Intro to Code-Based Schemes

Raul-Martin Rebane
Supervised by Vitaly Skachek

2018
November

Abstract

In the past decades a lot of work has been done in the field of quantum computation. Through the application of Shor's algorithm and others, a large-scale quantum computer could break many current cryptographic standards. While quantum-resistant cryptographic schemes exist, they have not seen much use due to computation cost or lack of standardization. In 2017 the National Institute of Science and Technology (NIST) began the process of standardizing post-quantum cryptographic primitives, as historically it has taken them almost two decades to deploy public key infrastructure.

Of the 59 submissions for public key encryption or key encapsulation schemes, lattice-based and code-based submissions were most popular with 21 and 18 submissions respectively. This report outlines the LAKE submission to a reader who is familiar with cryptography, but lacks coding theory knowledge.

1 Introduction

The first code-based cryptosystem defined by McEliece in 1978 was also one of the first public-key cryptosystems, and thus has been widely studied over the decades [McE78]. While variants of the system have been broken, the original McEliece system using binary Goppa codes has stood against cryptanalysis exceedingly well. Since 1978 there were 25 publications which specified increasingly sophisticated non-quantum attack algorithms, and the result is nearly negligible. To achieve 2^b security, the change in key size is below 0.1% for a large enough b [BCL$^+$17]. This stability is highly valuable in the transition to post-quantum cryptography.

However, the McEliece scheme suffers from some practical drawbacks which have kept it from being adopted as a widely used standard. Since the private and public keys are both large matrices, the number of bits required to represent them is much larger than

schemes which provide similar levels of security. Modern coding theory tackles this issue in a number of different ways. This report details the LAKE key encapsulation scheme, which is one of the NIST candidates for post-quantum standardization [ABD+17].

1.1 Linear codes

As a crash course into linear codes we will first need to establish the following basics of coding theory. Definitions from [cod].

An (n, M)-code $\mathcal{C}$ over an alphabet F is a set of $M > 0$ vectors (called codewords) of length n over F. Two different inputs are mapped onto two different outputs, otherwise the decoding will not be possible.

Let $\mathbb{F} = (F, +, \cdot)$ be a finite field. An (n, M, d) code over $\mathbb{F}$ is called *linear* if $\mathcal{C}$ is a nonempty linear vector subspace of $\mathbb{F}^n$. If $\mathcal{C}$ is a linear code of length n, dimension k and minimum distance d, we say that it is a $[n, k, d]$ code over $\mathbb{F}$.

A generator matrix of a linear $[n, k, d]$-code is a $k \times n$ matrix whose rows form a basis of the code.

A systematic generator matrix over $\mathbb{F}$ is a generator matrix which has the form

$$G = (I_k|A)$$

where A is a $k \times (n - k)$ submatrix and I_k is the $k \times k$ identity matrix. If a systematic generator matrix exists, then it is unique.

Let $\mathcal{C}$ be an $[n, k, d]$ code over a finite field $\mathbb{F}$. A parity-check matrix of a code $\mathcal{C}$ is an $r \times n$ matrix H over $\mathbb{F}$ such that for every $c \in \mathbb{F}^n$

$$c \in \mathcal{C} \Leftrightarrow H \cdot c^T = 0^T$$

A syndrome of a word $y \in \mathbb{F}^n$ is defined as

$$s^T = \begin{pmatrix} s_0 \\ s_1 \\ \vdots \\ s_{d-2} \end{pmatrix} = H \cdot y^T$$

where H is the parity-check matrix of $\mathcal{C}$. If no errors occurred during transmission then the syndrome will contain only zeroes.

2 LAKE Submission

A key drawback of the McEliece cryptosystem is that its public and private keys are both matrices, which require a lot of bits to store. LAKE is a Key Encapsulation Method

(KEM) which addresses this issue by using Ideal Low Rank Parity Check (I-LRPC) codes. In their construction they public and private keys are defined by one irreducible polynomial and three vectors, which is a significant improvement.

The way this key generation is achieved is as follows: a sampled irreducible polynomial $P \in F_q[X]$ defines a quotient ring. Then a subspace F of $\mathbb{F}_{q^m}$ is randomly chosen, as well as two vectors (x, y) from F such that x is invertible in the quotient ring defined by P, and $supp(x, y) = F$. The vectors (x, y) along with P define a parity check matrix of an ideal LRPC code - by taking the polynomials defined by each vector and multiplying them with X of varying degrees, they gain a double circulant generator. The systematic form of this code can then be published along with P, resulting in $pk = (x^{-1}y, P)$. The pair (x, y) is kept as the secret key.

The above paragraph hopefully illustrates why we must first tackle some definitions. We begin by defining the rank metric and LRPC codes, followed by an overview of the key generation and encryption methods. We use the notation provided in [GMRZ13] and [ABD$^+$17].

