QUANTUM RESISTANT REED-MULLER CODES ON McELIECE CRYPTOSYSTEM

by

Jasmine Elder

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Applied Mathematics

Charlotte

2020

Approved by:

_____
Dr. Gabor Hetyei

_____
Dr. Yongge Wang

_____
Dr. Evan Houston

_____
Dr. Manuel Pérez-Quiñones

ABSTRACT

JASMINE ELDER. QUANTUM RESISTENT REED MULLER CODES ON MCELIECE CRYPTOSYSTEM. (Under the direction of Dr. GABOR HETYEI)

Recently, Dr. Wang presented a new post quantum encryption scheme, Random Linear Code-Based Encryption scheme, RLCE, which is a variant to the McEliece encryption scheme. It is already well-known that the McEliece Encryption scheme based upon Reed Muller codes is not considered as a secure system for both classical and quantum computers. In this dissertation, we introduce and study the Reed-Muller code-based RLCE scheme. These successful attacks on the Reed Muller code based McEliece encryption scheme, namely, the Minder-Shokrollahi's attack, the Chizhov-Borodin's attack, and the Square Code attack, are proven to not work for the proposed Reed Muller code-based RLCE scheme. We determine the optimal method in preventing these known attacks against the new encryption system. Additionally, we suggest parameters needed for the 128, 192, and 256 bits security level.

DEDICATION

I dedicate this dissertation to my children, Kevin Junior and Kairo, for that they forever know that they can achieve the unimaginable. I hope they understand why Mommy spent so much time on the computer. I also dedicate this to my parents, Tonya and Gregory, because they instilled in me the desire to pursue my passions and reach my goals. To my ancestors, looking down on me, I thank you for your sacrifice, for it prepared and paved a way for me. To my sister, Chanelle, thank you for always being a role model, a teacher and a confident, indeed a blessing. To my brother Alexander, in memoriam, your drive to finish what you started, and your stick-to-itiveness has inspired me to stay the course. Lastly, to my dear husband, Kevin, I dedicate this to you because your love, support, patience and encouragement has allowed me to complete this journey.

ACKNOWLEDGEMENTS

I am thankful to God for this opportunity and allowing me to continue with this project until completion.

I would like to express my sincere gratitude and appreciation to my advisor Dr. Yongge Wang for his patience, time, knowledge and continuous support. He has enabled me to grow as a student and as a researcher.

I would like to also express my gratefulness to my co-advisor Dr. Hetyei for helping to facilitate my committee, asking relevant questions, and expressing concern when needed. I would like to also thank Dr. Evan Houston for teaching me and giving me the foundation needed to do my research. I would like to thank Dr. Pérez-Quiñones for joining me on my dissertation committee.

Additionally, I would like to thank my fellow cohort members for their friendship, and their support as we engaged in discussions and worked together. I want to thank my family for their love, continued support, encouragement, and sacrifice made to ensure I successfully accomplish this goal.

Special thanks to University of North Carolina at Charlotte for accepting me into the program and allowing myself to complete my research. Also, thank you to my department's graduate student coordinator, Dr. Deng. Dr. Deng has been very patient with my many questions and obstacles. He has allotted me with a teacher's assistantship that was very helpful throughout my tenure as a graduate student. Lastly, a very extraordinary thanks to the foundation of Lucille P. and Edward C Giles Dissertation

Fellowship. This fellowship afforded me the opportunity to solely focus on my dissertation.

TABLE OF CONTENTS

LIST OF TABLES viii
LIST OF ABBREVIATIONS ix
CHAPTER 1: INTRODUCTION/MOTIVATION 1
CHAPTER 2: REED MULLER CODES 4
2.1 Recursive Definition 4
2.2 Recursive Encode 6
2.3 Boolean Functions 6
2.4 Encode and Decode 8
CHAPTER 3: MCELIECE CRYPTOSYSTEM WITH REED MULLER CODES 12
CHAPTER 4: KNOWN ATTACKS ON MCELEICE CRYPTOSYSTEM 14
4.1 Square Code Attack 15
4.2 Minder - Shokrollahi Attack 18
4.3 Borodin-Chizhov Attack 20
4.4 ISD Attack 21
CHAPTER 5: REED MULLER – RLCE 23
5.1 Random Linear Code-based Encryption 23
5.2 Reed Muller – Reed Linear Code-based Encryption 25
5.2.1 SQUARE CODE ATTACK 27
5.2.2 MINDER-SHOKROLLAHI ATTACK 29
5.2.3 BORODIN-CHIZHOV ATTACK 30
5.2.4 SHORT KEYS ATTACK 31
5.2.5 ISD ATTACK 35
5.3 Parameters for RLCE Reed Muller 38
CHAPTER 6: CONCLUSION 39
REFERENCES 40
APPENDIX: MAPLE CODE 43

LIST OF TABLES

TABLE 1: Set of parameters for Reed Muller RLCE scheme 38

TABLE 2: Public-Key size comparison 38

LIST OF ABBREVIATIONS

ECC	ELLIPTIC CURVE CRYPTOLOGY
ISD	INFORMATION SET DECODING
KEM	KEY ENCAPSULATION MECHANISM
NIST	NATIONAL INSTITUTE of SCIENCE and TECHNOLOGY
RLCE	RANDOM LINEAR CODE BASED ENCRYPTION
RM	REED MULLER
RSA	RIVEST SHAMIR ADLEMAN
S.T.	SUCH THAT
SUPP	SUPPORT
WF	WORK FACTOR
WT	WEIGHT
∀	FOR ALL

CHAPTER 1: INTRODUCTION

Cryptology is the science of converting a plaintext, or message into a ciphertext, or scrambled message in order to provide secrecy. For example, when making purchases online, one may use their personal information from their credit card to finalize the transaction. Therefore, cryptology is important because it protects our private information from the non-intended receiver. In cryptology, we typically have two parties, Bob and Alice, who want to communicate with each other over a noisy channel. Ensuring that our communications or technology are secure have always been the essence of cryptology. That means making sure that the financial transactions we make online, any government telecommunications, and interactions on Facebook, to name a few, are protected.

With the development of quantum computers on the horizon, it poses a threat to the security of the current cryptosystems that are in place. Quantum computers are able to process complex algorithms and perform advance calculations like integer factorization and discrete logarithm problem that are expected to break all the current cryptosystems, like RSA, Rivest Shamir Adleman, and Elliptic Curve Cryptology. Thus, it is important to design solid and concrete quantum and classical secure cryptographic systems. Some categories for the candidates of post quantum cryptology include lattice, codes, hash, and others, but this dissertation will focus on code-based cryptology. Linear codes provide a technique to send information over a noisy channel and can be used to protect the information against eavesdroppers. A linear code, C, is a subspace of field F_q^n, with F_q a field. An (n, k) linear code has length n and dimension k on F_q. The

minimum distance of a code C, is the minimum number of digits component-wise that differ between two distinct codewords in C. The weight of a codeword, c, is the number of non-zero symbols in the codeword. If C is a linear code, then its dual, $C^{\perp}$, is also a linear code where

$$C^{\perp} = \{z \in F_q^n \mid < z \cdot c > = 0 \ \forall \ c \in C\} \text{ where } < z \cdot c > = \sum_{i=1}^{n} z_i c_i$$

is the dot product of $z \ and \ c$.

The goal of this dissertation is to develop a code based cryptographic system that is both secure against modern and quantum computers. One of the candidates for code-based quantum cryptography that is being studied is the McEliece encryption scheme. The McEliece cryptosystem is a public key cryptosystem that uses linear error correcting codes in order to create a public and private key (Hankerson et all). McEliece proposed a public key cryptosystem with the underlying fact that decoding random linear codes can be very difficult. This cryptosystem can be generalized to any linear codes with a good decoding algorithm. This encryption scheme has resisted existing quantum computer algorithm attacks. In 1978, Robert McEliece created the original McEliece cryptographic system which is based on binary Goppa codes. Several alternatives have been presented to replace these binary Goppa codes. In particular, Sidelnikov proposed the use of Reed Muller codes instead. Reed Muller codes are one of the oldest family of binary linear error correcting codes (V.M. Sidelnikov). Reed Muller codes are advantageous because of their straightforwardness in encoding and decoding. It, however, has been shown that McEliece Public Key Encryption based on Reed Muller codes can be broken by some attacks. It is important that there are new developments created to defeat all known methods of attacks. This dissertation study will employ Reed

Muller codes in a McEliece encryption scheme variant, the Random Linear Code based Encryption (RLCE) scheme, that can be resistant to quantum computing attacks. Supported by the NSA, National Security Agency, the National Institute of Science of Technology (NIST) has initiated a post-quantum cryptography project to solicit cryptographic techniques that are secure against quantum computers. My academic advisor, Dr. Yongge Wang, has developed a linear code-based quantum-safe technique RLCE, Random Linear Code based Encryption, scheme and has submitted it to NIST as a candidate for future Internet infrastructure protection. The RLCE scheme is valuable because allows for the use of any linear code in its construction. With this, we study the Reed Muller code based RLCE scheme, and that the Reed Muller RLCE scheme proves to be a contender for the post quantum cryptology era.

CHAPTER 2: REED MULLER CODES

Reed Muller codes are one of the oldest family of binary linear error correcting codes. Reed Muller codes, denoted $RM(r, m)$, follow a $[N, k, d]$ format where r represents the order of the code and m helps determine the block length and $r \leq m$. Reed Muller codes has a block length of $N = 2^m$, a message length, $k = \sum_{i=0}^{r} \binom{m}{i}$ and distance of $d = 2^{m-r}$. There are two ways that we will define the Reed Muller codes. We will consider the codes first as a recursive definition and then we will discuss the codes in terms of Boolean functions.

