Criptosistema McEliece.

Mario Muñoz Mesa
Adrián Rodríguez Montero
Carlota Valdivia Manzano

15 de diciembre de 2021

Índice

1. Introducción.

El criptosistema de McEliece fue presentado en 1978 por Robert McEliece como un criptosistema de clave pública basado en códigos correctores de errores. No ha sido un criptosistema muy popular, sin embargo no es susceptible de posibles ataques con el Algoritmo de Shor (algoritmo cuántico que permite encontrar la factorización en primos de un entero en tiempo polinomial, esto comprometería la seguridad de criptosistemas como RSA).

El criptosistema McEliece se ha mantenido seguro a pesar de múltiples ataques publicados por más de 40 años. Actualmente *Classic McEliece* se encuentra en la tercera ronda de NIST Post-Quantum Encryption.

Presentamos una breve visión general del criptosistema original así como las mínimas nociones de Teoría de Códigos que lo sustentan.

2. Nociones previas.

Como sabemos, un alfabeto $\mathcal{A}$ no es más que un conjunto finito no vacío de símbolos, y llamamos palabra a cada expresión construida con símbolos del alfabeto.

Definición 1. Llamamos bloque de código a un conjunto de palabras de longitud n. Esto es, se dice que C es un bloque de código de longitud n si es un subconjunto de $\mathcal{A}^n$. A cada elemento del bloque se le llamará palabra de código.

En Teoría de Códigos se pretende enviar un mensaje correctamente por un canal en el que puede haber ruido.

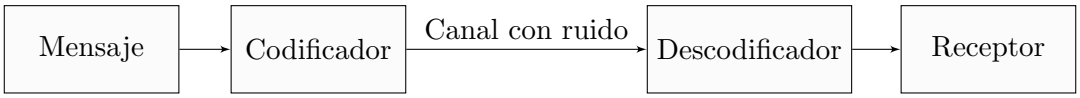

Figura 1: Transmisión a través de un canal con ruido.

Para ello el mensaje m de longitud k se transforma a una palabra de código de longitud n, añadiendo $n-k$ símbolos redundantes, se hace en el proceso de codificación. Esto permitirá que cuando se reciba el mensaje enviado con ciertos errores podamos recuperar el mensaje original mediante el proceso conocido como descodificación.

En adelante trabajaremos con el alfabeto $\mathbb{F}_q$, donde $\mathbb{F}_q$ denota un cuerpo finito de q elementos. Recordamos que todo cuerpo finito tiene número de elementos exponente de un primo, por lo que $q = p^m$, $m \in \mathbb{N}$

Definición 2. Diremos que un bloque de código C es corrector de t errores si cuando tomamos una palabra de código del mismo y cambiamos hasta un máximo de t símbolos, no alcanzamos una palabra de código diferente y tampoco alcanzamos una palabra que pueda ser obtenida a partir de una palabra de código cambiando un máximo de t símbolos.

Definición 3. La distancia de Hamming entre dos vectores de $\mathbb{F}_q^n$ se denota por $d_H(x,y) :=$ "número de componentes en las que difieren x e y"$= |\{i \in \{1,\dots,n\} \ : \ x_i \neq y_i\}|$. El peso de Hamming $wt(x)$ de $x \in \mathbb{F}_q^n$ es el número de coordenadas distintas de 0. La relación entre distancia y peso es $wt(x) = d_H(x,0)$. La distancia de Hamming de un bloque de código C de longitud n se define como $d_H(C) := \text{mín}\{d_H(x,y)\ \forall x,y \in C \text{ con } x \neq y\}$

Se le dice distancia pues $(\mathbb{F}_q^n, d_H)$ forma un espacio métrico.

