified Reed-Muller codes in pqsigRM[1]

The generator matrix of the modified Reed-Muller codes

$$\left(\begin{array}{cc|cc} G(r, m-2)^{\sigma_1^p} & G(r, m-2)^{\sigma_1^p} & G(r, m-2)^{\sigma_1^p} & G(r, m-2)^{\sigma_1^p} \\ 0 & G(r-1, m-2) & 0 & G(r-1, m-2) \\ \hline 0 & 0 & G(r-1, m-2) & G(r-1, m-2) \\ 0 & 0 & 0 & G(r-2, m-2)^{\sigma_2^p} \end{array} \right)$$

- ▶ For $\forall$ code $\mathcal{RM}(r, m)$
 $\forall r : 0 \leq r \leq m : \mathcal{RM}(r, m) = \{(u|u+v) : u \in \mathcal{RM}(r, m-1), v \in \mathcal{RM}(r-1, m-1)\}$
- ▶ The permutation σ_i^p is generated by first choosing $p \leq 2^{m-2}$ and then creating a permutation of random p variables

The cryptanalyst's problem

- ▶ $G_{\sigma_1, \sigma_2}(r, m)$ — code's generator matrix

Given

- ▶ Public key $G_{\text{pub}}(r, m) = M \cdot G_{\sigma_1, \sigma_2}(r, m) \cdot P$

To find

- ▶ Such γ', σ'_1 and σ'_2 that $G_{\text{pub}}(r, m) \cdot \gamma' = M \cdot G_{\sigma'_1, \sigma'_2}(r, m)$

Define the following notation:

$$\mathcal{C}_{(\alpha_1, \alpha_2, \alpha_3, \alpha_4)} = (\mathcal{C}_{\alpha_1} | \mathcal{C}_{\alpha_2} | \mathcal{C}_{\alpha_3} | \mathcal{C}_{\alpha_4})$$

The code $\mathcal{PQRM}_{\sigma_1, \sigma_2}(r, m)$ is a code of the following type:

$$\begin{aligned} \mathcal{PQRM}_{\sigma_1, \sigma_2}(r, m) = & \mathcal{RM}_{(1,1,1,1)}^{\sigma_1}(r, m-2) + \mathcal{RM}_{(0,1,0,1)}(r-1, m-2) + \\ & + \mathcal{RM}_{(0,0,1,1)}(r-1, m-2) + \mathcal{RM}_{(0,0,0,1)}^{\sigma_2}(r-2, m-2) \end{aligned}$$

The cryptanalyst's code problem

Given

► $\mathcal{C} = (\text{PQRM}_{\sigma_1, \sigma_2}(r, m))^\gamma$

To find

► Permutations $\gamma', \sigma'_1, \sigma'_2 : \mathcal{C}^{\gamma'} = \text{PQRM}_{\sigma'_1, \sigma'_2}(r, m)$

The attack algorithm

- ▶ Separation of blocks of the generating matrix;
- ▶ Using the Chizhov-Borodin attack to find permutations inside blocks;
- ▶ Clearing subblocks.

[4/8] Hadamard Square of $\mathcal{PQR}$ -code

Hadamard product

- ▶ The Hadamard product of vectors $c = (c_1, \dots, c_n) \in V_n$ и $b = (b_1, \dots, b_n) \in V_n$ is called the component-wise product of vectors c and b , i.e. the vector

$$c \circ b = (c_1 \cdot b_1, \dots, c_n \cdot b_n),$$

- ▶ Hadamard product of $[n]$ -codes $\mathcal{C}$ and $\mathcal{B}$ is a code

$$\mathcal{C} \circ \mathcal{B} = \{c \circ b \mid c \in \mathcal{C}, b \in \mathcal{B}\}$$

- ▶ The Hadamard square of the code $\mathcal{C}$ is the code $2\mathcal{C} = \mathcal{C} \circ \mathcal{C}$.
- ▶ Denote $q\mathcal{C} = (q - 1)\mathcal{C} \circ \mathcal{C}$.