2.1 Recursive Definition

Let us consider first the Reed Muller codes of 0^{th} order, $RM(0, m)$, $m \geq 0$. We define $RM(0,0) = \{0,1\}$, $RM(0,1) = \{00,11\}$, and $RM(0,2) = \{0000,1111\}$, and we can continue in this manner for higher values of m. Now let us define 1^{st} order Reed Muller code $RM(1,1) = \{00, 01, 10, 11\}$. Now, we can define the recursive nature of the Reed Muller codes, $RM(r, m) = \{(x, y + x) \mid x \in RM(r, m-1), y \in RM(r-1, m-1)\}$. Using this recursive definition, let us find RM (1, 2). From the definition we see that $RM(1,2) = \{(x, y + x) \mid x \in RM(1,1), y \in RM(0,1)\}$. So, we take all the values in $RM(1,1)$ and adjoin to the end of each of those values from RM (0, 1). Thus,

$RM(1,2) = \{0000, 0011, 0101, 0110, 1001, 1010, 1100, 1111\}$. From here we are able to use the recursive definition to build more Reed Muller codes. (Raaphorst, Sebastian)

Generator matrices are important for linear codes because it is used to transmit messages. The rows of generator matrix form a basis for the linear code. We can use a recursive construction for the generator matrices of $RM(r,m)$, which we will denote by $G(r,m)$. (Hankerson, D.R. et. Al)

$$G(0,m) = [1\ 1 \ldots 1]$$

$$G(r,m+1) = \begin{bmatrix} G(r,m) & G(r,m) \\ 0 & G(r-1,m) \end{bmatrix}$$

$$G(m,m) = \begin{bmatrix} G(m-1,m) \\ 0 \cdots 0\ \ 1 \end{bmatrix}$$

Let us explore some examples of generator matrices.

$$G(0,1) = [1\ 1]$$

$$G(0,2) = [1\ 1\ 1\ 1]$$

$$G(1,1) = \begin{bmatrix} G(0,1) \\ 0\ \ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

$$G(1,2) = \begin{bmatrix} G(1,1) & G(1,1) \\ 0 & G(0,1) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

2.2 Encoding Reed Muller Codes

To encode a message m into a codeword c, we multiply the message, m, by the generator matrix $G(r,m)$, i.e. $m * G(r,m) = c$. For example, let

$$m = [0\ 1\ 1\ 0]$$

and let

$$G(2,2) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Then

$$m * G(2,2) = 0 * [1\ 1\ 1\ 1] + 1 * [0\ 1\ 0\ 1] + 1 * [0\ 0\ 1\ 1] + 0 * [0\ 0\ 0\ 1]$$
$$= [0\ 1\ 1\ 0].$$

2.3 Boolean Functions

Now we will explore defining Reed Muller codes based on Boolean Functions. A Boolean monomial p is an element of $F_2[x_0, x_1, \dots, x_{m-1}]$, with m variables, of the form $p = x_0^{r_0} x_1^{r_1} \dots x_{m-1}^{r_{m-1}}$. A Boolean polynomial is a linear combination of the Boolean monomials. We are going to define codes of length $n = 2^m$, and to do so we need m

variables which take on values 0 or 1. $RM(r,m)$ is the set of all polynomials of degree $\leq r$ in the following ring $F_2[x_0, x_1, \dots, x_{m-1}]$.(Raaphorst, Sebastian)

We define the following mapping rule, where ψ: $F_2[x_0, x_1, \dots, x_{m-1}] \rightarrow F_2^{2^m}$

ψ (0) = $00 \cdots 0$ of length 2^m,

ψ (1) = $11 \cdots 1$ of length 2^m,

ψ (x_1) = $00 \cdots 011 \cdots 1$ two patterned sections of length 2^{m-1},

ψ (x_2) = $00 \cdots 011 \cdots 1$ four patterned sections of length 2^{m-2},

ψ (x_i) = $00 \cdots 011 \cdots 1$ 2^i patterned sections of length 2^{m-i-1}.

Now based upon Boolean functions we can define the generator matrix, $G(r,m)$ for Reed Muller codes.

$$G(r,m) = \begin{bmatrix} \psi(1) \\ \psi(x_1) \\ \psi(x_2) \\ \vdots \\ \psi(x_m) \\ \psi(x_1 x_2) \\ \psi(x_0 x_2) \\ \vdots \\ \psi(x_{m-1} x_m) \\ \psi(x_1 x_2 x_3) \\ \vdots \\ \psi(x_{m-r} x_{m-r+1} \cdots x_m) \end{bmatrix}.$$

For example,

$$G(1,3) = \begin{bmatrix} \psi(1) \\ \psi(x_1) \\ \psi(x_2) \\ \psi(x_3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

2.4 Encode and Decode

To encode a message m into a codeword c, we follow the same steps outlined above in Section 2.2 for encoding a message. We will simply replace the previous recursive generator matrix with the generator matrix constructed with the Boolean functions.

To decode a message, we first need to discuss characteristic vectors. Consider a monomial p of degree d in $RM(r, m)$. "The characteristic vectors of p are all the vectors corresponding to monomials of degree $m - d$". (Raaphorst, Sebastian) In short, the characteristic vectors of p are the monomials that are not in p and their complement. For example, in $RM(2,4)$, the characteristic vector to x_0, x_2 would contain the monomials $\{x_1x_3, \overline{x_1x_3}, \overline{x_1}x_3, x_1\overline{x_3}\}$ The steps to decode are as follows.

1. Starting with the bottom row in the generator matrix, find its characteristic vectors for that row and then take the dot product of each with the encoded message.
2. Take the majority of the values from the dot products in step one and assign that value as the coefficient of the row.
3. Complete steps 1 and 2 for each row (except for the top row), multiply each coefficient by its corresponding row and add the resulting vectors together to form the M_y vector. Add M_y to the received message. If the resulting vector has more ones than zeros than the top row's coefficient is 1, otherwise it is zero. Add the

top row, multiplied by its coefficient to M_y to get original message. The vector formed by the sequence of coefficients starting from the top row is the original message. (Raaphorst, Sebastian)

Let us consider an example. Consider we had a message $m = [0\ 1\ 1\ 0]$ using $RM(1,3)$ then we have that $mG(1,3) = [0\ 1\ 1\ 0\ 0\ 1\ 1\ 1]$, where

$$G(1,3) = \begin{bmatrix} \psi(1) \\ \psi(x_0) \\ \psi(x_1) \\ \psi(x_2) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Let us work on step number one and step number two simultaneously. Beginning with the bottom row, the characteristic vectors are: $x_2x_1, \overline{x_2x_1}, \overline{x_2}x_1, x_2\overline{x_1}$. So

$$x_2x_1 = [0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ ] \times [0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ ] = [0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ ]$$

$$\overline{x_2x_1} = [1\ 1\ 1\ 0\ 1\ 1\ 1\ 0]$$

$$\overline{x_2}x_1 = [1\ 1\ 0\ 0\ 1\ 1\ 0\ 0] \times [0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ ] = [0\ 1\ 0\ 0\ 0\ 1\ 0\ 0]$$

$$x_2\overline{x_1} = [0\ 0\ 1\ 1\ 0\ 0\ 1\ 1] \times [1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ ] = [0\ 0\ 1\ 0\ 0\ 0\ 1\ 0]$$

Now we take each vector above and multiply it by the received vector. Once we accomplish that then we will repeat those two steps for rows two and three and then proceed on to step 3.

$$[0\ 0\ 0\ 1\ 0\ 0\ 0\ 1] \cdot [01100110] = 0$$

$$[1\ 1\ 1\ 0\ 1\ 1\ 1\ 0] \cdot [01100110] = 0$$

$$[0\ 1\ 0\ 0\ 0\ 1\ 0\ 0] \cdot [01100110] = 0$$

$$[0\ 0\ 1\ 0\ 0\ 0\ 1\ 0] \cdot [01100110] = 0.$$

Thus, the coefficient of row four, bottom row, is 0.

For the third row, the characteristic vectors are: : $x_3x_1, \overline{x_3x_1}, \overline{x_3}x_1, x_3\overline{x_1}$. So

$$x_3x_1 = [0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ ] \times [0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ ] = [0\ 0\ 0\ 0\ 0\ 1\ 0\ 1]$$

$$\overline{x_3x_1} = [1\ 1\ 1\ 1\ 1\ 0\ 1\ 0]$$

$$\overline{x_3}x_1 = [1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ ] \times [0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ ] = [0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ ]$$

$$x_3\overline{x_1} = [0\ 0\ 0\ 0\ 1\ 1\ 1\ 1] \times [1\ 0\ 1\ 0\ 1\ 0\ 10\ ] = [0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ ]$$

Now we take each vector above and multiply it by the received vector.

$$[0\ 0\ 0\ 0\ 0\ 1\ 0\ 1] \cdot [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0] = 1$$

$$[1\ 1\ 1\ 1\ 1\ 0\ 1\ 0] \cdot [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0] = 1$$

$$[0\ 1\ 0\ 1\ 0\ 0\ 0\ 0] \cdot [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0] = 1$$

$$[0\ 0\ 0\ 0\ 1\ 0\ 1\ 0] \cdot [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0] = 1$$

Thus, the coefficient of row three, third row, is 1.

For the second row, the characteristic vectors are: : $x_2x_3, \overline{x_2x_3}, \overline{x_2}x_3, x_2\overline{x_3}$. So

$$x_2x_3 = [0\ 0\ 1\ 1\ 0\ 0\ 1\ 1] \times [0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ ] = [0\ 0\ 0\ 0\ 0\ 0\ 1\ 1]$$

$$\overline{x_2x_3} = [1\ 1\ 1\ 1\ 1\ 1\ 0\ 0]$$

$$\overline{x_2}x_3 = [1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ ] \times [0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ ] = [0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ ]$$

$$x_2\overline{x_3} = [0\ 0\ 1\ 1\ 0\ 0\ 1\ 1] \times [1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ ] = [0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ ].$$

Now, we take each vector above and multiply it by the received vector.

$$[0\ 0\ 0\ 0\ 0\ 0\ 1\ 1] \cdot [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0] = 1$$

$$[1\ 1\ 1\ 1\ 1\ 1\ 0\ 0] \cdot [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0] = 1$$

$$[0\ 0\ 1\ 1\ 0\ 0\ 0\ 0] \cdot [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0] = 1$$

$$[0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ ] \cdot [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0] = 1$$

Thus, the coefficient of row two, second row, is 1.

