Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA

Universidad Autónoma Metropolitana
Iztapalapa

Sistemas multivariables de cifrado PHFER y PR como aportación a la industria de la seguridad informática

Tesis que presenta

Susana Hernández Candelario

Para obtener el grado de

Maestra en Ciencias

Matemáticas Aplicadas e Industriales

Vo. Bo

Dr. José Noé Gutiérrez Herrera

Atte.

Susana Hernández Candelario

Ciudad de México

Diciembre 2020

Índice general

1. Introducción **4**
1.1. Funciones de variable vectorial 4
1.2. Una muestra de la criptografía post-cuántica 8
1.3. Criptosistemas de clave pública multivariados 10

2. Criptosistemas **14**
2.1. Sistema Matsumoto e Imai (MI) 14
2.1.1. El ataque de ecuaciones de linealización 18
2.2. Sistema HFE 20
2.2.1. Ataque Algebraico 25
2.2.2. El ataque de Kipnis y Shamir 28
2.3. Sistema Square 29
2.3.1. Criptoanálisis de Square 30
2.4. Sistemas de aceite y vinagre 36
2.4.1. Rainbow 36
2.4.2. Criptoanálisis 39

3. Nuevos modelos de cifrado **40**
3.1. Cifrado PHFER 40
3.2. Seguridad 42
3.3. Cifrado PR 42
3.4. Seguridad 44

4. Conclusiones y perspectivas **45**

A. Construcción de un campo finito **47**

B. Algoritmo Berlekamp **50**

C. Criptoanálisis de Rainbow **52**

Resumen

En septiembre del 2019, investigadores de Google lograron a través de su ordenador cuántico realizar una tarea en 3.2 minutos, cuyo trabajo costaría 10000 años con la súper computadora Submit. Empresas reconocidas mundialmente, IBM y Google, están invirtiendo grandes cantidades de dinero en el desarrollo de computadoras cuánticas ya que para el 2025 se completará el proceso estandarización y comenzará una migración a nuevos sistemas criptográficos. Por lo que estamos a muy poco tiempo de que las computadoras cuánticas estén a nuestro completo alcance y con ello romper varios de los sistemas que se consideran resistentes hasta el momento.

En el proceso de desarrollar sistemas criptográficos post-cuánticos seguros, parte de la criptografía Post-cuántica se encuentra dedicada al desarrollo y análisis de sistemas criptográficos multivariables, cuya construcción es a partir de polinomios multivariados que en su mayoría resultan ser cuadráticos. Este tipo de sistemas se consideran seguros ya que el problema de resolver un sistema de ecuaciones polinómicas multivariadas sobre un campo finito resulta ser un problema **NP-difícil**. Con esto en mente, la presente tesis tiene como objetivo desarrollar dos nuevos modelos que sean resistentes a los ataques criptográficos que se conocen hasta el momento. A lo largo del capítulo 1 se da una introducción a las funciones de variable vectorial y se muestra uno de los resultados más importantes para los sistemas criptográficos multivariables, el cual garantiza que los polinomios que definen la clave pública bastan ser de grado a lo más dos.

Los pioneros en el desarrollo de la criptografía multivariable basada en campos finitos fueron MatsumotO e Imai, quienes desarrollaron el sistema MI sobre el campo $\mathbb{F}_2$. Este sistema que se desarrolla en el capítulo 2, proporciona distintas herramientas para el desarrollo de otros sistemas seguros que se estudian en ese mismo capítulo: HFE, Rainbow y Square+. La clave pública para MI y para los otros 3 sistemas tiene la forma

$$P(m) = L_1 \circ \varphi \circ F \circ \varphi^{-1} \circ L_2(m)$$

donde L_i son transformaciones afines invertibles cuya función es ocultar la función central F, y φ es un isomorfismo de espacios vectoriales.

A lo largo del capítulo 3 se muestra el desarrollo y la implementación de dos nuevos sistemas criptográficos, PR y PHFER, los cuales se definen a partir de 3 de los sistemas criptográficos más seguros hasta el momento: Rainbow, Square+ y HFE. Se analiza el tiempo de ejecución de cada uno de ellos y estudia la seguridad de cada sistema a partir de la seguridad de cada una de sus capas.

Capítulo 1

Introducción

1.1. Funciones de variable vectorial

Sea $\mathbb{F}_q$ un campo finito de q elementos con q primo, cuya construcción se presenta en el Apéndice A. Con el siguiente lema se tiene la certeza de que todas las funciones sobre un campo finito serán polinomiales [1].

Lema 1.1.1. Sean n un número entero positivo y f una función de $\mathbb{F}_q^n$ en $\mathbb{F}_q$. Entonces f se expresa mediante un polinomio, es decir, f es polinomial.

Demostración Para cada $\bar{w} = (w_1, \ldots, w_n) \in \mathbb{F}_q^n$, sea $\delta_{\bar{w}} : \mathbb{F}_q^n \to \{0, 1\}$ definida como:

$$\delta_{\bar{w}}(\bar{x}) = \prod_{i=1}^{n} [1 - (x_i - w_i)^{q-1}].$$

Dado que cualquier elemento distinto de cero en $v \in \mathbb{F}_q$ satisface $v^{q-1} = 1$, entonces $1 - (x_i - w_i)^{q-1}$ es cero a menos que $x_i = w_i$ para todo i. $\delta_{\bar{w}}(\bar{x})$ también se puede escribir como

$$\delta_{\bar{w}}(\bar{x}) = \begin{cases} 0, \text{ si } \bar{x} \neq \bar{w} \\ 1, \text{ si } \bar{x} = \bar{w}. \end{cases}$$

Por otro lado, cualquier función $f : \mathbb{F}_q^n \to \mathbb{F}_q$, satisface

$$f(\bar{x}) = \sum_{\bar{a} \in \mathbb{F}_q^n} f(\bar{a}) \delta_{\bar{a}}(\bar{x}).$$

Así f es una función polinomial. □

Ejemplo 1.1.1. Se define $f : \mathbb{F}_2^3 \to \mathbb{F}_2$ de la siguiente forma

x	000	001	010	100	011	101	110	111
$f(x)$	1	1	0	1	0	1	1	0

Para hallar la expresión de f como un polinomio de tres variables, primero se calcula $\delta_{\bar{w}}(\bar{x})$ con

$\bar{w} \in \mathbb{F}_2^3$, cuyos valores resultan:

$$\begin{aligned}
\delta_{(0,0,0)}(x_1,x_2,x_3) &= x_1x_2x_3 + x_1x_2 + x_1x_3 + x_1 + x_2x_3 + x_2 + x_3 + 1\\
\delta_{(0,0,1)}(x_1,x_2,x_3) &= x_3 + x_1x_3 + x_2x_3 + x_1x_2x_3\\
\delta_{(0,1,0)}(x_1,x_2,x_3) &= x_2 + x_1x_2 + x_2x_3 + x_1x_2x_3\\
\delta_{(1,0,0)}(x_1,x_2,x_3) &= x_1 + x_1x_2 + x_1x_3 + x_1x_2x_3\\
\delta_{(0,1,1)}(x_1,x_2,x_3) &= x_2x_3 + x_1x_2x_3\\
\delta_{(1,0,1)}(x_1,x_2,x_3) &= x_1x_3 + x_1x_2x_3\\
\delta_{(1,1,0)}(x_1,x_2,x_3) &= x_1x_2 + x_1x_2x_3\\
\delta_{(1,1,1)}(x_1,x_2,x_3) &= x_1x_2x_3
\end{aligned}$$

De tal forma que

$$\begin{aligned}
f(\bar{x}) &= f(0,0,0)\delta_{(0,0,0)}(\bar{x}) + f(0,0,1)\delta_{(0,0,1)}(\bar{x}) + \cdots + f(1,1,1)\delta_{(1,1,1)}(\bar{x})\\
&= 1 + x_2 + x_1x_2 + x_1x_2x_3.
\end{aligned}$$

Definición 1.1.1. El *polinomio simétrico homogéneo de grado* d en n variables $x_1, x_2, \ldots, x_n$, que se escribe como h_d con $d = 0, 1, \ldots$, es la suma de todos los monomios de grado d en n variables.

$$h_d(x_1, x_2, \ldots, x_n) = \sum_{l_1+l_2+\cdots+l_n=d} x_1^{l_1} x_2^{l_2} \cdots x_n^{l_n}.$$

Ejemplo 1.1.2. Los polinomios de grado 0, 1, 2 en n variables respectivamente son:

$$\begin{aligned}
h_0(x_1, \ldots, x_n) &= 1,\\
h_1(x_1, \ldots, x_n) &= \sum_{1\le j\le n} x_j,\\
h_2(x_1, \ldots, x_n) &= \sum_{1\le j\le k\le n} x_j x_k.
\end{aligned}$$

Proposición 1.1.1. Para un n fijo, el número de términos posibles en un polinomio de grado d en n variables es

$$\binom{n+d}{d} = \frac{(n+d)!}{n!\ d!}.$$

Demostración. Para esta demostración, basta contar el número de monomios de grado menor o igual a d en un polinomio de grado d en n variables.

Por un lado el número total de polinomios simétricos homogéneos $h_i(x_1, \ldots, x_n)$ es $\binom{n+(i-1)}{i}$. El número total de polinomios simétricos homogéneos h_i con $0 \le i \le d$ resulta

$$1 + \binom{n}{1} + \binom{n+1}{2} + \cdots + \binom{n+(d-2)}{d-1} + \binom{n+(d-1)}{d}$$

y por otro lado, desarrollando $\binom{n+d}{d}$ usando la propiedad $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$, se obtiene

$$\binom{n+d}{d} = 1 + \binom{n}{1} + \binom{n+1}{2} + \cdots + \binom{n+(d-2)}{d-1} + \binom{n+(d-1)}{d}.$$

□

Estaremos trabajando con criptosistemas de clave pública multivariable, que se definen mediante funciones polinomiales multivariadas sobre un campo finito k. En estos sistemas, el emisor pondrá a disposición general una clave que sirve únicamente para cifrar (clave pública) y, por otro lado, mantendrá una segunda clave en secreto que usará para descifrar (clave secreta). El lema 1.1.1 permite al emisor definir ambas claves mediante funciones polinomiales cuyo grado deberá elegirse de tal forma que el número de términos en la clave pública no sea muy grande.

Para ilustrar como crece el número de monomios en la clave pública a medida que d crece, vea la siguiente tabla:

d	n	$\binom{n+d}{d}$
2	10	66
	15	136
	20	231
3	10	286
	15	816
	20	1771
4	10	1001
	15	3876
	20	10626
5	10	3003
	15	15504
	20	53130

Tabla 1.1: El número total de términos crece rápidamente cuando d crece. Si d es grande, entonces el tamaño de la clave pública será tan grande que impedirá un cálculo rápido y almacenamiento eficiente. Así, es suficiente trabajar con claves públicas polinomiales de grado a lo mas dos.

Usualmente la clave pública de un sistema criptográfico multivariable está formada por un conjunto de polinomios de grado dos, todos los polinomios cuadráticos son de la forma:

$$\sum_{i \leq j} a_{ij} x_i x_j + \sum b_i x_i + c.$$

Si al pasar de los años llegamos a enfrentar a las computadoras cuánticas, esto nos asegurará que algunos sistemas criptográficos serán definitivamente rotos, por ejemplo, RSA y DSA. Ello nos motiva a continuar con la búsqueda de sistemas criptográficos que no sólo resistan los ataques de computadoras clásicas sino, que a su vez, sean resistentes a esta nueva generación de computadoras. Cabe mencionar que no hay una forma específica de saber si un sistema criptográfico es seguro o no. Esta decisión se toma cuando dicho sistema no se puede romper al implementarle distintos ataques criptográficos, en ocasiones a los criptoanalistas les puede llevar años para creer que el sistema es resistente a dichos ataques. Así se llega a la conclusión de que aún no hay poder computacional que logre destruir el sistema criptográfico. En general, actualmente un sistema criptográfico se considera seguro cuando el mejor algoritmo conocido capaz de atacarlo requiere al menos 2^{80} operaciones para su ejecución.

Criptosistema RSA

Para el criptosistema *RSA* [9](primer sistema de cifrado de clave pública), se considera $n = pq$, donde p y q son primos distintos, se elige un exponente público e el cual es primo relativo con $p-1$ y $q-1$, y

entonces se calcula el exponente privado d como el inverso del exponente público módulo $\varphi(n)$, donde $\varphi(n)$ es la función de Euler. La clave pública es la pareja (e, n) y la clave privada (d, p, q). Dado el texto en claro m, con $0 \leq m < n$, el texto cifrado se calcula mediante $c = m^e \bmod n$, mientras que para descifrar se evalúa $m = c^d \bmod n$.

La seguridad del criptosistema RSA, recae en el exponente privado d de la pareja (e, n) que es pública. Si se logra encontrar la factorización de $n = pq$, entonces se puede calcular la clave privada (d, p, q) para cualquier clave pública (e, n). Consideremos la función ϕ de Euler $\phi(n) = (p-1)(q-1)$. Si se conoce n y $\phi(n)$, se puede factorizar n resolviendo el sistema de ecuaciones

$$\begin{aligned} n &= pq \\ \phi(n) &= (p-1)(q-1), \end{aligned}$$

para los primos p y q. Por un lado buscamos enteros p y q tales que

$$\begin{aligned} \phi(n) &= (p-1)(q-1) \\ &= pq - p - q + 1 \\ &= n - (p+q) + 1 \end{aligned}$$

por lo que $p + q = n - \phi(n) + 1$, luego p y q son raíces de la ecuación cuadrática

$$x^2 - [n - \phi(n) + 1]x + n = 0$$

obteniendo la factorización de n, con lo que se habrá roto el criptosistema RSA.

Cifrado ElGamal

El sistema de cifrado ElGamal fue descrito por Taher ElGamal en 1985 [21]. Es un algoritmo de cifrado de clave pública. Puede ser definido sobre cualquier grupo cíclico G, sin embargo en este apartado lo definiremos sobre $(\mathbb{Z}_p^*, \cdot)$, el grupo multiplicativo de los enteros módulo p. Su seguridad está basada en el *problema del logaritmo discreto* sobre $(\mathbb{Z}_p^*, \cdot)$, ya que hallar logaritmos discretos es difícil, pero la operación inversa (exponenciación) puede ser calculada eficientemente en tiempo polinomial.

Descripción

Sea $(G, \cdot)$ un grupo multiplicativo finito. Para un elemento $\alpha \in G$ de orden n, se define el *subgrupo cíclico de orden n*:

$$\langle \alpha \rangle = \{\alpha^i : 0 \leq i \leq n-1\}.$$

Definición 1.1.2. Dados un subgrupo $\langle \alpha \rangle$ de orden n de un grupo $(G, \cdot)$ y $\beta \in \langle \alpha \rangle$, el *logaritmo discreto de β en base α* es un único entero a con $0 \leq a \leq n-1$, tal que $\alpha^a = \beta$. Se denota a mediante $\log_\alpha \beta$.

A continuación se describe el sistema ElGamal.

Criptosistema ElGamal en $\mathbb{Z}_p^*$

Sea p primo tal que el *problema de logaritmo discreto* en $(\mathbb{Z}_p^*, \cdot)$ no se resuelve en tiempo polinomial, y sea $\alpha \in \mathbb{Z}_p^*$ un elemento primitivo. Considere $\mathcal{P} = \mathbb{Z}_p^*$ el conjunto de textos en claro y $\mathcal{C} = \mathbb{Z}_p^* \times \mathbb{Z}_p^*$ el conjunto de posibles textos cifrados. Se define

$$\mathcal{K} = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a (\text{ mod p})\},$$

el conjunto de llaves posibles. La clave pública son valores de p, α y β, la clave privada es el valor a. Para $K = (p, \alpha, a, \beta)$, y para un número secreto aleatorio $y \in \mathbb{Z}_{p-1}$, se define

$$e_K(x, y) = (y_1, y_2), \text{ dicha pareja representa el mensaje cifrado,}$$

donde

$$y_1 = \alpha^y \bmod p$$

y

$$y_2 = x\beta^y \bmod p.$$

Para $y_1, y_2 \in \mathbb{Z}_p^*$, se define

$$d_K(y_1, y_2) = y_2(y_1^a)^{-1} \bmod p.$$

Ejemplo 1.1.3. Sean $(197, 2, 82)$ la clave pública y $a = 111$ la clave privada de un usuario que llamaremos Bob para el criptosistema ElGamal. Él da a conocer su clave pública y se conserva el valor de a en privado. Supongamos que Alicia desea enviar a Bob el mensaje $x = 123$, eligiendo $y = 191$, entonces Alicia envía la pareja $e_K(123, 191) = (y_1, y_2)$, donde

$$y_1 = 2^{191} \bmod 197 = 117$$
$$y_2 = 123(82^{191}) \bmod 197 = 175$$

Para descifrar, Bob calcula $d_K(117, 175)$ mediante

$$d_K(117, 175) = 175(2^{191})^{-1} \bmod 197 = 123.$$

Si un atacante logra calcular $a = \log_\alpha \beta$, entonces podrá descifrar textos cifrados de la misma forma que Bob. Por lo que una condición necesaria para que el criptosistema ElGamal sea seguro es que el *problema de logaritmo discreto* en $\mathbb{Z}_p^*$ no sea soluble. De hecho, no se conoce un algoritmo que resuelva el *problema de logaritmo discreto* en tiempo polinomial. Para la resistencia del sistema ElGamal se pide que p tenga al menos 300 dígitos y $p - 1$ tenga al menos un factor primo grande.
Una aplicación que resulta factible del sistema ElGamal es sobre las curvas elípticas, ya que si el grupo G es el conjunto de puntos racionales de una curva elíptica, el sistema de ElGamal requiere longitudes de clave considerablemente más pequeñas para tener la misma seguridad que ElGamal sobre $\mathbb{Z}_p^*$ o RSA.

1.2. Una muestra de la criptografía post-cuántica

Existen sistemas criptográficos que han llegado a ser muy difíciles de romper, incluso si se tuviera a la mano una computadora cuántica. Esta resistencia es primordial para que sea un sistema factible. Un ejemplo de éstos es un sistema de cifrado de clave pública el cual es parametrizado por el valor de b, que nos indica el nivel de seguridad deseado por el usuario, aunque es importante mencionar que existen más parámetros a considerar los cuales permiten un cifrado, descifrado, firma y verificación más rápidos con claves más pequeñas y firmas más pequeñas.

Un sistema de firma de clave pública basado en hash

Este sistema requiere de una función hash, la cual producirá $2b$ bits de salida. Brevemente, una función hash es una función H que transforma los datos de entrada de tamaño arbitrario a un resultado de tamaño fijo (por ejemplo, 256 bits), que se denomina valor hash. Ejemplo de tales funciones son SHA-256 y SHA3-256, que transforman la entrada en salida de 256 bits.
Por ejemplo, para b = 128 se podría elegir la función hash a SHA-256, la clave pública del firmante tiene $8b^2$ bits, es decir: 16 kilobytes para $b = 128$. La llave consiste de $4b$ cadenas

$$y_1[0], y_1[1], y_2[0], y_2[1], \ldots, y_{2b}[0], y_{2b}[1],$$

donde cada cadena consiste de $2b$ bits.

La firma de un mensaje m tiene $2b(2b + 1)$ bits, es decir: 8 kilobytes para $b = 128$. La firma consiste de cadenas con $2b$-bits $r, x_1, \ldots, x_{2b}$ tales que los bits $(h_1, \ldots, h_{2b})$ de $H(r, m)$ satisfacen $y_1[h_1] = H(x_1), y_2[h_2] = H(x_2), \ldots, y_{2b}[h_{2b}] = H(x_{2b})$, donde

$$H : \{ \text{ cadena de } r \text{ bits}\} \times \{mensajes\} \to \{1, 2, \ldots, 2^K\}.$$

Por ejemplo, para $b = 0$ y $K = 128$, la función H asigna un elemento de $\{1, 2, \ldots, 2^K\}$ a cada mensaje. La cuestión ahora es dado el valor $H(x) = y$, cómo se encuentra el valor de x. Para esto el firmante inicia generando un secreto x y se calcula $y = H(x)$. La llave secreta del firmante tiene $8b^2$ bits, a saber, $4b$ cadenas aleatorias uniformemente independientes $x_1[0], x_1[1], x_2[0], x_2[1], \ldots, x_{2b}[0], x_{2b}[1]$, cada cadena tiene $2b$ bits. El firmante calcula la llave pública $y_1[0], y_1[1], y_2[0], y_2[1], \ldots, y_{2b}[0], y_{2b}[1]$ como

$$H(x_1[0]), H(x_1[1]), H(x_2[0]), H(x_2[1]), \ldots, H(x_{2b}[0]), H(x_{2b}[1])$$

Para firmar un mensaje m, el firmante genera una cadena aleatoria uniforme r, calcula los bits $(h_1, \ldots, h_{2b})$ de $H(r, m)$, y resulta $(r, x_1[h_1], \ldots, x_{2b}[h_{2b}])$ como una firma de m. El firmante descarta los valores de x restantes y se niega a firmar más mensajes (sistema de firma única).
Si el firmante deseara firmar más de un mensaje, lo que se hace es incluir el mensaje firmado y una nueva clave que se usará para firmar el siguiente mensaje, así lo que se hará es, verificar el primer mensaje firmado y la nueva clave pública, y después verificar la firma del siguiente mensaje. Siguiendo este procedimiento la firma del enésimo mensaje incluirá todos $n - 1$ mensajes firmados anteriores.

