Avances en el Criptosistema de Clave Pública Basado en una Variedad Algebraica

Contreras G., José A.[*]
Londoño R., Anastacia[**]

20 de abril de 2016

Resumen

En 2004, Akiyama K. y Goto Y. [1] propusieron un criptosistema de clave pública de resistencia postcuántica basado en una variedad algebraica. Específicamente, el criptosistema se basa en un problema poco común en la criptografía multivariable: el Problema de Búsqueda Secciones. Este esquema de cifrado asimétrico usa de tamaños razonables de las claves: para parámetros recomendados, el tamaño de la clave secreta es de 102 bits y el de la clave pública es de 500 bits. Dado el evidente atractivo del sistema, este ha sido examinado exhaustivamente por la comunidad criptográfica, en este artículo, se presentan los ataques publicados al criptosistema y los avances realizados a éste en respuesta.

1. Introducción

En 1994, gracias a los trabajos de Shor [9], quedó demostrado que los algoritmos de factorización de números enteros y de logaritmo discreto podrían ser resueltos de manera efectiva mediante computadores cuánticos. Esto significa que los criptosistemas RSA y de curvas elípticas, entre otros, ya no serán seguros si un computador cuántico es construido. Por lo tanto, es importante la búsqueda de criptosistemas basados en problemas matemáticos más complejos.

En este orden, la primera versión del criptosistema de variedad algebraica (CVA) fue presentada por Akiyama-Goto [1] basada en un problema NP-Completo de geometría algebraica, al que llamaremos *el Problema de Búsqueda de Secciones*, que describiremos luego y que puede ser visto como el problema de resolver un sistema de ecuaciones polinómicas multivariables (de grado alto) el cual se sabe que es NP-Completo. Esta versión dió lugar a los ataques de Voloch [10] y Uchiyama-Tokunaga [11] (ver también [7]).

La nueva versión del criptosistema fue publicada en 2008 por Akiyama-Goto [2], y extendida por Akiyama-Goto-Miyake en 2009 [3]. En 2010, Faugere-Spaenlehauer [6] publicaron un criptoanálisis algebraico que rompe por completo el sistema, La idea principal de este ataque es descomponer los ideales deducidos del texto cifrado con el fin de evitar el problema de la búsqueda de secciones.

[*] jancontreras@cenditel.gob.ve
[**] anas.2720@gmail.com

En 2015 Okumura [8] presentó un criptosistema de clave pública basado en ecuaciones diofánticas de grado creciente (EDC), usando un método análogo al de Akiyama-Goto. En el artículo se dan algunas discusiones sobre cómo asegurar este criptosistema en contra de los ataques conocidos (incluyendo el de Faugere-Spaenlehauer), sin embargo no se alcanzó una demostración de la seguridad del mismo. Recientemente, en Enero de 2016, Ding et al. [5] publicaron un criptoanálisis efectivo contra [8], mostrando que la seguridad de EDC depende de la dificultad de encontrar ciertos vectores (relativamente) cortos a partir de la clave pública y el texto cifrado.

En 2015 Contreras [4] presentó, un criptosistema de clave pública basado en una variedad algebraica, siguiendo lo establecido por [3], en el marco de un proyecto de investigación de la Fundación CENDITEL. Este artículo se presenta como una compilación de las investigaciones relacionadas al criptosistema con el fin de darle continuidad y actualizar el proyecto mencionado.

El resto del artículo está organizado como sigue; en la sección 2, presenta algunos hechos importantes sobre las superficies algebraicas junto con el criptosistema original [1], en la sección 3 se describen los ataques de Voloch [10], Uchiyama-Tokunaga [11] y su versión generalizada, publicada por Iwami [7]. En la sección 4 se presenta brevemente la última versión del criptosistema de variedad algebraica (ver [3]) y se describe el ataque de Faugere-Spaenlehauer. Finalmente en la sección 5 se describe el criptosistema basado en ecuaciones diofánticas de [8] y su respectivo criptoanálisis.