Now, we can start step three.

So after multiplying the coefficients of the rows by the rows we get that

$M_y = [0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ ] + [0\ 0\ 1\ 1\ 0\ 0\ 1\ 1] = [0\ 1\ 1\ 0\ 0\ 1\ 1\ 0]$.

Now we take the vector M_y and add it to the original received message. So,

Since we got a vector of all zeros, then the top row's coefficient must be zero. Thus, with the coefficient of row one is 0, coefficient of row two is 1, coefficient of row three is 1 and coefficient of row four is 0. Hence, our original message must have been

$[0\ \ 1\ \ 1\ \ 0]$.

CHAPTER 3: McELIECE CRYPTOSYSTEM WITH REED MULLER CODES

With the development of quantum computers, widely known cryptosystems like elliptic curve cryptology and RSA can be broken. Thus, it has become important to develop cryptosystems that are resistant to the attacks of quantum computers. One well known code-based post quantum cryptosystem is the McEliece cryptosystem, originally based on binary Goppa codes due to their fast decoding algorithm. In 1978, McEliece proposed a public key cryptosystem with the underlying fact that decoding random linear codes can be very difficult. One element of the McEliece cryptosystem is that the public code does not have any known structure, thus the potential is extremely high for it to be considered as a secure system. The cryptosystem is alike to a random code. This cryptosystem can be generalized to any linear codes with a good decoding algorithm other than just the binary Goppa codes. There have been several proposals to replace the binary Goppa codes with other codes, like Generalized Reed Solomon codes, binary Reed Muller codes, polar codes and others, which have been proven insecure. It is our goal to replace the binary Goppa codes with Reed Muller codes to make a more efficient quantum resistant cryptosystem.

Let us describe how the McEliece Cryptosystem works with the Reed Muller codes in place of the binary Goppa codes. Let C be a linear code with block length n and dimension k that can correct up to t errors. Let G be the $k\ x\ n$ generator matrix for C. Let S be a $k\ x\ k$ nonsingular scrambler matrix. And let P be a $n\ x\ n$ permutation matrix.

Then, let Bob calculate $G' = SGP$, where G' is the public key, and $S, G,$ and P are Bob's private keys.

Algorithm for encryption:

To encrypt a message m, with length k, Alice would take her message and

1. Calculate $c' = mG'$.

2. Select random n dimensional vector e with $weight(e) = \lfloor \mathrm{d} - 1 \rfloor / 2$.

3. Calculate $c = c' + e$ and sends c to Bob.

Algorithm for decryption:

Bob firsts

1. Calculate $c' = cP^{-1} = (mG' + e)P^{-1} = mSG + + eP^{-1} = mSG + e'$

2. By using a decoding algorithm for Reed Muller codes we can strip off e' and get mSG.

Bob finds mS by multiplying by G^{-1}, Bob can recover m by multiplying by S^{-1}. Bob could have written G in standard form, $[I_k A]$ and mS would be the first k positions of mSG so the multiplication of G^{-1} would not have been needed. Bob can decrypt the message since he knows a decoding algorithm for Reed Muller codes. However, an attacker has to try to recover the structure of the code given by the public keys. (V.M. Sidelnikov)

CHAPTER 4: KNOWN ATTACKS ON MCELEICE CRYPTOSYSTEM

There are a few main cryptanalytic techniques that are used to attack the cryptosystem to recover the message. The two main attacks are the message recovery attack and the key recovery attack. The message recovery attack is based on generic decoding algorithms. The Information Set Decoding, ISD, is a decoding algorithm that searches for an information set such that the error positions are all out of the information set. The ISDs uses information from the ciphertext and public key. The key recovery attack idea is to recover private key from public key. In order withstand these types of attacks we want our codes large enough, as well as, we desire that the structure of the code be hidden. Below is a description of the main attacks on Reed Muller codes. These techniques include finding small weight code words in C that will help reveal the underlying structure of the key, algebraic attacks, considering the products of codewords, and others.

4.1 Square Code Attack

A major concern for code-based cryptosystems is the square code. The square code is an important tool because it can be used to classify between random codes and private codes.

Definition: Consider A and B are two linear codes of length n. Then $A \star B$, called the star product, is a vector space spanned by all products $a \star b = (a_1 b_1, \dots, a_n b_n)$ where $a \in A$ and $b \in B$. If $A = B$, then $A \star A$ is called the square code, commonly written as A^2.

The square code is composed by the component-wise products of codewords of the public code. The comparison of the dimension of the code versus the dimension of its square code is important because the square code attack relies on the use of random columns. It is therefore easy to identify the random columns by computing the square code of the code generated by the public matrix G. The dimension of the square of a linear code, C, is known to be $\dim(C^2) \leq \min\{n, \frac{1}{2}k(k+1)\}$. The dimension of a random linear code will achieve its upper bound with high probability. With the use of random insertion alone, it has been proven that the system is not secure against the square code attack.

The following few propositions are crucial for Reed Muller codes because they are used in the square code attack to distinguish a random code from itself.

Proposition: "Let r and m be two integers such that $0 \leq r < m$. Then the square of a Reed Muller code is also a reed muller code. In fact, $RM(r,m)^2 = RM(2r,m)$."

The proof from (Otmani, Ayoub & Talé Kalachi, Hervé) is below

"**Proof**. Let $c_1 = (f(a_1), \dots, f(a_n))$ and $c_2 = (g(a_1), \dots, g(a_n))$ be elements of $RM(r,m)$ with $\deg f \leq r$. and $\deg g \leq r$. Hence, $c_1 \star c_2$ is the vector $(fg(a_1), \dots, fg(a_n))$ which corresponds to polynomial fg. This means $c_1 \star c_2 \in RM(2r,m)$. Conversely, each monomial $x_1^{e_1}, \dots x_m^{e_m}$ with $e_i \geq 0$ and $\sum_i e_i \leq 2r$ is the product of two polynomials of degree less than or equal to r. This proves that a basis of $RM(2r,m)$ is contained in $RM(r,m)^2$."

■

Another proposition, by Ayoub Otmani & Hervé Talé Kalachi, is seen below.

Proposition: "Let G be a $k \times (n+l)$ matrix obtained by inserting l random columns in the generating matrix of a Reed Muller code $RM(r,m)$ and let C be the code spanned by the rows of G. Assume that $l \leq \binom{k+1}{2}$ and $\sum_{i=0}^{2r}\binom{m}{i} \leq n$. Then $\sum_{i=0}^{2r}\binom{m}{i} \leq \dim C^2 \leq \sum_{i=0}^{2r}\binom{m}{i} + l$." (Herve Tale Kalachi)

Thus, if C is a $(n+l,k)$ code and D is a (n,k) code then

$$\dim C^2 \leq \dim D^2 + l$$

For Reed Muller codes, we have that the upper bound is reached. So, we have the following

$$\dim C^2 = \dim D^2 + l.$$

Proposition: "For any $i \in \{1, \dots, n\}$, we have the following:

$$\dim C_i^2 = \begin{cases} \dim C^2 - 1 & if \quad i \in I \\ \dim C^2 & if \quad i \notin I \end{cases}$$

where C_i is the punctured code at index i." (Herve Tale Kalachi)

The author claims that from the prior proposition that the square code can distinguish the random positions of the public code if $\sum_{i=0}^{2r} \binom{m}{i} + l \leq n$. The set I could be discovered and then we would apply the typical attacks on the Reed Muller code to recover message, which we will discuss next.

4.2 Minder-Shokrollahi Attack

In 1994 Sidelnikov proposed the use of Reed Muller codes for building the McEliece cryptosystem. However, their proposed McEliece cryptosystem was shown to be broken by an attack. This idea of the attack is to compute two minimum weight codewords that have close support. The attack main tools are built upon the following

two theorems and definitions. The following theorems and definitions come from the Lorenz Minder and Amin Shokrollahi in *Cryptanalysis of the Sidelnikov Cryptosystem*.

Theorem: "*For any integer, m, we have RM (0, m)* $\subset$ *RM (1, m)* $\subset \cdots \subset$ *RM (m, m)".* (Lorenz Minder and Amin Shokrollahi)

Definition: "*The support of a codeword c* $\in$ *RM (r, m) is defined as the set of indices* i *such that* $c_i \neq 0$*, which is denoted supp(c)".* (Lorenz Minder and Amin Shokrollahi)

Definition: "*Let* c *be a codeword of C and L be an index set. Then,* $proj_L(c)$ *is a sub-codeword which is composed of the components with indices in L from c. Also, for a linear code C, we define* $proj_L(C) = \{proj_L(c) \mid c \in C\}$*".* (Lorenz Minder and Amin Shokrollahi)

For example, Let $c = [0\ \ 0\ \ 1\ \ 1\ \ 0\ \ 0\ \ 1\ \ 1\ ]$ and $L = 3, 4, 6, 7$. Then,

$$proj_L(c) = [1\ \ 1\ \ 0\ \ 1].$$

Theorem: "Let x be a codeword with minimum weight in $RM(r, m)$. Then, there exists $x_1, x_2 \ldots, x_r \in RM(1, m)$ such that $x = x_1 \cdot x_2 \cdots x_r$ where x_i is a codeword with the minimum weight in $RM(1, m)$ and $x_i \cdot x_j$ denotes the component-wise multiplication." (Lorenz Minder and Amin Shokrollahi)

The reason the attack works is because minimum weight words in the r^{th} order Reed Muller code of length 2^m are products of r minimum weight words in $RM(1, m)$.