Un sistema de firma de clave pública cuadrática multivariable

La clave pública en este sistema es una sucesión de $2b$ polinomios en $4b$ variables con coeficientes en $\mathbb{F}_2$, es decir $P_1, P_2, \ldots, P_{2b} \in F_2[w_1, \ldots, w_{4b}]$. Cada polinomio tiene grado a lo más 2, donde además no tiene términos cuadráticos, y es representado como una secuencia de $1+4b+4b(4b-1)/2$ bits, a saber, los coeficientes de $1, w_1, \ldots, w_{4b}, w_1w_2, w_1w_3, \ldots, w_{4b-1}w_{4b}$. La clave pública tiene $16b^3+4b^2+2b$ bits. La firma de un mensaje m tiene $6b$ bits, a saber, $4b$ valores $w_1, \ldots, w_{4b} \in F_2$ y una cadena r de $2b$-bits que satisface

$$H(r, m) = (P_1(w_1, \ldots, w_{4b}), \ldots, P_{2b}(w_1, \ldots, w_{4b})).$$

Donde H es una función hash estándar. Verificar una firma hace uso de una evaluación de H y aproximadamente b^3 operaciones de bits para evaluar $P_1, P_2, \ldots, P_{2b}$.
Una importante ventaja de este sistema de firma con respecto al sistema de firma basada en hash es el tamaño de firma, el cual es más corto. Otros sistemas cuadráticos multivariables suelen tener firmas aún más pequeñas, y en muchos casos, claves públicas aún más pequeñas.

Desafíos en la criptografía post-cuántica

A pesar de que aún no se ha construido una computadora cuántica viable, una de las razones por las cuales debemos de concentrarnos en la criptografía post-cuántica es que si en unos años nos enfrentamos a la criptografía cuántica, se habrán perdido tantos años de investigación.
La criptografía cuántica, expande una clave compartida corta en una secuencia compartida efectiva. Un requisito previo para la criptografía cuántica es que los usuarios, digamos, Alice y Bob conozcan 256 bits de clave secreta. El resultado de la criptografía cuántica es que Alice y Bob conocen un flujo de 1012 bits secretos que se usarán para cifrar mensajes. La longitud del flujo de salida aumenta linealmente con la cantidad de tiempo que Alice y Bob pasan en criptografía cuántica.

Los esquemas de firma basados en criptosistemas de clave pública vienen con dos claves: una pública y se usa para verificar firmas, mientras que la otra es privada, y se usa para producir una firma electrónica.

Criptosistemas simétricos y asimétricos

Retomando el tema de las computadoras cuánticas. La aparición de la primera computadora cuántica a pequeña escala en 2001, la cual factorizó el número 15 en 3x5 usando el algoritmo de Shor, nos motiva a encontrar sistemas criptográficos de clave pública que sean seguros y eficientes.

Los *criptosistemas simétricos* son aquellos que usan la misma clave tanto para el cifrado como para el descifrado, y, por lo tanto requieren un intercambio previo de la clave secreta para comunicarse en un canal de comunicación abierto.
Los *criptosistemas asimétricos* requieren dos claves, una pública y una secreta. La clave pública sirve para cifrar, y la clave privada, la cual sólo el emisor conoce, sirve para descifrar. La criptografía asimétrica resulta ser más segura que la simétrica, ya que se requiere hallar dos claves distintas para romper el sistema de cifrado.

1.3. Criptosistemas de clave pública multivariados

En la búsqueda de criptosistemas de clave pública que sean seguros y eficientes nos encontramos con los criptosistemas de clave pública multivariable, los cuales surgen del problema de resolver un conjunto de ecuaciones polinomiales multivariadas sobre un campo finito, cuya complejidad resulta ser NP-difícil [16]. Con esto en mente la construcción de dichos criptosistemas puede realizarse a partir de polinomios multivariados sobre un conjunto finito, que en su mayoría resultan ser polinomios cuadráticos.

Sistemas bipolares

Sea k un campo finito. En un criptosistema de clave pública multivariable bipolar, el cifrado está dado por una función $F : k^n \to k^m$

$$F(x_1, \ldots, x_n) = (f_1, \ldots, f_m),$$

donde cada f_i es un polinomio en $k[x_1, \ldots, x_n]$. Una construcción típica de este tipo de sistemas comienza construyendo una función $F : k^n \to k^m$ tal que

1. $F(x_1, \ldots, x_n) = (f_1, \ldots, f_m)$, donde $f_i \in k[x_1, \ldots, x_n]$
2. Cualquier ecuación
$$F(x_1, \ldots, x_n) = (y'_1, \ldots, y'_m)$$

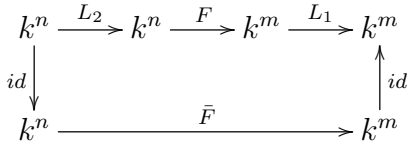
se puede resolver fácilmente. Es decir, podemos encontrar eficientemente una preimagen de $(y'_1, \ldots, y'_m)$, la cual será única para el caso de encriptación, y es denotada por

$$F^{-1}(y'_1, \ldots, y'_m)$$

Una vez que se encuentra la función, el cifrado $\bar{F}$ se construye como una composición de tres funciones:

$$\bar{F} = L_1 \circ F \circ L_2$$

donde L_1 es una transformación afín invertible elegida aleatoriamente de k^m a k^m y L_2 es una transformación afín invertible elegida aleatoriamente de k^n a k^n. En este caso la clave pública consiste de las m componentes polinomiales de $\bar{F}$ y la estructura de campo de k, mientras la clave secreta consiste de L_1 y L_2. La función F podrá ser o no parte de la clave secreta.



Para cifrar el mensaje $X' = (x'_1, \ldots, x'_n)$, se calcula $\bar{F}(X')$. Para descifrar un texto cifrado $Y' = (y'_1, \ldots, y'_m)$, se resuelve el sistema de ecuaciones definidas por

$$\bar{F}(x_1, \ldots, x_n) = Y'. \tag{1.1}$$

Esto se logra al calcular primero $Y_1 = L_1^{-1}(Y')$, luego $Y_2 = F^{-1}(Y_1)$, seguido de $L_2^{-1}(Y_2)$.

En los esquemas multivariables bipolares, L_1 y L_2 sirven para ocultar la función F que de otra forma podría invertirse.

Sistemas mixtos

Un sistema de clave pública multivariado mixto usa una función $\bar{H} : k^{n+m} \to k^l$ con clave pública

$$\bar{H}(x_1, \ldots, x_n, y_1, \ldots, y_m) = (\bar{h}_1, \ldots, \bar{h}_l). \tag{1.2}$$

Donde cada $\bar{h}_i$ es un polinomio en $k[x_1, \ldots, x_n, y_1, \ldots, y_m]$. Para encontrar tal esquema, se usa una función $H : k^{n+m} \to k^l$

$$H(x_1, \ldots, x_n, y_1, \ldots, y_m) = (h_1, \ldots, h_l)$$

donde cada h_i es un polinomio en $k[x_1, \ldots, x_n, y_1, \ldots, y_m]$ tal que:

1. Para cualquier $x'_1, \ldots, x'_n$ dado, el sistema de ecuaciones

$$H(x'_1, \ldots, x'_n, y_1, \ldots, y_m) = (0, \ldots, 0) \tag{1.3}$$

puede resolverse. En la mayoría de los casos, la Eq. (1.3) es un sistema de ecuaciones lineales afines en las variables $y_1, \ldots, y_m$.

2. Para cualquier elemento $(y'_1, \ldots, y'_m)$, el sistema de ecuaciones

$$H(x_1, \ldots, x_n, y'_1, \ldots, y'_m) = (0, \ldots, 0) \tag{1.4}$$

puede ser resuelto con facilidad. La Eq. (1.4) es un sistema de ecuaciones lineales.

Una vez que se halla tal función, $\bar{H}$ se construye como

$$\bar{H} = L_3 \circ H \circ (L_1 \times L_2)$$

donde $L_1 : k^n \to k^n$ y $L_2 : k^m \to k^m$ son definidas como un caso bipolar y $L_3 : k^l \to k^l$ es una transformación lineal invertible.

Para cifrar un mensaje $X' = (x'_1, \ldots, x'_n)$, sustituimos en la ecuación (1.2) y resolvemos el sistema de ecuaciones

$$\bar{H}(x'_1, \ldots, x'_n, y_1, \ldots, y_m) = (0, \ldots, 0),$$

denotando la solución por $Y' = (y'_1, \ldots, y'_m)$. Esta Y' es el mensaje cifrado. Para descifrar un texto cifrado $Y' = (y'_1, \ldots, y'_m)$, primero calculamos

$$\bar{Y} = L_3^{-1}(Y').$$

Entonces, sustituimos $\bar{Y}$ en la ecuación (1.4), y resolvemos el sistema de ecuaciones

$$H(x_1, \ldots, x_n, \bar{y}_1, \ldots, \bar{y}_m) = (0, \ldots, 0).$$

Si denotamos a la solución de esta ecuación como $\bar{X}$, entonces el texto en claro está dado por

$$X' = L_1^{-1}(\bar{X}).$$

La clave pública consiste de los l componentes polinomiales de $\bar{H}$ y la estructura del campo k. La clave secreta consiste de L_1, L_2 y L_3. La ecuación $H(X, Y) = (0, \ldots, 0)$, dependiendo del caso, podrá formar parte o no de la clave pública. Si se diera el valor de Y, y no se contaran con las funciones L_1, L_2, L_3 que sirven para ocultar la función, se podría resolver fácilmente la ecuación $H(X, Y) = (0, \ldots, 0)$.

Esquemas IP

El problema del isomorfismo de polinomios (IP) que se describe a continuación se originó al tratar de atacar a los Criptosistemas de clave pública multivariable (MPKC, por sus siglas en inglés) para encontrar las claves secretas. Sean F_1, F_2 tales que

$$F_i(x_1, \ldots, x_n) = (f_{i1}, \ldots, f_{im})$$

son funciones polinomiales de k^n a k^m. El *problema IP* es buscar dos transformaciones lineales afines e invertibles L_1 sobre k^n y L_2 sobre k^m (si existen) tales que

$$F_1(x_1, \ldots, x_n) = L_2 \circ F_2 \circ L_1(x_1, \ldots, x_n).$$

A lo largo de esta tesis se trabajan con distintos criptosistemas de clave pública multivariable. En cada uno de ellos se utilizan dos transformaciones afines invertibles L_1 y L_2, las cuales sirven para "disfrazar" la función central de cada sistema, la cual definiremos más adelante. Estas dos transformaciones fortalecen la seguridad al realizar combinaciones lineales de la clave pública, ya que al ser desconocidas dificultan realizar diversos tipos de ataques criptográficos.

Seguridad básica y eficiencia

Cualquier proceso de descifrado que hemos comentado hasta el momento hace uso de las funciones polinomiales, que en su mayoría resultan tener componentes cuadráticos (es innecesario trabajar con funciones que sean de grado mayor, ya que para el caso cuadrático ya se tiene un problema NP-difícil). La seguridad recae en que la Eq. (1.1) no sea fácil de resolver para un atacante que no

tenga información adicional a la clave pública. También es importante que, aunque se tengan varias parejas de texto cifrado y texto en claro, no se pueda hacer uso de la clave pública para encontrar el inverso de la función de cifrado, y con ello romper el sistema. Así mismo debe de ser complicada la factorización de P, de lo contrario será fácil recuperar la clave secreta.

Un poco de historia
El primer intento de firma digital multivariada fue dado en 1984 [17]. Este sistema está basado sobre una ecuación cuadrática

$$y = x_1^2 + \alpha x_2^2 \bmod n, \quad \text{con } \alpha = -u^{-2} (\bmod n). \tag{1.5}$$

donde n es un entero compuesto grande difícil de factorizar y $u \in \mathbb{Z}_n^*$. La clave pública consiste de n y α, y la clave privada es u. Para firmar un mensaje y se requiere encontrar una solución de Eq.(1.5), la cual es sencilla de obtener si se conoce la factorización de n.
Para el año 1987, Pollard y Schnorr rompieron dicho sistema [20], al encontrar un algoritmo que resuelve Eq.(1.5) para cualquier y, sin necesidad de tener la factorización de n.

Complejidad de algoritmos

En esta sección se recuerdan brevemente las definiciones de las clases de complejidad de algoritmos. Muchos sistemas criptográficos basan su seguridad en problemas que pertenecen a alguna de estas clases.

Se dice que un problema pertenece a la clase computacional **P** si puede ser resuelto en *tiempo polinomial*, es decir, en un número de pasos que a lo más es polinomial en el tamaño de entrada. La clase computacional **NP**, en su lugar, es definida como la clase de problemas cuya solución puede ser verificada en tiempo polinomial. Es claro que **P** es subconjunto de **NP**. Esto es un problema fundamental abierto de matemáticas el hecho que existan problemas en **NP** que no se encuentren en **P**. Esto es una conjetura, que $\mathbf{P} \neq \mathbf{NP}$. Si es el caso, habrían problemas difíciles de resolver pero cuya solución podría ser verificada fácilmente. Por ejemplo, el problema de factorización de números enteros es un problema **NP**, ya que es fácil verificar que un número m es un factor primo de un entero n, pero no se conoce ningún algoritmo que calcule (en una computadora convencional) los factores primos de n.

Se dice que un problema **NP** es **NP-completo** si está en la clase **NP** y todo problema en la clase **NP** es reducible polinomialmente a él. Esto significa que, dado un problema **NP-completo**, para cualquier problema **NP** hay una función que puede ser calculada con recursos polinomiales y que transforman el problema en la clase **NP** al problema en la clase **NP-completo**, de forma que conocida una solución de este último es posible obtener, también en tiempo polinomial, una solución del primero. Finalmente, un problema B se dice **NP-difícil** si para algún problema **NP-completo** A, se cumple que el problema A se puede resolver en tiempo polinomial utilizando un algoritmo polinomial hipotético para el problema B.

Capítulo 2

Criptosistemas

2.1. Sistema Matsumoto e Imai (MI)

Matsumoto e Imai hicieron una gran aportación en la criptografía basada en campos finitos [Ap. A]. El sistema MI se desarrolló en extensiones del campo $\mathbb{F}_2$, el cual consiste esencialmente en evaluaciones de polinomios cuadráticos en varias variables y fue roto 7 años después de su publicación (1995) [19]. Sin embargo las ideas de Matsumoto e Imai han originado gran cantidad de investigación en la creación de sistemas de cifrado y sus ideas principales han sido utilizadas para el mismo fin.

Descripción

Sea k un campo de cardinalidad q, y n un número natural. Supongamos que K es una extensión de grado n de k ($K = k[y]/\langle g(y)\rangle$, donde $k[y]$ es el anillo de polinomios con coeficientes en k y $g(y)$ un polinomio irreducible sobre k). Sea $\varphi : K \to k^n$ el isomorfismo de espacios vectoriales

$$a_1 + a_2 y + \cdots + a_n y^{n-1} \mapsto (a_1, \ldots, a_n).$$

Se considera, además, una *función central* la cual dependerá de cada criptosistema, se le nombra de esta forma ya que se encuentra precisamente en medio de la composición de transformaciones que se presenta más adelante para formar la clave pública. Esta función tiene que ser polinomial, invertible y de grado a lo más dos, esto es por practicidad, ya que si la clave consta de polinomios de grado mayor que dos, esta será más grande.
Para el sistema MI se tiene como función central $F(X) = X^{1+q^\theta}$, con $MCD(q^\theta + 1, q^n - 1) = 1$.
La inversa de la función F resulta:

$$F^{-1}(X) = X^t$$

dicha t existe puesto que $MCD(q^\theta + 1, q^n - 1) = 1$ y satisface $(q^\theta + 1)t \equiv 1 \bmod (q^n - 1)$. Esto se cumple cuando q es par, de lo contrario el polinomio no resulta ser invertible.
Sean L_1, L_2 transformaciones afines invertibles en k^n. La clave pública está dada por

$$P = L_1 \circ \varphi \circ F \circ \varphi^{-1} \circ L_2.$$

También se requiere conocer la aritmética del campo.

$F(X)$ actúa en elementos de K (polinomios), vea Fig. 2.1. Veamos que $\bar{F}$ da como resultado polinomios cuadráticos:

Si K es un campo de característica p, la función

$$x \mapsto x^p$$

es un homomorfismo de K en sí mismo, tiene como núcleo al elemento cero, por lo que es inyectiva. Entonces, para cada entero $r \geq 1$, la función $x \mapsto x^{p^r}$ es un endomorfismo de K, llamado el automorfismo de Frobenius.
De la función central

$$\begin{aligned} F(X) &= X^{1+q^\theta} \\ &= X \cdot X^{q^\theta} \text{ es el producto de dos } k\text{-transformaciones lineales (de Frobenius)} \\ &= (a_0 + a_1\beta + \cdots + a_{n-1}\beta^{n-1})(a_0 + a_1\beta + \cdots + a_{n-1}\beta^{n-1})^{q^\theta} \\ &= (a_0 + a_1\beta + \cdots + a_{n-1}\beta^{n-1})(a_0 + a_1\beta^{q^\theta} + \cdots + a_{n-1}\beta^{(n-1)q^\theta}) \quad \text{pues } a_i \in k \\ &= F_0(a_0, \cdots, a_{n-1}) + F_1(a_0, \cdots, a_{n-1})\beta + \ldots \end{aligned}$$

con lo que $\bar{F} = (F_0, \ldots, F_n)$ se compone de polinomios cuadráticos. Por otro lado, la clave privada consta de la descomposición de P.

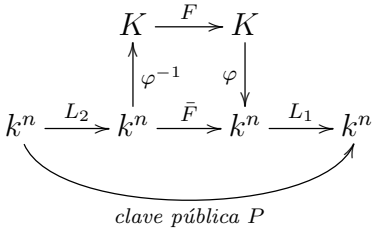


Figura 2.1: El sistema MI

Cifrado Suponga que $P = (P_0, P_1, \ldots, P_n)$. Dado un mensaje $m = (m_1, \ldots, m_n) \in k^n$, se cifra mediante la evaluación $P(m)$

$$P(m_1, \ldots, m_n) = \begin{pmatrix} P_1(m_1, \ldots, m_n) \\ \vdots \\ P_n(m_1, \ldots, m_n) \end{pmatrix}$$

Sea $c = (c_1, \ldots, c_n)$ un mensaje cifrado. Para descifrar se evalúa

$$P^{-1}(c_1, \ldots, c_n) = L_2^{-1} \circ \varphi \circ F^{-1} \circ \varphi^{-1} \circ L_1^{-1}(c_1, \ldots, c_n).$$

Ejemplo 2.1.1. Considere $k = \mathbb{F}_2$ y $n = 5$. Sea K una extensión de grado 5 de k, $\varphi : K \to k^5$ el isomorfismo de espacios vectoriales dado por:

$$a_0 + a_1 y + a_2 y^2 + a_3 y^3 + a_4 y^4 \mapsto (a_0, a_1, a_2, a_3, a_4)$$

Se toma $\theta = 2$, con lo que

$$MCD(2^\theta + 1, 2^n - 1) = MCD(5, 31) = 1$$

Para obtener la función inversa, se resuelve la congruencia

$$5 \cdot t \equiv 1 \pmod{31}$$

con el valor hallado para t se obtiene la función inversa $F^t(x) = F^{25}(x)$. Se consideran,

$$L_1(x) = A_1 x^T + B_1, \; L_2(x) = A_2 x^T + B_2$$

transformaciones afines invertibles, con

$$A_1 = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Dado un mensaje $x = (a, b, c, d, e)$, entonces la clave pública $P(x)$ está dada por

$$P(\bar{x}) = \begin{pmatrix} ac + ad + bd + be + b + cd + ce + c + e \\ ab + ac + ad + ae + bc + be + b + cd + ce + c + d + e \\ ac + ad + ae + a + bd + b + ce + c + de + 1 \\ ab + ac + a + be + de + d \\ ab + ac + a + be + de + d \end{pmatrix}$$

Por ejemplo si desea cifrar $m = (1, 1, 0, 0, 1)$, basta con evaluar $P(m)$, obteniendo $c = (1, 1, 0, 1, 1)$.

Para descifrar se calcula

$$P^{-1}(c) = L_1^{-1} \circ \varphi^{-1} \circ F^{-1} \circ \varphi \circ L_2^{-1}(c) = m$$

Ejemplo 2.1.2. Sean $k = \mathbb{F}_2$ y $n = 10$. El campo k se hace público, y sea K una extensión de k de grado 10, $\varphi : K \to k^{10}$ el isomorfismo de espacios vectoriales dado por:

$$a_0 + a_1 y + \ldots + a_{n-1} y^{n-1} \mapsto (a_0, \ldots, a_{n-1})$$

el cual permite ir de un espacio a otro. Tomamos $\theta = 4$, así

$$MCD(2^\theta + 1, 2^n - 1) = MCD(17, 1023) = 1$$

Para hallar el valor de t, se resuelve la congruencia

$$17 \cdot t \equiv 1 \pmod{1023}$$

cuya solución es $t = 662$, por lo tanto $F^{-1} = X^{662}$.