2. Preliminares

2.1. Superficies Algebraicas

Sea $\mathbb{F}_p$ un cuerpo finito con p elementos. Una variedad algebraica sobre $\mathbb{F}_p$ se define como el conjunto de soluciones de ecuaciones algebraicas que tiene dos grados de libertad. Para construir nuestro criptosistema usaremos una variedad algebraica afín X, en el espacio afín $\mathbb{A}^3$, definida de manera natural mediante la ecuación $X(x, y, t) = 0$ sobre $\mathbb{F}_p$. En nuestro caso, no reviste mayor importancia si X es suave, en el sentido de poseer puntos singulares, pero el polinomio $X(x, y, t)$ necesariamente debe ser irreducible.

Sobre X pueden definirse ciertas curvas, por ejemplo si tomamos otra variedad Y, la interseccion $X \bigcap Y$ es una curva sobre X. Tales curvas pueden ser encontradas fácilmente, pero encontrar todas las curvas sobre X no es una tarea sencilla. Un tipo de curvas sobre la variedad X, particularmente difícil de encontrar explícitamente, son las curvas parametrizadas, definidas por ecuaciones de tipo $(x, y, t) = (u_x(t), u_y(t), t)$, donde $u_x(t)$ y $u_y(t)$ son polinomios en t sobre $\mathbb{F}_p$.

Si definimos la relación $\sigma : X \to \mathbb{A}^1$ como $\sigma(x, y, t) = t$, entonces la curva parametrizada induce una relación inversa $\tau : \mathbb{A}^1 \to X$, tal que $\sigma \circ \tau = \mathsf{id}_{\mathbb{A}^1}$. La función σ es llamada el fibrado de X en $\mathbb{A}^1$ y τ es llamada una sección de σ.

Podemos ver una sección como sigue: si reescribimos $X(x, y, t)$ como un polinomio sobre $\mathbb{F}_p[t]$, podemos ver a X como una curva sobre el cuerpo $\mathbb{F}_p$ (o sobre el anillo $\mathbb{F}_p[t]$). Luego, una sección será un punto racional sobre esta curva. Encontrar tales puntos es uno de los diez problemas de Hilbert y no existe algoritmo que lo resuelva en tiempo polinomial.

Definición 2.1.1 *Sea $X(x, y, t) = 0$ una variedad algebraica sobre $\mathbb{F}_p$. El problema de encontrar curvas parametrizadas $(x, y, t) = (u_x(t), u_y(t), t)$ sobre X se llama* ***Problema de Búsqueda de Secciones en*** *X.*

La variedad algebraica está definida por el polinomio $X(x, y, t)$, el objetivo del problema de búsqueda de secciones es encontrar dos polinomios $u_x(t)$ y $u_y(t)$ sobre $\mathbb{F}_p(t)$, tales que $X(u_x(t), u_y(t), t) = 0$. Este problema conduce a la resolución de un sistema de ecuaciones polinomiales no lineales en varias variables (ver [3] ó [4]). Como es sabido, la resolución de tal sistema es un problema que tiene complejidad NP-completo.

2.2. Versión Original del Criptosistema (CVA04)

describiremos ahora la primera versión del criptosistema anunciado en 2004, para más detalles vea [1].

2.2.1. Parámetros

1. El tamaño del cuerpo base: p.
2. El grado máximo de las secciones: d.
3. El número de bloques en un texto plano: l, (con $d < l$).

2.2.2. Detalles del las Claves

1. La clave secreta está formada por dos de secciones.

$$D_1 : (x, y, t) = (u_x(t), u_y(t), t) \text{ y } D_2 : (x, y, t) = (v_x(t), v_y(t), t)$$

con

$$d = \text{máx}\left\{u_x(t), u_y(t), v_x(t), v_y(t)\right\}.$$

2. La clave pública es una superficie X que contiene a D_1 y D_2 como secciones.

Notación De aquí en adelante, dado un polinomio A nos referiremos a Λ_A como su soporte.

2.2.3. Descripción del Criptosistema

El criptosistema está dividido en 3 etapas, como sigue:

1. Generación de Claves Elegir dos polinomios

$$D_1(x, y, t) = (u_x(t), u_y(t), t) \text{ y } D_2(x, y, t) = (v_x(t), v_y(t), t)$$

tales que $(u_x(t) - v_x(t))|(u_y(t) - v_y(t))$, y construir la superficie $X(x, y, t)$ que los contenga como secciones.

