Universidad de Valladolid
Facultad de Ciencias

Trabajo Fin de Grado

Grado en Matemáticas

Curvas elípticas: su uso en criptografía

Autora: Celia Rodríguez Muñoz

Tutor/es: Félix Delgado de la Mata

Índice general

Introducción

El objetivo de este trabajo es presentar una primera aproximación al uso de las curvas elípticas en criptografía. El desarrollo de la criptografía de clave pública a partir de los años 80 del pasado siglo se fundamenta en la simbiosis de tres ingredientes. Estos tienen como objeto dar respuesta al problema de conseguir transmisiones seguras a través de canales inseguros. El primer ingrediente no es otro que la teoría de números, rama fundamental de las matemáticas, que hasta el momento, se desarrolló a un nivel esencialmente teórico. Su aplicación en el universo de la transmisión digital requirió incorporar a la teoría de números la visión algorítmica y computacional. Este aspecto no era completamente nuevo: muchos matemáticos se habían ocupado anteriormente del cálculo concreto y del desarrollo de algoritmos, pero ahora estos jugarán un papel esencial. El tercer ingrediente, claro está, es la organización de métodos y algoritmos en protocolos y piezas de software que den la respuesta final a las necesidades individuales, empresariales o sociales de trasmitir la información de forma segura.

El uso de las curvas elípticas en criptografía encaja perfectamente en este modelo, tal y como ocurre con otros temas más tradicionales de la teoría de números como la aritmética modular, los residuos cuadráticos y muchos otros. El énfasis de este trabajo se ha puesto en alcanzar a comprender cómo la teoría de curvas elípticas (sobre cuerpos finitos) se aplica con éxito en el desarrollo de técnicas criptográficas. Teniendo en cuenta la amplitud del estudio de las curvas elípticas y la complejidad matemática de aspectos fundamentales de las mismas, ésto obliga a que el trabajo no se ocupe de la prueba de algunos resultados claves, sino que se centra en el uso de los mismos.

La memoria se organiza en tres capítulos. El primero de ellos está dedicado a definir y desarrollar los aspectos fundamentales de las curvas elípticas desde un punto de vista geométrico y aritmético. Los puntos principales son la reducción de la ecuación de una curva elíptica a la forma de Weierstrass y la estructura de grupo. El teorema de Hasse y el de la estructura del grupo de puntos son dos resultados clave en las aplicaciones, posteriormente se incluyen condiciones suficientes para disponer de curvas elípticas con una estructura de grupo 'agradable' para su uso criptográfico.

El segundo capítulo se dedica a la revisión de algunos métodos 'clásicos' (sobre cocientes de $\mathbb{Z}$, especialmente en $\mathbb{Z}/p\mathbb{Z}$, p primo). Se centra en sistemas desarrollados en base a los dos problemas más conocidos: el problema del logaritmo discreto y el problema de factorización. En ambos casos se trata de problemas aceptados como intratables computacionalmente, es decir, de los que no se conoce un algoritmo eficiente (polinomial) que los resuelva, siendo exponenciales o subexponenciales los únicos métodos conocidos. Por supuesto, hay casos especiales que se pueden resolver de forma eficiente. Dentro de este contexto se han incluido los sistemas de ElGamal (logaritmo discreto) y RSA (factoriza-

ción). Se incluyen también un estándar de firma digital y un algoritmo de factorización, el $p-1$ de Pollard.

El tercer y último capítulo, objetivo final de la memoria, se dedica al uso de las curvas elípticas. En primer lugar veremos como se asigna a un mensaje un punto de una curva elíptica. Posteriormente se desarrollan métodos basados en el problema del logaritmo elíptico (análogo al problema del logaritmo discreto) y el problema de factorización. En este último aspecto se precisa la extensión del concepto de curva elíptica con coordenadas en $\mathbb{Z}/n\mathbb{Z}$, n entero pero no primo (curvas pseudo-elípticas). En ambos casos se comparan con los análogos 'clásicos' viendo que, en general, son más rápidos y más seguros. El capítulo se cierra con el algoritmo de factorización de Lenstra que se basa en curvas elípticas y que es, a día de hoy, uno de los algoritmos más eficaces de factorización.

Las referencias principales para elaborar este trabajo han sido las siguientes: [9], [12], [14], [17], [26], [27] y [29].

Capítulo 1

Curvas elípticas

1.1 Notaciones previas

A lo largo de la memoria $\mathbb{K}$ denotará un cuerpo. Así mismo denotamos por $\mathbb{A}^2(\mathbb{K})$ (o simplemente $\mathbb{A}^2$ si el cuerpo está claro en el contexto) al plano afín sobre $\mathbb{K}$. Fijado un sistema de referencia, un punto de $\mathbb{A}^2$ tiene coordenadas (x, y) y, por tanto, $\mathbb{A}^2$ se puede identificar con $\mathbb{K}^2$. $\mathbb{P}^2(\mathbb{K})$ (o simplemente $\mathbb{P}^2$) denota el plano proyectivo sobre $\mathbb{K}$. Fijado un sistema de referencia cada punto $P \in \mathbb{P}^2$ tiene coordenadas homogéneas $P = [x, y, z]$ entendiendo que $[x, y, z] = [\lambda x, \lambda y, \lambda z]$ para cualquier $\lambda \in \mathbb{K} - \{0\}$.