Se consideran, $L_1(x) = Ax^T + B_1,\ L_2(x) = Ax^T + B_2$ transformaciones invertibles, donde

$$A = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Si $\bar{x} = (a, b, c, d, e, f, g, h, i, j)$, entonces la clave pública P está dada por

$$P(\bar{x}) = \begin{pmatrix} 1 + c + bc + ae + be + de + f + af + ag + bg + cg + dg + eg + fg \\ +h + ch + dh + eh + fh + ai + bi + ci + di + fi + aj + bj + dj + ej + fj + hj \\ \\ a + ab + c + d + ad + bd + e + ae + ce + bf + df + bg + fg + ah \\ +bh + ch + dh + fh + gh + i + bi + ci + di + hi + j + cj + dj + gj + hj \\ \\ 1 + a + b + ab + c + ac + d + ad + bd + ae + ce + de + bf + cf \\ +ag + bg + cg + dg + eg + h + ch + fh + gh + i + ai + ci + fi + gi + dj + ej + hj \\ \\ a + ab + ac + ad + cd + e + ae + cf + ag + bg + dg + eg + fg + ch \\ +bi + ci + ei + fi + gi + hi + bj + dj + ej + fj + hj \\ \\ a + ab + c + ac + d + ad + cd + e + ae + de + f + cf + df + ef \\ +g + bg + cg + dg + fg + ah + bh + eh + gh + ci + fi + bj + dj + fj + gj \\ \\ a + b + ab + c + cd + ae + be + ce + de + cf + df + g + ag \\ +cg + eg + bh + ch + eh + ci + bj + dj + fj + gj + ij \\ \\ ab + be + af + bf + g + ag + cg + dg + eg + fg + ah + bh \\ +ch + dh + eh + ci + di + fi + gi + bj + cj + hj \\ \\ 1 + a + ab + c + ae + be + ce + de + df + g + ag + dg + eg + fg \\ +h + ch + dh + eh + fh + gh + i + ai + bi + di + ei + gi + hi + aj + fj + gj + hj \\ \\ 1 + b + bc + d + ad + f + af + bf + df + ef + g + ag + cg + dg \\ +eg + h + bh + dh + eh + gh + i + ai + ei + fi + j + bj + cj + ej + hj + ij \\ \\ b + ac + bc + d + bd + cd + ae + be + ce + de + af + bf \\ +dg + ah + ch + eh + fh + gh + i + ai + gi + hi + aj + cj + fj + gj \end{pmatrix}$$

y la clave secreta está formada por la descomposición de P.

Se toma como mensaje $m = (1, 1, 1, 1, 1, 0, 0, 0, 1, 1)$.

Cifrado. Para cifrar simplemente se evalúa el mensaje en la clave pública P

$$c = P(m) = (0, 0, 1, 1, 1, 1, 0, 0, 1, 0)$$

Descifrado. Se calcula $P^{-1}(c)$ donde

$$P^{-1}(x) = L_1^{-1} \circ \varphi^{-1} \circ F^{-1} \circ \varphi \circ L_2^{-1}(x)$$

entonces

$$P^{-1}(c) = P^{-1}(0,0,1,1,1,1,0,0,1,0) = m$$

Obs. El tamaño de la clave varía de acuerdo al valor de n y θ. En el ejemplo (2.1.1) se considera $n = 5$, $\theta = 2$. La clave $P(x)$ resulta de 5 polinomios cuadráticos. A su vez, para el ejemplo (2.1.2) se tomaron $n = 10$ y $\theta = 4$. En este caso la clave pública $P(x)$ se compone de 10 polinomios cuadráticos en 10 variables, la cual es, en gran medida, más grande que la del primer ejemplo. Es por ello que se recomienda considerar el valor de θ mínimo con la condición $MCD(2^\theta + 1, 2^n - 1) = 1$.

2.1.1. El ataque de ecuaciones de linealización

Patarin rompió por primera vez el sistema MI en 1995 [18]. El ataque no busca recuperar la clave privada, si no busca descifrar sin ella, ya que la clave privada satisface relaciones entre la imagen y la preimagen en forma de ecuaciones de linealización.

Definición 2.1.1. Sea $\mathcal{G} = \{g_1, \ldots, g_m\}$ cualquier conjunto de polinomios en $k[x_1, \ldots, x_n]$. Una *ecuación de linealización* para $\mathcal{G}$ es cualquier polinomio en $k[x_1, \ldots, x_n, y_1, \ldots, y_m]$ de la forma

$$\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}x_i y_j + \sum_{i=1}^{n} b_i x_i + \sum_{j=1}^{m} c_j y_j + d = 0 \tag{2.1}$$

tal que se obtiene la función cero en $k[x_1, \ldots, x_n]$ al sustituir y_j en g_j, para $j = 1, \ldots, n$. El conjunto de ecuaciones de linealización de $\mathcal{G}$ forma un k- espacio vectorial.

Sea $\{f_1, \ldots, f_n\}$ el conjunto de componentes de la clave pública, y supongamos que se tiene una ecuación de linealización para este conjunto de la forma (2.1). Dado un texto cifrado $(y_1, \ldots, y_n)$, sustituyendo f_i en y_i produce una ecuación no lineal en las variables $x_1, \ldots, x_n$ cuyo conjunto de soluciones contiene al texto cifrado. Con las suficientes ecuaciones de linealización se pueden producir las ecuaciones lineales linealmente independientes necesarias, cuyo sistema resultante tenga como solución única al texto en claro o bien permita reducir el número de variables. Es necesario hallar el número de ecuaciones de linealización linealmente independientes que se derivan del espacio de ecuaciones de linealización y estudiar la viabilidad de dicho ataque. Este problema se analiza en el siguiente resultado.

Teorema 2.1.1. Sea $\mathcal{L}$ el espacio de ecuaciones de linealización de los componentes de $\bar{F}$ y suponga que $Y' = (y_1', \ldots, y_n') \in k^n$. Si $\mathcal{L}_{Y'}$ es el espacio de ecuaciones lineales que resulta al sustituir y_i en y_i' para $i = 1, \ldots, n$ en cada elemento de $\mathcal{L}$, entonces $\dim_k \mathcal{L}_{Y'}$ es al menos

$$n - MCD(\theta, n) \geq \frac{2n}{3},$$

excepto cuando $Y' = (0, \ldots, 0)$.

Demostración. Para la construcción de las ecuaciones de linealización, considerar $X, Y \in K$ tales que

$$Y = F(X) = X^{q^\theta + 1}.$$

Entonces

$$\begin{aligned} Y^{q^\theta-1} &= (X^{q^\theta+1})^{q^\theta-1} \\ &= X^{(q^\theta+1)(q^\theta-1)} \\ &= X^{q^{2\theta-1}} \end{aligned}$$

Multiplicando en ambos lados por XY, se obtiene

$$XY^{q^\theta} = X^{q^{2\theta}}Y.$$

Se define $R(X,Y) \in K[X,Y]$ mediante

$$R(X,Y) = XY^{q^\theta} - X^{q^{2\theta}}Y,$$

y

$$\bar{R} = \varphi \circ R \circ (\varphi^{-1} \times \varphi^{-1})$$

De $\bar{R}$ se pueden derivar n ecuaciones de linealización para los componentes de $\bar{F}$. Dichos n componentes de $\bar{R}(x_1,\ldots,x_n,y_1,\ldots,y_n)$ son de la forma (2.1) y por construcción, sustituyendo y_i en f_i produce el polinomio cero en $k[x_1,\ldots,x_n]$.

Sean $(x'_1,\ldots,x'_n) = \bar{F}^{-1}(y'_1,\ldots,y'_n) \in k^n$, $Y' = \varphi^{-1}(y'_1,\ldots,y'_n)$ y $X' = \varphi^{-1}(x'_1,\ldots,x'_n)$. Entonces X' debe ser solución de

$$X^{q^{2\theta}}Y' = X(Y')^{q\theta}, \tag{2.2}$$

o

$$X^{q^{2\theta}-1} = (Y')^{q^\theta} - 1,$$

si $Y' \neq 0$. La segunda ecuación tiene a lo más $MCD(q^{2\theta}-1, q^n-1)$ soluciones en K. Además, puesto que $MCD(q^\theta+1, q^n-1) = 1$, se tiene

$$MCD(q^{2\theta-1}, q^n-1) = MCD(q^\theta-1, q^n-1),$$

por lo que se tienen a lo más $MCD(q^\theta-1, q^n-1)+1$ soluciones, incluyendo la solución trivial.

La ecuación $X(Y')^{q^\theta} = X^{q^{2\theta}}Y'$ tiene a lo más $q^{MCD(\theta,n)}$ soluciones debido a que

$$MCD(q^a-1, q^b-1) = q^{gcd(a,b)} - 1$$

para cualesquiera enteros a y b. Si β es el número de ecuaciones lineales linealmente independientes que se obtienen de Eq. (2.2), entonces habrán $q^{n-\beta}$ soluciones para el sistema de ecuaciones lineales correspondientes. Por tanto $q^{n-\beta} \leq q^{MCD(\theta,n)}$, y entonces, $\beta \geq n - MCD(\theta,n)$.

Los tres valores más grandes posibles de $MCD(\theta,n)$ son n, $n/2$ y n si es par y $n/3$ si 3 divide a n. Por tanto, si demostramos que los dos primeros casos son imposibles, se podrá concluir que

$$n - MCD(\theta,n) \leq \frac{2n}{3}.$$

El valor de $MCD(\theta,n)$ no puede ser n ya que el valor de θ es mayor que cero y menor que n. Ahora si $MCD(\theta,n) = n/2$, significa que θ debe valer $n/2$. Sabemos que

$$MCD(q^{n/2}+1, q^n-1) = q^{n/2}+1 > 1,$$

lo cual no es posible. Por tanto $MCD(\theta,n)$ no puede ser $n/2$ y el valor más grande que puede tener es $n/3$.

Con esto hemos verificado que el sistema Matsumoto e Imai no es tan seguro, ya que dado un texto cifrado se pueden hallar al menos $2n/3$ ecuaciones lineales que satisface el texto en claro, lo cual es equivalente a filtrar $2/3$ de la información. Además, aquellas ecuaciones podrán ser usadas para eliminar $2/3$ de las variables de las ecuaciones cuadráticas públicas derivadas de la clave pública y del texto cifrado, las cuales deberían ser más fáciles de resolver.

Recientemente, en diciembre del 2015, fue creado un algoritmo capaz de resolver un sistema de ecuaciones cuadráticas sobre los binarios en tiempo polinomial en una computadora cuántica [8]. Con ello, una computadora cuántica puede romper el cifrado multivariable sobre $\mathbb{F}_2$. Esto condujo a considerar sistemas de cifrado sobre campos de característica impar.

2.2. Sistema HFE

El sistema HFE (Hidden Field Equations) por sus siglas en inglés, fue propuesto por Patarin en 1996. El sistema es una generalización del sistema Matsumoto-Imai. HFE fue roto con un ataque algebraico. Mediante el texto cifrado se desea obtener el texto en claro haciendo uso de la clave pública resolviendo el sistema de ecuaciones polinomiales:

$$\begin{aligned} P_1(m_1, \ldots, m_n) - c_1 &= 0 \\ &\vdots \\ P_n(m_1, \ldots, m_n) - c_n &= 0 \end{aligned}$$

Antes de comenzar con la descripción del sistema HFE, daremos la definición de q-grado de un polinomio y después se introduce un lema que nos permite asegurar que la transformación $\varphi^{-1} \circ L \circ \varphi$ tiene q-grado igual a uno, donde L es una transformación lineal en k^n y φ es un isomorfismo de espacios vectoriales k^n en K.

Definición 2.2.1. El *q-peso de Hamming* de un entero es el número de dígitos distintos de cero en su expansión q-aria.

Definición 2.2.2. El *q-grado del polinomio* $G(x) = \sum_{j=1}^{t} \alpha_j X^{d_j}, \alpha_j \neq 0$, es

$$qHWD(G) = \max_{1 \leq j \leq t} \{q - \textit{peso de Hamming } (d_j)\}$$

De hecho existe una correspondencia uno a uno entre las funciones cuadráticas $k^n \to k^n$ y las funciones $K \to K$ de $q-$grado igual a dos. En general, el $q-$grado de la función central determina el grado de la clave pública.

Lema 2.2.1. $L : k^n \to k^n$ es una función lineal si y sólo si $\widehat{L} = \varphi^{-1} \circ L \circ \varphi : K \to K$ tiene q-grado igual a uno, en otras palabras $\widehat{L}$ es de la forma

$$\widehat{L}(X) = \sum_{j=0}^{n-1} \alpha_j X^{q^j}.$$

Demostración.

($\Leftarrow$) Sean $L : k^n \to k^n$ una función lineal y $\widehat{L} : K \to K$ definida como arriba. $\widehat{L}$ es homogénea ya que para todo $a \in K$, y $0 < j < n$, se satisface que $\widehat{L}(aX) = a\widehat{L}(X)$

$$\begin{aligned} \widehat{L}(aX) &= \alpha_0(aX) + \alpha_1(aX)^q + \ldots + \alpha_{n-1}(aX)^{q^{n-1}} \\ &= a(\alpha_0 X + \alpha_1 X^q + \ldots + \alpha_{n-1}(X)^{q^{n-1}}) \quad \text{ya que } a^{q^j} = a, \\ &= a\widehat{L}(X). \end{aligned}$$

Por otro lado, por la propiedad del automorfismo de Frobenius, $(X_1+X_2)^{q^j} = X_1^{q^j} + X_2^{q^j}$, se tiene que $\widehat{L}(X)$ es una función $k-$lineal. Además, $L = \varphi \circ \widehat{L} \circ \varphi^{-1}$ también resulta ser $k-$lineal debido al isomorfismo φ.

($\Rightarrow$) Recíprocamente sea $S = \{\varphi^{-1} \circ L \circ \varphi : L \in \mathcal{M}_{n\times n}(k)\}$. Puesto que la composición de funciones lineales resulta una función lineal, entonces S es el conjunto de todas las funciones k lineales de K en K.

Ya que $\{X \mapsto X^{q^j} : \ 0 < j < n\}$ es un conjunto de funciones k-lineales, entonces

$$\{X \mapsto X^{q^j} : \ 0 < j < n\} \subseteq S.$$

Queda por demostrar que $S = gen_K\{X \mapsto X^{q^j} : \ 0 < j < n\}$.

Por un lado, $\mathcal{B} = \{1, x, x^2, \ldots, x^{n-1}\}$ es una base de K. Considerando las posibles combinaciones lineales de los elementos de la base $\mathcal{B}$, y puesto que hay n elementos de la base, entonces hay q^{n^2} transformaciones en S, por tanto $|S| = q^{n^2}$.
Por otro lado, una combinación K-lineal de monomios X^{q^j} tiene q^n posibilidades para cada uno de los n coeficientes, por lo que hay $(q^n)^n$ combinaciones. Estas combinaciones deben de ser distintas, pues si existieran dos iguales, entonces la diferencia sería el polinomio cero, así las combinaciones son distintas. Por lo que $|gen_K\{X \mapsto X^{q^j} : \ 0 < j < n\}| = q^{n^2}$, concluyendo que $S = gen_K\{X \mapsto X^{q^j} : 0 < j < n\}$.

□

Lema 2.2.2. La función $F : k^n \to k^n$ pueden ser expresada como una n-tupla de polinomios de grado dos si y sólo si $\widehat{F} = \varphi^{-1} \circ F \circ \varphi : K \to K$ tiene q-grado igual a dos; en otras palabras $\widehat{F}$ es de la forma

$$\widehat{F}(X) = \begin{cases} \sum_{\substack{j=0 \\ i\leq j}}^{n-1} \alpha_{ij} X^{q^i+q^j} + \sum_{j=0}^{n-1} \beta_j X^{q^j} + \gamma & q > 2 \\ \\ \sum_{\substack{j=1 \\ i<j}}^{n-1} \alpha_{ij} X^{q^i+q^j} + \sum_{j=0}^{n-1} \beta_j X^{q^j} + \gamma & q = 2. \end{cases}$$

Demostración
($\Leftarrow$) Ya que φ es un isomorfismo de espacio vectoriales

$$\begin{aligned} F(x_1, \ldots, x_n) &= \varphi \circ \widehat{F} \circ \varphi^{-1} \\ &= \varphi \left(\sum \alpha_{ij} (x_1 + x_2 y^{q^i} + \cdots + x_n y^{(n-1)q^i})(x_1 + x_2 y^{q^j} + \cdots + x_n y^{(n-1)q^j}) \right. \\ &\quad \left. + \sum \beta_i (x_1 + x_2 y^{q^i} + \cdots + x_n y^{(n-1)q^i} + \gamma) \right) \end{aligned}$$

dado que y^{q^j} es un k-combinación lineal de $1, y, \ldots, y^{n-1}$, se tiene

$$F(x_1, \ldots, x_n) = \begin{pmatrix} k - \text{combinación cuadrática de } x_i \\ \vdots \\ k - \text{combinación cuadrática de } x_i \end{pmatrix}.$$

Es decir, F es una tupla de polinomios cuadráticos de grado 2.

($\Rightarrow$) Ya que φ y la función de permutación son isomorfismos de espacios vectoriales, es suficiente mostrar que la función

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \mapsto \begin{pmatrix} x_t x_s \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

se evalúa en la forma buscada. Primero, se cumple

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \mapsto \begin{pmatrix} x_t \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \mapsto \begin{pmatrix} x_s \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

son transformaciones lineales y por el lema 2.2.1 sus elevaciones son de la forma

$$M_t(X) = \sum \alpha_{ti} X q^i,$$

$$M_s(X) = \sum \alpha_{si} X q^i,$$

Así

$$\begin{pmatrix} x_{ts} \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \varphi \left(M_t(\varphi^{-1} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} M_s(\varphi^{-1} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}) \right)$$

se sigue que la elevación de la función es $M_t M_s$.

□

A continuación se describe el sistema HFE.

Descripción

Sean k un campo finito de cardinalidad q, K una extensión de grado n de k y φ el isomorfismo de espacios vectoriales descrito antes.

Definición 2.2.3. Un *polinomio HFE con límite D* es un polinomio de q-grado igual a 2 y grado total a lo más D. En otras palabras $G \in K[X]$ es un polinomio HFE si es de la forma

$$G(X) = \sum_{q^i + q^j \leq D} \alpha_{ij} X^{q^i + q^j} + \sum_{q^j \leq D} \beta_j X^{q^j} + \gamma$$

con $\alpha_{ij}, \beta_j, \gamma \in K$.

Para el sistema HFE se usa un polinomio HFE como función central en lugar de X^{1+q^θ}, el resto de la construcción es la misma que para el sistema MI. Se usan dos transformaciones afines invertibles L_1 y L_2 junto con el isomorfismo de espacios vectoriales $\varphi : K \to k^n$ para generar la clave pública como

$$P = L_1 \circ \varphi \circ F \circ \varphi^{-1} \circ L_2$$

note que P es la tupla

$$P = \begin{pmatrix} P_1(x_1, \ldots, x_n) \\ \vdots \\ P_n(x_1, \ldots, x_n) \end{pmatrix}$$

donde cada P_i es un polinomio cuadrático. La clave privada es la descomposición de P.

Cifrado. Para cifrar un mensaje $m = (x_1, \ldots, x_n) \in k^n$, el emisor usa la clave pública P para calcular $P(x_1, \ldots, x_n) = (y_1, \ldots, y_n)$ y la envía al receptor.

Descifrado. Dado el texto cifrado $(y'_1, y'_2, \ldots, y'_n)$,

(i) Calcular $(\bar{y}_1, \ldots, \bar{y}_n) = L_1^{-1}(y'_1, y'_2, \ldots, y'_n)$,

(ii) Sea $\bar{Y} = \varphi^{-1}(\bar{y}_1, \ldots, \bar{y}_n) \in K$. Calcular el conjunto

$$Z = F^{-1}(\bar{Y}) = \{\bar{Z} \in K : F(\bar{Z}) = \bar{Y}\}.$$

Para calcular Z se usa el algoritmo Berlekamp (Apen. B).

(iii) Para cada $Z_i \in Z$ calcular

$$(x_{ij}, \ldots, x_{in}) = L_2^{-1} \circ \varphi(Z_i)$$

Es posible que se obtengan más de un elemento en Z, de ser así se cifra cada elemento en Z y se elige el correcto.

En la Figura 2.2 se muestra el diagrama del sistema HFE. A diferencia del sistema MI, en este caso es muy complicado encontrar las preimagenes de $F(X)$, si el polinomio HFE es de grado alto, pero de igual forma como en el diagrama MI, L_1 y L_2 sirven para ocultar el polinomio HFE.

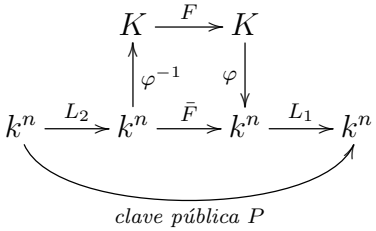


Figura 2.2: El sistema HFE

Variaciones de HFE

El sistema HFE fue roto en 1999 por Kipnis-Shamir [11], es por ello que surgieron distintas variaciones de esta construcción con el fin de mejorar su seguridad. Dos de ellas, HFE+ y HFE-, que se describen a continuación fueron construidas con el fin de imposibilitar ciertos ataques, por ejemplo el ataque Kipnis-Shamir, siempre y cuando el número de polinomios que se agreguen u oculten de la clave pública no sea demasiado grande.