Products of r linearly independent first order codewords are minimum weight in $RM(r,m)$. There are $2^{mr-r(r-1)}$ minimum weight codewords in $RM(1,m)$.

One of the main ideas on attacking the McEliece cryptosystems is to find the permutation matrix P. Let σ be any permutation on the set $\{1,2,\dots,n\}$. For any code C of length n, denote c^{σ} to be the code obtained from C with positions permuted according to σ. Given the permuted scrambled Reed Muller code, C, construct σ such that resulting code is also a Reed Muller code. (Lorenz Minder and Amin Shokrollahi)

Below is an outline of the Minder-Shokrollahi attack.

Let C=$RM(r,m)^{\sigma}$ for some unknown σ, given by an arbitrary generator matrix.

1. "Find codewords in C with high probability to also belong to $RM(r-1,m)^{\sigma}$. Find enough of such vectors to build a basis of $RM(r-1,m)^{\sigma}$.
2. Iterate the previous step while decreasing the value of r until reach $RM(1,m)^{\sigma}$
3. Find a permutation σ' such that $RM(1,m)^{\sigma\sigma'} = RM(1,m)$. Then σ' will be found and satisfy $RM(r,m)^{\sigma\sigma'} = RM(r,m)$.

Then $\sigma' = P^{-1}$." (Lorenz Minder and Amin Shokrollahi)

So, from the outline of the attack, we need to be able to find the subcode $RM(r-1,m)^{\sigma} \subset RM(r,m)^{\sigma}$, find the factors of minimum weight words, and find inner words in the shortened code.

To find the subcode, $RM(r-1,m)^{\sigma} \subset RM(r,m)^{\sigma}$, we start off by finding a codeword for which we know is a product of other codewords, i.e minimum weight codewords. From there we then split the codeword off by a factor of the word. This is done by shortening the code on $supp(x)$, and use the structure of the shortened word to find a factor of x, which is in $RM(r-1,m)^{\sigma}$. Finding enough of these words in $RM(r-1,m)^{\sigma}$ will result in a basis of $RM(r-1,m)^{\sigma}$. (Lorenz Minder and Amin Shokrollahi)

So, breaking this cryptosystem is roughly equivalent to recovering a single Reed Muller code.

4.3 Borodin and Chizhov Attack

The paper of Chizhov-Borodin discusses a new algorithm for the attack on the McEliece cryptosystem based on the $RM(r,m)$ code. The idea that Borodin and Chizhov had was to use two effortless calculations in order to discover the first order Reed Muller code given the r^{th} order Reed Muller code. We learn that given Reed Muller code $RM(r,m)$ we can construct the square code, $RM(2r,m)$, and $RM(kr,m)$, for some constant k, easily. They noticed that the dual code of $RM(r,m)$, which is $RM(m-r-1,m)$, can also be constructed given $RM(r,m)$. Thus, we can obtain $RM(kr+l(m-1),m)$. And finally, we can find a permutation σ‘ if $gcd(r,m-1)=1$ such that $RM(r,m)^{\sigma\sigma'} = RM(r,m)$. If the $gcd(r,m-1)=1$, then we can find a permutation σ‘ such that

$RM(r,m)^{\sigma\sigma'} = RM(1,m)$, from $RM(r,m)^{\sigma'}$. Otherwise, $RM(r-1,m)$ can be obtained by the Minder-Shokrollahi attack that we discussed in previous subsection. By iterating this procedure until we have $gcd(r-k, m-1) = 1$, $RM(1,m)$ can be found. Thus, it is a straightforward process to find the permutation σ‘, that is P^{-1}.

4.4 ISD Attack

In the McEliece encryption scheme, one of the most effective attack on the system involves the Information Set Decoding algorithm because the algorithm does not use any information about the configuration of the code. Thus, the private code is unknown and to succeed with the ISD attack, the random looking code will need to be decoded with no information in regard to its structure. In essence, the ISD algorithm randomly examine codewords until a codeword of a desired weight is obtained. We want to “find a set of coordinates of a jumbled vector which are error-free and such that the restriction of the code’s generator matrix to these positions is invertible.” (Minder) The steps are as follows:

1. We have an $[n,k]$ code C with generator matrix G, and we want to find a word of weight at most t. So, we take the encrypted vector $y \in F_q^n$ which is known to have distance t from C.
2. Denote the closest codeword by $\boldsymbol{c}$. Let $I \subset \{1, \dots, n\}$ of size k be an information set. Assume that y and $\boldsymbol{c}$ agree on the positions indexed by I.

3. Multiply the encrypted vector by the inverse of the submatrix to get the original message. Then, $y_I G_I^{-1}$ is the preimage of $\boldsymbol{c}$ and we find $\boldsymbol{c}$ as $(y_I G_I^{-1})G$. (Christiane Peters)

We will briefly discuss the Stern's ISD in Chapter 5.2.5.

CHAPTER 5: RANDOM LINEAR CODE BASE ENCRYPTION

We have discussed the importance of designing quantum safe cryptographic techniques to protect our internet infrastructure in the quantum computing age and my advisor, Dr. Yongge Wang has designed one that works. He has developed a linear code-based quantum safe technique RLCE (Random Linear Code based Encryption) scheme. The goal of this project is to develop more RLCE based cryptographic systems and solutions that are secure against both quantum and classical computers. My work will be investigating the security of RLCE-Reed Muller and give exact parameters choices for 128, 192, and 256 bits level security. Thus, any brute force attack would need 2^{128}, 2^{192} and 2^{256} operations to attempt to break it.

Now, let us define the RLCE scheme.

5.1 RLCE Scheme

Random Linear Code based Encryption (RLCE) Scheme is believed to be immune to existing attacks on linear code-based encryption schemes, and that the security of the encryption scheme does not depend on the structure of the linear code. Copied below is the protocol for the RLCE scheme by Yongge Wang. The idea of this

scheme is to juxtapose a linear code with a random code to obtain overall a random like code.

Let $n, k, d, t > 0$, and $w \in \{1, 2, \dots, n\}$ be parameters such that $n - k + 1 \geq d \geq 2t + 1$. Also let $G_s = [g_0, g_1, \dots, g_{n-1}]$ be a $k \times n$ generator matrix for an $[n, k, d]$ linear code, C. Let P_1 be a randomly chosen $n \times n$ permutation matrix and $G_s P_1 = [g_0, g_1, \dots, g_n]$.

1. Let $r_0, r_1, \dots, r_{w-1} \in GF(q)^k$ be column vectors drawn uniformly at random and let $G_1 = [g_0, \dots, g_{n-w}, r_0, \dots, g_{n-1}, r_{w-1}]$ be the $k \times (n + w)$ matrix obtained by inserting the column vectors r_i into G_s.

2.Let $A_0 = \begin{bmatrix} a_{0,00} & a_{0,01} \\ a_{0,10} & a_{0,11} \end{bmatrix}, \dots, A_{w-1} = \begin{bmatrix} a_{w-1,00} & a_{w-1,01} \\ a_{w-1,10} & a_{w-1,11} \end{bmatrix} \in GF(q)^{2\times 2}$

be nonsingular 2×2 matrices chosen uniformly at random and such that $a_{i,00}\, a_{i,01}\, a_{i,10}\, a_{i,11} \neq 0$ for all $0, \dots, w - 1$ and let

$$A = diag[I_{n-w}, A_0, \dots, A_{w-1}] = \begin{bmatrix} I_{n-w} & 0 & 0 & 0 & 0 \\ 0 & A_0 & 0 & 0 & 0 \\ 0 & 0 & A_1 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & A_{w-1} \end{bmatrix}$$

be an $(n + w) \times (n + w)$ nonsingular matrix.

3. Let S be a random dense $k \times k$ nonsingular matrix and P_2 be an $(n + w) \times (n + w)$ permutation matrix.
4. The public key is the $k \times (n + w)$ matrix $G = SG_1AP_2$ and the private key is (S, G_s, P_1, P_2, A).

To encrypt a message, $m \in GF(q)^k$, we choose row vector $e \in GF(q)^{n+w}$ such that the $wt(e) \leq t$ and compute the cipher text $c = mG + e$.

To decrypt a received message $c = [c_0, \dots, c_{n+w-1}]$, calculate $cP_2^{-1}A^{-1} = mSG_1 + eP_2^{-1}A^{-1} = [c'_0, \dots, c'_{n+w-1}]$.

Let $c' = [c'_0, \dots, c'_{n-w}, c'_{n-w+2}, c'_{n-w+4}, \dots, c'_{n+w-2}]$ be the row vector of length n selected from the length $n + w$ row vector $cP_2^{-1}A^{-1}$. Then $c'P_1^{-1} = mSG_s + e'$ for some error vector $e' \in GF(q)^n$ where its weight is at most t. Using the efficient decoding algorithm, one can compute mSG_s from $c'P^{-1}$. Let D be a $k \times k$ inverse matrix of SG'_s, where G'_s is the first k columns of G_s. Then, $m = c_1 D$, where c_1 is the first k elements of mSG_s. Finally, calculate the weight $e = wt(c - mG)$. If $wt(e) \leq t$, then output m as the decrypted plaintext. Otherwise, output error. (Wang)

5.2 RLCE-Reed Muller

Now, I will describe how to apply the RLCE scheme to Reed Muller codes. We will see that the steps are the same, just with Reed Muller generator matrix included. Let $n, k, d, t > 0$, and $w \in \{1, 2, \dots, n\}$ be parameters such that $n - k + 1 \geq d \geq 2t + 1$. Also let $G_s = [g_0, g_1, \dots, g_{n-1}]$ be a $k \times n$ generator matrix for the $[2^m, \sum_{i=0}^{r} \binom{m}{i}, 2^{m-r}]$ Reed Muller linear code, C. Let P_1 be a randomly chosen $n \times n$ permutation matrix and $G_s P_1 = [g_0, g_1, \dots, g_n]$.