2.1 The Rank Metric and Support

The field $\mathbb{F}_{q^m}$ can be viewed as an m-dimensional vector space over $\mathbb{F}_q$ with a basis $(\beta_1, \ldots, \beta_m) \in \mathbb{F}_q^m$. We can then express every element $x \in \mathbb{F}_{q^m}$ as a vector $(k_1, \ldots k_n)$ such that $x = \sum_{i=0}^{m} k_i\beta_i$. This is similar to viewing a number as a vector of digits. As an element defines a vector, a vector $v \in \mathbb{F}_{q^m}^n$ defines an $m \times n$ matrix. The rank weight of a vector $rank(v)$ is defined as the rank of the resulting vector, and the rank distance is defined as $rd(x, y) = rank(x - y)$.

Let $x \in (x_1, \ldots, x_n) \in \mathbb{F}_{q^m}^n$ be a vector of rank r. We denote by E the F_q-sub vector space of F_{q^m} generated by $x_1, \ldots, x_n$. The vector space E is called the **support** of x.

2.2 LRPC and key generation

2.2.1 Circulant matrices and polynomial form

To reduce the size of the key, LAKE uses codes whose generator matrices are double circulant. A circulant matrix is a square matrix in which every row is the preceeding row shifted one index to the right, that is of the form

$$M = \begin{pmatrix} m_0 & m_1 & \ldots & m_{n-1} \\ m_{n-1} & m_0 & \ddots & m_{n-2} \\ \vdots & \ddots & \ddots & \vdots \\ m_1 & m_2 & \ldots & m_0 \end{pmatrix}$$

Note that the set of circulant matrices $M_n(\mathbb{F}_{q^m})$ is isomorphic to the set of polynomials with coefficients in $\mathbb{F}_{q^m}$ modulo $X^n - 1$. A double circulant matrix is of the form $M = (A|B)$ where A and B are both circulant. This allows us to efficiently define a generator matrix using only a pair of vectors.

With the link between circulant matrices and polynomial residue fields, we can define a code that is generated by a pair of polynomials (g_1, g_2).

Let $P(X) \in \mathbb{F}_q[X]$ be a polynomial of degree n and $g_1, g_2 \in \mathbb{F}_{q^m}^n$. Let $G_1(X)$ and $G_2(X)$ be the polynomials associated with g_1 and g_2 respectively.

A $[2n, n]_{q^m}$ ideal code $\mathcal{C}$ of generator (g_1, g_2) is the code with the generator matrix

$$G = \left(\begin{array}{cc|cc} G_1(X) & mod\ P & G_2(x) & mod\ P \\ XG_1(X) & mod\ P & XG_2(x) & mod\ P \\ \vdots & & \vdots & \\ X^{n-1}G_1(X) & mod\ P & X^{n-1}G_2(x) & mod\ P \end{array}\right)$$

When switching between the matrix form and the polynomial form, we say that (h_1, h_2) define a parity-check matrix of a code $\mathcal{C}$ if $(H_1^T|H_2^T)$ is a parity check matrix of $\mathcal{C}$ where

$$H_1 = \begin{pmatrix} h_1 & mod\ P \\ Xh_1 & mod\ P \\ \vdots & \\ X^{n-1}h_1 & mod\ P \end{pmatrix}^T \quad H_2 = \begin{pmatrix} h_2 & mod\ P \\ Xh_2 & mod\ P \\ \vdots & \\ X^{n-1}h_2 & mod\ P \end{pmatrix}^T$$

Take note of the transposes. Additionally, if we have a syndrome in matrix form $\sigma^T = (H_1|H_2)(e_1|e_2)^T$ then this is defined in polynomial form as $\sigma = e_1h_1 + e_2h_2\ mod\ P$. If g_1 is invertible, then under systematic form the code can be defined as

$$\mathcal{C} = \{(x, xg), x \in \mathbb{F}_{q^m}^n\}, g = g_1^{-1}g_2\ mod\ P$$

2.2.2 Ideal-LRPC codes

From these types of ideal codes, we use a subset called ideal LRPC codes. Let F be a $\mathbf{F}_q$-subspace of dimension d of $\mathbb{F}_{q^m}$, (h_1, h_2) two vectors of $\mathbb{F}_{q^m}^n$ of support F and $P \in \mathbb{F}_q[X]$ a polynomial of degree n. A code $\mathcal{C}$ with parity check matrix $H = (H_1|H_2)$ (H_1, H_2 defined same as above) is an ideal LRPC code of type $[2n, n]$.

A key feature of this definition is the subspace F. The LAKE protocol relies on the fact that when encoding a couple of vectors (e_1, e_2) such that $Supp(e_1, e_2) = E$ where E is a random subspace of $\mathbb{F}_{q^m}$ of some dimension r, the result has a product subspace of $P =< E.F >$. Knowing the whole space P and the space F allows recovery of the space E, which is then taken as shared key material.