HFE-

Si $(P_1, \ldots, P_n)$ es la clave pública del sistema HFE, es posible mantener en secreto algunos de estos polinomios, usualmente los últimos polinomios de P. Sea ℓ el número de dichos polinomios P_i que no se dan en la clave pública. Entonces solamente $P_1, \ldots, P_{n-l}$. son públicos.

HFE +

Sean P_i los polinomios de la clave pública para el esquema original HFE. Se añaden ℓ polinomios cuadráticos Q_i en $x_1, \ldots, x_n$ y se revuelven los polinomios P_i con los polinomios Q_i mediante una función biyectiva afín, lo cual genera la clave pública.

Observación Es posible combinar estas dos variaciones. Por ejemplo diseñar un esquema de cifrado usando HFE original con polinomios $P_1, \ldots, P_n$ manteniendo P_n en secreto, e introduciendo un polinomio aleatorio Q_n en lugar de P_n, y calcular la clave pública como una transformación afín de $P_1, \ldots, P_{n-1}, Q_n$.

Ejemplo 2.2.1. Se considera el campo $\mathbb{F}_{11}$, los mensajes a cifrar son elementos de $\mathbb{F}_{11}^6$ ($n = 6$). Para la construcción de la extensión K de grado 6 de $\mathbb{F}_{11}$ consideramos el polinomio irreducible de grado 6, $p(y) = y^6 + 3y^4 + 4y^3 + 6y^2 + 7y + 2$. Sea

$$F(x) = \alpha^3 x^{132} + \alpha^{10} x^{121} + x^{22} + \alpha^5 x^2 + \alpha^4 x + 1$$

como polinomio HFE (función central), donde α es raíz $p(x)$. Se consideran

$$A_1 = \begin{pmatrix} 5 & 1 & 5 & 8 & 4 & 2 \\ 0 & 8 & 1 & 2 & 8 & 7 \\ 9 & 6 & 3 & 5 & 9 & 10 \\ 0 & 5 & 6 & 10 & 10 & 9 \\ 1 & 6 & 6 & 10 & 6 & 2 \\ 4 & 2 & 6 & 1 & 5 & 0 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 5 \\ 8 \\ 7 \\ 9 \\ 4 \\ 6 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} 7 & 10 & 8 & 1 & 5 & 0 \\ 3 & 7 & 3 & 9 & 1 & 8 \\ 10 & 7 & 8 & 6 & 8 & 2 \\ 7 & 4 & 0 & 9 & 8 & 6 \\ 7 & 6 & 10 & 3 & 0 & 8 \\ 9 & 3 & 8 & 5 & 1 & 10 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 8 \\ 4 \\ 5 \\ 2 \\ 1 \\ 5 \end{pmatrix}$$

donde las matrices A_1 y A_2 son invertibles. Se construyen L_1 y L_2 transformaciones afines invertibles

$$L_1(x) = A_1 x^T + B_1 \quad L_2(x) = A_2 x^T + B_2$$

con B_1 y B_2 elementos aleatorios de $\mathbb{F}_{11}^6$. La clave pública resulta ser muy grande es por ello que evitamos escribirla específicamente, pero con la información proporcionada es fácil calcularla.

Cifrado Se considera el mensaje $m = (1, 4, 7, 8, 5, 6)$. Para cifrar se evalúa $P(m)$

$$P(m) = L_1 \circ \varphi \circ F \circ \varphi^{-1} \circ L_2(m) = (7, 8, 9, 3, 2, 1)$$

Descifrado

(i) $L_1^{-1}(1, 4, 7, 8, 5, 6) = (9, 9, 0, 8, 2, 4)$

(ii) $\bar{Y} = \varphi^{-1}(9, 9, 0, 8, 2, 4) = 9 + 9\alpha + 8\alpha^3 + 2\alpha^4 + 4\alpha^5$.
Usando el algoritmo Berlekamp (Vea el Apéndice B), se obtiene

$$Z = \{9\alpha^4 + 8\alpha^3 + \alpha^2 + 5\alpha + 1, 7\alpha^5 + 7\alpha^4 + 10\alpha^3 + 3\alpha^2 + 2\alpha,$$
$$9\alpha^5 + 10\alpha^4 + 5\alpha^3 + 6\alpha^2 + \alpha, 5\alpha^5 + 3\alpha^4 + 3\alpha^3 + 9\alpha^2 + 7\alpha + 2\}$$

(iii) Finalmente se calcula L_2^{-1} de cada uno de los elementos de Z y posteriormente se cifran nuevamente, para decidir cual fue el mensaje cifrado. En este caso el mensaje es

$$m = L_2^{-1}(7\alpha^5 + 7\alpha^4 + 10\alpha^3 + 3\alpha^2 + 2\alpha)$$

como se esperaba.

2.2.1. Ataque Algebraico

Un ataque que puede ser empleado en contra de los criptosistemas de clave pública multivariable es el ataque algebraico. Actualmente los algoritmos de bases de Gröbner F_4 y F_5 [8] son los más eficientes en cuanto a estos ataques. Si se trabaja en un campo k de característica 2, y el valor de D es pequeño, el criptosistema HFE se puede romper mediante ataques algebraicos, es por ello que se sugiere que el valor de q sea 13 ó 31 ya que proporcionan una defensa contra un ataque algebraico de bases de Gröbner.

Suponga que un atacante desea recuperar el texto en claro de un texto cifrado $(y_1, \ldots, y_n)$, sin el conocimiento de la clave privada, usando la clave pública. En este tipo de ataque se intenta resolver el sistema de ecuaciones

$$\begin{aligned} P_1(x_1, \ldots, x_n) - y_1 &= 0 \\ P_2(x_1, \ldots, x_n) - y_2 &= 0 \\ &\vdots \\ P_n(x_1, \ldots, x_n) - y_n &= 0 \end{aligned}$$

sin la ayuda de las propiedades del sistema, pero haciendo uso de la Geometría Algebraica y con ello, hallar todas las soluciones de dicho sistema utilizando las bases de Gröbner para el ideal formado por los componentes de la clave pública. Para entrar en contexto, se dará una introducción a la Geometría Algebraica.

Variedades afines e ideales

Definición 2.2.4. Sea R un anillo y sea I un subconjunto distinto del vacío. Decimos que I es un *ideal* de R si

- $0 \in I$.
- I es un subgrupo aditivo de R,
- para todo $a \in R$ y para todo $x \in I$, se tiene ax, $xa \in I$,

Definición 2.2.5. Sea k un campo y sean $f_1, \ldots, f_m$ polinomios en $k[x_1, \ldots, x_n]$. Se define

$$\mathbf{V}(f_1, \ldots, f_m) = \{(a_1, \ldots, a_n) \in k^n : f_i(a_1, \ldots, a_n) = 0 \text{ para todo } 1 \leq i \leq m\},$$

llamamos $\mathbf{V}(f_1, \ldots, f_m)$ la *variedad afín* definida por $f_1, \ldots, f_m$.

Definición 2.2.6. Sean $f_1, \ldots, f_m$ polinomios en $k[x_1, \ldots, x_n]$. Se define el ideal

$$\langle f_1, \ldots, f_m \rangle = \{\sum_{i=1}^{m} h_i f_i : h_1, \ldots, h_m \in k[x_1, \ldots, x_n]\}.$$

Si bien cada ideal definido como arriba es generado por un conjunto finito de polinomios, es posible que se tengan más de una base. Las variedades que generan por separado resultan ser la misma variedad. Esta propiedad es importante al momento de resolver ecuaciones polinomiales ya que se desea hallar una mejor base (base de Gröbner) para hallar todas las soluciones del sistema de ecuaciones polinomiales.
Consideremos el conjunto de todos los polinomios que se anulan en una variedad dada.

Definición 2.2.7. Sea $V \subset k^n$ una variedad afín. Entonces el conjunto

$$\mathbf{I}(V) = \{f \in k[x_1, \ldots, x_n] : f(a_1, \ldots, a_n) = 0 \text{ para todo } (a_1, \ldots, a_n) \in V\}$$

es llamado *el ideal asociado a* V.

Hasta el momento hemos visto la forma de generar un ideal a partir de un conjunto finito de polinomios $f_1, \ldots, f_m$ (todas las posibles combinaciones lineales). También que con un conjunto finito de polinomios se crea una variedad afín $\mathbf{V}(f_1, \ldots, f_m)$ (las n-adas que anulan todos los polinomios f_i) y, a su vez, se puede construir el ideal asociado a una variedad $\mathbf{I}(\mathbf{V}(f_1, \ldots, f_m)$. La pregunta que puede surgir es, ¿Es posible tener la igualdad $\mathbf{I}(\mathbf{V}(f_1, \ldots, f_m)) = \langle f_1, \ldots, f_m \rangle$? La respuesta es que no siempre se tiene dicha igualdad.

Lema 2.2.3. Si $f_1, \ldots, f_m \in k[x_1, \ldots, x_n]$, entonces $\langle f_1, \ldots, f_m \rangle \subset \mathbf{I}(\mathbf{V}(f_1, \ldots, f_m))$.

Demostración. Sea $f \in \langle f_1, \ldots, f_m \rangle$, es decir que $f = \sum_{i=1}^{m} h_i f_i$ para algunos polinomios $h_i \in k[x_1, \ldots, x_n]$. Ya que los polinomios $f_1, \ldots, f_m$ se anulan en $\mathbf{V}(f_1, \ldots, f_m)$, también se debe anular cualquier combinación lineal de ellos. Por tanto f se anula en $\mathbf{V}(f_1, \ldots, f_m)$ lo cual prueba que $f \in \mathbf{I}(\mathbf{V}(f_1, \ldots, f_m))$.

Bases de Gröbner

Tanto en el algoritmo de la división como en el algoritmo de eliminación Gaussiana para sistemas de ecuaciones lineales, el ordenar los términos (de manera creciente o decreciente) de los polinomios o las entradas de la matriz según sea el caso, facilita la ejecución de dichos algoritmos. Análogamente se busca definir un orden en $k[x_1, \ldots, x_n]$, digamos ordenar los términos de los polinomios. Existen distintas forma de definir el ordenamiento de los polinomios en $k[x_1, \ldots, x_n]$.

Definición 2.2.8. Un *orden monomial* en $k[x_1, \ldots, x_n]$ es cualquier relación $>$ en $\mathbb{Z}^n_{\geq 0}$, o equivalentemente, cualquier relación en el conjunto de monomios x^α, $\alpha \in \mathbb{Z}^n_{\geq 0}$, que satisface:

(i) $>$ es un orden total (o lineal) en $\mathbb{Z}^n_{\geq 0}$,

(ii) Si $\alpha > \beta$ y $\gamma \in \mathbb{Z}^n_{\geq 0}$, entonces $\alpha + \gamma > \beta + \gamma$,

(iii) $>$ es un buen orden en $\mathbb{Z}^n_{\geq 0}$. Esto significa que cada subconjunto distinto del vacío de $\mathbb{Z}^n_{\geq 0}$ tiene un elemento mínimo bajo $>$.

Ahora se definen tres de los más importantes ordenes monomiales que sirven para la construcción de las bases de Gröbner, en particular se trabaja con el orden lexicográfico para encontrar una base de Gröbner reducida respecto a este orden.

Definición 2.2.9. (Orden lexicográfico). Sea $\alpha = (\alpha_1, \ldots, \alpha_n)$ y $\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{Z}^n_{\geq 0}$. Se dice que $\alpha >_{lex} \beta$ si, en el vector $\alpha - \beta \in \mathbb{Z}^n_{\geq 0}$, la entrada distinta de cero más a la izquierda es positiva. Se escribe $x^\alpha >_{lex} x^\beta$ si $\alpha >_{lex} \beta$.

Por ejemplo:

1. Si $\alpha = (3, 5, 9), \beta = (1, 8, 4)$, entonces $(3, 5, 9) >_{lex} (1, 8, 4)$ ya que $\alpha - \beta = (2, -3, 5)$.

2. Las variables $x_1, \ldots, x_n$ son ordenadas de la forma usual por el orden lex. Como

$$(1, 0, \ldots, 0) >_{lex} (0, 1, 0, \ldots, 0) >_{lex} \ldots >_{lex} (0, 0, \ldots, 0, 1).$$

Entonces $x_1 >_{lex} x_2 >_{lex} \ldots >_{lex} x_n$.

Definición 2.2.10. (Orden lexicográfico graduado). Sea $\alpha, \beta \in \mathbb{Z}^n_{\geq 0}$. Se dice que $\alpha >_{grlex} \beta$ si

$$|\alpha| = \sum_{i=1}^{n} \alpha_i > \sum_{i=1}^{n} \beta_i = |\beta|, \text{ o } \alpha >_{lex} \beta \text{ si } |\alpha| = |\beta|.$$

Definición 2.2.11. (Orden lexicográfico inverso graduado). Sea $\alpha, \beta \in \mathbb{Z}^n_{\geq 0}$. Decimos que $\alpha >_{grevlex} \beta$ si

$$|\alpha| = \sum_{i=1}^{n} \alpha_i > |\beta| = \sum_{i=1}^{n} \beta_i, \text{ o } |\alpha| = |\beta| \text{ y la entrada distinta de cero más a la derecha de}$$
$$\alpha - \beta \in \mathbb{Z}^n_{\geq 0} \text{ es negativa.}$$

Hay algunas definiciones para términos especiales de polinomios con respecto a los ordenamientos.

Definición 2.2.12. Sea $f = \sum_\alpha a_\alpha x^\alpha$ un polinomio distinto de cero en $k[x_1, \ldots, x_n]$ y sea $>$ un orden monomial.

(i) El **multigrado** de f es

$$multigr(f) = max\{\alpha \in \mathbb{Z}^n_{\geq 0} : a_\alpha \neq 0\}.$$

(el máximo se toma respecto a $>$).

(ii) El **coeficiente principal** de f es

$$\mathbf{LC}(f) = a_{multigr(f)} \in k.$$

(iii) El **monomio principal** de f es

$$\mathbf{LM}(f) = x^{multigr(f)}$$

(con coeficiente 1).

(iv) El **término principal** de f es

$$\mathbf{LT}(f) = \mathbf{LC}(f) \cdot \mathbf{LM}(f).$$

Ilustremos esta última definición con un ejemplo. Sea $f = 11xy^2z^2 + 9z^2 + 3y^4 + 4x^3z^2$ y sea $>$ el orden *lex*. Entonces

$$\begin{aligned} multigr(f) &= (0, 4, 0), \\ \mathbf{LC}(f) &= 3, \\ \mathbf{LM}(f) &= y^4, \\ \mathbf{LT}(f) &= 3y^4. \end{aligned}$$

Gracias a la teoría de la Geometría Algebraica que hasta el momento se introdujo, se sabe que resolver el sistema de ecuaciones lineales definida al inicio de la sección, es equivalente a encontrar un conjunto de generadores para el ideal $\langle P_1, \ldots, P_n \rangle$. De las maneras posibles, los mejores conjuntos generadores de este ideal son las bases de Gröbner.

Definición 2.2.13. Dado un ideal $I \leq k[x_1, \ldots, x_n]$ y un orden monomial $>$ una *base de Gröbner* es una colección finita $g_1, \ldots, g_t \subset I$ tales que los términos principales de los g_i generan todos los términos principales en I, esto es, si $\mathbf{LT}(f)$ es el término principal de f respecto a $>$, entonces

$$\langle \mathbf{LT}(g_1), \ldots, \mathbf{LT}(g_t) \rangle = \langle \mathbf{LT}(f) : f \in I \rangle.$$

En especial nos interesan las bases de Gröbner respecto al orden lexicográfico (lex), específicamente se busca la base única de Gröbner reducida respecto al orden lex.

Definición 2.2.14. Una *Base de Gröbner reducida* para un ideal polinomial I es una base de Gröbner G para I tal que

(i) $\mathbf{LC}(p) = 1$ para todo $p \in G$.

(ii) Para todo $p \in G$, ningún monomio de p está en $\langle \mathbf{LT}(G - \{p\})\rangle$.

Con ello, la base de Gröbner para el ideal $\langle P_1 - y_1, \ldots, P_n - y_n\rangle$ revelará el texto en claro, así

$$\langle P_1(x_1, \ldots, x_n) - y_1, \ldots, P_n(x_1 \ldots, x_n) - y_n\rangle = \langle g_1(x_1, \ldots, x_{n-1}, x_n), \ldots, g_n(x_n)\rangle$$

la cual podrá ser resuelta mediante sustitución hacia atrás. Aunque usar esta base de Gröbner resulta fácil, hallarla es extremadamente difícil. Los métodos más conocidos son los algoritmos F_4 y F_5, aunque utilizan mucha memoria y su tiempo de ejecución es considerablemente largo. Si se tienen memoria y tiempos ilimitados siempre se puede hallar una base de Gröbner, aunque por su gran costo, muchas veces es preferible el ataque a fuerza bruta para resolver el sistema de ecuaciones.

2.2.2. El ataque de Kipnis y Shamir

Otro ataque en contra del criptosistema HFE fue el ataque de Kipnis y Shamir el cual fue propuesto en 1992 [12]. Dicho ataque pretende transformar L_1 y L_2 de su representación matricial a una representación polinomial y convertir los n polinomios cuadráticos de P en una representación matricial para después resolver la ecuación fundamental asociada, la idea principal es asociar L_1, L_2 y P a funciones $\widehat{L}_1$, $\widehat{L}_2$ y $\widehat{P}$ sobre K. Finalmente usar la condición del rango de polinomio HFE, $F(x)$ para determinar L_1, L_2 y $F(x)$

Recordemos que la clave pública P para el criptosistema HFE está dada de la siguiente forma

$$P = L_1 \circ \varphi \circ F \circ \varphi^{-1} \circ L_2$$

donde L_1, L_2 son trasformaciones afines invertibles, F es un polinomio HFE con límite D y φ es un isomorfismo de espacios vectoriales. Definimos $r = \log_q D$ y supongamos además que q es impar.

Definición 2.2.15. Para una función G con $qHWD(G) = 2$, es decir si G es de la forma

$$G(X) = \sum_{j=0}^{n-1} \alpha_{ij} X^{q^i+q^j} + \sum_{j=0}^{n-1} \beta_j X^{q^j} + \gamma \in K[x],$$

y A es una matriz de dimensiones $n \times n$ con entradas α_{ij} cuando estén definidas y 0 en otro caso, la *matriz asociada a* G es

$$M = \frac{1}{2}(A + A^T).$$

La matriz asociada a un polinomio HFE tiene la forma $\left(\begin{smallmatrix} * & 0 \\ 0 & 0 \end{smallmatrix}\right)$, donde * es un bloque de dimensiones $r \times r$ de entradas distintas de cero.

A continuación se describe el ataque. Dado que P es pública, se conoce k, n y $|K|$. Podemos suponer conocidos K y φ, ya que todos los campos finitos de la misma cardinalidad son isomorfos. Además los campos de cardinalidad q^n tienen un único subcampo para cada divisor de n, así la estructura del subcampo se preserva [14].

Se define una función $\widehat{P} : K \to K$, mediante

$$\widehat{P} = \varphi^{-1} \circ P \circ \varphi = (\varphi^{-1} \circ L_1 \circ \varphi) \circ F \circ (\varphi^{-1} \circ L_2 \circ \varphi)$$

Sean $\widehat{L}_1 = \varphi^{-1} \circ L_1 \circ \varphi$ y $\widehat{L}_2 = \varphi^{-1} \circ L_2 \circ \varphi$ entonces

$$\widehat{P} = \widehat{L}_1 \circ F \circ \widehat{L}_2. \tag{2.3}$$

Por el lema 2.2.1 $\widehat{L}_1$ y $\widehat{L}_2$ tienen q grado igual a uno, es decir

$$\widehat{L}_1(X) = \sum_{j=0}^{n-1} l_{1j} X^{q^j},$$

$$\widehat{L}_2(X) = \sum_{j=0}^{n-1} l_{2j} X^{q^j},$$

para algunos $l_{ij} \in K$. Así $\widehat{P}$ tiene q grado igual a 2, con lo que

$$\widehat{P}(X) = \sum_{j=0}^{n-1} a_{ij} X^{q^i+q^j} + \sum_{j=0}^{n-1} b_j X^{q^j} + c \in K[X].$$

Debido a Eq. (2.3) se tiene $\widehat{L}_1^{-1} \circ \widehat{P}(X) = F \circ \widehat{L}_2(X)$ y

$$\widehat{P} = \sum_{k=0}^{n-1} s_k \widehat{G}^k = W \cdot \mathcal{F} \cdot W^T \tag{2.4}$$

donde $\widehat{g}_{ij}^k = (\widehat{p}_{i-k \bmod n,\ j-k \bmod n})^{q^k}$, $w_{ij} = s_{j-i \bmod n}^{q^i}$ y $\mathcal{F}$ es la representación matricial del polinomio HFE. Por lo que el rango de la matriz $W \cdot \mathcal{F} \cdot W^T$ es menor o igual a r, lo cual indica que se pueden calcular los coeficientes s_k de la Eq. (2.4), esto equivale al problema del rango mínimo el cual se considera NP-difícil.

2.3. Sistema Square

En el año 2009, Crystal Clough desarrolló el sistema Square [4]. Para la construcción del sistema se hacen uso de las fortalezas de algunos sistemas que hemos visto, digamos tomar como función central un polinomio de grado pequeño, así como quitar debilidades que llegaron a tener dichos sistemas. Por ejemplo, para el descifrado habrán dos soluciones de $X = F^{-1}(Y)$, si consideramos L_2 una transformación afín inyectiva, sólo una raíz produce el mensaje m.