Properties of Hadamard product

- ▶ For any codes $\mathcal{C}$ and $\mathcal{B}$ of length n hold $\mathcal{C}^\sigma \circ \mathcal{B}^\sigma = (\mathcal{C} \circ \mathcal{B})^\sigma$;
- ▶ For any codes $\mathcal{C}_\alpha$ and $\mathcal{B}_\beta$ of length n hold $\mathcal{C}_\alpha \circ \mathcal{B}_\beta = (\mathcal{C} \circ \mathcal{B})_{\alpha \circ \beta}$
- ▶ $\mathcal{RM}(r_1, m) \circ \mathcal{RM}(r_2, m) = \mathcal{RM}(r_1 + r_2, m)$;
- ▶ $q\mathcal{RM}(r, m) = \mathcal{RM}(qr, m)$.

Hadamard square of $\mathcal{PQR}$ M-code

Generator matrix of modified Reed-Muller code $2\mathcal{PQR}_{\sigma_1, \sigma_2}(r, m)$:

$$\left(\begin{array}{c|c|c|c} (2r, m-2)^{\sigma_1^p} & (2r, m-2)^{\sigma_1^p} & (2r, m-2)^{\sigma_1^p} & (2r, m-2)^{\sigma_1^p} \\ 0 & A \circ (r-1, m-2) & 0 & A \circ (r-1, m-2) \\ 0 & 0 & A \circ (r-1, m-2) & A \circ (r-1, m-2) \\ 0 & 0 & 0 & A \circ B \\ \hline 0 & 0 & 0 & B \circ (r-1, m-2) \\ 0 & 0 & 0 & (2r-4, m-2)^{\sigma_2^p} \\ \hline 0 & (2r-2, m-2) & 0 & 0 \\ 0 & 0 & (2r-2, m-2) & 0 \\ 0 & 0 & 0 & (2r-2, m-2) \end{array} \right),$$

$$A = G(r, m-2)^{\sigma_1}, B = G(r-2, m-2)^{\sigma_2}$$

Hadamard power of $\mathcal{PQR}\mathcal{M}$ -code

Proposition

For any $\mathcal{PQR}\mathcal{M}_{\sigma_1, \sigma_2}(r, m)$ and any natural q , exist $[2^{m-2}]$ -codes $\mathcal{A}^q$ and $\mathcal{B}^q$, such that the equality holds

$$q\mathcal{PQR}\mathcal{M}_{\sigma_1, \sigma_2}(r, m) = \mathcal{RM}_{(1,1,1,1)}^{\sigma_1}(\min(qr, m-2), m-2) \oplus \\ \oplus \mathcal{A}_{(0,1,0,1)}^q \oplus \mathcal{A}_{(0,0,1,1)}^q \oplus \mathcal{B}_{(0,0,0,1)}^q.$$

[5/8] Attack steps

Separation of first block

Теорема

Let $q = \lceil \frac{m-2}{r} \rceil - 1$. Then if the permutations $\sigma_1, \sigma_2 \in S_{2^{m-2}}$ are such that $\mathcal{A}^q = \mathcal{B}^q = \mathcal{V}_{2^{m-2}}$, then the next equality is true

$$q\mathcal{PQR}\mathcal{M}_{\sigma_1, \sigma_2}(r, m) = \mathcal{RM}^{\sigma_1}(qr, m-2) \times \mathcal{V}_{3 \cdot 2^{m-2}}$$

and at the same time $q\mathcal{PQR}\mathcal{M}_{\sigma_1, \sigma_2}(r, m) \neq \mathcal{V}_{2^m}$.

$$\left(\begin{array}{c|ccc} G'(qr, m-2)^{\sigma_1} & & & \\ \hline & I_{k_2 \times n} & & \\ & & I_{k_2 \times n} & \\ & & & I_{k_3 \times n} \end{array} \right),$$

where $k_i = k(r+1-i, m-2)$, $n = 2^{m-2}$.