2. Cifrado Este paso consiste en convertir un texto plano (de elementos en $\mathbb{F}_p$) en un texto cifrado.

1. Dividir el texto plano m en l bloques, de la forma $m = m_0||\cdots||m_{l-1}$ e incrustar m en un polinomio en t como

$$m(t) = m_{l-1}t^{l-1} + \cdots + m_1 t + m_0 \ (0 \leq m_i < p, i = 0, \ldots, l-1).$$

2. Elegir un polinomio irreducible $f(t)$ de grado l.
3. Elegir un polinomio aleatorio

$$r(x,y,t) = \sum_{(i,j)\in\Lambda_r} r_{ij}(t)x^i y^i$$

y escribir

$$X(x,y,t)r(x,y,t) = \sum_{(i,j)\in\Lambda_{Xr}} a_{ij}(t)x^i y^i$$

donde $\Lambda_{X_r} := \left\{(i,j) \in \mathbb{N}^2 | a_{ij}(t) \neq 0\right\}$

4. Elegir un polinomio aleatorio

$$s(x,y,t) = \sum_{(i,j)\in\Lambda_{Xr}} s_{ij}(t)x^i y^i$$

tal que $\deg(s_{ij}(t)) = \deg(a_{ij}(t) - 1)$. Esto hace que fs y Xr tengan la misma forma como polinomios en x y y sobre $\mathbb{F}_p[t]$.

5. Establecer el polinomio cifrado $F(x,y,t)$ como

$$F(x,y,t) = m(t) + f(t)s(x,y,t) + X(x,y,t)r(x,y,t) \tag{2.1}$$

3. Decifrado Este paso consiste en recuperar el mesaje a partir del texto cifrado y la clave privada.

1. Sustituir las secciones D_i en el polinomio $F(x,y,t)$ obteniendo:

$$\begin{aligned} h_1(t) &= F(u_x(t), u_y(t), t) = m(t) + f(t)s(u_x(t), u_y(t), t) \\ h_2(t) &= F(v_x(t), v_y(t), t) = m(t) + f(t)s(v_x(t), v_y(t), t) \end{aligned}$$

2. Calcular $h_1(t) - h_2(t)$ para obtener $f(t)\left\{s(u_x(t), u_y(t), t) - s(v_x(t), v_y(t), t)\right\}$.
3. Factorizar $f(t)\left\{s(u_x(t), u_y(t), t) - s(v_x(t), v_y(t), t)\right\}$ y encontrar $f(t)$ como un polinomio irreducible de grado l.
4. Obtener $m(t)$ como el resto de dividir $h_1(t)$ por $f(t)$ y recuperar el texto plano m de $m(t)$.

3. Ataques a CVA04

Fueron presentados dos ataques a esta versión del criptosistema (considerando [7] como una generalización de [11]).

3.1. Ataque de Reducción de Uchiyama y Tokunaga

Uchiyama y Tokunaga anunciaron su ataque al criptosistema en 2007 [11]. Su algoritmo es el siguiente:

1. Dado un texto cifrado $F(x,y,t)$ como en 2.1, calcular el resto

$$R(x,y,t) = \sum_{(i,j)\in\Lambda_R} g_{i,j}(t)x^i y^y$$

de la división de $F(x,y,t)$ por la clave pública $X(x,y,t)$.

2. Establecer G como el conjunto de todos los factores irreducibles de $g_{i,j}(t)$ de grado mayor o igual l.

3. Para cada $f_i(t) \in G$, calcular el resto $m_i(t)$ de la división por $g_{00}(t)$. Luego, alguno de los $m_i(t)$ coincide con el texto plano $m(t)$.

Para que este algoritmo funcione, es necesario que G contenga a $f(t)$ y que $g_{00}(t)$ tenga la forma $m(t) + f(t)s(t)$, para algún $s(t)$. En [11] demuestran que, estas condiciones se satisfacen, siempre que el término líder de $X(x, y, t)$ de un orden monomial, sea de la forma $cx^{\alpha}y^{\beta}$ con $c \in \mathbb{F}_p$.

3.1.1. Refinamiento de Iwami

Para usar el algoritmo de Uchiyama y Tokunaga, es necesaria una suposición sobre la forma del término líder de un un orden monomial. En 2008 Iwami [7], encontró una manera de evitar esa suposición. La idea general, es considerar $X(x, y, t)$ como un polinomio en dos variables sobre el cuerpo $\mathbb{F}_p[t]$, en lugar de como un polinomio de tres variables en $\mathbb{F}_p$. Luego, dividiendo por el coeficiente del término líder, se puede tener siempre que este tenga la forma $x^{\alpha}y^{\beta}$. Finalmente, aplicando el algoritmo de reducción a $X(x, y, t)$ sobre $\mathbb{F}_p$, el método de Uchiyama y Tokunaga se recupera el pólinomio $f(t)$.