Descripción

Sea k un campo de cardinalidad q, con $q \equiv 3 \pmod 4$. El texto en claro será un vector en k^n. Se incrusta k^n en un espacio $k^{n+\ell}$ vía una función afín inyectiva $L_2 : k^n \to k^{n+\ell}$. Se eligen n y ℓ de tal forma que $n + \ell$ sea impar, se sugiere que ℓ tome valores pequeños, por ejemplo $\ell = 1$ o $\ell = 3$. Los valores recomendados son:

$$q = 31, 43 \quad n \in [20, 35] \quad \ell = 1, 3.$$

Sean $K \cong k[y]/\langle g(y) \rangle$ un extensión de grado $n + \ell$ de k, $\varphi : K \to k^{n+\ell}$,

$$\varphi(a_1 + a_2 y + \ldots + a_{n+\ell} y^{n+\ell-1}) = (a_1, \ldots, a_{n+\ell}).$$

y una función afín invertible $L_1 : k^{n+\ell} \to k^{n+\ell}$. Como función central

$$F(X) = X^2$$

y como clave pública

$$P = L_1 \circ \varphi \circ F \circ \varphi^{-1} \circ L_2.$$

P será una $(n+\ell)$-tupla de polinomios cuadráticos en n variables

$$P(x_1, \ldots, x_n) = \begin{pmatrix} P_1(x_1, \ldots, x_n) \\ P_2(x_1, \ldots, x_n) \\ \vdots \\ P_{n+\ell}(x_1, \ldots, x_n) \end{pmatrix}$$

Cifrado Para cifrar un texto en claro $(m_1, \ldots, m_n) \in k^n$ únicamente calculamos

$$(c_1, \ldots, c_{n+\ell}) = P(m_1, \ldots, m_n).$$

Descifrado Para descifrar un texto cifrado $(c_1, \ldots, c_{n+\ell}) \in k^{n+\ell}$ se calcula, en primer lugar,

$$Y = \varphi^{-1} \circ L_1^{-1}(c_1, \ldots, c_{n+\ell}) \in K.$$

Es necesario resolver $X^2 = Y$. Puesto que $q \equiv 3 \pmod 4$ y $n + \ell$ es impar, entonces $|K| \equiv 3 \pmod 4$. Para hallar las raíces se utiliza la fórmula

$$X = \pm Y^{\frac{q^{n+\ell}+1}{4}}. \tag{2.5}$$

Sean X_1 y X_2 soluciones de (2.5). Si $L_1 \circ \bar{F}(X_i) = (c_1, \ldots, c_{n+\ell})$, se considera a X_i como la raíz correcta. Por otro lado si $L_2(x) = A_2 x + B_2$, con A_2 una matriz de $(n+\ell) \times n$, otra forma de decidir la raíz correcta es considerar una matriz cuyos renglones formen una base de $A_2(\mathbb{F}_q^n)^\perp$, denotada por H. Si el producto de H por $\varphi(X_i) - B_2$ resulta ser cero, X_i se toma como la raíz correcta.

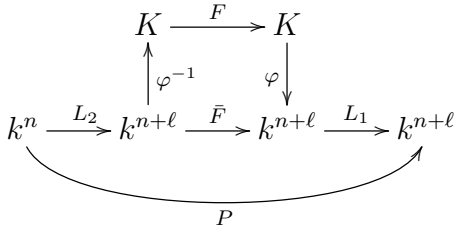


Figura 2.3: El sistema Square

2.3.1. Criptoanálisis de Square

Después del desarrollo de Square, distintos ataques criptográficos fueron usados para intentar romper este sistema y la mayoría no obtuvo éxito . Uno de ellos, el ataque de ecuaciones de linealización, el cual sirvió para romper MI, no logró un ataque satisfactorio pese a la cercanía en la descripción de los dos sistemas.
Otro ataque que fue usado contra Square, es el ataque algebraico. El cual tiene una seguridad mayor a 2^{80} con los valores estándar que son $n > 32$ y $q = 31$.
Por otro lado el ataque Kipnis-Shamir que fue usado para romper HFE, tiene la principal desventaja que para su implementación se requiere matrices cuadradas y en este caso se trabaja con matrices $n \times n + \ell$. Y aunque considerando las últimas ℓ variables como desconocidas o con un valor fijo, digamos cero, es imposible romper Square con este ataque. Es decir, considerar los $n + \ell$ polinomios de la clave pública de n variables como polinomios en $n + \ell$ variables y asociar una función $K \to K$. Sin embargo esto resulta más complicado que X^2, ya que no se cuenta con un rango bajo.

El ataque SFlash

El esquema de firma digital SFlash es una versión modificada de MI, en el cual se considera como clave pública los primeros $n-r$ polinomios de la clave pública de MI. El ataque para SFlash consiste en recuperar los r polinomios eliminados usando las propiedades diferenciales de la función central de MI. A continuación se describe el ataque SFlash contra Square.

Definición 2.3.1. Para una función f, el diferencial de f es

$$Df(a,x) = f(a+x) - f(a) - f(x) + f(0).$$

La clave pública de Square es una función $k^n \to k^{n+l}$. Podemos pensar que la clave pública P viene de un sistema MI de característica impar haciendo las últimas ℓ coordenadas igual a cero. De esta forma, supondremos que las dos transformaciones lineales son invertibles. Así, dicha transformación invertible la denotaremos por L_2'. Por lo que todos lo polinomios son parcialmente conocidos ya que las piezas faltantes son aquellas que involucran las variables $x_{n+1}, \ldots, x_{n+l}$.

El ataque SFlash explota las propiedades de la función central del sistema MI, primero se cumple la siguiente propiedad:

$$DF(\xi A, X) + DF(A, \xi X) = (\xi + \xi^{q^\theta})DF(A,X),$$

donde $DF(A,X)$ es el diferencial $DF(A,X) = F(A+X) - F(A) - F(X) + F(0)$.
Cuando $\theta = 0$ y la característica de k es mayor que dos, se habla del sistema Square, y la anterior propiedad se sigue cumpliendo.

Sean

$$\overline{F} = L_1 \circ \varphi \circ F \circ \varphi^{-1} \circ L_2',$$
$$\overline{N}_\xi = L_2'^{-1} \circ \varphi \circ M_\xi \circ \varphi^{-1} \circ L_2',$$

donde $M_\xi : K \to K$ es una multiplicación por $\xi \in K$; es decir, $M_\xi(X) = \xi X$.

Entonces para $\vec{a} = (a_1, \ldots, a_{n+\ell})$, $\vec{x} = (x_1, \ldots, x_{n+\ell})$, se consideran

$$A' = \varphi^{-1} \circ L_2'(\vec{a}),\ X' = \varphi^{-1} \circ L_2'(\vec{x}).$$

Así

$$\begin{aligned} D\overline{F}(\overline{N}_\xi(\vec{a}), \vec{x}) + D\overline{F}(\vec{a}, \overline{N}_\xi(\vec{x})) &= L_1 \circ \varphi(DF(M_\xi A', X') + DF(A', M_\xi X')) \\ &= L_1 \circ \varphi(M_{2\xi} \circ DF(A', X')) \\ &= L_1 \circ M_{2\xi} \circ \varphi^{-1} \circ L_1^{-1} \circ L_1 \circ \varphi(DF(A', X')) \\ &= L_1 \circ \varphi \circ M_{2\xi} \circ \varphi^{-1} \circ L_1^{-1}(D\overline{F}(\vec{a}, \vec{x})). \end{aligned}$$

Por lo tanto

$$D\overline{F}(\overline{N}_\xi(\vec{a}), \vec{x}) + D\overline{F}(\vec{a}, \overline{N}_\xi(\vec{x})) \subset gen_k\{D\overline{F}_i : i = 1, \ldots, n+\ell\}.$$

Concluyendo que $\overline{F}$ no está disponible para un atacante ya que se obtienen más ecuaciones y más incógnitas que en el sistema original.

Ataque Square

Este ataque usa diferenciales de forma distinta al ataque SFlash, y en este caso el ataque Square rompe el sistema [3]. Para explicar este ataque, se da una versión distinta de la clave pública.

La clave pública se puede definir mediante

$$P = L_1 \circ \overline{F} \circ L_2$$

donde

$$\begin{aligned}
&L_1 : k^{n+\ell} \to k^{n+\ell} \text{ es una transformación afín invertible,}\\
&\overline{F} = \varphi \circ F \circ \varphi^{-1} \text{ y } F(X) = X^2,\\
&L_2 : k^n \to k^{n+\ell} \text{ es una transformación afín inyectiva.}
\end{aligned}$$

Lo que sigue es trabajar sobre la extensión del campo k. Puesto que L_1 y L_2 son afines, se afirma que $\varphi^{-1} \circ L_1 \circ \varphi$ y $\varphi^{-1} \circ L_2 \circ \varphi'$ son afines, donde $\varphi' : \mathbb{F}_{q^n} \to k^n$.

Sean $L_1(x) = A_1 x^T + B_1$ la transformación afín invertible y $L_2 = A_2 x^T + B_2$ la transformación afín inyectiva. Se denotan $[a_1, \ldots, a_{n+\ell}]$ a las columnas de la matriz A_1. Considere α^j un elemento de la base para el campo K, con $0 \leq j \leq (n + \ell - 1)$. Entonces si para un j fijo $\varphi(\alpha^j) = (\beta_1, \ldots, \beta_{n+\ell})$, se tiene

$$\begin{aligned}
\varphi^{-1} \circ L_1 \circ \varphi(\alpha^j) &= \varphi^{-1} \circ L_1(\beta_1, \ldots, \beta_{n+\ell}) \text{ con } \beta_i \in k\\
&= \varphi^{-1}\left(A_1(\beta_1, \ldots, \beta_{n+\ell})^T + B_1\right)\\
&= \varphi^{-1}\left(\beta_1[a_1] + \ldots + \beta_{n+\ell}[a_{n+\ell}] + B_1\right)\\
&= \left(\beta_1[a_1] + \ldots + \beta_{n+\ell}[a_{n+\ell}] + B_1\right) \cdot (1, \alpha, \ldots, \alpha^{n+\ell-1}) \text{(producto punto)}\\
&= \left(\beta_1[a_1] + \ldots + \beta_{n+\ell}[a_{n+\ell}]\right) \cdot (1, \alpha, \ldots, \alpha^{n+\ell-1}) + B_1 \cdot (1, \alpha, \ldots, \alpha^{n+\ell-1})\\
&= A_1(\beta_1, \ldots, \beta_{n+\ell})^T \cdot (1, \alpha, \ldots, \alpha^{n+\ell-1})^T + B_1(1, \alpha, \ldots, \alpha^{n+\ell-1})\\
&= (A_1\varphi(\alpha^j)) \cdot (1, \alpha, \ldots, \alpha^{n+\ell-1})^T + B_1 \cdot (1, \alpha, \ldots, \alpha^{n+\ell-1})\\
&= \varphi^{-1}(A_1\varphi(\alpha^j)) + \varphi^{-1}(B_1).
\end{aligned}$$

Entonces $\varphi^{-1} \circ L_1 \circ \varphi(x) = \varphi^{-1}(A_1\varphi(x)) + \varphi^{-1}(B_1)$, por tanto $\varphi^{-1} \circ L_1 \circ \varphi = \hat{L}_1 + \hat{l}_1$ es afín tomando

$$\hat{L}_1(x) = \varphi^{-1}(A_1\varphi(x))$$

$$\hat{l}_1 = \varphi^{-1}(B_1).$$

Análogamente la transformación $\varphi^{-1} \circ L_2 \circ \varphi' = \hat{L}_2 + \hat{l}_2$ resulta afín, con

$$\hat{L}_2(x) = \varphi^{-1}(A_2\varphi'(x))$$

$$\hat{l}_2 = \varphi'^{-1}(B_2).$$

Sean

$$\hat{P} = \varphi^{-1} \circ P \circ \varphi',$$

$$X = \varphi^{-1}(\vec{x}) \text{ y } Y = \varphi^{-1}(\vec{y}).$$

Usando esta notación,

$$\begin{aligned}
\hat{P}(X) =& \hat{L}_1(\hat{L}_2(X)^2) + 2\hat{L}_1(\hat{l}_2\hat{L}_2(X)) + \hat{L}_1(\hat{l}_2^2) + \hat{l}_1\\
&= \text{cuadrático} + \text{lineal} + \text{constante.}
\end{aligned}$$

Ahora se consideran la parte lineal de $\hat{P}(X)$

$$\hat{P}_1(X) = 2\hat{L}_1(\hat{l}_2\hat{L}_2(X)), \text{ y el diferencial}$$

$$D\hat{P}(X,Y) = \hat{L}_1(2\hat{L}_2(X)\cdot\hat{L}_2(Y)).$$

Dado $A \in K$, si $\tilde{M}_A(W)$ denota la multiplicación por una constante la cual depende de A. Fijando X para algún $A \in K$, se define

$$D\hat{P}_A(X) = D\hat{P}(X,A) = \hat{L}_1 \circ \tilde{M}_A \circ \hat{L}_2(X),$$

con $\tilde{M}_A(W) = 2\hat{L}_2(A)W$, entonces $\{D\hat{P}_A : A \in K\}$ son funciones lineales y forman un espacio vectorial sobre k de dimensión n cuya base resulta $\{\hat{L}_1 \circ M_{\alpha^i} \circ \hat{L}_2 : \ 0 \leq i \leq n+\ell-1\}$.

Puesto que $\hat{P}_1(X) = 2\hat{L}_1(\hat{l}_2\hat{L}_2(X))$ también tiene la forma $\hat{L}_1 \circ \tilde{M}_A \circ \hat{L}_2$ con $\tilde{M}_A(W) = \hat{l}_2(W)$, entonces se obtiene el siguiente conjunto

$\Delta = \{\hat{L}_1 \circ M_{\alpha^i} \circ \hat{L}_2 : \ 0 \leq i \leq n+\ell-1\} \cup \{\hat{P}_1\} = \{D_i = \hat{L}_1 \circ M_{\lambda_i} \circ \hat{L}_2\}$ para algunos $\lambda_i = \alpha^i, i \neq n+\ell$.

Las funciones de Δ resultan útiles ya que ayudan a identificar $\hat{L}_1$. Dado que se satisface la siguiente ecuación

$$(\hat{L}_1 \circ M_\lambda \circ \hat{L}_1^{-1}) \circ (\hat{L}_1 \circ M_{\lambda_i} \circ \hat{L}_2) = \hat{L}_1 \circ M_{\lambda\lambda_i} \circ \hat{L}_2. \tag{2.6}$$

Por lo que se busca solución para L para el sistemas de ecuaciones

$$L \circ D_i \in gen\{D_j : j > m\}, \ i \leq m. \tag{2.7}$$

Si $L \circ D_i \in gen\{D_j : j > m\}$, entonces con $D_i = \hat{L}_1 \circ M_1 \circ \hat{L}_2 = \hat{L}_1 \circ \hat{L}_2$

$$\begin{aligned} L \circ D_i &= L \circ (\hat{L}_1 \circ \hat{L}_2) \\ &= \hat{L}_1 \circ M_\beta \circ \hat{L}_2, \quad \text{así} \end{aligned}$$

$$\begin{aligned} L &= \hat{L}_1 \circ M_\beta \circ \hat{L}_2 \circ \hat{L}_2^{-1} \circ \hat{L}_1^{-1} \\ &= \hat{L}_1 \circ M_\beta \circ \hat{L}_1^{-1} \end{aligned}$$

Una vez conocida $L = \hat{L}_1 \circ M_\beta \circ \hat{L}_1^{-1}$, M_λ puede ser hallada. Calculando los valores propios de L

$$\begin{aligned} |[L] - xI| &= |[\hat{L}_1 \circ M_\lambda \circ \hat{L}_1^{-1}] - x[\hat{L}_1 \circ \hat{L}_1^{-1}]| \\ &= |[\hat{L}_1][M_\lambda][\hat{L}_1^{-1}] - x[\hat{L}_1][\hat{L}_1^{-1}]| \\ &= |[\hat{L}_1]||[M_\lambda] - xI||[\hat{L}_1^{-1}]| \\ &= |[M_\lambda] - xI| \end{aligned}$$

así λ es un valor propio de L y por tanto $\lambda, \lambda^q, \ldots, \lambda^{q^{n+\ell-1}}$ resultan ser valores propios de L.

$\hat{L}_1$ puede ser calculada a partir de λ porque $\hat{L}_1$ es una solución de la ecuación

$$L \circ \hat{L}_1 = \hat{L}_1 \circ M_\lambda.$$

Una vez que $\hat{L}_1$ es conocido, $\hat{L}_2$ puede ser determinado. Si $\hat{P}_2$ es la parte cuadrática de $\hat{P}$, entonces para cualquier $a \in K$, como

$$\hat{L}_1^{-1}(\hat{P}_2(a)) = (\hat{L}_2(a))^2,$$
$$\hat{L}_1^{-1}(\tfrac{1}{2}\hat{P}_1(a)) = \hat{l}_2\hat{L}_2(a),$$

se cumple que

$$\hat{L}_2(X) = \frac{\sqrt{\hat{L}_1^{-1}(\hat{P}_2(a))}}{\hat{L}_1^{-1}(\frac{1}{2}\hat{P}_1(a))}\hat{L}_1^{-1}(\tfrac{1}{2}\hat{P}_1(X)).$$

En el momento que $\hat{L}_1$ y $\hat{L}_2$ se encuentran, L_1 y L_2 se pueden calcular y el sistema es roto.

Square+

La modificación + en los Criptosistemas multivariables de clave pública, se realiza agregando algún número p de polinomios cuadráticos a la clave pública antes de la transformación final. Enseguida se describe este proceso en el caso de Square+.

Sea k un campo de cardinalidad q, donde $q \equiv 3 \pmod 4$. Los textos en claro son vectores en k^n, se incrusta el espacio de textos en claro en $k^{n+\ell}$, y supóngase que K es una extensión de grado $n+\ell$ de k. Al igual que Square, se hace uso de las funciones: el isomorfismo de espacios vectoriales $\varphi : K \to k^{n+\ell}$ dada por

$$a_1 + a_2 y + \cdots + a_{n+\ell} y^{n+\ell-1} \mapsto (a_1, \ldots, a_{n+\ell}),$$

la función central $F : K \to K$ dada por

$$F(X) = X^2$$

y una función afín inyectiva $L_2 : k^n \to k^{n+\ell}$. También se usan p polinomios cuadráticos en $n+\ell$ variables,

$$g_1, \ldots, g_p \in k[x_1, \ldots, x_{n+\ell}]$$

y una función afín invertible $L_1 : k^{n+\ell+p} \to k^{n+\ell+p}$. Ya que $\varphi \circ F \circ \varphi^{-1}$ es una $(n+\ell)-$tupla de polinomios cuadráticos, agregando $g_1, \ldots, g_p$ se crea una función $\bar{F}^+ : k^{n+\ell+p} \to k^{n+\ell+p}$. La clave pública será

$$P^+ = L_1 \circ \bar{F}^+ \circ \ L_2.$$

P^+ está compuesta por una $(n+\ell+p)-$tupla de polinomios cuadráticos

$$P^+(x_1, \ldots, x_n) = \begin{pmatrix} P_1^+(x_1, \ldots, x_n) \\ P_2^+(x_1, \ldots, x_n) \\ \vdots \\ P_{n+\ell+p}^+(x_1, \ldots, x_n) \end{pmatrix}$$

Cifrado Un texto en claro $(m_1, \ldots, m_n) \in k^n$ se cifra mediante el cálculo

$$(c_1, \ldots, c_{n+\ell+p}) = P^+(m_1, \ldots, m_n).$$

Descifrado Para un texto cifrado $(c_1, \ldots, c_{n+\ell+p})$, se descifra como sigue: primero, sean

$$(y_1, \ldots, y_{n+\ell+p}) = L_1^{-1}(c_1, \ldots, c_{n+\ell+p})$$
$$Y = \varphi^{-1}(y_1, \ldots, y_{n+\ell}) \in K.$$

En seguida se resuelve $X^2 = Y$, cuyas soluciones se hallan usando la fórmula de la raíz

$$X = \pm Y^{\frac{q^{n+\ell}+1}{4}},$$

donde sólo una de ellas será la correcta.

Seguridad

Para la creación de los modelos que se proponen en esta tesis (PR y PHFER) se trabaja con Square+. La ventaja que tiene este sistema es que al ser aleatorios los polinomios cuadráticos que se añaden a clave pública, éstos permiten crear ruido ante un ataque. Por un lado, únicamente el usuario legítimo puede separar dichos polinomios de la clave para realizar un descifrado más rápido y por otro lado se logra que el texto cifrado sea más largo que el texto en claro. La pregunta que surge es qué reacción tienen dichos polinomios a los ataques ya conocidos.