1. Let $r_0, r_1, \ldots, r_{w-1} \in GF(q)^k$ be column vectors drawn uniformly at random and let $G_1 = [g_0, \ldots, g_{n-w}, r_0, \ldots, g_{n-1}, r_{w-1}]$ be the $k \times (n+w)$ matrix obtained by inserting the column vectors r_i into G_s.

2. Let $A_0 = \begin{bmatrix} a_{0,00} & a_{0,01} \\ a_{0,10} & a_{0,11} \end{bmatrix}, \ldots, A_{w-1} = \begin{bmatrix} a_{w-1,00} & a_{w-1,01} \\ a_{w-1,10} & a_{w-1,11} \end{bmatrix} \in GF(q)^{2\times 2}$ be nonsingular 2×2 matrices chosen uniformly at random and such that $a_{i,00}\, a_{i,01}\, a_{i,10}\, a_{i,11} \neq 0$ for all $0, \ldots, w-1$ and let

$$A = diag[I_{n-w}, A_0, \ldots, A_{w-1}] = \begin{bmatrix} I_{n-w} & 0 & 0 & 0 & 0 \\ 0 & A_0 & 0 & 0 & 0 \\ 0 & 0 & A_1 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & A_{w-1} \end{bmatrix}$$

be an $(n+w) \times (n+w)$ nonsingular matrix.

3. Let S be a random dense $k \times k$ nonsingular matrix and P_2 be an $(n+w) \times (n+w)$ permutation matrix.

4. The public key is the $k \times (n+w)$ matrix $G = SG_1AP_2$ and the private key is (S, G_s, P_1, P_2, A).

To encrypt a message, $m \in GF(q)^k$, we choose row vector $e \in GF(q)^{n+w}$ such that the $wt(e) \leq t$ and compute the cipher text $c = mG + e$.

To decrypt a received message $c = [c_0, \ldots, c_{n+w-1}]$, calculate $cP_2^{-1}A^{-1} = mSG_1 + eP_2^{-1}A^{-1} = [c'_0, \ldots, c'_{n+w-1}]$.

Let $c' = [c'_0, \ldots, c'_{n-w}, c'_{n-w+2}, c'_{n-w+4}, \ldots, c'_{n+w-2}]$ be the row vector of length n selected from the length $n+w$ row vector $cP_2^{-1}A^{-1}$. Then $c'P_1^{-1} = mSG_s + e'$

for some error vector $e' \in GF(q)^n$ where its weight is at most t. Using the efficient decoding algorithm, one can compute mSG_s from $c'P^{-1}$. Let D be a $k \times k$ inverse matrix of SG_s', where G_s' is the first k columns of G_s. Then, $m = c_1 D$, where c_1 is the first k elements of mSG_s. Finally, calculate the weight $e = wt(c - mG)$. If $wt(e) \leq t$, then output m as the decrypted plaintext. Otherwise, output error. (Wang)

The security of this codes-based cryptosystems depends on the difficulty of the following attacks. There are two different ways to find the original system: find the random matrix directly or distinguish the added random columns from the others. I used maple, a mathematical software, to compute the value for w , such that $\binom{n+w}{w} > 2^{k_c}$, where the value of $k_c = \{128, 192, 256\}$. Also, my value for w is always such that $w \ngeq n - k$. I used the smallest value for w for which securely satisfied all the below attacks.

5.2.1 SQUARE CODE ATTACK

We have seen above that the square code of a Reed Muller code, $RM(r, m)^2$, is $RM(2r, m)$. Thus, we have that the dimension of the square code to be $k = \sum_{i=0}^{2r} \binom{m}{i}$.

Let us now consider the new generator matrix G that now has w randomly inserted columns with C being the code spanned by the rows of G. To withstand the square code

attack, we validate that the dimensions of the square code is satisfied by the below inequality.

$$\sum_{i=0}^{2r} \binom{m}{i} \leq DimC^2 \leq \min\left\{n + w, \sum_{i=0}^{2r} \binom{m}{i} + w\right\}$$

To show this, let G be the public key for an (n, k, d, t, w) RLCE encryption scheme based on a Reed Muller code. Let C be the code generated by the rows of G. Let D_1 be the code with a generator matrix G_1 obtained from G by replacing the randomized w columns with all-zero columns and let D_2 be the code with a generator matrix G_2 obtained from G by replacing the n non-randomized columns with zero columns. Note that $G = G_1 + G_2$. Thus, we have that

$C \subset D_1 + D_2$. Since $D_1 \star D_2 = 0$, combined with the previous sentence, we learn that $C^2 \subseteq D_1^2 + D_2^2$.

For the parameters that we use, we have $2r > m$ and $\mathrm{n} + \mathrm{w} = \dim RM(2r, m) + w$. In this case, the dimension of the square code reaches the maximum since the RLCE scheme with Reed Muller codes behaves like random linear codes. We therefore have $\dim C^2 = D_1^2 + D_2^2 = n + D_2^2 = n + w$. Note that $\dim D_1^2 = \dim RM(2r, m)$ and $\dim D_2^2 = \min\{w, \binom{k}{2}\} = w$. Thus, square code attack will not identify the randomized columns. If the attack were able to be successful, then the attacker would need to employ one of the following additional attacks like the Minder Shokrollahi to recover message.

5.2.2 MINDER SHOKROLLAHI ATTACK

For the Minder Shokrollahi attack, the basis of the attack and the costliest is based on the ability to find low weight codewords. For linear codes, finding low weight words is generally hard. My goal was to limit the ability to find low weight codewords. Their paper suggests for their attack having low rate codes, i.e., for Reed Muller codes, $RM(r, m)$, we have that $r < \frac{m}{2}$. So, if I choose high rate codes such that $r \geq \frac{m}{2}$, it will be harder to successfully attack the system. In the first proposal of Sidelnikov scheme, r is proposed to be small number, thus having a low rate. It was mentioned that, "It is worth noting that the attack of Minder and Shokrollahi becomes infeasible in the high-rate case where *r* is large, due to the difficulty of finding minimum-weight codewords" (Hang Dinh et al). Couple that with the probability that a word of weight t shows up as a row in the diagonalized matrix is $\frac{\binom{k}{1}\binom{n+w-k}{t-1}}{\binom{n+w}{t}}$. (Minder) With our suggested values of k being large in comparison to n then we have that this probability of finding the low weight code words is negligible. (Minder L., Shokrollahi A).

5.2.3 BORODIN CHIZHOV ATTACK

The Borodin Chizhov attack is similar to the Minder Shokrollahi attack in that given a Reed Muller code $RM(r, m)$, the attacker could retrieve $RM(1, m)$, by using square code, dual code and greatest common divisor. However, as we seen above, the Minder Shokrollahi and the Borodin Chizhov attack relies on finding minimum weight codewords. We saw that the probability of doing so was negligible. Also, the attack that Borodin and Chizhov recommend uses the property that the dual of a Reed Muller code is a Reed Muller code. However, the dual of the RLCE-RM code is not necessarily a Reed Muller code due to the randomness property that the RLCE scheme employs. Thus, this attack is not applicable.

5.2.4 ALAIN SHORT KEY ATTACK

Alain Couvreur developed a key recovery attack on the RLCE scheme based on the Generalized Reed Solomon codes. The attack on this cryptosystem is successful for only certain amounts of the added columns, w. We explore whether this attack can be successfully applied to the Reed Muller RLCE system. Recall the definition of the square code,

Definition: Consider A and B are two linear codes of length n. Then $A \star B$, called the star product, is a vector space spanned by all products $a \star b = (a_1 b_1, \dots, a_n b_n)$ where $a \in A$ and $b \in B$. If $A = B$, then $A \star A$ is called the square code, commonly written as A^2.

The dimension of the square code, C^2, with C being a random code of length n and dimension k is

$$\dim\ C^2 = \min\left(\binom{k+1}{2}, n\right) = \min\left(\frac{k(k+1)}{2}, n\right).$$

Whereas, the dimension of the square code of the GRS code is $\dim\ C^2 = 2k - 1$. The square code distinguisher works if the $\dim\ C^2 < n$ or if $\dim C^2 < \frac{k(k+1)}{2}$.

We also define the following two words, puncturing and shortening, which are some traditional ways to obtain some new codes from existing ones. These two methods are

techniques to use the square code as a distinguisher of a random code and a non-randomized code. For $c \in F_q^n$,

Definition: Let $C \subseteq F_q^n$ and $L \subseteq [1, n]$. The puncturing of C at L is the code

$$P_L(C) = \{(c_i)_{i \in [1,n] \setminus L} \; s.t. \; c \in C\}.$$

We puncture C by deleting the coordinates represented in L for all codewords of C. The resultant code is called the punctured *code* of C. The generator matrix of $P_L(C)$is obtained from the generator matrix of C by deleting the columns in L.

Definition: Let $C \subseteq F_q^n$ and $L \subseteq [1, n]$. The shortening of C at L is the code

$$S_L(C) = P_L\{ \, c \in C \; s.t. \forall i \in L, c_i = 0\}.$$

(Alain Couvreur, et al.)

Let C be a linear code and consider the set of coordinates L of i elements and select all the codewords of C that have 0 in the coordinates of L, this set is a subcode of C. Puncturing the subcode on L will produce the shortened code $S_L(C)$, where $|S_L(C)| = n - i$. (Mohamed Saeed Taha)

Example:

Let G be a generator matrix such that $G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$

Let $L = \{5, 6\}$. Then the generator matrix for $P_L(C)$ is $\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$

The $S_L(C) = \{001111, 111100, 110011, 101001\}$.

The author found that they can distinguish some random codes from the GRS code by computing the square code of a shortened code. For this attack to work, the author wants that the square of the shortenings of the code do not behave as random codes. The nature of a GRS code allows for its positions $\{1, \dots, n + w\}$ to be split into four separate categories.