Nos centramos en los bloques de código de longitud n sobre $\mathbb{F}_q$, esto es, con $C \subseteq \mathbb{F}_q^n$

Sabemos que $\mathbb{F}_q^n$ es un $\mathbb{F}_q$-espacio vectorial de dimensión n con la suma y la multiplicación por escalar habitual. Nos fijaremos en aquellos subconjuntos de $\mathbb{F}_q^n$ con este tipo de estructura.

Definición 4. Sea $C \subseteq \mathbb{F}_q^n$, se dice que C es un código lineal si C es un subespacio vectorial de $\mathbb{F}_q^n$. Si C es un subespacio k-dimensional de $\mathbb{F}_q^n$ también nos referimos a C como un código $[n, k]$. Si además la distancia mínima de Hamming $d_H(C) = d$, nos referiremos como un código $[n, k, d]$

Valery Denisovich Goppa, matemático ruso, presenta en 1970 los códigos Goppa llamados así en su nombre. Estos códigos, en el caso binario, tienen un algoritmo de decodificación eficiente presentado por N. Patterson en 1975.

Definición 5. Sea $g(x) = g_0+g_1x+g_2x^2+\cdots+g_tx^t \in \mathbb{F}_{p^m}[x]$ con p primo, y sea $L = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in \mathbb{F}_{p^m}^n$ tal que $g(\alpha_i) \neq 0 \quad \forall \alpha_i \in L$. Llamaremos código de Goppa correspondiente a L y $g(x)$, denotado por $\Gamma(L, g)$, al código corrector de errores formado por

$$\left\{ \mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{F}_p^n \ : \ \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \text{ mód } g(x) \right\}$$

Cuando $p = 2$ decimos que se trata de un código Goppa binario y lo denotamos por $\Gamma_2(L, g)$.

Nota: si el polinomio $g(x)$ es irreducible entonces el código de Goppa $\Gamma(L, g)$ se dice que es irreducible, y se verifica $g(\alpha_i) \neq 0$ ya que si α_i fuera raíz de $g(x)$ entonces $g(x)$ ya no sería irreducible.

Proposición 2.1. Sea $\Gamma(L, g)$ un código Goppa, entonces $\Gamma(L, g)$ es un código lineal.

Que sea un código lineal implica que como subespacio vectorial tiene una base, y supone una ventaja pues podremos codificar de forma eficiente como veremos a continuación.

Definición 6. Sea C un código lineal $[n, k]$ sobre $\mathbb{F}_q$. Se llama matriz generadora de C a la matriz $k \times n$ formada por los elementos de la base de C como filas.

Veamos cómo podríamos codificar y descodificar con códigos Goppa:

- **Codificación.** Codificar un mensaje de longitud k, $\mathbf{m} = (m_1, \ldots, m_k) \in \mathbb{F}_q^k$, simplemente consiste en multiplicar por la matriz generadora G, la codificación del mensaje será $\mathbf{m} \cdot G$. Mensajes de mayor longitud se particionan en bloques de longitud k previamente.
- **Descodificación.** Sea $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ el vector recibido con r errores

$$\mathbf{y} = (y_1, y_2, \ldots, y_n) = \underbrace{(c_1, c_2, \ldots, c_n)}_{\text{palabra de código}} + \underbrace{(e_1, e_2, \ldots, e_n)}_{\text{vector error}}$$

  con $e_i \neq 0$ en exactamente r posiciones. Para descodificar, si el código Goppa es binario, podemos aplicar el algoritmo de Patterson, que dado $\mathbf{y}$ nos devolverá el vector de error. Por lo que ya podemos recuperar el mensaje enviado por la relación: $(m_1, m_2, \ldots, m_k) \cdot G = (c_1, c_2, \ldots, c_n)$

3. Intercambio de mensajes.

En esta sección se va a explicar el procedimiento de comunicación entre dos personas mediante el criptosistema de McEliece. Se explicará tanto el cifrado de un mensaje por parte de un emisor, como su correspondiente descifrado por parte de un receptor.