En primera instancia, el valor de p no debe de ser muy grande (2 o 3), de lo contrario el sistema se puede romper mediante el ataque de bases de Gröbner ya que el sistema estaría sobredeterminado. Puesto que Square es resistente al ataque Kipnis-Shamir, es de esperarse que Square+ sea también resistente a este ataque. Ya que la clave pública consiste de $n + \ell + p$ polinomios en n variables, al realizar este ataque se incrementa el rango de las forma cuadráticas que se desarrollan en el ataque ya que la diferencia entre el número de polinomios y el número de variables incrementa. Finalmente, Square+ resiste el ataque diferencial ya que los polinomios cuadráticos que se añaden a la clave pública ejercen el ruido suficiente a las funciones diferenciales de dicho ataque, con lo que resiste también el ataque SFlash y el ataque Square. Así concluimos que Square+ es una variación segura de Square.

Square-

Esta variación del criptosistema Square oculta las propiedades diferenciales que posee, eliminando r polinomios de la clave pública. Estas construcciones suelen no ser inyectivas, sin embargo, la colección de todos los textos claros posibles es sólo una fracción del tamaño del espacio vectorial. Por lo que al considerarse parámetros adecuados se puede lograr que el sistema sea viable.

Cifrado Un texto en claro $(m_1, \ldots, m_n) \in k^n$ se cifra mediante el cálculo

$$(c_1, \ldots, c_{n+\ell-r}) = P^-(m_1, \ldots, m_n)$$

Descifrado Para un texto cifrado $(c_1, \ldots, c_{n+\ell-r}) \in k^{n+\ell-r}$, se debe adivinar el valor de

$$(c_{n+\ell-r+1}, \ldots, c_{n+\ell}).$$

Se recomienda que r sea pequeño. Para cada suposición $\tilde{c}_1, \ldots, \tilde{c}_r$ de los componentes perdidos, se intenta descifrar $(c_1, \ldots, c_{n+\ell-r}, \tilde{c}_1, \ldots, \tilde{c}_r)$. En promedio se tienen que hacer $\frac{q^r}{2}$ valores supuestos antes de hallar el correcto.

Ejemplo 2.3.1. A continuación, daremos un ejemplo del sistema Square+. Los mensajes m son de longitud $n = 22$, $m = (m_1, \ldots, m_{22})$, con $m_i \in \mathbb{F}_{31}$; daremos como parámetro $\ell = 3$, así $n + \ell$ es impar. Además sean K una extensión de grado 25 de $\mathbb{F}_{31}$, φ el isomorfismo que hemos considerado anteriormente, como función central $F(X) = X^2$, $L_2 : \mathbb{F}_{31}^{22} \to \mathbb{F}_{31}^{25}$ una transformación inyectiva afín, $L_1 : \mathbb{F}_{31}^{28} \to \mathbb{F}_{31}^{28}$ una transformación biyectiva afín, y 3 polinomios aleatorios en 25 variables con coeficientes en $\mathbb{F}_{31}$. No se muestran L_1 y L_2 por motivos de espacio.

Consideramos $m \in \mathbb{F}_{31}^{22}$ de la siguiente forma

$$m = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22)$$

Cifrado Para cifrar simplemente se evalúa $c = P^+(m)$,

$$P^+(m) = (18, 2, 10, 27, 1, 3, 6, 13, 26, 23, 23, 29, 10, 7, 12, 30, 27, 6, 0, 29, 30, 12, 4, 15, 6, 27, 29, 15)$$

Descifrado

- Como un primer paso para descifrar, al recibir c se calcula $L_1^{-1}(c)$,

$$L_1^{-1}(c) = (7, 11, 13, 0, 29, 27, 8, 3, 2, 17, 25, 19, 13, 17, 24, 19, 6, 9, 0, 17, 3, 22, 10, 9, 26, 0, 11, 28)$$

- evaluar φ^{-1} a las primeras $n+\ell$ entradas del resultado anterior, denotado por w. Calcular w^{exp} y $-w^{exp}$, donde $exp = (q^{n+\ell}+1)/4 = 4808198122482839583459328499691967688$.
- Evaluar $\varphi(\pm w^{exp})$ para obtener las dos preimagenes de X^2. Sólo queda decidir cual de las dos raíces que la correcta (cual de ellas cifra c). Donde $-w^{exp}$ es la raíz correcta.

2.4. Sistemas de aceite y vinagre

En seguida se presenta un sistema de cifrado multivariable, este criptosistema se compone de un sistema de ecuaciones cuadráticas multivariables fáciles de resolver, disfrazadas por dos transformaciones afines invertibles. El sistema inicial consiste de o ecuaciones polinomiales en n variables sobre un campo finito k. Las primeras v variables se refieren a las *variables de vinagre* y el resto de ellas son las *variables de aceite*. Este nombre lo lleva debido a que inicialmente las variables son separadas en distintas capas, y después son mezcladas mediante una transformación afín.

Sea k un campo finito con q elementos y sea $n = o + v$ un entero con o, v enteros positivos.

Definición 2.4.1. Un *polinomio de aceite y vinagre* es cualquier polinomio de grado dos $f \in k[x_1, \ldots, x_o, x'_1, \ldots, x'_v]$ de la forma

$$f = \sum_{i=1}^{o}\sum_{j=1}^{v} a_{ij}x_i x'_j + \sum_{i=1}^{v}\sum_{j=1}^{v} b_{ij}x'_i x'_j + \sum_{i=1}^{o} c_i x_i + \sum_{j=1}^{v} d_j x'_j + e,$$

con $a_{ij}, b_{ij}, c_i, d_j, e \in k$, donde x_i, $i = 1, \ldots, o$, son las variables de aceite y x'_j, $j = 1, \ldots, v$ son las variables de vinagre.

De ahí viene el nombre de polinomio de aceite y vinagre, ya que los términos cuadráticos de las variables aceite y vinagre no se encuentran completamente mezcladas (como el aceite al revolverlo con vinagre).

Definición 2.4.2. Sea $F : k^n \to k^o$ una función polinomial de la forma

$$F(x_1, \ldots, x_o, x'_1, \ldots, x'_v) = (f_1, \ldots, f_o),$$

donde $f_1, \ldots, f_o \in k[x_1, \ldots, x_o, x'_1, \ldots, x'_v]$ son polinomios de aceite y vinagre. Entonces F es llamada una *función de aceite y vinagre*.

El sistema de aceite y vinagre balanceado es caracterizado por $o = v$, para el caso del sistema de aceite y vinagre no balanceado se debe de cumplir $v > o$.

2.4.1. Rainbow

El sistema Rainbow es una generalización de los sistemas de aceite y vinagre no balanceados multicapas. A continuación se presenta la construcción de este sistema y después se presenta un ejemplo.

Descripción

Para un entero positivo n sea $S = \{1, 2, \ldots, n\}$. Se eligen $0 < u < n$ y $v_1, v_2, \cdots, v_u$ enteros tales que $0 < v_1 < v_2 < \cdots < v_u = n$. Se definen $S_\ell = \{1, \ldots, v_l\}$ donde $\ell \in \{1, \ldots, u\}$.
Sean $o_i = v_{i+1} - v_i$ para cada $i \in \{1, \ldots, u-1\}$ y $\mathcal{O}_i = S_{i+1} - S_i$ para cada $i \in \{1, \ldots, u-1\}$. Notemos en primera instancia que

$$S_1 \subset S_2 \subset \cdots \subset S_u = S.$$

Por otro lado se observa que $|S_i| = v_i$.
Sea P_l el k-espacio vectorial de polinomios cuadráticos generados por polinomios de siguiente forma

$$f(x_1, \ldots, x_n) = \sum_{i \in \mathcal{O}_l, j \in \mathcal{S}_l} \alpha_{i,j} x_i x_j + \sum_{i,j \in \mathcal{S}_l} \beta_{i,j} x_i x_j + \sum_{i \in \mathcal{S}_{l+1}} \gamma_i x_i + \eta$$

Llamamos a cualquier polinomio en P_l una *l-ésima capa polinomial de aceite y vinagre.*
La función central para el sistema Rainbow $F : \mathbb{F}^n \to \mathbb{F}^{n-v_1}$ es definida como (denotando $(x_1, \ldots, x_n)$ como $\bar{x}$):

$$F(\bar{x}) = (\tilde{F}_1(\bar{x}), \ldots, \tilde{F}_{u-1}(\bar{x})) = (F_1(\bar{x}), \ldots, F_{n-v_1}(\bar{x}))$$

donde cada $\tilde{F}_i$ consiste de o_i polinomios elegidos aleatoriamente de P_i.
F tiene $u-1$ capas de aceite y vinagre. La primera capa consiste de o_1 polinomios $F_1, \ldots, F_{o_1}$ tal que $x_j, j \in O_1$ son las variables de aceite y $x_j, j \in S_1$ son las variables de vinagre, esto para cada i-ésima capa.

F es un polinomio Rainbow con $u-1$ capas. La clave pública es generada de la forma habitual aplicando dos transformaciones afines, L_1 y L_2, donde

$$L_1 : \mathbb{F}^{n-v_1} \to \mathbb{F}^{n-v_1},\ L_2 : \mathbb{F}^n \to \mathbb{F}^n,\ P = L_1 \circ F \circ L_2$$

Cifrado Dado un mensaje m, se cifra mediante la evaluación $P(m)$.

Descifrado Dado que los coeficientes de F son elegidos aleatoriamente, entonces dado un mensaje cifrado $(y'_1, \ldots, y'_o) \in k^o$ se puede invertir F mediante una elección aleatoria $(\bar{x}'_1, \ldots, \bar{x}'_v) \in k^v$ de variables de vinagre y resolver el sistema de ecuaciones lineales resultantes dado por

$$F(x_1, \ldots, x_o, \bar{x}'_1, \ldots, \bar{x}'_v) = (y'_1, \ldots, y'_o).$$

Si el sistema no tiene solución, se seleccionan distintos valores para las variables de vinagre hasta encontrar una solución. La función F se transforma mediante dos funciones afines invertibles L_1 y L_2. Ya que L_1 va de k^o en sí mismo y L_2 va de k^{o+v} en sí mismo, esto genera una función cuadrática:

$$\bar{F} = L_1 \circ F \circ L_2.$$

Ejemplo 2.4.1. Consideramos el campo $\mathbb{F}_{11}$ y $n = 10$. Siguiendo la descripción del sistema Rainbow, se elige $u = 4$, $v_1 = 2, v_2 = 3, v_3 = 7$ y $v_4 = 10$, entonces $F(x)$ es un polinomio Rainbow de tres capas con 8 polinomios, es decir $F(x) = \left(\tilde{F}_1(x), \tilde{F}_2(x), \tilde{F}_3(x)\right)$ con

$$\tilde{F}_1(\bar{x}) = \left[\ 2x_3x_1 + x_3x_2 + x_1x_2 + x_2x_1 + 7x_1 + 10x_2 + x_3 + 1\ \right]$$

$$\tilde{F}_2(\bar{x}) = \begin{bmatrix} 3x_4x_3 + 10x_7x_1 + 2x_1x_2 + 7x_2x_3 + 5x_4 + 3x_5 + x_6 + 8x_7 \\ 4x_5x_1 + 9x_7x_2 + 7x_6x_3 + x_1x_3 + 3x_2x_1 + 2x_2 + 5x_4 + 5 \\ x_4x_1 + 5x_6x_3 + 8x_5x_2 + 9x_1x_2 + 3x_2x_3 + x_1x_3 + 7x_7 + 5x_6 \\ 5x_5x_3 + 10x_6x_1 + x_7x_2 + 8x_1x_3 + x_3x_2 + 10x_7 + 3x_5 + 9 \end{bmatrix}$$

$$\tilde{F}_3(\bar{x}) = \begin{bmatrix} x_{10}x_7 + 2x_9x_4 + 8x_8x_1 + x_9x_6 + 10x_3x_4 + 5x_6x_2 + x_{10}x_1 + 4 \\ 9x_{10}x_6 + 3x_9x_4 + 8x_8x_1 + x_9x_6 + 10x_3x_4 + 5x_6x_2 + x_{10}x_1 + 4 \\ 4x_8x_7 + 2x_{10}x_3 + 5x_8x_1 + 3x_3x_5 + 5x_7x_1 + 7x_3x_2 + 4x_9x_5 \end{bmatrix}$$

Análogamente se eligen $L_1(x) = A_1x^T + B_1$ y $L_2(x) = A_2x^T + B_2$ afines invertibles para transformar $F(x)$, de tal forma que $P(x) = L_1 \circ F \circ L_2(x)$, donde:

$$A_1 = \begin{pmatrix} 2 & 6 & 10 & 2 & 10 & 1 & 6 & 3 \\ 4 & 4 & 10 & 2 & 10 & 1 & 6 & 3 \\ 6 & 5 & 4 & 9 & 1 & 10 & 5 & 8 \\ 8 & 3 & 5 & 8 & 1 & 10 & 5 & 8 \\ 10 & 1 & 9 & 4 & 7 & 8 & 4 & 2 \\ 1 & 10 & 2 & 7 & 2 & 7 & 1 & 6 \\ 3 & 8 & 5 & 10 & 6 & 5 & 4 & 4 \\ 5 & 6 & 10 & 2 & 10 & 1 & 6 & 6 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 1 \\ 4 \\ 5 \\ 6 \\ 3 \\ 2 \\ 10 \\ 6 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} 2 & 6 & 10 & 2 & 10 & 1 & 6 & 3 & 4 & 9 \\ 4 & 4 & 10 & 2 & 10 & 1 & 6 & 3 & 4 & 9 \\ 6 & 5 & 4 & 9 & 1 & 10 & 5 & 8 & 7 & 2 \\ 8 & 3 & 5 & 8 & 1 & 10 & 5 & 8 & 7 & 2 \\ 10 & 1 & 9 & 4 & 7 & 8 & 4 & 2 & 10 & 6 \\ 1 & 10 & 2 & 7 & 2 & 7 & 1 & 6 & 8 & 7 \\ 3 & 8 & 6 & 10 & 6 & 5 & 4 & 4 & 9 & 1 \\ 5 & 6 & 10 & 2 & 10 & 1 & 6 & 6 & 9 & 1 \\ 7 & 4 & 3 & 5 & 3 & 8 & 4 & 2 & 7 & 0 \\ 4 & 7 & 8 & 6 & 8 & 3 & 7 & 9 & 1 & 3 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 10 \\ 5 \\ 3 \\ 2 \\ 1 \\ 7 \\ 8 \\ 9 \\ 4 \\ 9 \end{pmatrix}$$

Cifrado Si $m = (2, 3, 4, 7, 10, 1, 5, 5, 10, 8)$, entonces para cifrar simplemente se evalúa $P(m)$

$$P(x) = L_1 \circ F \circ L_2(x) = (4, 6, 8, 4, 6, 0, 1, 3)$$

Descifrado Dado el mensaje cifrado $c = (4, 6, 8, 4, 6, 0, 1, 3)$. El proceso de descifrado comienza al calcular $L_1^{-1}(c) = (y_1', y_2', \ldots, y_8')$, entonces

- $L_1^{-1}(c) = (6, 1, 8, 1, 1, 10, 1, 3)$.
- Se eligen (x_1', x_2') variables de vinagre y se resuelve el sistema de ecuaciones $\tilde{F}_1(x_1', x_2', x_3)$=6 (si es soluble). Que en este caso se compone de una ecuación con una incógnita, x_3. Si consideramos como variables de vinagre la pareja $(1, 8)$ se obtiene que el primer sistema es indeterminado por lo que es necesario ocupar distintas variables de vinagre, digamos $(3, 7)$, para el cual $\tilde{F}_1(x_1', x_2', x_3) = 6$ si tiene solución.
- En seguida, las variables de vinagre y el valor obtenido, $(3, 7, 5)$, serán sustituidos en la siguiente capa $\tilde{F}_2$ y se resuelve el sistema de ecuaciones asociado (de ser soluble), cuya solución obtenida es $(10, 3, 8, 5)$.
- Los valores que se han obtenido $(3, 7, 5, 10, 3, 8, 5)$ serán sustituidos en $\tilde{F}_3$ y se resuelve el sistema de 3 ecuaciones con 3 incógnitas asociado, obteniendo $(3, 7, 5, 10, 3, 8, 5, 9, 6, 1)$.
- Finalmente se calcula $L_2^{-1}(3, 7, 5, 10, 3, 8, 5, 9, 6, 1)$, obteniendo el mensaje original m.

2.4.2. Criptoanálisis

A continuación se describen distintos ataques en contra del criptosistema Rainbow. Uno de ellos, el método de reducción de rango, no es aplicable en contra del sistema y se explicarán brevemente las razones. En otros ataques, por ejemplo el método de ataque para esquemas de aceite y vinagre o el método de rango mínimo, se verán su nivel de seguridad. Finalmente se concluirá que el sistema Rainbow es unos de los criptosistemas de clave multivariable más seguros que existen en la actualidad.

Método de reducción de rango

Jintai Ding y Dieter Schmidt en 2005 desarrollaron el sistema Rainbow [7], en cuya implementación lograron concluir que no se puede encontrar un ataque de complejidad inferior a 2^{80} (número de operaciones necesarias para ejecutar un algoritmo). En ese trabajo se consideraron los valores $q = 2^8$, $n = 33$. El polinomio Rainbow se compone de 4 capas con un total de 27 polinomios, además, $v_1 = 6$, $v_2 = 12$, $v_3 = 17$, $v_4 = 22$, $v_5 = 33$, para más detalles vea el apéndice C. Podríamos pensar que es posible usar el ataque de reducción de rango en contra del sistema Rainbow ya que el espacio generado por los componentes polinomiales del cifrado de Shamir (cifrado en el cual se uso por primera vez este ataque) se comportan de la siguiente manera

$$V_1 \subset V_2 \subset \ldots \subset V_t,$$

donde V_i es el espacio generado por los componentes polinomiales del cifrado, cada V_i es subconjunto propio de V_{i+1}, tal como sucede en el sistema Rainbow. Sin embargo, la diferencia de dimensiones entre ellos es uno para el sistema Shamir y es mayor que uno en el caso Rainbow. Esta es la razón más importante por lo cual es imposible usar dicho ataque en contra de Rainbow ya que se requiere que la diferencia de dimensiones sea uno.

El ataque que usa la estructura de múltiples capas

En el cifrado Matsumoto e Imai, Patarin se percató que si un cifrado estaba formado por varias capas independientes y paralelas, se puede hacer una separación de variables de tal forma que todos los polinomios sean derivados como combinaciones lineales de polinomios sobre cada grupo de variables. Sin embargo en el sistema las capas del polinomio Rainbow no son independientes, ya que cada capa está relacionada a la anterior. Es por ello que no se puede hacer una separación de variables. Así este ataque tampoco puede ser usado en contra del sistema Rainbow.

Otros dos ataques que se usaron en contra de Rainbow son: el ataque de rango mínimo y el ataque para sistemas de aceite y vinagre. El primero tiene una complejidad mayor a 2^{100} y el segundo tiene una complejidad de 2^{81} para los valores que se dieron en Apen. C.

En general, se puede atacar al sistema desde la primera capa o desde la última. La seguridad de la última capa depende de la eficiencia del método del rango mínimo. La complejidad del ataque es $q^{(v_2-1)}o_{u-1}^3$ si $v_1 > o_1$ o $q^{2v_1}o_{u-1}^3$ si $v_1 \leq o_1$, con lo que $v_2 = o_1 + v_1$ no debe ser muy pequeño. La seguridad del sistema es al menos $(n - v_1) \times n^3 \times q^{o_1+v_1} \times u$. Para el caso de un ataque desde la primera capa, el ataque de aceite y vinagre no balanceado indica que $v_{u-1} - o_{u-1}$ no debe ser muy pequeño.
Si se considera un campo finito de cardinalidad q^r, entonces el nivel de seguridad viene dado por el valor $2^{3r(v_2-1)}$, se debe elegir $v_1 > o_1$ para hacer más eficiente el sistema. Si se requiere un nivel de seguridad de 2^θ, se debe cumplir que $v_2 = o_1 + v_1$ sea al menos $1 + \theta/3r$. Por último si se desea tener la clave privada lo más pequeña posible para facilitar los cálculos del descifrado, la diferencia entre o_1 y v_1 debe se 0 o 1.

Capítulo 3

Nuevos modelos de cifrado

3.1. Cifrado PHFER

En seguida se presenta un nuevo modelo de cifrado, al cual denominamos PHFER. Este sistema, como sus siglas lo indican, es desarrollado con tres distintos cifrados, Square+, HFE y Rainbow. Dado un mensaje m de tamaño $n = s + \beta + r$ (s, β, r enteros positivos), el proceso de cifrado se realizará por partes. En la figura 3.1 se muestra un diagrama del sistema. La primera parte del mensaje se cifra usando Square+ (las primeras s entradas), la segunda parte, las siguientes $\beta + 2$ entradas, se cifra con HFE y las últimas r entradas se cifrarán usando Rainbow. Esta elección de cifrado permite que ante un ataque, si se logra descifrar una parte del cifrado, no todo el mensaje sea descubierto, a no ser que se rompa todo el sistema, junto con dos transformaciones afines e invertibles L_1 y L_2.

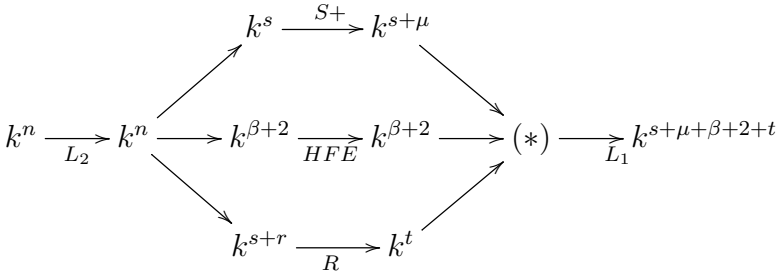


Figura 3.1: Cifrado PHFER, donde $t = s + r - v$ y $(*)$ es la concatenación de los tres cifrados.