3.1.2. Condición para Evitar el Ataque de Reducción

Para evitar este ataque, basta con modificar el CVA04 de tal forma que ningún orden monomial sea efectivo para extraer suficiente información de $m(t)$ y $f(t)$ cuando $F(x, y, t)$ sea dividido por $X(x, y, t)$. Es decir, se debe construir el criptosistema, de tal manera que $m(x, y, t)$ y $f(x, y, t)$ contengan al menos un monomio que sea divisible por todos los monomios de $X(x, y, t)$.

3.2. Ataque de Voloch

La idea del ataque de Voloch [10] es considerar una extensión de $\mathbb{F}_p(t)$ y usar la función traza T. De nuevo, consideremos F como en 2.1, el algoritmo del ataque de Voloch se describe a continuación:

1. Sustituir y por algún polinomio $c(t)$, tal que $X(x, c(t), t)$ se vuelva irreducible.

2. Sea α una solución de $X(x, c(t), t) = 0$ sobre $\mathbb{F}_p(t)$, encontrar $\beta \in \mathbb{F}_p(t)(\alpha)$ tal que $T_{\mathbb{F}_p(t)(\alpha)/\mathbb{F}_p(t)}(\beta) = 0$.

3. Calcular $T(\beta F(\alpha, c(t), t))$ y notese que

$$T(\beta F(\alpha, c(t), t)) = T(\beta m(t) + \beta f(t)s(\alpha, c(t), t)) = f(t)T(\beta s(\alpha, c(t), t)).$$

4. Factorizar $T(\beta F(\alpha, c(t), t))$ y obtener $f(t)$.

5. Encontrar $\beta_1 \in \mathbb{F}_p(t)(\alpha)$ tal que $T_{\mathbb{F}_p(t)(\alpha)/\mathbb{F}_p(t)}(\beta_1) \in \mathbb{F}_p^{\times}$ y calcular

$$T(\beta_1 F(\alpha, c(t), t)) = m(t)T(\beta_1) + f(t)T(\beta_1 s(\alpha, c(t), t))$$

6. Dividir $T(\beta_1 F(\alpha, c(t), t))$ entre $f(t)$, para encontrar $m(t)T(\beta_1)$, y por lo tanto $m(t)$.

3.2.1. Ideas para evitar este ataque

Existen, al menos, dos maneras de evitar el ataque de Voloch: bien haciendo el cálculo de la traza extremadamente ineficaz, o bien cambiando la forma de $m(t)$ y $f(t)$. Ambas ideas se pueden aplicar simultáneamente simplemente añadiendo variables independientes a $m(t)$ y $f(t)$. En [3] establecen como caso seguro, el caso en el que m y f son polinomios en tres variables.

4. Versión Final del Criptosistema (CVA09)

4.1. Descripción del Criptosistema

Damos aquí una breve descripción de CVA. Para una presentación más detallada, se remite al lector a [3]. Mantendremos en esta sección la notación establecida en 2.1. Esta versión del CVA mantiene los parámetros p y d descritos en 2.2.1. Estos son especialmente importantes para la seguridad del criptosistema, ya que tienen un impacto directo en el tamaño binario de la clave secreta, que se determina por $2d\log(p)$. Los otros parámetros considerados son w, el grado de la superficie de clave pública X en x e y, y k la cardinalidad de Λ_X. Los parámetros w, d y p tienen un impacto considerable en el tamaño de la clave pública, que tiene, aproximádamente, $dw\log(p)$ bits.