Antes de comenzar, cabe recordar que en la Criptografía de Clave Pública todos los usuarios tienen una clave pública y una privada. Como sus nombres indican la clave pública está al alcance del resto de usuarios, mientras que la clave privada solo la posee el propio usuario en cuestión.

En primer lugar, partimos del supuesto en el que un emisor quiere enviar un mensaje confidencial a un receptor. Para llevar a cabo dicho procedimiento, el emisor utilizará la clave pública del receptor para cifrar el mensaje, de forma que solamente él, con su respectiva clave privada, pueda descifrarlo.

Así pues, si un tercer usuario intercepta el mensaje podemos asumir que no puede descifrarlo al no poseer la clave privada del receptor.

A continuación, veremos como se generan las claves públicas y privadas en el criptosistema McEliece, y la relación que existe entre ellas, para ello vamos a hacer uso de códigos Goppa explicados previamente.

Para obtener las claves privadas y públicas es necesaria la generación de tres matrices:

- Una matriz G en $\mathbb{F}_2^{k\times n}$, que va a ser la matriz generadora de un código de Goppa binario [n,k] elegido. Dicho código es capaz de corregir de manera eficiente hasta t errores para algún parámetro t.
- Una matriz invertible aleatoria S en $\mathbb{F}_2^{k\times k}$.
- Una matriz de permutaciones aleatoria P en $\mathbb{F}_2^{n\times n}$, es decir, una matriz con todos sus elementos nulos salvo un elemento por cada fila y columna que vale 1, de forma que cuando multipliques P por un vector permute sus coordenadas.

A partir de estas tres matrices se puede obtener la clave pública y privada de un usuario de la siguiente manera:

- **Clave privada.** La clave privada se corresponde con la terna formada por las tres matrices definidas anteriormente: $k_s = (G, S, P)$.
- **Clave pública.** La clave pública es el par formado por el producto resultante de los elementos de la terna de la clave privada $G' = S \cdot G \cdot P$ y el parámetro t: $k_p = (G', t)$

Ahora veremos como funciona realmente la comunicación entre el emisor y el receptor:

- **Cifrado de un mensaje:** Para llevar a cabo el cifrado de un mensaje $\mathbf{m} = (m_1, \ldots, m_k) \in \mathbb{F}_2^k$ dirigido a un receptor con clave pública $k_p = (G', t)$, el emisor necesita:

  1. Elegir un vector aleatorio de errores $\mathbf{e} = (e_1, \ldots, e_k) \in \mathbb{F}_2^n$ con peso $wt(e) = t$.
  2. Multiplicar el mensaje m por la matriz G' y obtener un vector de tamaño n: $c = m \cdot G'$
  3. Añadir el vector de errores e al vector c: $c' = m \cdot G' + e$.

  De esta forma el mensaje cifrado que quiere enviar el emisor sería c'.

- **Descifrado de un mensaje:** Para llevar a cabo el descifrado de un mensaje $c' = mG' + e$ recibido por un receptor con clave pública $k_p = (G', t)$, dicho receptor necesita:

  1. En primer lugar, multiplicar dicho vector c' por la matriz inversa de la matriz de permutación, P^{-1}: $\hat{c} = c'P^{-1} = mSGPP^{-1} + eP^{-1} = mSG + eP^{-1}$. Dicho paso va a ser siempre posible, puesto que toda matriz de permutación es regular.
  2. A continuación, usar el algoritmo eficiente de descodificación de t errores, el algoritmo de Patterson, para obtener mSG ya que es el motivo de usar códigos binarios. Dicho algoritmo se puede usar ya que la matriz SG es una matriz generadora del código Goppa, y que tanto el vector e como eP^{-1} tienen el mismo peso: $wt(e) = wt(eP^{-1}) = t$.
  3. Una vez se ha obtenido mSG, puede obtener mS resolviendo el sistema de ecuaciones correspondiente a llevar a cabo el descifrado del código de Goppa.
  4. Finalmente, obtener el código descifrado m a partir de la inversa de la matriz aleatoria S, $m = mSS^{-1}$.