Construcción de la clave. Sean m un mensaje de longitud n con entradas en $\mathbb{F}_q$. La función central de PHFER es una concatenación del cifrado Square+, HFE y Rainbow. Donde las primeras s entradas del mensaje son cifradas usando Square+. Para el cifrado HFE se consideran las entradas $s-2, s+1, \ldots, \beta$. Finalmente usando Rainbow se cifran las últimas r entradas, las cuales se concatenan con las primeras s entradas de β. La descripción de cada uno de estos sistemas se muestra a continuación.

- La función Square+ se define como $\mathcal{F}_{S^+} : k^n \to k^{n+\ell+p}$, que resulta de la siguiente composición:

$$k^n \xrightarrow{L_2} k^{n+\ell} \xrightarrow{\varphi_1} K \xrightarrow{\bar{F}^+} K \xrightarrow{\varphi_2^{-1}} k^{n+\ell+p}$$

  con $\bar{F}^+$ la función central de Square+.

- La función HFE es definida como $\mathcal{F}_{HFE} : k^n \to k^n$ que resulta de la siguiente composición:

$$k^n \xrightarrow{\varphi} K \xrightarrow{f} K \xrightarrow{\varphi^{-1}} k^n$$

  donde f es una función HFE.

- La función Rainbow
$$\mathcal{F}_R = (g_1, \ldots, g_{n-v_1}) : k^n \to k^{n-v_1}$$
haciendo uso de la construcción del sistema de cifrado Rainbow, donde $S = \{1, 2, \ldots, n\}$ y $\mathcal{O}_i = S_{i+1} - S_i$ y cada polinomio cuadrático $g_l \in P_l$.

Se utilizan dos transformaciones afines invertibles $L_1 : k^{s+\mu+\beta+2+t} \to k^{s+\mu+\beta+2+t}$ y $L_2 : k^n \to k^n$. La clave pública P está dada por la concatenación de los tres cifrados antes mencionados, es decir:

$$P = L_2 \circ F \circ L_1,$$

donde $F = F_{S+}||F_{HFE}||F_R$.

Cifrado Dado $m \in k^n$, el texto cifrado se obtiene calculando $c = P(m)$.

Descifrado Sea $c = P(m) \in k^{s+\mu+\beta+t}$ un mensaje cifrado, el proceso de descifrado es:

1. Se calcula $L_1^{-1}(c)$, el resultado será separado nuevamente en tres partes.
2. Las primeras $s + \mu$ entradas de $L_1^{-1}(c)$, se descifran con Square+, obteniendo s variables de vinagre, que denotaremos por $(y_1, \ldots, y_s)$.
3. Las siguientes $\beta + 2$ entradas de $L_1^{-1}(c)$ se descifran usando HFE (se consideran únicamente las últimas β entradas para el resto del descifrado) denotadas por y'.
4. Sustituir las s variables de vinagre en cada g_i del cifrado Rainbow, resolver los sistemas de ecuaciones asociados y hallar las r entradas faltantes, las cuales denotaremos mediante y''.
5. Sea $y = (y_1, \ldots, y_s, y', y'')$. Calcular el texto en claro mediante $L_2^{-1}(y)$.

En el descifrado HFE, si el conjunto Z contiene más de un elemento y varios de ellos son solución correcta al cifrado HFE, el elemento que se considera será aquel que tenga como 2 primeras entradas, las últimas 2 coordenadas que se obtuvieron del descifrado Square+.

Resultados experimentales

Se usaron distintos parámetros para implementar PHFER. Por ejemplo, el valor de n (la longitud del mensaje m) se encuentra entre 20 y 36, el valor de q es 31 o 43. Los valores de μ, v, s, β y r se eligen de acuerdo a cada capa del cifrado y con ello lograr que la clave no sea muy grande de manera que sea factible comparar el tiempo de ejecución para el cifrado y descifrado.

(A) $q = 31$, $n = 20$, $s = 6$, $\beta = 7$, $r = 7$,

(B) $q = 31$, $n = 25$, $s = 8$, $\beta = 9$, $r = 8$,

(C) $q = 43$, $n = 30$, $s = 9$, $\beta = 10$, $r = 11$,

(D) $q = 43$, $n = 25$, $s = 9$, $\beta = 8$, $r = 8$,

(E) $q = 43$, $n = 35$, $s = 10$, $\beta = 15$, $r = 10$,

	(A)	(B)	(C)	(D)	(E)
Cifrado	0.0075s	0.0145469s	0.028875s	0.01515625 s	0.02940625 s
Descifrado	75.0013s	89.0014s	183.00353s	364.0010997s	589.001695 s

Tabla 3.1: Resultados experimentales (PHFER).

3.2. Seguridad

Recordemos que el sistema PHFER, se define a partir de tres sistemas criptográficos seguros: Square+, HFE y Rainbow. La seguridad de cada uno de ellos ha sido estudiada de manera independiente, donde se concluye que Square+ es una versión segura del sistema Square. También se conocen los valores para hacer el sistema HFE seguro, los cuales fueron usados al implementar PHFER; así mismo sabemos que para el sistema Rainbow aún no se conoce un ataque que sea capaz de romperlo. De este modo lo que se logró al desarrollar este modelo es que cada una de sus capas fueran seguras.

Por lo que si un atacante intenta romper este nuevo cifrado, además de intentar hallar las transformaciones L_1 y L_2, que como ya vimos no es un trabajo fácil de hacer, debe de lograr romper cada una de las capas. En principio si él tiene acceso a cada una de las dimensiones de los campos que se consideran en cada una de las capas, podría intentar atacar cada una de ellas e ir recuperando el mensaje. Supongamos que el atacante de alguna forma logró recuperar L_1 y decide atacar desde la capa superior y rompe Square+, entonces habría recuperado s coordenadas y podrá usarlas como variables de vinagre para recuperar la parte del mensaje que fue cifrado con Rainbow, recuperando $s + r$ entradas de $L_2(m)$ de este modo utilizaría dos entradas que se recuperan en Square+ las cuales fueron cifradas usando HFE para intentar romper dicho cifrado. De cualquier modo restaría hallar las β entradas restantes usando el cifrado HFE. Para finalmente recuperar, si es posible, la transformación afín invertible L_2 y con ello romper el cifrado.

Se tiene la certeza que cada una de las capas tiene un nivel de seguridad mayor a 2^{80} y en caso de Rainbow no hay ataque que logre romperlo. Si en un futuro se encuentra una debilidad en el sistema Square, se puede puede proponer una versión segura y con ello mejorar el modelo PHFER.

3.3. Cifrado PR

En esta sección, se presenta un nuevo sistema de cifrado, el cifrado PR. El texto en claro es sometido a dos distintos cifrados; el cifrado Square+ y el cifrado Rainbow. En la primera capa de este sistema únicamente se consideran las primeras d entradas del texto en claro, para posteriormente usar el cifrado Square$^+$. Al final de esta primera parte del cifrado el mensaje cifrado tendrá $n-d+(l+p)$ entradas, las cuales son concatenadas con la salida del cifrado Rainbow como se muestra en la figura 3.2. Para la segunda capa del cifrado, se cifrarán las n entradas del mensaje con el cifrado Rainbow.

Construcción de la clave. Sean n, l, d, p, m_1, m_2 enteros positivos tales que $m_1 = n - d + (l + p)$ y $m_2 = 2n - d + (l + p)$ con $n > d$. La función central de PR es una concatenación del cifrado Square+ y del cifrado Rainbow. Recordemos la definición de cada uno de ellos.

- La función Square+ se define como $\mathcal{F}_{S^+} : k^n \to k^{n+\ell+p}$, que resulta de la siguiente composición:

$$k^n \xrightarrow{L_2} k^{n+\ell} \xrightarrow{\varphi_1} K \xrightarrow{\bar{F}^+} K \xrightarrow{\varphi_2^{-1}} k^{n+\ell+p}$$

  con $\bar{F}^+$ la función central de Square+.

- La función Rainbow es definida como

$$\mathcal{F}_R = (g_1, \ldots, g_{n-v_1}) : k^n \to k^{n-v_1}$$

con la construcción del sistema de cifrado Rainbow, donde $S = \{1, 2, \ldots, n\}$ y $\mathcal{O} = S_{i+1} - Si$ y donde cada polinomio cuadrático g_l es de la forma

$$g_l = \sum_{i \in \mathcal{O}_l, j \in S_l} \alpha_{i,j} x_i x_j + \sum_{i,j \in S_l} \beta_{i,j} x_i x_j + \sum_{i \in S_{l+1}} \gamma_i x_i + \eta$$

con $\alpha_{i,j}, \beta_{i,j}$, y γ_i, elegidos aleatoriamente del campo k.

Se utilizan dos transformaciones afines invertibles $L_2 : k^n \to k^n$ y $L_1 : k^{n+m_1} \to k^{m_2}$. La clave pública es dada por $P = L_2 \circ F \circ L_1 : k^n \to k^{m_2}$, donde $F = F_{S^+} \parallel F_R$ ($\parallel$ es la función concatenación), y la clave privada es representada en la figura 3.2.

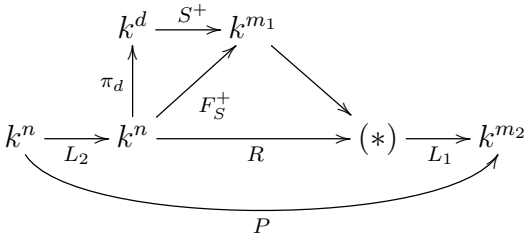


Figura 3.2: π_d es la proyección de las primeras d entradas, $m_1 = d + (l + p)$ y $m_2 = m_1 + n$, y * es la función concatenación

Cifrado Dado $m \in k^n$, el texto cifrado es calculado evaluando $c = P(m) \in k^{m_2}$

Descifrado

1. Se calcula $L_1^{-1}(c)$
2. Considerar las primeras m_1 entradas y descifrar usando Square$^+$ obteniendo $n - d$ variables de vinagre, que se denotan por $(y_1, \ldots, y_{n-d})$.
3. Sustituir las $n - d$ variables de vinagre en cada g_i del cifrado Rainbow, resolver los sistemas de ecuaciones asociados y hallar las d entradas faltantes.
4. Finalmente calcular el texto en claro mediante $L_2^{-1}(y_1, \ldots, y_n)$.

Resultados experimentales

Para esta sección se usan distintos parámetros para el cifrado PR, y se muestra una tabla con los tiempos de ejecución para el cifrado y descifrado.

(A) $q = 31$, $n = 20$, $l = 3$, $d = 16$

(B) $q = 31$, $n = 30$, $l = 3$, $d = 22$

(C) $q = 43$, $n = 25$, $l = 3$, $d = 20$

(D) $q = 43$, $n = 30$, $l = 1$, $d = 24$

(E) $q = 43$, $n = 35$, $l = 3$, $d = 20$

	(A)	(B)	(C)	(D)	(E)
Cifrado	0.00055625s	0.000867813s	0.000767188s	0.000892188s	0.00106563s
Descifrado	0.000811472s	0.00161988s	0.00125737s	0.00166749s	0.00195322s

Tabla 3.2: Resultados experimentales (PR).

3.4. Seguridad

A diferencia del modelo PHFER, el cifrado PR usa el mensaje en su totalidad en cada una de las capas. En la primer parte del cifrado se considera una proyección del mensaje m y esta se cifra usando Square+, el cual tiene un nivel de seguridad mayor a 2^{80}. En la segunda capa se cifra el mensaje m usando Rainbow, el cual es uno de los sistemas criptográficos multivariables más seguros de la actualidad. También gran parte de la seguridad recae nuevamente en las transformaciones L_1 y L_2, las cuales disfrazan la clave pública y si un atacante lograra hallarlas, restaría romper cada una de las capas, de aquí la seguridad recae en el sistema Square. Pero ya vimos que estos dos sistemas son seguros y cada uno de los parámetros que se usaron para este modelo fueron elegidos de acuerdo a cada unos de ellos para seguir conservando su nivel de seguridad.

Si en un futuro se logra romper Square+, el modelo PR habrá sido roto. Pero nuevamente, se podría modificar el modelo y usar una versión segura la cual garantice su seguridad una vez más.

Capítulo 4

Conclusiones y perspectivas

Después de que el sistema MI, creado por Matsumoto e Imai, fuera roto en 1995, gran parte de sus ideas fueron utilizadas para el desarrollo de sistemas criptográficos multivariables. Por ejemplo, tener como función central una función polinomial de grado a lo más dos y hacer uso de dos transformaciones afines e invertibles L_1 y L_2 como disfraz para la clave pública.

La seguridad de estos sistemas recae en la capacidad computacional para resolver un conjunto de ecuaciones polinomiales multivariadas sobre un campo finito, el cual resulta ser un problema NP-difícil. Varios de los sistemas que usan las ideas del sistema MI son HFE, Square y Rainbow, los cuales son la principal herramienta para los modelos que se proponen en esta tesis.

El sistema HFE, el cual fue desarrollado en el año 1996 y roto tres años más tarde, tiene como función central un polinomio HFE con límite D, un polinomio de q-grado igual a dos y de grado total a lo más D, y dos transformaciones laterales L_1 y L_2 afines e invertibles. A pesar de ser un sistema que ha sido roto, se usa en el modelo PHFER con parámetros que aseguran la resistencia a diversos ataques criptográficos, por ejemplo, se usa el valor de $q = 31$ para tener una resistencia a un ataque de bases de Gröbner. Además en dicho modelo la seguridad no recae en HFE, sino en Square+ y en Rainbow, los cuales son hasta el momento los sistemas más seguros.

El sistema Square, desarrollado en 2009, tiene como función central $F(X) = X^2$, la cual proporciona en el descifrado dos posibles soluciones. Hace uso de una transformación afín invertible L_1, y de otra función lateral L_2 que resulta ser afín inyectiva. Se creía que Square era un sistema seguro, sin embargo fue roto años más tarde y a partir de ese momento se desarrollaron versiones más seguras, Square+ y Square-. En nuestros modelos se uso la versión más segura para Square, Square+, la cual consiste en la adición de un número pequeño de polinomios aleatorios, digamos a lo más 3 a la clave pública. Y se consideraron los parámetros que aseguran tener un sistema Square+ seguro, por ejemplo considerar la longitud del mensaje $n > 32$ y $q = 31$, donde $q \equiv 3 \pmod 4$.

Finalmente el sistema Rainbow, para el cual no existe algoritmo capaz de romperlo hasta el momento, tiene como función central un polinomio Rainbow. Dicho sistema fue empleado en ambos modelos propuestos en este trabajo ya que brindan una seguridad de no recuperar el mensaje en su totalidad por parte de un atacante.

El modelo PHFER, resulta ser un modelo seguro ya que es la concatenación de tres de los sistemas más resistentes y además se hace uso de dos transformaciones afines e invertibles. En un posible ataque, este deberá romper cada una de sus capas. Aunque se logre romper una capa o incluso dos, no se podrá recuperar por completo ya que el mensaje m, se divide en tres partes y cada una se cifra usando un sistema diferente. A pesar de que la capa cifrada con Square y la capa cifrada con Rainbow están conectadas por s entradas del mensaje, resulta casi imposible recuperar en su totalidad el mensaje. Esto es por el hecho que se usaron los parámetros correctos para que la capa de cifrado con HFE resista un ataque de bases de Gröbner.

El modelo PR tiene un cifrado diferente, ya que la capa Square+ usa una proyección de las primeras d entradas del mensaje, y en la capa Rainbow el mensaje se cifra por completo. Ésta probablemente sea una debilidad del sistema PR, que una capa dependa de la otra. Si se rompiera el sistema Square+, se recuperan las d entradas y éstas entrarían como variables de vinagre para recuperar el resto del mensaje. Sin embargo, esto sucede si se cumplen dos cosas: se rompe el sistema Square+ y se hallan las dos transformaciones L_1 y L_2. Pero aún no se conoce un algoritmo que permita encontrar L_1 y L_2 de manera efectiva y rápida.

Ambos modelos, PR y PHFER tienen tiempos de ejecución pequeños. Por ejemplo en la tabla 3.1 se muestran los tiempos de cifrado y descifrado del sistema PHFER. Se observa que el tiempo de cifrado no es más de 3 milésimas de segundo, considerando un mensaje de longitud 35 cuyas entradas se encuentran en el campo $\mathbb{F}_{43}$. En el descifrado, pese a que el tiempo es considerablemente mayor, resulta ser un tiempo razonable, el cual oscila entre 75 segundos y 9 minutos. Para el sistema PR, la tabla 3.2 muestra que ambos tiempos de cifrado y descifrado son muy pequeños ya que no exceden 2 milésimas de segundo para ambos procesos. Pero a pesar de que el tiempo de descifrado para PHFER resulta ser mucho mayor que para PR podemos pensar que es el más seguro, puesto que para el usuario no resulta complicado descifrar ya que se posee toda la información necesaria para recuperar un mensaje y las capas no son dependientes entre ellas.

Finalmente se puede agregar como conclusión, que si en un futuro se le encuentran debilidades a algunos de los sistemas Square+, Rainbow o HFE, no se tendrán problemas en modificar cada uno de los modelos por una versión más segura con el fin de mantener su seguridad.

Apéndice A

Construcción de un campo finito

A lo largo de este trabajo se tuvo la necesidad de construir extensiones de campos de cierto grado n, es por ello que dedicaremos el presente Apéndice para desarrollar la construcción de campos finitos. Comenzaremos este capítulo recordando un poco de la teoría de los anillos conmutativos.

Definición A.0.1. Un *anillo conmutativo con unidad* es un conjunto R con dos operaciones binarias (suma +, multiplicación *) tales que

- ambas operaciones son conmutativas y asociativas,
- la operación suma tiene una identidad 0, y la multiplicación tiene una identidad 1,
- todo elemento en R tiene inverso aditivo y
- la multiplicación se distribuye bajo la suma.

Un **campo** es un anillo con la propiedad de que todo elemento distinto de cero tiene un inverso multiplicativo. En seguida, se define anillo de polinomios sobre un campo k.

Definición A.0.2. Sea R un anillo y sea x indeterminada. Para todo $n \in \mathbb{Z}$ y $a_0, a_1, \cdots, a_n \in R$ con $a_n \neq 0$, una expresión de la forma $a_0 + a_1x + \ldots + a_nx^n$ es llamada *un polinomio en x sobre R*. Un polinomio es mónico si el coeficiente líder es 1. El *anillo de polinomios en x sobre R*, es denotado por $R[x]$, es el conjunto de todos los polinomios con coeficientes en R. La adición en $R[x]$ es definida componente a componente y la multiplicación se define como

$$f(x) * g(x) = \sum_k x^k \sum_{i=0}^{k} a_i b_{k-i}$$

con $f(x) = \sum_i a_i x^i$ y $g(x) = \sum_j b_j x^j$.

Sea k un campo. Consideramos el anillo de polinomios con coeficientes en k, el cual es denotado

$$k[x] = \{a_n x^n + \ldots + a_1 x_1 + a_0 : \ a_0, \ldots, a_n \in k\}$$

Recordemos que $k[x]$ tiene un comportamiento parecido a los enteros, por ejemplo, se cumple el algoritmo de la división, y también es posible calcular el máximo común divisor entre dos polinomios $f(x)$ y $g(x)$. Si este valor es 1, diremos que $f(x)$ y $g(x)$ son coprimos.

Definición A.0.3. Un elemento de un campo finito cuyas potencias generan los elementos distintos de cero del campo es llamado *primitivo*.

Definición A.0.4. Si R es un anillo y existe un entero positivo n tal que $nr = 0$ para cada $r \in R$, entonces n es llamado la *característica de R*. Si tal entero n no existe, R se dice de *característica 0*.

Congruencia módulo un polinomio

Definición A.0.5. Sea $p(x) \in k[x]$ con k un campo. Diremos que $f(x), g(x)$ son *congruentes módulo* $p(x)$, denotado por $f(x) \equiv g(x) \bmod p(x)$, si $p(x)$ divide $f(x) - g(x)$.

Proposición A.0.1. La congruencia modulo $p(x)$ es una relación de equivalencia.

Demostración. Si $f(x) \equiv g(x) \bmod p(x)$, entonces existe $h(x)$ tal que $p(x)h(x) = f(x) - g(x)$.

1. Sea $h(x) = 0$ (el polinomio idénticamente cero). Se cumple $f(x) \equiv f(x) \bmod p(x)$, así la relación es simétrica.

2. Si $f(x) \equiv g(x) \bmod p(x)$, entonces existe $h(x)$ tal que

$$\begin{aligned} & p(x)h(x) = f(x) - g(x) \\ \text{así} \quad & -(p(x)h(x) = f(x) - g(x)) \\ \text{resulta} \quad & p(x)(-h(x)) = g(x) - h(x) \end{aligned}$$

   por lo que $g(x) \equiv h(x) \bmod p(x)$.

3. Finalmente si $f(x) \equiv g(x) \bmod p(x)$ y $g(x) \equiv r(x) \bmod p(x)$, existen $h_1(x)$ y $h_2(x)$ tales que

$$\begin{aligned} p(x)h_1(x) &= f(x) - g(x) \\ p(x)h_2(x) &= g(x) - r(x) \end{aligned}$$

   sumando estas dos últimas ecuaciones se obtiene que $h(x) = h_1(x) + h_2(x)$, así se cumple la transitividad, concluyendo la demostración.