Definition: "The set of twin positions corresponds to columns that result in a mix of a random column and a GRS one." (Alain Couvreur)

The outline of this attack is as follows.

1. "Choose the value of L.
2. Shorten on L positions. Identify pairs of twin positions and repeat.
3. Puncture the twin positions to get a GRS code and apply the Sidelnikov Shestakov attack.

4. For each pair of twin positions, recover the mixing matrix, by de-randomization.
5. Finish to recover the structure of the GRS code." (Alain Couvreur)

The formula that the author Alain Couvreur uses in his "Recovering Short Secret Keys of RLCE in Polynomial Time", for his distinguisher attack is the following:

$$\dim\left(S_L(C)\right)^{*2} < \binom{k+1-|L|}{2}$$

and

$$\dim(S_L(C))^{*2} < \mathrm{n} + \mathrm{w} - |\mathrm{L}|.$$

Then inserting the Reed Muller code information, we have that,

$$\dim(S_L(C))^{*2} = \min\left(\sum_{i=0}^{2r}\binom{m}{i} + w - |L|, n + w - |L|\right) < \binom{k+1-|L|}{2}$$

and

$$\dim(S_L(C))^{*2} = \min\left(\sum_{i=0}^{2r}\binom{m}{i} + w - |L|, n + w - |L|\right) < n + w - |L|.$$

We choose our parameters such that we have

$$\mathrm{n} + \mathrm{w} = \dim RM(2r, m) + w.$$

Thus,

$$\dim(S_L(C))^{*2} = n + w - |L|.$$

Moreover, we see that for our Reed Muller code C of length n, methods of puncturing and shortening result in a code C' with length n' where we have that the dim $C'^2 = n'$. Hence, for this attack, the author says that

> "For this distinguisher to work we need to shorten the code enough so that its square does not fill in the ambient space, but not too much since the square of the shortened code should have a dimension strictly less than the typical dimension of the square of a random code " (Alain Couvreur)

But since the square of the Dimension of the shortened/punctured code fills the space, this attack will not work.

5.2.5 ISD ATTACK

In this nonstructural ISD attack, we use Sterns algorithm, that attempts to solve the problem of finding low weights codewords.

Although there are many variations of the ISD algorithm, most modern and optimal ISD algorithms use or rely on Stern's algorithm. The basis of Stern's algorithm uses the following two parameters, p and l, such that $0 \leq p \leq t$ and $0 \leq l \leq n - k$. Let $I \subset \{1, \dots, n\}$, an information set of size k. "Stern's algorithm divides the information set I into two equal-size subsets X and Y and looks for words having exactly weight p among the columns indexed by X, exactly weight p among the columns indexed by Y, and exactly weight 0 on a fixed uniform random set of l positions outside the I-indexed columns." (Christiane Peters) In every iteration round, an information set I is selected.

We omit some details and just take the result, refer to *RLCE Key Encapsulation Mechanism* paper for more information.

The success probability of one iteration of the algorithm for the Reed Muller RLCE code is

$$\frac{\binom{n+w-t}{k/2-p}\binom{t}{p}\binom{n+w-t-k/2-p}{k/2-p}\binom{t-p}{p}\binom{n+w-k-t+2p)}{l}}{\binom{n+w}{k/2}\binom{n+w-k/2}{k/2}\binom{n+w-k}{l}}.$$

Thus, for this ISD with the Reed Muller RLCE added columns we have the average number of iterations of Stern's algorithm, which is the multiplicative inverse of the success probability of the first round, is the following

$$S_I = \frac{\binom{n+w}{\frac{k}{2}}\binom{n+w-\frac{k}{2}}{\frac{k}{2}}\binom{n+w-k}{l}}{\binom{n+w-t}{\frac{k}{2}-p}\binom{t}{p}\binom{n+w-t-\frac{k}{2}-p}{\frac{k}{2}-p}\binom{t-p}{p}\binom{n+w-k-t+2p}{l}};$$

$where\ n = 2^m\ and\ t = (2^{m-r})\ and\ k = \sum_{i=0}^{r}\binom{m}{i}, with\ p = 1\ and\ l = 3$.

(J. Stern.)

Therefore, we have the following formula to compute the strengths for Stern's ISD attack.

$$\kappa_c = \log_2(S_I\Bigg((2n+2w-k)k^2 + 2l\binom{\frac{k}{2}}{p}(q-1)^p(k+1) + (n-k-l)(k+1)(q-1)^{2p-l}\binom{\frac{k}{2}}{p}^2\Bigg))$$

where, $(2n + 2w - k)k^2 + 2l\binom{\frac{k}{2}}{p}(q-1)^p(k+1) + (n-k-l)(k+1)(q-1)^{2p-l}\binom{\frac{k}{2}}{p}^2$ represents the number of operations needed per iteration for various steps $with\ p = 1\ and\ l = 3$. (Wang)

We omit many details and refer reader to read *Coding theory and applications*, we just simply take the result. For the quantum version of Stern's ISD algorithm, the Grover's algorithm could be used to reduce the iteration steps to the square root of S_I . Grover's algorithm is the fastest searching quantum algorithm. In the worse-case scenario, the Grover algorithm would only need to compute in comparison to classical ISD, the square root iterations. Thus, we have

$$\kappa_q = \log_2(\sqrt{S_I}\left((2n + 2w - k)k^2 + 2l\binom{\frac{k}{2}}{p}(q-1)^p(k+1) + (n-k-l)(k+1)(q-1)^{2p-l}\binom{\frac{k}{2}}{p}^2\right)) \ with\ p = 1\ and\ l = 3.$$ (Wang)

5.3 PARAMETERS FOR RLCE REED MULLER

Thus, to defeat the above attacks I recommend the following:

For 128 bits security, I recommend $RM(7, 13)$, and $w = 13$.

For 192 bits security, I recommend $RM(7, 14)$ and $w = 18$.

For 256 bits security, I recommend $RM(9, 16)$ and $w = 21$.

For example, the RLCE scheme with Reed Muller codes in Table 1 has 128 bits security strength under ISD attacks and 90-bits security strength under quantum ISD attacks.

These recommendations are based upon providing a secure system, defeating all known attacks against the Reed Muller code based RLCE scheme.

Table 1. Set of parameters for Reed Muller RLCE scheme

κ_c	κ_q	$[n, k, t]$	w	$P\kappa$ in bytes
128	90	$[2^{13}, 5812, 64]$	13	216133
192	121	$[2^{14}, 9908, 128]$	18	1002565
256	178	$[2^{16}, 50643, 128]$	21	11784784

Table 2: Public -Key size comparison (in bytes)

κ	RM-RLCE	Polar-RLCE	GRS-RLCE	Classic-McEliece
128	216133	97530	188001	255000
192	1002565	256080	450761	511880
256	11784784	379220	1232001	1326000

SECTION 6: CONCLUSION

We know that post quantum cryptology is an important security priority. In this dissertation, we study Reed Muller Codes in the Random Linear Code-Based Encryption scheme. Although previous Reed Muller codes based on the McEliece cryptosystem have been proven to be vulnerable to the attacks of Minder-Shokrollahi and Borodin-Chizhov and so forth, to our knowledge, this RLCE scheme allows us to prevent all previously known effective attack against it using the Reed Muller codes. Adding additional columns makes for longer codes but does not pose a problem for the success of the cryptosystem. We suggest parameters that optimize security strengths for bit levels 128, 192, and 256. Thus, Reed Muller codes proves to be a suitable contender for post quantum cryptology.

REFERENCES

Adkins, William. *LSU Mathematics*, 2012, www.math.lsu.edu/~adkins/m4023/4023u12ps5a.pdf.

AL-Ashker, Mohammed M. "Coding Theory Lectures." *IUGAZA*, 2010, site.iugaza.edu.ps/mashker/files/2010/02/coding_theory_lecturs_for_m_Al-Ashker.pdf.

Chizhov I.V., Borodin M.A. (2013) The Failure of McEliece PKC based on Reed-Muller codes.

Couvreur, Alain. "Cryptanalysis Techniques in Algebraic Code–Based Cryptography." *Number-Theoretic Methods in Cryptology 2019*, 2019, nutmic2019.imj-prg.fr/slides/AlainCouvreur.pdf.

Couvreur, Alain, et al. "CRYPTANALYSIS OF PUBLIC-KEY CRYPTOSYSTEMS THAT USE SUBCODES OF ALGEBRAIC GEOMETRY CODES." *Eindhoven University of Technology*, 2014, www.win.tue.nl/~ruudp/lectures/14-09-17-4ICMCTA-29.pdf.

Couvreur, Alain, et al. "Recovering Short Secret Keys of RLCE in Polynomial Time." *IACR*, 2018, eprint.iacr.org/2018/528.pdf.

De Giorgi, Andrea. "McEliece-Type Cryptosystems: Costs, Security and an Attack to a Recent Variant." *Universitat Zurich*, 2012, user.math.uzh.ch/rosenthal/masterthesis/03916947/DeGiorgi_2012.pdf.

Dinh, Hang, et al. "Quantum Fourier Sampling, Code Equivalence, and the Quantum Security of the McEliece and Sidelnikov Cryptosystems." *ArXiv*, 2018, arxiv.org/pdf/1111.4382.pdf.

Dumer, Ilya. "Recursive Decoding of Reed-Muller Codes." *ArXiv*, 2017, arxiv.org/pdf/1703.05303.pdf.

F. J. MacWilliams and N. J. A. loane, "The Theory of Error-Correcting Code", North-Holland, (1978).

Guruswami, Venkatesan. "Notes 1: Introduction, Linear Codes." *Carnegie Melon University School of Computer Science*, 2010, www.cs.cmu.edu/~venkatg/teaching/codingtheory/notes/notes1.pdf.