Conditions of the attack

1. The dimensions of code must satisfy $2r < m - 2$, otherwise the first block will contain $I_{n \times n}$;
2. Permutation σ_1 must differ significantly from the automorphism of the Reed-Muller code.

Probability of unsuccessful separation of the first block

r	m	N_{exp}	P
2	8	350	24%
2	9	350	12.29%
3	9	350	3.14%
2	10	150	7.14%
3	10	150	8.67%

In cases where the algorithm does not work, the permutation σ_1 is such that the code $\mathcal{RM}^{\sigma_1}(r, m - 2)$ is close to the code $\mathcal{RM}(r, m - 2)$.

Experiments with fixed p

p	7	8	9	10	11	12	13	14	15
$\mathcal{PQR}\mathcal{M}(2, 8)$	0	0	0	0	2	19	87	157	228
$\mathcal{PQR}\mathcal{M}(2, 9)$	0	0	0	0	0	0	0	0	16
$\mathcal{PQR}\mathcal{M}(3, 9)$	0	67	224	319	337	348	350	350	350

p	16	17	18	19	20	21	22	23	24
$\mathcal{PQR}\mathcal{M}(2, 8)$	255	303	323	331	344	346	350	350	350
$\mathcal{PQR}\mathcal{M}(2, 9)$	71	136	221	291	313	321	338	345	349
$\mathcal{PQR}\mathcal{M}(3, 9)$	350	350	350	350	350	350	350	350	350

[6/8] Further separation of blocks

Original matrix

Let apply the Gaussian transform to the original matrix over the first block

$$\left(\begin{array}{c|ccc} G(r, m-2)^{\sigma_1^P} & G(r, m-2)^{\sigma_1^P} & G(r, m-2)^{\sigma_1^P} & G(r, m-2)^{\sigma_1^P} \\ \hline 0 & G(r-1, m-2) & 0 & G(r-1, m-2) \\ 0 & 0 & G(r-1, m-2) & G(r-1, m-2) \\ 0 & 0 & 0 & G(r-2, m-2)^{\sigma_2^P} \end{array} \right)$$

Generator matrix of code

$$\begin{pmatrix} G(r, m) & 0 & G(r, m) \\ 0 & G(r, m) & G(r, m) \\ 0 & 0 & G(r-1, m)^{\sigma_2} \end{pmatrix}$$

The code $\text{QRM}_{\sigma_2}(r, m)$ is a code of the following form:

$$\text{QRM}_{\sigma_2}(r, m) = \mathcal{RM}_{(1,0,1)}(r, m) + \mathcal{RM}_{(0,1,1)}(r, m) + \mathcal{RM}_{(0,0,1)}^{\sigma_2}(r-1, m)$$

$2r < m$ case

Generator matrix of code $2\mathcal{QR}\mathcal{M}_{\sigma_2}(r, m)$ has the following form:

$$\begin{pmatrix} G(2r, m) & 0 & 0 \\ 0 & G(2r, m) & 0 \\ 0 & 0 & G_{\mathcal{C}} \end{pmatrix},$$

where $\mathcal{RM}(2r, m) \subset \mathcal{C}$.

[7/8] Attack step after block separation

Public key

$$M \cdot \begin{pmatrix} G(r, m-2)^{\sigma_1} & G(r, m-2)^{\sigma_1} & G(r, m-2)^{\sigma_1} & G(r, m-2)^{\sigma_1} \\ 0 & G(r-1, m-2) & 0 & G(r-1, m-2) \\ 0 & 0 & G(r-1, m-2) & G(r-1, m-2) \\ 0 & 0 & 0 & G(r-2, m-2)^{\sigma_2} \end{pmatrix} \cdot P$$