4.1.1. Detalles de las Claves

1. La clave secreta está formada por una sección.

$$D : (x, y, t) = (u_x(t), u_y(t), t)$$

con

$$d = \text{máx}\left\{u_x(t), u_y(t)\right\}.$$

2. La clave pública es una superficie X, determinada por:

   a) Un polinomio irreducible $X(x, y, t) \in \mathbb{F}_p[x, y, t]$ tal que $|\Lambda_X| = k$ y X contiene a D como sección, con $d_{ij}^{(X)} \in \mathbb{N}$ el grado (en t) del coeficiente del monomio $x^i y^j$.

   b) $m(x, y, t)$, un polinomio deducido del texto a cifrar, con $d_{ij}^{(m)} \in \mathbb{N}$ el grado (en t) del coeficiente del monomio $x^i y^j$.

   c) $f(x, y, t)$, un polinomio divisor (equivalente a $f(t)$ en CVA04), con $d_{ij}^{(f)} \in \mathbb{N}$ el grado (en t) del coeficiente del monomio $x^i y^j$.

Para el proceso de cifrado/descifrado, es necesario que se cumplan las siguientes condiciones

$$\begin{aligned}
\Lambda_m \subset \Lambda_f\Lambda_X &= \left\{(i_1 + i_2, j_1 + j_2) : (i_1, j_1) \in \Lambda_f, (i_2, j_2) \in \Lambda_X\right\}. \\
\text{máx}\left\{i : (i, j) \in \Lambda_X\right\} &< \text{máx}\left\{i : (i, j) \in \Lambda_m\right\} < \text{máx}\left\{i : (i, j) \in \Lambda_f\right\}. \\
\text{máx}\left\{j : (i, j) \in \Lambda_X\right\} &< \text{máx}\left\{j : (i, j) \in \Lambda_m\right\} < \text{máx}\left\{j : (i, j) \in \Lambda_f\right\}. \\
\text{máx}\left\{d_{i,j}^{(X)}\right\}_{(i,j)\in\Lambda_X} &< \text{máx}\left\{d_{i,j}^{(m)}\right\}_{(i,j)\in\Lambda_m} < \text{máx}\left\{d_{i,j}^{(f)}\right\}_{(i,j)\in\Lambda_f}.
\end{aligned}$$

4.1.2. Criptosistema

Igual que en su primera versión, el criptosistema está dividido en 3 etapas, como sigue:

1. Generación de Claves Elegir dos polinomios

$$u_x(t) \text{ y } u_y(t)$$

y construir la superficie $X(x, y, t)$ que los contenga como secciones.

2. Cifrado .

1. Consideremos el texto plano m introducido en un polinomio

$$m(x, y, t) = \sum_{(i,j)\in\Lambda_m} m_{i,j}(t)x^i y^j$$

2. Elegir un polinomio irreducible

$$f(x, y, t) = \sum_{(i,j)\in\Lambda_f} f_{i,j}(t)x^i y^j$$

3. Elegir 4 polinomios aleatorios r_0, r_1, s_0, s_1 de la forma:

$$r_k(x, y, t) = \sum_{(i,j)\in\Lambda_f} r_{ij}^{(k)}(t)x^i y^i \ , \ s_k(x, y, t) = \sum_{(i,j)\in\Lambda_X} s_{ij}^{(k)}(t)x^i y^i$$

donde $k \in \{0, 1\}$ y $\forall i, j$ se tiene que $\deg(r_{i,j}^{(k)}(t)) = d_{i,j}^{(f)}$ y $\deg(s_{i,j}^{(k)}(t)) = d_{i,j}^{(X)}$.

4. Establecer el polinomio cifrado $F(x, y, t)$ como

$$F_0(x, y, t) = m(x, y, t) + f(x, y, t)s_0(x, y, t) + X(x, y, t)r_0(x, y, t) \tag{4.1}$$
$$F_1(x, y, t) = m(x, y, t) + f(x, y, t)s_1(x, y, t) + X(x, y, t)r_1(x, y, t) \tag{4.2}$$

3. Decifrado .

1. Sustituir la sección $D = (u_x(t), u_y(t), t)$ en el polinomio $F(x, y, t)$ obteniendo:

$$\begin{aligned} h_0(t) &= F_0(u_x(t), u_y(t), t) = m(u_x(t), u_y(t), t) + f(u_x(t), u_y(t), t)s_0(u_x(t), u_y(t), t) \\ h_1(t) &= F_1(u_x(t), u_y(t), t) = m(u_x(t), u_y(t), t) + f(u_x(t), u_y(t), t)s_1(u_x(t), u_y(t), t) \end{aligned}$$