Gueye, Cheikh Thiecoumba, and EL. Hadji Modou Mboup. “Secure Cryptographic Scheme Based on Modified Reed Muller Codes.” *ResearchGate*, 2013, www.researchgate.net/profile/Cheikh_Thiecoumba_Gueye/publication/316455284_Secure_Cryptographic_Scheme_based_on_Modified_Reed_Muller_Codes/links/592c2fa90f7e9b9979adf914/Secure-Cryptographic-Scheme-based-on-Modified-Reed-Muller-Codes.pdf?origin=publication_detail.

Hankerson, D.R., Hoffman D.G., Leonard D.A., Lindner C.C., Phelps K.T., Rodger C.A., Wall J. R., Coding Theory and Cryptography The Essentials. New York: Marcel Dekker, 2000. Print.

Herve Tale Kalachi. Security of cryptographic protocols based on coding theory. Cryptography and Security [cs.CR]. Normandie Université; Université de Yaoundé I, 2017. English. NNT: 2017NORMR045 . tel-01689877

J. Stern. A method for finding codewords of small weight. In *Coding theory and applications*, pages 106–113. Springer, 1989.

Khalifa, O., Abdullah, A., Suriyana, N., Zawanah, S., Hameed, S.,(2008) Reed-Muller Code Simulation Performance.

Lee W., No J., Kim Y. (2017) Punctured Reed Muller code based McEliece cryptosystems.

Minder, Lorenz. “Cryptography Based on Error Correcting Codes.” *Algo.epfl.ch*, 2007, algo.epfl.ch/_media/en/projects/lorenz_thesis.pdf.

Minder L., Shokrollahi A. (2007) Cryptanalysis of the Sidelnikov Cryptosystem. In: Naor M. (eds) Advances in Cryptology - EUROCRYPT 2007. EU- ROCRYPT 2007. Lecture Notes in Computer Science, vol 4515. Springer, Berlin, Heidelberg

Otmani, Ayoub & Talé Kalachi, Hervé. (2015). Square Code Attack on a Modified Sidelnikov Cryptosystem. 9084. 173-183. 10.1007/978-3-319-18681-8_14.

Overbeck, Raphael. “Public Key Cryptography Based on Coding Theory.” *TUprints*, 2007, tuprints.ulb.tu-darmstadt.de/epda/000823/Overbeck_Dissertation_ULB.pdf.

Peters, Christiane. “Information-Set Decoding for Linear Codes over Fq.” *IACR*, 2010, eprint.iacr.org/2009/589.pdf.

Raaphorst, Sebastian. “Reed Muller Codes.” *Citeseerx*, 2003, citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.115.3214&rep=rep1&type=pdf.

Rodolfo Canto Torres and Nicolas Sendrier. Analysis of information set decoding for a sub-linear error weight. In *Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016, Proceedings*, pages 144–161, 2016. (Cited on pages 1 and 14.)

Saeed Taha, Mohamed. “Approche Algébrique Sur L'Equivalence Des Codes.” *Tel Serveur De Theses Multidisciplinaire*, 2017, tel.archives-ouvertes.fr/tel-01678829v1/document.

Seltzer, Larry. “Security and Quantum Computing: Planning next Generation Cryptography.” *HPE*, 28 Aug. 2018, www.hpe.com/us/en/insights/articles/security-and-quantum-computing-planning-beyond-public-key-cryptography-1808.html.

V.M. Sidelnikov,” A public-key cryptosystem based on binary Reed–Muller codes”, Discrete Math. Appl., vol. 4, no. 3, pp. 191-207, 1994.

Wang, Yongge. (2016). Quantum resistant random linear code based public key encryption scheme RLCE. 2519-2523. 10.1109/ISIT.2016.7541753.

Wang, Yongge. (2017). RLCE Key Encapsulation Mechanism (RLCE-KEM) Specification.

APPENDIX: MAPLE CODE

Maple Code for Square Code Attack for security bit levels 128, 192, 256

Square code for 128

$$m := 13; r := 7; w := 13;$$

$$m := 13$$
$$r := 7$$
$$w := 13 \quad \textbf{(1)}$$

$$n := 2^m;\ k := sum\left(\binom{m}{i}, i = 0..9\right); t := \left(2^{m-r}\right); filtrationpa := sum\left(\binom{m}{i}, i = 0..2 \cdot r\right);$$
$$minDim1 := n + w; minDim2 := w + sum\left(\binom{m}{i}, i = 0..2 \cdot r\right);$$

$$n := 8192$$
$$k := 7814$$
$$t := 64$$
$$filtrationpa := 8192$$
$$minDim1 := 8205$$
$$minDim2 := 8205 \quad \textbf{(2)}$$

Square code for 192

$$m := 14; r := 7; w := 18;$$

$$m := 14 \quad \textbf{(3)}$$
$$r := 7 \quad \textbf{(3)}$$
$$w := 18 \quad \textbf{(3)}$$

$$n := 2^m;\ k := sum\left(\binom{m}{i}, i = 0..9\right); t := \left(2^{m-r}\right); filtrationpa := sum\left(\binom{m}{i}, i = 0..2 \cdot r\right);$$
$$minDim1 := n + w; minDim2 := w + sum\left(\binom{m}{i}, i = 0..2 \cdot r\right);$$

$$n := 16384$$
$$k := 14913$$
$$t := 128$$
$$filtrationpa := 16384$$
$$minDim1 := 16402$$
$$minDim2 := 16402 \quad \textbf{(4)}$$

Square code for 256

$m := 16; r := 9; w := 21;$

$$m := 16$$
$$r := 9$$
$$w := 21 \quad \textbf{(5)}$$

$n := 2^m;\ k := sum\left(\binom{m}{i}, i = 0..9\right); t := \left(2^{m-r}\right); filtrationpa := sum\left(\binom{m}{i}, i = 0..2 \cdot r\right);$
$minDim1 := n + w; minDim2 := w + sum\left(\binom{m}{i}, i = 0..2 \cdot r\right);$

$$n := 65536$$
$$k := 50643$$
$$t := 128$$
$$filtrationpa := 65536$$
$$minDim1 := 65557$$
$$minDim2 := 65557 \quad \textbf{(6)}$$

Maple code for Minder – Shokrollahi's Attack for security bit levels 128, 192, 256

MS attack for 128

need

$r \geq \frac{m}{2}$;

$7 \geq \frac{13}{2}$

$$\frac{m}{2} \leq r$$

$$\frac{13}{2} \leq 7 \quad \textbf{(1)}$$

prob of finding a single word of min weight t

2^{13-7}

$$64 \quad \textbf{(2)}$$

$\left(1 - \frac{64}{2^{13}+13}\right)^{5812}$: *evalf* (%);

$$1.715427165\ 10^{-20} \quad \textbf{(3)}$$

THE PROBABILITY THAT A FIXED WORD OF WEIGHT T SHOWS UP AS A ROW IN THE DIAGONALISED MATRIX IS

$\frac{\binom{5812}{1}\binom{2^{13}+16-5812}{63}}{\binom{2^{13}}{64}}$: *evalf* (%)

$$5.905503102\ 10^{-33} \quad \textbf{(4)}$$

$5.905503102\ 10^{-33} \cdot 2^{49}$

$$3.324502696\ 10^{-18} \quad \textbf{(5)}$$

MS attack for 192

$7 \geq \frac{14}{2}$

$$7 \leq 7 \quad \textbf{(6)}$$

prob of finding a single word of min weight t

2^{14-7}

$$128 \tag{7}$$

$$\left(1 - \frac{128}{2^{14} + 18}\right)^{9908} : evalf(\%)$$

$$1.941201493\ 10^{-34} \tag{8}$$

THE PROBABILITY THAT A FIXED WORD OF WEIGHT T SHOWS UP AS A ROW IN THE DIAGONALISED MATRIX IS

$$\frac{\binom{9908}{1}\binom{2^{14} + 18 - 9908}{127}}{\binom{2^{14}}{128}} : evalf(\%)$$

$$3.341412393\ 10^{-50} \tag{9}$$

$$3.341412393\ 10^{-50} \cdot 2^{56}$$

$$2.407741377\ 10^{-33} \tag{10}$$

MS attack for 256

$$9 \geq \frac{16}{2}$$

$$8 \leq 9 \tag{11}$$

prob of finding a single word of min weight t

$$2^{16 - 9}$$

$$128 \tag{12}$$

$$\left(1 - \frac{128}{21 + 2^{16}}\right)^{50643} : evalf(\%)$$

$$1.034663767\ 10^{-43} \tag{13}$$

THE PROBABILITY THAT A FIXED WORD OF WEIGHT T SHOWS UP AS A ROW IN THE DIAGONALISED MATRIX IS

$$\frac{\binom{50643}{1}\binom{2^{16}+21-50643}{127}}{\binom{2^{16}}{128}} : evalf(\%)$$

$$1.475711753\ 10^{-80} \quad \textbf{(14)}$$

$$1.475711753\ 10^{-80} \cdot 2^{72}$$

$$6.968851721\ 10^{-59} \quad \textbf{(15)}$$

Maple code for ISD attack for security bit level 128

2^{128}

$$340282366920938463463374607431768211456 \quad (1)$$

$w := 13$

$$w := 13 \quad (2)$$

$\binom{2^{13}+w}{w};$

$$12150960779637692686972646833809433815142 5 \quad (3)$$

ReedMuller ISD for kc=128

$m := 13; r := 7; w := 13;$

$$m := 13$$
$$r := 7$$
$$w := 13 \quad (4)$$

$n := 2^m; k := sum\left(\binom{m}{i}, i = 0..r\right); t := \left(2^{m-r}\right); evalf(\%); n \cdot m; n - k$

$$n := 8192$$
$$k := 5812$$
$$t := 64$$
$$64.$$
$$106496$$
$$2380 \quad (5)$$

$l := 3; p := 1;$

$$l := 3$$
$$p := 1 \quad (6)$$

$$\kappa c := \log_2\left(\left(\left(\frac{(n+w)!}{\left(\frac{k}{2}\right)! \cdot \left(n+w-\frac{k}{2}\right)!} \cdot \frac{\left(n+w-\frac{k}{2}\right)!}{\left(\frac{k}{2}\right)! \cdot (n+w-k)!} \cdot \left(\binom{n+w-k}{l}\right)\right) \Big/ \left(\frac{(n+w-(t))!}{\left(\frac{k}{2}-p\right)! \cdot \left(n+w-(t)-\left(\frac{k}{2}-p\right)\right)!} \cdot \binom{t}{p} \cdot \frac{\left(n+w-(t)-\frac{k}{2}-p\right)!}{\left(\frac{k}{2}-p\right)! \cdot (n+w-(t)-k)!} \cdot \binom{t-p}{p} \cdot \binom{n+w-(t)-k+2p}{l}\right)\right) \cdot \left((2n+2w-k)k^2 + 2\binom{\frac{k}{2}}{p}(q-1)^p l(k+1)\right)\right)$$

$$+\left(\frac{k}{2}\atop p\right)^2 (q-1)^{2p-l}(n-k-l)(k+1)\Bigg)\Bigg) : \textit{evalf}(\%);$$