4. Seguridad de McEliece.

La seguridad del criptosistema McEliece consiste en la capacidad de recuperar textos sin formato a partir de textos cifrados utilizando un código oculto de corrección de errores, que el remitente al inicio distorsiona con errores aleatorios.

En esta sección nos centraremos en la seguridad del método original de McEliece. Sea G un código de Goppa binario $\Gamma_2(L, g) \subseteq \mathbb{F}_2^n$, con $g(x) = g_0 + g_1x + g_2x^2 + \cdots + g_tx^t \in \mathbb{F}_{2^m}[x]$ y $L = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in \mathbb{F}_{2^m}^n$, el cual puede corregir a lo sumo t errores. La dimensión del código de Goppa para aplicaciones criptográficas se toma como $k = n - tm$. Sea G una matriz generadora $k \times n$ del código de Goppa y $G' = SGP$ una clave pública del criptosistema McEliece con $k \times k$ matriz de permutación binaria no singular P.

La seguridad del criptosistema de McEliece es consecuencia de los siguientes problemas de decisión de la teoría de códigos:

- Sea C un $[n, k]$ código lineal sobre $\mathbb{F}_q$ y $\alpha \in \mathbb{F}_q^n$. Encontrar una palabra de código $c \in C$ tal que la distancia $d(\alpha, c)$ sea mínima.
- Sea C un $[n, k]$ código lineal sobre un cuerpo $\mathbb{F}_q$ y $w \in \mathbb{N}$. Encontrar una palabra de código $c \in C$ tal que su peso $wt(c) = w$.

Estos dos problemas son NP-Completos, puesto que los mejores algoritmos conocidos para tratar de resolverlos son del orden exponencial. Por tanto, no es posible computacionalmente resolver los problemas tomando valores grandes en los parámetros del código. Como estos dos problemas son NP-Completos el problema de romper el criptosistema de McEliece también es NP-Completo ya que los códigos irreducibles binarios de Goppa solo son una fracción de todos los posibles códigos lineales.

5. Ventajas y Desventajas.

El criptosistema de McEliece tiene como una de sus principales ventajas el rápido proceso de codificación y descodificación, los cuales son más rápidos que los procesos que realizan hoy en día la mayoría de los sistemas más empleados como puede ser RSA. Otra de las ventajas que presenta es que es un criptosistema post-cuántico, es resistente contra ataques realizados por ordenadores cuánticos que emplean el algoritmo Shor.

McEliece también presenta una serie de desventajas aunque son pocas. La más importante es el tamaño de las claves públicas y privadas, ya que estas son muy grandes, puesto que son matrices de elevadas dimensiones. Otra de las desventajas que presenta es que no se puede emplear para producir firmas digitales. Este problema ha sido solucionado mediante el criptosistema dual a McEliece llamado Niederreiter. Este criptosistema usa una matriz de paridad en lugar de una matriz de generación.

Vamos a poner un ejemplo de lo grandes que son las claves públicas y privadas y vamos a ver lo que pasa si aumentamos el tamaño de los parámetros los cuales están relacionados directamente con el tamaño de las claves públicas y privadas. Por tanto, si estos parámetros aumentan su tamaño el tamaño de las claves también se verá aumentado.

En sus inicios, los tamaños de parámetros que sugirió McEliece fueron $n = 1024$, $k = 524$, $t = 50$, los cuales dan lugar a una clave pública de tamaño cercano a 2^{19} bits. En estos años, por el avance tecnológico que se ha llevado a cabo, el tamaño de los parámetros sugeridos ha sido elevado a $n = 2048$, $k = 175$, $t = 27$, así obtenemos 80 bits de seguridad, necesitamos 2^{80} operaciones para romper el criptosistema. Sin embargo, para poder resistir a ataques realizados por ordenadores cuánticos el tamaño de los parámetros debe ser más alto llegando a, $n = 6960$, $k = 5400$, $t = 119$, por tanto, el tamaño de la clave pública es aproximadamente de 2^{23} bits.