2. Calcular $h_0(t) - h_1(t)$ para obtener

$$f(u_x(t), u_y(t), t)s_0(u_x(t), u_y(t), t) - f(u_x(t), u_y(t), t)s_1(v_x(t), v_y(t), t).$$

3. Factorizar $f(u_x(t), u_y(t), t)\{s_0(u_x(t), u_y(t), t) - s_1(v_x(t), v_y(t), t)\}$ y encontrar $\tilde{f}(u_x(t), u_y(t), t)$ como un polinomio irreducible de grado $\deg(f(u_x(t), u_y(t), t))$.
4. Obtener $\tilde{m}(u_x(t), u_y(t), t) = h_0(t)$ (mód $\tilde{f}(t)$) y recuperar $\tilde{m}(x, y, t)$ mediante el siguiente sistema lineal:

$$\tilde{m}(u_x(t), u_y(t), t) = \sum \tilde{m}_{ijk} u_x(t)^i u_y(t)^j t^k$$

4.1.3. Seguridad del Criptosistema

Akiyama, Goto y Miyake proponen en [3], los siguientes parámetros, para asegurar el criptosistema ante los ataques conocidos hasta la publicación de ese artículo.

- $p = 2$.
- $d > 50$.
- $w > 5$.
- $k > 3$.

En este caso, el tamaño de la clave privada sería de aproximadamente 100 bits y el tamaño de la clave pública sería cercano a 500 bits. Puesto que no se conocía, para esta versión del criptosistema (y usando esos parámetros), un ataque más rápido que la búsqueda exhaustiva, entonces el nivel de seguridad esperado vendría dado por p^{2d+2}.

4.2. Ataque Faugere-Spaenlehauer

El punto principal del ataque es descomponer ideales, en lugar de factorizar los polinomios univariados obtenidos mediante la evaluación de $F_0 - F_1$ en la sección $(u_x(t), u_y(t), t)$. De esta manera, se puede manipular implícitamente el llamado *polinomio divisor* f durante el proceso de descifrado. En consecuencia, se evita resolver el *Problema de Búsqueda de Secciones*.

En [6], se presentan 3 versiones del ataque, una primera versión determinista basada en dos lemas fundamentales, el primero, muestra que, una vez descompuesto el ideal $\langle F_0, F_1, X\rangle = \langle f(s_0 - s_1), X\rangle$, se puede manipular f implícitamente, mediante $\langle f, X\rangle$, el segundo, muestra que se puede calcular explícitamente un ideal multivariado que contenga a m.

La segunda versión acelera el proceso considerando el campo de cocientes $\mathbb{F}_p(t)$. En efecto, los polinomios de CVA tienen un alto grado de t. Puesto que la complejidad de las bases de Gröbner es lineal en la complejidad de la aritmética en el campo base, parece natural calcular en el campo de cocientes $\mathbb{F}_p(t)$.

Por último, se utiliza un sistema modular para implementar eficazmente el ataque: se realizan cálculos en algunos campos finitos (bien elegidos) $\mathbb{F}_p[t]/(P)$ y se recuperan los resultados utilizando el Teorema Chino del Resto. Haciendo esto, el tamaño de los coeficientes de los valores intermedios están acotados (estos coeficientes pueden ser enormes cuando los cálculos se realizan en el campo de cocientes). En particular, este ataque es capaz de romper el criptosistema con los parámetros recomendados en intervalos de $0,05$ segundos. Esto permite realizar un análisis preciso de complejidad para demostrar que este ataque es casi lineal en el tamaño de la clave secreta. Experimentalmente, en [6] fueron capaces de romper con esta técnica algunos casos en que el tamaño de la clave secreta era mayor de 10000 bits.

4.2.1. Algoritmo

Ataque nivel 3 (con cá lculos en $\mathbb{F}_q$)