Proposición A.0.2. Si $f(x) \equiv g(x) \bmod p(x)$ y $f_1(x) \equiv g_1(x) \bmod p(x)$, entonces

(a) $f_1(x) + f(x) \equiv g_1(x) + g(x) \bmod p(x)$.

(b) $f_1(x)f(x) \equiv g_1(x)g(x) \bmod p(x)$.

Demostración. Para demostrar el inciso a, si se considera $h'(x) = h(x) + h_1(x)$ se obtiene el resultado.
Para el inciso b, si $h'(x) = h(x)f_1(x) + h_1(x)g(x)$, se llega a la congruencia (b).

Las dos proposiciones anteriores muestran que hay dos operaciones bien definidas sobre el conjunto de clases de equivalencia de $f[x]$ módulo $p(x)$, cumpliendo también las propiedades de los anillos. Se denota por $k[x]/ < p(x) >$ al anillo de polinomios módulo $p(x)$.

Proposición A.0.3. El conjunto de polinomios de grado menor que grad$(p(x))$ forma un sistema completo de residuos modulo $p(x)$.

Demostración. Sea $p(x)$ de grado d con coeficientes en k. Si dos polinomios distintos tienen grado menor que d, entonces su diferencia no es divisible por $p(x)$, es decir, son no congruentes módulo $p(x)$. Cada polinomio es equivalente a su residuo cuando es dividido por $p(x)$ y el residuo tiene grado menor que d. Así cada polinomio en $k[x]$ es equivalente exactamente a un polinomio de grado menor que d.

Definición A.0.6. Un polinomio $p(x) \in k[x]$ se dice irreducible si para cualquier factorización $p(x) = a(x)b(x)$, se tiene que $a(x)$ o $b(x)$ es un elemento de k.

Teorema A.0.1. Sea $p(x)$ un polinomio irreducible sobre el campo k. Entonces $k[x]/\langle p(x) \rangle$ es un campo.

Demostración Sea $g(x)$ un polinomio cuya clase de equivalencia es distinta de cero módulo $p(x)$. Entonces $g(x)$ no es divisible por $p(x)$, es decir son primos relativos, por lo que existe $r(x)$ y $h(x)$ tales que $g(x)r(x) + p(x)h(x) = 1$. de este modo $g(x)r(x) \equiv 1 \bmod p(x)$. Consecuentemente $g(x)$ es una unidad modulo $p(x)$, y por tanto $k[x]/\langle g(x)\rangle$ es un campo.

Con este último teorema, ya se tiene una forma de construir un campo finito de la forma deseada $k[x]/\langle g(x)\rangle$. Usaremos la letra α, para denotar la clase de equivalencia de x, α puede pensarse como una raíz de $p(x)$. Recordemos además que bajo un campo finito de característica p se tiene la siguiente propiedad

$$(a+b)^{p^r} = a^{p^r} + b^{p^r}.$$

Ejemplo A.0.1. Consideremos el anillo de polinomios sobre $\mathbb{Z}_2$, denotado por $\mathbb{Z}_2[x]$. El campo $\mathbb{F}_8$ se construye de la siguiente manera: puesto que el polinomio $x^3 + x + 1$ resulta ser irreducible sobre dicho campo, entonces

$$\mathbb{F}_8 = \mathbb{Z}_2[x]/(x^3 + x + 1) \cong \mathbb{Z}_2[\alpha]$$

donde $\alpha^3 + \alpha + 1 = 0$, es un campo con 8 elementos de característica 2.

Los elementos de $\mathbb{F}_8$ son:

$$0, 1, \alpha, \alpha^2, 1+\alpha, 1+\alpha^2, \alpha+\alpha^2, 1+\alpha+\alpha^2$$

Un estudio más detallado sobre campos finitos puede encontrarse por ejemplo en [15].

Apéndice B

Algoritmo Berlekamp

El algoritmo Berlekamp fue creado por Elwyn Berlekamp en 1967. Proporciona un método para factorizar polinomios con coeficientes en un campo finito $\mathbb{F}_q$, con q primo, de alto grado en un tiempo razonable. Es decir, si $f(x) \in \mathbb{F}_q[x]$, que podemos suponer mónico, a través del algoritmo Berlekamp $f(x)$ se podrá expresar como

$$f(x) = f_1(x)^{r_1} \cdots f_k(x)^{r_k}$$

donde $f_i(x)$ son polinomios mónicos irreducibles distintos en $\mathbb{F}_q[x]$ y $r_i \geq 1$ para cada $i = 1, \ldots, k$.

Descripción

Supongamos que f no tiene factores repetidos, es decir $f = f_1 \cdot f_2 \cdots f_k[x]$, donde cada f_i es un polinomio mónico irreducible distinto sobre $\mathbb{F}_q$. Si $(c_1, c_2, \ldots, c_k)$ es cualquier k-tupla de elementos de $\mathbb{F}_q$, entonces hay una única $h \in \mathbb{F}_q[x]$ con $h(x) \equiv c_i \ mod \ f_i(x)$ para $1 \leq i \leq k$ y $grad(h) < grad(f)$.

El polinomio $h(x)$ satisface la condición

$$h(x)^q \equiv c_i^q = c_i \equiv h(x) \ mod \ f_i(x) \text{ para } 1 \leq i \leq k$$

y por tanto

$$h(x)^q \equiv h(x) \ mod \ f(x), \ grad(h) < grad(f) \tag{B.1}$$

Si h es solución de Eq.(B.1), entonces

$$h(x)^q - h(x) = \prod_{c \in \mathbb{F}_q} (h(x) - c)$$

implica que cada factor irreducible de f divide a alguno de los polinomios $h(x) - c$.
Esto es, todas las soluciones de la ecuación (B.1) satisfacen

$$h(x) \equiv c_i \ mod \ f_i(x), \ 1 \leq i \leq k,$$

para la k-tupla $(c_1, \ldots, c_k)$ de elementos de $\mathbb{F}_q$.

Con lo que hay exactamente q^k soluciones de la ecuación (B.1). Reduciendo a un sistema de ecuaciones lineales, se encuentran dichas soluciones.

Si $grad(f) = n$, entonces se construye una matriz $B = (b_{ij}), \ 0 \leq i, j \leq n-1$, de $n \times n$, donde las potencias de $x^{iq} \ mod \ f(x)$ satisfacen

$$x^{iq} \equiv \sum_{j=0}^{n-1} b_{ij} x^j \text{ mod } f(x) \text{ para } 0 \leq i \leq n-1$$

donde $b_{ij} \in \mathbb{F}_q$ y $h(x) = a_0 + a_1 x + \ldots + a_{n-1}x^{n-1} \in \mathbb{F}_q[x]$ satisface la ecuación (B.1) si y sólo si

$$(a_0, \ldots, a_{n-1})B = (a_0, \ldots, a_{n-1}) \tag{B.2}$$

con $B = (b_{ij})$ y $0 \leq i, j \leq n-1$, esto es $\bar{a}^t = (a_0, \ldots, a_{n-1})^t$ es un vector propio de B^t. La ecuación (B.1) se cumple si y sólo si

$$\begin{aligned} h(x) &= \sum_{j=0}^{n-1} a_j x^j = \sum_{j=0}^{n-1} \sum_{i=0}^{n-1} a_i b_{ij} x^j \\ &\equiv \sum_{i=0}^{n-1} a_i x^{iq} \equiv h^q \bmod f(x) \end{aligned}$$

El sistema (B.2) se puede reescribir como

$$(a_0, \ldots, a_{n-1})(B - I) = (0, \ldots, 0), \tag{B.3}$$

cuyo sistema tiene exactamente q^k soluciones, por lo que el rango de $B-I$ es k, el número de factores irreducibles de grado 1 de f. Además el rango de $B - I$ es $n - k$. Ya que el polinomio $h_1(x) = 1$ es una solución trivial de la ecuación (B.1), entonces el vector $(1, 0, \ldots, 0)$ es solución de (B.3). Existirán los polinomios $h_2(x), \ldots, h_k(x)$ con grado $\leq n-1$ tales que los vectores correspondientes a $h_2(x), \ldots, h_k(x)$ forman la base del núcleo de $B - I$. Entonces los polinomios $h_2(x), \ldots, h_k(x)$ son la factorización de f. Una vez encontrado el rango r, sabemos que el número de polinomios irreducibles será $k = n - r$.

Si $k = 1$, entonces $f(x)$ es irreducible y terminará el procedimiento. También para $k = 1$, las soluciones de $h^q \equiv h \bmod f$ son los polinomios constantes y el núcleo de $B-I$ contendrá los vectores $(c, 0, \ldots, 0)$ donde $c \in \mathbb{F}_q$. Si $k \geq 2$, entonces se obtendrá el polinomio $h_2(x)$ el cual es un factor de la factorización de f. Entonces se halla el $MCD(f(x), h_2(x) - c)$, para todo $c \in \mathbb{F}_q$; el resultado será una factorización no trivial de $f(x)$, aunque no necesariamente con factores irreducibles.
Si no se hallan k factores de f mediante $h_2(x)$, se encuentra el $MCD(g(x), h_3(x) - c)$, para toda $c \in \mathbb{F}_q$ y cada factor $g(x)$ obtenido con $h_2(x)$, y el proceso continuará hasta hallar k factores de $f(x)$.

A continuación se presenta el algoritmo de factorización Berlekamp:

Algoritmo I. Factorización de Berlekamp

1. Dado $f(x)$ de grado n, se toman
$$x^{iq} \equiv \textstyle\sum_{j=0}^{n-1} b_{ij} x^j \bmod f(x)$$
donde $0 \leq i < n$. Se calcula el rango de la matriz $(B - I)$, denotado por r,
de donde se obtiene el número de factores irreducibles de $f(x)$, que será igual a $k = n - r$.

2. Si $k = 1$ entonces $f(x)$ es irreducible. Si $k > 1$, se resuelve el sistema (B.3),
se considera el polinomio $h_2(x)$ asociado a un vector solución y se calcula
$MCD(f(x), h_2(x) - c)$ para toda $c \in \mathbb{F}_q$.

3. Si la factorización obtenida consta de k factores, el proceso acaba. En caso contrario
se calcula el $MCD(g(x), h_3(x) - c)$ para toda $c \in \mathbb{F}_q$, con $h_3(x)$ asociado a un vector solución
(como en 2) y cada factor $g(x)$ obtenido con $h_2(x)$.

Apéndice C

Criptoanálisis de Rainbow

Jintai Ding y Dieter Schmidt presentaron el criptoanálisis con base a un ejemplo que ellos mismos implementaron en [7]. Este apéndice está dedicado a la explicación del mismo y al estudio de su seguridad respecto a dos ataques específicos. El desarrollo del sistema Rainbow se encuentra en la sección 2.4.1, donde se incluye un ejemplo detallado del cifrado y del descifrado del criptosistema, considerando $k = \mathbb{F}_{11}$ y $n = 10$.

Considere un campo finito k de cardinalidad $q = 2^8$, un mensaje m de longitud $n = 33$. Sea $S = \{1, 2, \ldots, 33\}$. Se eligen $u = 5$, $v_1 = 6$, $v_2 = 12$, $v_3 = 17$, $v_4 = 22$ y $v_5 = 33$. Puesto que $o_i = v_{i+1} - v_i$ se tiene $o_1 = 6$, $o_2 = 5$, $o_3 = 5$, $o_4 = 11$. Así la clave pública consiste de 27 polinomios cuadráticos en 33 variables. El número máximo de monomios en la clave pública es $(27 \times 35 \times 34)/2 = 16065$.

Ataque para sistemas de aceite y vinagre

Como ya se ha explicado, la función L_1 mezcla los componentes del polinomio Rainbow. Por tanto cada componente de la clave pública pertenece a la capa superior de polinomios de aceite y vinagre, es decir todos pertenecen a P_4 ya que son polinomios de aceite y vinagre con 22 variables de vinagre y 11 variables de aceite. Lo que se necesita para atacar el sistema es separar la última capa de 11 variables de aceite y 22 variables de vinagre. La complejidad de dicho ataque, esto es el número de operaciones necesarias, de este primer paso es $q^{22-11-1} \times 11^4 > 2^{90}$, usando el ataque para sistemas de firma de aceite y vinagre no balanceados [13].

Ataque del rango mínimo

Hay dos formas de usar el método del rango mínimo. Una forma de atacar es buscar una combinación lineal de los polinomios multivariados públicos que tenga un rango más bajo. El conjunto de polinomios que cumple tener el rango más bajo son los polinomios con 6 variables de aceite y 6 variables de vinagre pertenece a la primer capa, esto es, $\tilde{F}_1$. La forma para representar un polinomio cuadrático multivariable es mediante su matriz simétrica asociada.

Para los parámetros v, o_1 y o_2 de Rainbow, las matrices $A_{\tilde{F}_1}, \ldots, A_{\tilde{F}_{o_1+o_2}}$ corresponden a los polinomios cuadráticos centrales $\{F_1, \ldots, F_{o_1+o_2}\}$ que son de la forma

$$A_{F_i} = \begin{cases} \begin{pmatrix} *_{v\times v} & *_{v\times o_1} & 0_{v\times o_2} \\ *_{o_1\times v} & 0_{o_1\times o_1} & 0_{o_1\times o_2} \\ 0_{o_2\times v} & 0_{o_2\times o_1} & 0_{o_2\times o_2} \end{pmatrix} & \text{si } 1 \leq i \leq o_1, \\ \begin{pmatrix} *_{v\times v} & *_{v\times o_1} & *_{v\times o_2} \\ *_{o_1\times v} & 0_{o_1\times o_1} & *_{o_1\times o_2} \\ *_{o_2\times v} & *_{o_2\times o_1} & 0_{o_2\times o_2} \end{pmatrix} & \text{si } o_1+1 \leq i \leq o_1+o_2, \end{cases}$$

donde $*_{i\times j}$ son matrices de $i \times j$ sobre k. Por lo que las matrices para este ejemplo resultan ser de tamaño 33×33 y haciendo combinaciones lineales de ellas se busca una matriz cuyo rango sea 12.

Para atacar el sistema, el problema se convierte en buscar un matriz de rango 12 entre un grupo de 27 matrices de tamaño 33×33. La complejidad para encontrar dicha matriz es $q^{12} \times 27^3$, el cual es más grande que 2^{100}.

Otra posibilidad para buscar polinomios que correspondan a polinomios en la segunda capa, a saber el único que pertenece a P_3, y viene de las combinaciones lineales de $\tilde{F}_i$, $i > 4$. Una manera es buscarlos aleatoriamente, ya que la dimensión de P_3 es 16, esto se convierte en un problema de buscar un elemento en un subespacio de dimensión 16 en un espacio de dimensión 27. Por lo que, la búsqueda aleatoria necesita mínimo q^{11} búsquedas, también se necesita determinar si el rango es menor que 22 para cada búsqueda. Así la complejidad es al menos $q^{11} \times (22 \times 33^2/3) > 2^{100}$.

Los valores sugeridos para el sistema Rainbow para el conjunto de $v_1 = 6$ variables tiene una probabilidad menor que 0.37 de tener solución única al realizar el descifrado, para este ejemplo se debe de resolver un sistema de 27 ecuaciones con 33 incógnitas, las cuales se pueden pensar que son aleatorias. Con esto se concluye que se tiene una complejidad para atacar este ejemplo de al menos 2^{80}.

Bibliografía

[1] **Assmus, E. and Key, J.** Designs and their Codes. Cambridge, 1992.

[2] **Daniel J. Bernstein, Johannes Buchmann y Erik Dahmen**. Post-Quantum Cryptography. Springer, 2009.

[3] **Crystal Clough**. Square: A new Family of Multivariate Encryption Schemes. Tesis para obtener el grado de Doctor en Ciencias Matemáticas. University of Cincinnati, 2009.

[4] **Clough C., Baena J., Ding J., Yang BY., Chen M.**. Square, a new multivariate encryption scheme. CT-RSA 2009, Vol. 5473 of Lecture Notes in Computer Science. Springer, 2009.

[5] **Whitfield Diffie and Martin E. Hellman**. New directions in Cryptography, IEEE Transactions on information theory, Vol. IT-22, 1976.

[6] **Jintai Ding, Jason E. Gower y Dieter S. Schmidt**. Multivariate Public Key Cryptosystems. Springer, 2006.

[7] **Jintai Ding y Dieter Schmidt**. Rainbow, a New Multivariable Polynomial Signature Scheme. Department of Mathematical Sciences, Vol. 3531 of Lecture Notes in Computer Science. Springer, 2005.

[8] **Jean-Charles Faugère, Kelsey Horan and Marc Kaplan**. Fast Quantum Algorithm for Solving Multivariate Quadric Equations. Vol. 1712 of dblp computer science bibliography, 2017. Disponible en arXiv:1712.07211.

[9] **M. Jason Hinek**. Cryptanalysis of RSA and its variants. Taylor and Francis Group, LLC, Chapman and Hall, 2009.

[10] **Yasuhiko Ikematsu, Ray Perlner, Daiel Smith-Tone, Tsuyoshi Takagi y Jeremy Vates**, HFERP- A New Multivariate Encryption Scheme, Vol. 10786 of Lecture Notes in Computer Science, 2018.

[11] **Aviad Kipnis and Adi Shamir**. Cryptanalysis of the HFE public key cryptosystem by relinearization, Vol. 1666 of Lecture Notes in Compututer Science, pages 19–30. Springer, 1999.

[12] **Aviad Kipnis and Adi Shamir** Cryptanalysis of the HFE public key cryptosystem by relinearization. In Advances in cryptology—CRYPTO '99, Vol. 1666 of Lecture Notes in Computer of Science, pages 19–30, 1999.

[13] **A. Kipnis , Patarin J., Goubin L.** Unbalanced oil and vinegar signature schemes. In Eurocrypt'99, Vol. 1592 of Lecture Notes in Computer of Science, pages 206–222. Springer, 1999.

[14] **Neal Koblitz** A Course in Number Theory and Cryptography, 2ed. Springer, 1991.

[15] **Rudolf Lidl, Harald Niederreiter** Introduction to finite fields and their applications. Cambridge University Press, 1986.

[16] **V. V. Muniswamy**. Design and Analysis of Algorithms. I K International Publishing House, 2009.

[17] **H. Ong, Schnorr, Claus P., and Shamir, Adi**. Efficient signature schemes based on polynomial equations, Vol. 196 of Lecture Notes in Computer Science, pages 37-46. Springer,1985

[18] **Jacques Patarin**. Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88. Vol. 20 of Designs, Codes and Cryptography, pages 175-209, 2000.

[19] **Jacques Patarin**. Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): Two new families of asymmetric algorithms, Vol. 1070 of Lecture Notes in Computer Science, pages 33–48. Springer, 1996.

[20] **John M. Pollard and Claus P. Schnorr** An efficient Solution of the Congruence $x^2 + ky^2 = m(\text{mod } n)$. IEE Transactions on Information Theory, Vol. IT-33, No.5, 1987.

[21] **Douglas R. Stinson**. Cryptography Theory and Practice, 3ed. Chapman and Hall/CRC, 2005.

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA

ACTA DE EXAMEN DE GRADO

No. 00212
Matrícula: 2183802540

Sistemas multivariables de cifrado PHFER y PR como aportación a la industria de la seguridad informática.

Con base en la Legislación de la Universidad Autónoma Metropolitana, en la Ciudad de México se presentaron a las 12:00 horas del día 27 del mes de enero del año 2021 POR VÍA REMOTA ELECTRÓNICA, los suscritos miembros del jurado designado por la Comisión del Posgrado:

DR. CARLOS JOSE ENRIQUE SIGNORET POILLON
DR. JUAN CARLOS KU CAUICH
DR. JOSE NOE GUTIERREZ HERRERA

Bajo la Presidencia del primero y con carácter de Secretario el último, se reunieron para proceder al Examen de Grado cuya denominación aparece al margen, para la obtención del grado de:

MAESTRA EN CIENCIAS (MATEMÁTICAS APLICADAS E INDUSTRIALES)

DE: SUSANA HERNANDEZ CANDELARIO

y de acuerdo con el artículo 78 fracción III del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

Aprobar

Acto continuo, el presidente del jurado comunicó a la interesada el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.

UNIVERSIDAD AUTONOMA METROPOLITANA
Casa abierta al tiempo
DIRECCION DE SISTEMAS ESCOLARES

SUSANA HERNANDEZ CANDELARIO
ALUMNA

REVISÓ

MTRA. ROSALIA SERRANO DE LA PAZ
DIRECTORA DE SISTEMAS ESCOLARES

DIRECTOR DE LA DIVISIÓN DE CBI

DR. JESUS ALBERTO OCHOA TAPIA

PRESIDENTE

DR. CARLOS JOSE ENRIQUE SIGNORET POILLON

VOCAL

DR. JUAN CARLOS KU CAUICH

SECRETARIO

DR. JOSE NOE GUTIERREZ HERRERA

El presente documento cuenta con la firma –autógrafa, escaneada o digital, según corresponda- del funcionario universitario competente, que certifica que las firmas que aparecen en esta acta – Temporal, digital o dictamen- son auténticas y las mismas que usan los c.c. profesores mencionados en ella