6. Posibles Ataques.

Vamos a mostrar de manera muy breve los dos posibles tipos de ataques que se pueden realizar contra el criptosistema de McEliece:

- **Ataques genéricos de descodificación**: consisten en intentar recuperar el mensaje original, $\mathbf{m}$, que ha sido cifrado obteniendo el vector $\mathbf{c}'$ considerando la matriz pública del criptosistema G', como matriz generatriz para dicho código. Los mejores resultados se obtienen utilizando algoritmos que mejoran la descodificación del conjunto de información. Esta descodificación consiste en encontrar k coordenadas del vector $\mathbf{c}'$ que no presenten ningún error. Entonces tomando esas coordenadas del vector $\mathbf{c}'$ y la matriz inversa formada por las k columnas seleccionadas de la matriz G' podemos recurperar el mensaje original $\mathbf{m}$.
- **Ataques contra la estructura del código**: este tipo de ataques consisten en intentar recuperar las matrices G, S y P a partir de las cuales se ha construido el criptosistema de McEliece. Para este tipo de ataques, aquellos criptosistemas de McEliece que emplean códigos Goppa son los que presentan una mayor seguridad.

7. Conclusiones.

Una vez estudiado el criptosistema de McEliece podemos sacar varias conclusiones y hacer un par de apuntes de cara al futuro.

Los sistemas criptográficos más populares de la actualidad, por ejemplo, RSA que es uno de los más utilizados y uno de los vistos en la asignatura serán muy vulnerables y dejarán de ser seguros cuando aparezcan en el mercado los ordenadores cuánticos. Debido a esto, es necesario que estemos preparados, puesto que la seguridad a la hora de transmitir información es de vital importancia para cualquier persona que realiza multitud de comunicaciones empleando dispositivos electrónicos.

La criptografía basada en códigos presenta una posible solución al problema, ya que contiene diferentes familias de códigos y estas pueden ser implementadas en el criptosistema McEliece.

McEliece es un criptosistema resistente a los ataques actuales y también a los ataques realizados por ordenadores cuánticos ya que es un criptosistema post-cuántico pero presenta algunas desventajas con respecto a otros criptosistemas debido al gran tamaño de sus claves, al ser matrices de elevadas dimensiones.

Por todo esto, es importante ir pensando en desarrollar nuevos algoritmos criptográficos más resistentes que los actuales para poder hacer frente a los ordenadores cuánticos aunque puedan tener ciertos puntos débiles en sus inicios como le pasa al criptosistema de McEliece con la eficiencia. Conforme pasen los años y se realicen más investigaciones y avances posiblemente se dará solución a este tipo de errores o vulnerabilidades y a otros que se irán descubriendo obteniendo criptosistemas más seguros y con menos debilidades.

8. Anexo.

Adjuntamos ejemplo de uso del Criptosistema McEliece con Sagemath, basado en el código de "Enlaces y referencias".

9. Enlaces y referencias.

- https://arxiv.org/pdf/1907.12754.pdf
- https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions
- https://ocw.ehu.eus/pluginfile.php/50556/mod_page/content/35/Resumen_total_con_portada.pdf
- https://doc.sagemath.org/html/en/reference/coding/sage/coding/goppa_code.html
- https://uvadoc.uva.es/bitstream/handle/10324/40282/TFG-G4106.pdf
- https://upcommons.upc.edu/bitstream/handle/2117/133124/136200.pdf
- https://github.com/extracru/correction-code-cryptography/blob/master/Ejemplos%20SageMath/algoritmoMcEliece.ipynb
- http://repositori.uji.es/xmlui/bitstream/handle/10234/181824/TFM_Ever_final.pdf
- https://eprint.iacr.org/2009/187.pdf