2.2.3 LAKE Key generation

The key generation process of LAKE is as follows

- Choose irreducible polynomial $P \in \mathbb{F}_q[X]$ of degree n.
- Choose a subspace F of $\mathbb{F}_{q^m}$ with dimension d uniformly at random and sample the vectors $(x, y) \xleftarrow{\$} F^n \times F^n$ such that x is invertible mod P with $Supp(x, y) = F$.
- Compute $h = x^{-1}y \ mod \ P$. Recall that this defines the systematic version of the parity-check matrix of an ideal code. In a subsequent section we will discuss how this provides indistiguishability.
- $pk = (h, P)$ and $sk = (x, y)$

2.2.4 LAKE Key Encapsulation

To encapsulate a shared key, a subspace E of $\mathbb{F}_{q^m}$ with dimension r is uniformly chosen. From that subspace two vectors (e_1, e_2) are sampled such that $Supp(e_1, e_2) = E$. The vectors are then encoded, $c = e_1 + e_2 h \ mod \ P$ and sent. The shared key $K := H(E)$ is taken as a hash of E, as the owner of the secret key is able to decode E from the message. It is unclear from the submission of the protocol how one is supposed to hash the subspace. The hash function in the submission is modeled as a random oracle, but in the real world there needs to be an agreed representation for the subspace that could then be hashed.

The owner of the secret key can uncover E from c using an efficient constant-time decoding algorithm. The description of the decoding algorithm is beyond the scope of this work.

2.3 Hardness

LAKE requires that an adversary can't recover the subspace E efficiently without knowledge of F. The argument why this could be hard comes from the Rank Syndrom Decoding (RSD) problem, which states that given a matrix H, a syndrome σ and a weight ω it is hard to sample a vector x with weight lower that ω such that $Hx^T = \sigma^T$. This problem has been studied with some positive results - namely if there exists a ZPP (Zero-error probabilistic polynomial-time) algorithm for solving RSD, then ZPP=NP [GZ16].

The version of the above problem for ideal codes can be shown to be equivalent to another problem called the Ideal-Rank Support Recovery problem. The problem states that given a vector $h \in \mathbb{F}_{q^m}$, a polynomial $P \in \mathbb{F}_q[X]$ of degree n, a syndrome σ and a weight ω, it is hard to recover the support E of dimension lower than ω such that $e_1 + e_2 h = \sigma \ mod \ P$ where e_1 and e_2 are sampled from E. Due to the simplicity of the scheme this problem almost directly corresponds to the LAKE protocol.

2.4 Indistinguishability

In the key generation section we stated that the systematic version of the parity-check matrix provides indistinguishability. The authors of LAKE don't seem to have reduced this to any proven hard problem, but instead simply claim that given P and the systematic form h it is hard to determine whether h is an ideal LRPC code of weight d or a random ideal code.

2.5 Performance

Below is a table comparing different security levels of the LAKE key encapsulation method and the Classic McEliece cryptosystem, another NIST submission which is the Niederreiter dual version of the McEliece system with emphasis on long-term security. The data is taken from the first round of performance benchmarks of the NIST submissions [oST18]. While McEliece has benefits in its very high encapsulation/decapsulation speed and small ciphertext size, these slight gains are offset by the vast difference in both public and private key sizes.

Name	Keygen cycles	EC	DC	SK	PK	CT
lake1	2561456	448068	1945452	40	423	423
lake2	2918264	478172	3158524	40	636	636
lake3	2768236	530908	4328524	40	826	826
mceliece6960119	839556968	174276	321580	13908	1047319	226
mceliece839556968	1198956300	185368	342640	14080	1357824	240

EC	Encapsulation cycles
DC	Decapsulation cycles
SK	Secret key size (bytes)
PK	Public key size (bytes)
CT	Ciphertext size (bytes)

3 Summary

Coding theory is currently one of the best ways to achieve post-quantum security, second only to perhaps lattices. This report detailed the design of a modern coding theory based scheme, LAKE, and outlined its strengths compared to the classic McEliece cryptosystem.

References

[ABD+17] Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich, and Gilles

Zemor. Lake. Technical report, National Institute of Standards and Technology, 2017. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[BCL+17] Daniel J. Bernstein, Tung Chou, Tanja Lange, Ingo von Maurich, Rafael Misoczki, Ruben Niederhagen, Edoardo Persichetti, Christiane Peters, Peter Schwabe, Nicolas Sendrier, Jakub Szefer, and Wen Wang. Classic mceliece. Technical report, National Institute of Standards and Technology, 2017. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[cod] Lecture notes of introduction to coding theory. available at https://courses.cs.ut.ee/MTAT.05.082/2014_spring/uploads/Main/lecture-notes.

[GMRZ13] Philippe Gaborit, Gaetan Murat, Olivier Ruatta, and Gilles Zemor. Low Rank Parity Check codes and their application to cryptography. 04 2013.

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

[McE78] Robert J. McEliece. A public-key cryptosystem based on algebraic coding theory. Technical report, NASA, 1978.

[oST18] National Institute of Science and Technology. Supercop benchmarks, October 2018. available at https://bench.cr.yp.to/results-kem.html#amd64-titan0.