The matrix after block separation

$$\begin{pmatrix} G(r, m-2)^{\sigma_1 \gamma_1} & G(r, m-2)^{\sigma_1 \gamma_2} & G(r, m-2)^{\sigma_1 \gamma_3} & G(r, m-2)^{\sigma_1 \gamma_4} \\ 0 & G(r-1, m-2)^{\gamma_2} & 0 & G(r-1, m-2)^{\gamma_4} \\ 0 & 0 & G(r-1, m-2)^{\gamma_3} & G(r-1, m-2)^{\gamma_4} \\ 0 & 0 & 0 & G(r-2, m-2)^{\sigma_2 \gamma_4} \end{pmatrix}$$

In case $2r < m - 2$ the complexity of the separation algorithm is $O(n^4 \cdot \log_2(\log_2(n)))$

- ▶ For a full key recovery it is necessary to clear the blocks above the diagonal and find permutations $\sigma_1, \sigma_2, (\gamma_1|\gamma_2|\gamma_3|\gamma_4)$;
- ▶ For this purpose, the Chizhov-Borodin attack with complexity $O(L(n))$ and the Reed-Muller code decoding algorithm, which corrects 2^{m-r} errors with complexity $O(\text{DecRM}(n))$.

[8/8] Results of the work

Results of the work

Теорема

The cryptanalyst's problem is solved in

$O(n^4 \cdot \log_2(\log_2(n)) + n^2 \cdot \text{DecRM}(n) + L(n))$ operations, where $\text{DecRM}(n)$ is the complexity of decoding the Reed-Muller code and $L(n)$ is the complexity of reconstructing the structure of the Reed-Muller code, if the following conditions are met:

- ▶ *The dimensions r and m of the code $\mathcal{PQR}_{\sigma_1, \sigma_2}(r, m)$ satisfy the condition $2r < m - 2$;*
- ▶ *Permutations $\sigma_1, \sigma_2 \in S_{2^{m-2}}$ are such that $\mathcal{A}^q = \mathcal{B}^q = \mathcal{V}_{2^{m-2}}$, where $q = \lceil \frac{m-2}{r} \rceil - 1$.*

Algorithm running time

$\mathcal{PQR}_{\sigma_1\sigma_2}(r, m)$	Block separation	Cleaning the subblocks
(3, 9)	00 : 00 : 01	00 : 00 : 01
(4, 11)	00 : 04 : 08	00 : 00 : 15
(5, 13)	18 : 04 : 09	00 : 33 : 03

Bibliography I

-  Yongwoo Lee, Wijik Lee, Young Sik Kim, and Jong-Seon No.
Modified pqsigrm: Rm code-based signature scheme.
IEEE Access, 8:177506–177518, 2020.
-  Lorenz Minder and Amin Shokrollahi.
Cryptanalysis of the sidelnikov cryptosystem.
4515, 05 2007.
-  Михаил Бородин and Иван Чижов.
Эффективная атака на криптосистему Мак-Элиса, построенную на основе кодов Риды - Маллера.
Дискретная математика, 26:10–20, 01 2014.



Ready to answer your questions

Andrei Rubenkov

Специалист-исследователь
лаборатории криптографии

a.rubenkov@kryptonite.ru



[9/8] Attack step after block separation

The matrix after block separation

$$\begin{pmatrix} G(r, m-2)^{\sigma_1 \gamma_1} & G(r, m-2)^{\sigma_1 \gamma_2} & G(r, m-2)^{\sigma_1 \gamma_3} & G(r, m-2)^{\sigma_1 \gamma_4} \\ 0 & G(r-1, m-2)^{\gamma_2} & 0 & G(r-1, m-2)^{\gamma_4} \\ 0 & 0 & G(r-1, m-2)^{\gamma_3} & G(r-1, m-2)^{\gamma_4} \\ 0 & 0 & 0 & G(r-2, m-2)^{\sigma_2 \gamma_4} \end{pmatrix}$$