1. Elegir $n \approx \deg_t(m)\frac{\log(p)}{C}$ polinomios irreducibles P_i tales que $\deg(P_i) \approx \frac{C}{\log(p)}$, $\forall i \in \{1, \ldots, n\}$ y $\sum_{i=1}^{n} \deg(P_i) > \deg_t(m)$.
2. Para i desde 1 hasta n:
   a) considerar $\mathbb{K} = \mathbb{F}_p[t]/(P_i)$.
   b) calcular el resultante $R = \mathsf{Res}_x(F_0 - F_1, X) \in \mathbb{K}[y]$.
   c) Factorizar $R = \prod Q_i(y)$, tal que $Q_0(y) \in \mathbb{K}[y]$ denota el factor irreducible de mayor grado en y.
   d) calcular una base de Gröbner de orden lexicográ fico reverso del ideal $J = \langle F_0 + z, F_1 + z, X, Q_0 \rangle \subset \mathbb{K}[x, y, z]$.
   e) considerar el siguiente sistema lineal sobre $\mathbb{K}$:

$$NF_J(z) + \sum_{(i,j)\in\Lambda_m} m_{ij}(t) NF_J(x^i y^j) = 0$$

   Si el sistema no tiene solución, entonces regresar al paso 2 y elegir otro factor del resultante.
   f) Dada la solución del sistema $M_{ij}(t)$, calcular m' (mód $P_i = \sum_{(i,j)\in\Lambda_m} m_{ij}(t)x^i y^j$) donde $(m_{ij}(t))$ es la solución del sistema lineal.
3. Recuperar $m = m'$ (mód $\prod_{i=1}^{n} P_i$) usando el Teorema Chino del Resto.

Note que, el sistema lineal, en el paso 7 tiene solamente $|\Lambda_m|$ incógnitas y $\deg(J) \approx \deg_{xy}(m)\deg_{xy}(f)\deg_{xy}(X)$ ecuaciones. Luego, para parámetros prácticos, $|\Lambda_m| \approx k$ es menor que $\deg(J)$, por lo que el sistema lineal esta sobredeterminado y tiene en general una solución única.

Finalmente, el valor $\sum \deg(Pi) \approx \deg_t(m)$ depende solamente del tamaño del texto plano. Por lo tanto, el número de veces que se tiene que ejecutar el bucle principal del algoritmo es lineal en el tamaño del mensaje. Dado que el coste de las operaciones aritméticas en $\mathbb{F}_p[t]/(P)$ depende sólo de C (que es una constante elegida por el atacante), se espería que este ataque sea lineal o casi lineal en el tamaño del texto plano. De hecho, esta expectativa se confirmó mediante un análisis de complejidad y por resultados experimentales como se puede ver en [6].

4.2.2. Una Cota Inferior de la Complejidad del Algoritmo de Descifrado

La complejidad de este ataque tiene que ser comparada con un límite inferior del costo del proceso de descifrado. Durante el algoritmo de descifrado, se requiere factorizar $(F_0 - F_1)(u_x(t), u_y(t), t)$ sobre $\mathbb{F}_p[t]$. El grado de este polinomio es de al menos dw. Hasta donde saben los autores, los mejores algoritmos de factorización probabilísticos tienen una complejidad aritmética de $\tilde{\mathcal{O}}(d^2w^2 + dw\log(p))$. Por otra parte, también hay un problema de mochila para resolver después de la factorización. La complejidad de este paso es difícil estimar por lo que no se considerará aquí (ya que lo que se quiere es establecer una cota inferior). El último paso del proceso de descifrado es la resolución de un sistema lineal con $\mathcal{O}(dw)$ variables: la complejidad aritmética de este paso es $\mathcal{O}(w^3d^3)$. Por último, la complejidad binaria total del algoritmo de descifrado está acotada inferiormente por

$\mathcal{O}(\log(p)(w^3d^3 + dw\log(p)))$, que es cúbico en los parámetros d y w, y cuadrático en $\log(p)$. En comparación, el ataque Faugere-Spaenlehauer es casi lineal en d y $\log(p)$, y polinomial de grado 7 en w.

5. Criptosistema Basado en Ecuaciones Diofánticas de Tipo Grado Creciente

5.1. Preliminares Matemáticos

5.1.1. Polinomios de Tipo Grado Creciente

Definición 5.1.1 *Un polinomio $X(x) \in \mathbb{Z}[x]\setminus\{0\}$ es de **Tipo Grado Creciente** si la relación*

$$\Lambda_X \to \mathbb{Z}_{\geq 0}; i \mapsto \sum i$$

es inyectiva.