142.8587286 **(7)**

quantum

$$\kappa c := \log_2\left(\left(\left(\left(\left(\frac{(n+w)!}{\left(\frac{k}{2}\right)!\cdot\left(n+w-\frac{k}{2}\right)!}\cdot\frac{\left(n+w-\frac{k}{2}\right)!}{\left(\frac{k}{2}\right)!\cdot(n+w-k)!}\right.\right.\right.\right.\right.$$

$$\left.\left.\left.\left.\left.\cdot\left(\binom{n+w-k}{l}\right)\right)\right/\left(\frac{(n+w-(t))!}{\left(\frac{k}{2}-p\right)!\cdot\left(n+w-(t)-\left(\frac{k}{2}-p\right)\right)!}\cdot\binom{t}{p}\right.\right.\right.\right.$$

$$\left.\left.\left.\left.\cdot\frac{\left(n+w-(t)-\frac{k}{2}-p\right)!}{\left(\frac{k}{2}-p\right)!\cdot(n+w-(t)-k)!}\cdot\binom{t-p}{p}\cdot\binom{n+w-(t)-k+2p}{l}\right)\right)\right)^{1/2}\cdot\Bigg((2n+2w-k)k^2$$

$$+2\left(\frac{k}{2}\atop p\right)(q-1)^p l(k+1)+\left(\frac{k}{2}\atop p\right)^2(q-1)^{2p-l}(n-k-l)(k+1)\Bigg) : \textit{evalf}(\%);$$

90.61993553 **(8)**

Maple code for ISD attack for security bit level 192

2^{192}

$$6277101735386680763835789423207666416102355444464034512896 \quad (1)$$

$w := 18$

$$w := 18 \quad (2)$$

$\binom{2^{14}+w}{w};$

$$1142217786244604289353595911474955393279637248783111648717825 \quad (3)$$

ReedMuller ISD for kc=192

$m := 14; r := 7; w := 18;$

$$m := 14$$
$$r := 7$$
$$w := 18 \quad (4)$$

$n := 2^m; k := sum\left(\binom{m}{i}, i = 0..r\right); t := \left(2^{m-r}\right); evalf(\%); n \cdot m; n - k$

$$n := 16384$$
$$k := 9908$$
$$t := 128$$
$$128.$$
$$229376$$
$$6476 \quad (5)$$

$l := 3; p := 1;$

$$l := 3$$
$$p := 1 \quad (6)$$

$$\kappa c := \log_2\left(\left(\left(\frac{(n+w)!}{\left(\frac{k}{2}\right)! \cdot \left(n+w-\frac{k}{2}\right)!} \cdot \frac{\left(n+w-\frac{k}{2}\right)!}{\left(\frac{k}{2}\right)! \cdot (n+w-k)!} \cdot \binom{n+w-k}{l}\right) \Big/ \left(\frac{(n+w-(t))!}{\left(\frac{k}{2}-p\right)! \cdot \left(n+w-(t)-\left(\frac{k}{2}-p\right)\right)!} \cdot \binom{t}{p} \cdot \frac{\left(n+w-(t)-\frac{k}{2}-p\right)!}{\left(\frac{k}{2}-p\right)! \cdot (n+w-(t)-k)!} \cdot \binom{t-p}{p} \cdot \binom{n+w-(t)-k+2p}{l}\right)\right) \cdot \left((2n+2w-k)k^2 + 2\binom{\frac{k}{2}}{p}(q-1)^p l(k+1)\right)\right)$$

$$+\left(\binom{\frac{k}{2}}{p}^2 (q-1)^{2p-l}(n-k-l)(k+1)\right)\Bigg) : \textit{evalf}(\%);$$

201.7097217 **(7)**

quantum

$$\kappa c := \log_2\left(\left(\left(\left(\frac{(n+w)!}{\left(\frac{k}{2}\right)!\cdot\left(n+w-\frac{k}{2}\right)!}\cdot\frac{\left(n+w-\frac{k}{2}\right)!}{\left(\frac{k}{2}\right)!\cdot(n+w-k)!}\right.\right.\right.\right.$$

$$\left.\left.\left.\left.\cdot\binom{n+w-k}{l}\right)\right)\Bigg/\left(\frac{(n+w-(t))!}{\left(\frac{k}{2}-p\right)!\cdot\left(n+w-(t)-\left(\frac{k}{2}-p\right)\right)!}\cdot\binom{t}{p}\right)^{1/2}\right.$$

$$\left.\cdot\frac{\left(n+w-(t)-\frac{k}{2}-p\right)!}{\left(\frac{k}{2}-p\right)!\cdot(n+w-(t)-k)!}\cdot\binom{t-p}{p}\cdot\binom{n+w-(t)-k+2p}{l}\right)\Bigg)\Bigg)\Bigg) \cdot\Bigg((2n+2w-k)k^2$$

$$+2\binom{\frac{k}{2}}{p}(q-1)^p l(k+1)+\binom{\frac{k}{2}}{p}^2(q-1)^{2p-l}(n-k-l)(k+1)\Bigg) : \textit{evalf}(\%);$$

121.3706430 **(8)**

Maple code for ISD attack for security bit level 256

$$2^{256}$$

$$115792089237316195423570985008687907853269984665640564039457584007913129639936 \quad (1)$$

$$w := 21$$

$$w := 21 \quad (2)$$

$$\binom{2^{16}+w}{w};$$

$$27495730141255267930317983319447910806767174706917919654591251462254797347605\backslash 6785 \quad (3)$$

ReedMuller ISD for kc=256

$$m := 16; r := 9; w := 21;$$

$$m := 16$$
$$r := 9$$
$$w := 21 \quad (4)$$

$$n := 2^m; k := sum\left(\binom{m}{i}, i = 0..r\right); t := \left(2^{m-r}\right); evalf(\%); n \cdot m; n - k$$

$$n := 65536$$
$$k := 50643$$
$$t := 128$$
$$128.$$
$$1048576$$
$$14893 \quad (5)$$

$$l := 3; p := 1;$$

$$l := 3$$
$$p := 1 \quad (6)$$

$$\mathbf{kc} := \log_2\left(\left(\left(\frac{(n+w)!}{\left(\frac{k}{2}\right)! \cdot \left(n+w-\frac{k}{2}\right)!} \cdot \frac{\left(n+w-\frac{k}{2}\right)!}{\left(\frac{k}{2}\right)! \cdot (n+w-k)!} \cdot \left(\binom{n+w-k}{l}\right)\right) \Big/ \left(\frac{(n+w-(t))!}{\left(\frac{k}{2}-p\right)! \cdot \left(n+w-(t)-\left(\frac{k}{2}-p\right)\right)!} \cdot \binom{t}{p} \cdot \frac{\left(n+w-(t)-\frac{k}{2}-p\right)!}{\left(\frac{k}{2}-p\right)! \cdot (n+w-(t)-k)!} \cdot \binom{t-p}{p}\right)\right)$$

$$\cdot\binom{n+w-(t)\text{-}k+2p}{l}\Bigg)\Bigg)\cdot\Bigg((2n+2w-k)k^2+2\binom{\frac{k}{2}}{p}(q-1)^p l(k+1)$$

$$+\binom{\frac{k}{2}}{p}^2(q-1)^{2p-l}(n-k-l)(k+1)\Bigg)\Bigg):\ \textbf{evalf}(\%);$$

308.9555159 **(7)**

quantum

$$\kappa c :=$$

$$\log_2\Bigg(\Bigg(\Bigg(\Bigg(\frac{(n+w)!}{\left(\frac{k}{2}\right)!\cdot\left(n+w-\frac{k}{2}\right)!}\cdot\frac{\left(n+w-\frac{k}{2}\right)!}{\left(\frac{k}{2}\right)!\cdot(n+w-k)!}$$

$$\cdot\left(\binom{n+w-k}{l}\right)\Bigg)\Bigg/\Bigg(\frac{(n+w-(t))!}{\left(\frac{k}{2}-p\right)!\cdot\left(n+w-(t)-\left(\frac{k}{2}-p\right)\right)!}\cdot\binom{t}{p}^{1/2}$$

$$\cdot\frac{\left(n+w-(t)\text{-}\frac{k}{2}-p\right)!}{\left(\frac{k}{2}-p\right)!\cdot(n+w-(t)\text{-}k)!}\cdot\binom{t-p}{p}\cdot\binom{n+w-(t)\text{-}k+2p}{l}\Bigg)\Bigg)\Bigg)\Bigg)\cdot\Bigg((2n+2w-k)k^2$$

$$+2\binom{\frac{k}{2}}{p}(q-1)^p l(k+1)+\binom{\frac{k}{2}}{p}^2(q-1)^{2p-l}(n-k-l)(k+1)\Bigg):\ evalf(\%);$$

178.2539239 **(8)**