In case $2r < m - 2$ the complexity of the separation algorithm is $O(n^4 \cdot \log_2(\log_2(n)))$

Application of the Chizhov-Borodin attack

- ▶ We obtain the permutations $\sigma_1\gamma_1$, γ_2 , γ_3 and $\sigma_2\gamma_4$ up to automorphism.
- ▶ Matrix after applying the permutation $(\sigma_1^{-1}\gamma_1^{-1}|\gamma_2^{-1}|\gamma_3^{-1}|\sigma_2^{-1}\gamma_4^{-1})$:

$$\begin{pmatrix} G(r, m-2) & G(r, m-2)^{\sigma_1} & G(r, m-2)^{\sigma_1} & G(r, m-2)^{\sigma_1\sigma_2^{-1}} \\ 0 & G(r-1, m-2) & 0 & G(r-1, m-2)^{\sigma_2^{-1}} \\ 0 & 0 & G(r-1, m-2) & G(r-1, m-2)^{\sigma_2^{-1}} \\ 0 & 0 & 0 & G(r-2, m-2) \end{pmatrix}$$

[10/8] Cleaning the matrix subblocks

Block cleaning idea

Given the generating matrix of the code $\mathcal{D}$ and the matrix L , which was multiplied by this matrix from the right:

$$\begin{pmatrix} G(r, m)^\sigma & G(r, m)^\sigma \\ 0 & G(r-1, m) \end{pmatrix}, \begin{pmatrix} P_{\sigma^{-1}} & M \\ 0 & I_{2^m \times 2^m} \end{pmatrix}$$

The matrix of the obtained code $\mathcal{D}'$ has the form:

$$\begin{pmatrix} G(r, m) & G_{\mathcal{C}} \\ 0 & G(r-1, m) \end{pmatrix},$$

where code $\mathcal{C} = \mathcal{RM}^\sigma(r, m) + E$, $E \subset \mathcal{RM}(r-1, m)$.

Then there exists an algorithm that cleans the code $\mathcal{C}$ from the term $\mathcal{RM}(r-1, m)$ in $O(n^2 \text{DecRM}(n))$ operations.

Block cleaning algorithm

$$\begin{pmatrix} G(r, m) & G_e \\ 0 & G(r-1, m) \end{pmatrix}$$

1. Applying the Gaussian transworn to obtain a basis of words of minimum weight in the subblock (1, 1);
2. In the subblock (1, 2) the words have the form $c = f + e$, where $|e| = d(r, m) = 2^{m-r}$, $e \in \mathcal{RM}^\sigma(r, m)$, $f \in \mathcal{RM}(r-1, m)$;
3. We use the matrix lying in the subblock (2, 2) to decode c . The Reed-Muller code decoder corrects at least $t = 2^{m-r} - 1$ errors;

Block cleaning algorithm

$$\begin{pmatrix} G(r, m) & G_e \\ 0 & G(r-1, m) \end{pmatrix}$$

- 4 After finding vectors e we also find the vectors $f = c + e$;
- 5 Finding each vector e requires no more than $2^m - 2^{m-r}$ decoding c times by replacing 1 coordinate;
- 6 Solving a system of linear equations we find σ .

The final stage

$$\begin{pmatrix} G(r, m-2) & G(r, m-2)^{\sigma_1} & G(r, m-2)^{\sigma_1} & G(r, m-2)^{\sigma_1 \sigma_2^{-1}} \\ 0 & G(r-1, m-2) & 0 & G(r-1, m-2)^{\sigma_2^{-1}} \\ 0 & 0 & G(r-1, m-2) & G(r-1, m-2)^{\sigma_2^{-1}} \\ 0 & 0 & 0 & G(r-2, m-2) \end{pmatrix}$$

- ▶ Applying the algorithm to the submatrices (2×2) , we find the permutations σ_1 and σ_2 and clear the remaining blocks;
- ▶ Knowing the permutations σ_1 and σ_2 we completely restore the secret key.