Observación 5.1.1 *Sea $X(x) \in \mathbb{Z}[x]\setminus\{0\}$, las siguientes afirmaciones son válidas:*

1. *El polinomio $X[x]$ es de tipo grado creciente si, y sólo si, el grado total de todos los monomios de $X(x)$ difiere.*
2. *Si X es de tipo grado creciente, Λ_X es un conjunto totalmente ordenado por el siguiente orden $\succ$: dados dos elementos $i = (i_1, \ldots, i_n)$, $i = (j_1, \ldots, j_n)$ en Λ_X, tenemos que $i \succ j$ si $i_1 + \cdots + i_n > j_1 + \cdots + j_n$*

En adelante, dado un polinomio X de tipo grado creciente, asumiremos que Λ_X, está dotada del orden descritos en la observación anterior.

5.1.2. Peso Ponderado y Reducción LLL

Cuando nos referimos a una norma sobre $\mathbb{R}^m$ en general nos referimos a la norma euclidiana o una norma ℓ_p para algún $p > 0$ (note que la norma ℓ_2 es la misma que la norma euclidiana). Por otro lado, un espacio ponderado se define como una espacio normado dotado de una norma especial que llamamos una norma ponderada. formalmente:

Definición 5.1.2 *Para un vector $\mathbf{w} = (w_1, \ldots, w_m) \in (\mathbb{R}^+)^m$ se define una función $||\cdot||_{\mathbf{w}} : \mathbb{R}^m \to \mathbb{R}$ como sigue:*

$$||a||_{\mathbf{w}} := \sqrt{(a_1w_1)^2 + \cdots + (a_mw_m)^2} \text{ donde } a = (a_1 \ldots, a_m) \in \mathbb{R}.$$

*No es complejo probar que $||\cdot||_{\mathbf{w}}$ es una norma en $\mathbb{R}^m$, más aún, definiremos $||\cdot||_{\mathbf{w}}$ como la **Norma Ponderada** de peso $\mathbf{w}$.*

Observación 5.1.2 *Para cualesquiera $\mathcal{L} \subset \mathbb{R}^m$ (subespacio) y $\mathbf{w} = (w_1, \ldots, w_m) \in (\mathbb{R}^+)^m$ se tiene que, el espacio $\mathcal{L}$ dotado de la norma ponderada de peso $\boldsymbol{w}$, forma un espacio ponderado de peso $\boldsymbol{w}$ al que denotaremos por $\mathcal{L}^{\mathbf{w}}$.*

5.2. Descripción del Criptosistema

5.3. Descripción del Ataque

Referencias

[1] Akiyama K., and Goto Y. (2004). Algebraic surfaces a new public key encryption. Electronics, Information and Communication Engineers Technical report. OIS, Office Information Systems, 104 (423), 13-20.

[2] Akiyama, K., and Goto, Y. (2008). An improvement of the algebraic surface public-key cryptosystem. In Proceedings of SCIS.

[3] Akiyama, K., Goto, Y., and Miyake, H. (2009). An algebraic surface cryptosystem. In Public Key Cryptography–PKC 2009 (pp. 425-442). Springer Berlin Heidelberg.

[4] Contreras, J. (2015). Implementación de un Criptosistema de Clave Pública Basado en una Variedad Algebraica y un Estudio de su Espacio de Clave.

[5] Ding, J., Kudo, M., Okumura, S., Takagi, T., and Tao, C. (2015). Cryptanalysis of a public key cryptosystem based on Diophantine equations via weighted LLL reduction.

[6] Faugere, J. C., and Spaenlehauer, P. J. (2010). Algebraic cryptanalysis of the PKC'2009 algebraic surface cryptosystem. In Public Key Cryptography–PKC 2010 (pp. 35-52). Springer Berlin Heidelberg.

[7] Iwami, M. (2008). A reduction attack on Algebraic Surface public-key Cryptosystems. In Computer Mathematics (pp. 323-332). Springer Berlin Heidelberg.

[8] Okumura, S. (2015). A public key cryptosystem based on diophantine equations of degree increasing type. Pacific Journal of Mathematics for Industry, 7(1), 1-13.

[9] Shor, P. W. (1994, November). Algorithms for quantum computation: Discrete logarithms and factoring. In Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on (pp. 124-134). IEEE.

[10] Voloch, F. (2007). Breaking the Akiyama-Goto algebraic surface cryptosystem. In Arithmetic, Geometry, Cryptography and Coding Theory, CIRM meeting.

[11] Uchiyama, S., and Tokunaga, H. (2007, January). On the security of the Algebraic Surface public-key Cryptosystems. In Proceedings of SCIS.