Si $L \subset \mathbb{P}^2$ es una recta proyectiva, es decir, $L \simeq \mathbb{P}^1$, $U = \mathbb{P}^2 - L$ tiene estructura de plano afín. Habitualmente usaremos $L = \{[x, y, z] \in \mathbb{P}^2 \mid z = 0\}$ y $U = \mathbb{P}^2 - L = \{[x, y, z] \in \mathbb{P}^2 \mid z \neq 0\}$. Se suele decir que L es la recta del infinito de $U \subset \mathbb{P}^2$. Un punto $P \in U$ de coordenadas $[x, y, z]$ con $z \neq 0$ se identifica con el punto de coordenadas afines $(\frac{x}{z}, \frac{y}{z}) \in \mathbb{A}^2 \simeq U$. Recíprocamente, si $Q = (x, y) \in \mathbb{A}^2 \simeq U$, el punto Q tiene coordenadas proyectivas $[x, y, 1]$.

Sea $F(X, Y, Z) \in \mathbb{K}[X, Y, Z]$ un polinomio homogéneo de grado n. Dado $(x, y, z) \in \mathbb{K}^3$ se tiene que $F(tx, ty, tz) = t^n F(x, y, z)$ para todo $t \in \mathbb{K} - \{0\}$. Por lo tanto, si $P = [x, y, z] \in \mathbb{P}^2$ se tiene que $F(x, y, z) = 0 \Leftrightarrow F(tx, ty, tz) = 0$ y tiene sentido decir que $F(P) = 0$.

Dado $f(X, Y) \in \mathbb{K}[X, Y]$ su homogeneizado respecto de la indeterminada Z es $f^*(X, Y, Z) = Z^n f(\frac{X}{Z}, \frac{Y}{Z})$ siendo n el grado del polinomio f. Así pues, f^* es el polinomio resultante de añadir a cada monomio la potencia de Z mínima necesaria para conseguir un polinomio homogéneo de grado n. Con frecuencia pondremos $F(X, Y, Z)$ para denotar $f^*(X, Y, Z)$. Recíprocamente, si $G(X, Y, Z) \in \mathbb{K}[X, Y, Z]$ es un polinomio homogéneo de grado n, a $g(X, Y) = G_*(X, Y) = G(X, Y, 1)$ se le llama el deshomogeneizado de G respecto de Z. Nótese que dado $f \in \mathbb{K}[X, Y]$ y $G \in \mathbb{K}[X, Y, Z]$ $(f^*)_* = f$ pero, en general, no es verdad que $(G_*)^* = G$

Una curva plana afín sobre $\mathbb{K}$ es un polinomio $f(X, Y) \in \mathbb{K}[X, Y]$. Identificaremos las curvas f y λf donde $\lambda \in \mathbb{K} - \{0\}$ y también polinomios, es decir curvas, que difieran en un cambio de coordenadas afines. De forma análoga, una curva plana proyectiva sobre $\mathbb{K}$ es un polinomio homogéneo $F(X, Y, Z) \in \mathbb{K}[X, Y, Z]$ con identificaciones similares, es decir, el producto por escalares no nulo y el cambio de coordenadas proyectivas.

Dada un curva proyectiva $F \in \mathbb{K}[X,Y,Z]$ denotaremos por $F(\mathbb{K})$ al conjunto sus puntos en $\mathbb{P}^2(\mathbb{K})$, es decir, $F(\mathbb{K}) = \{P = [x,y,z] \mid F(P) = 0\} \subset \mathbb{P}^2(\mathbb{K})$. Diremos que $F(\mathbb{K})$ son los puntos racionales de la curva F. Si $\mathbb{K}'$ es una extensión de $\mathbb{K}$, es decir, $\mathbb{K} \subset \mathbb{K}'$ es un subcuerpo, tiene sentido considerar también $F(\mathbb{K}')$, los puntos de F en $\mathbb{P}^2(\mathbb{K}')$. En particular, tiene especial relevancia $F(\bar{\mathbb{K}})$ siendo $\bar{\mathbb{K}}$ la clausura algebraica en $\mathbb{K}$.

Teorema 1. *(Teorema de Bézout) Dos curvas algebraicas proyectivas planas C y D de grados m y n, definidas sobre un cuerpo algebraicamente cerrado $\mathbb{K}$ y sin componente irreducible común, tienen exactamente mn puntos de intersección contados con su multiplicidad.*

Demostración. Consultar [29] anexo A4 y [9] capítulo 5. También para el concepto y definición de multiplicidad de intersección. □

Observemos que si $F \in \mathbb{K}[X,Y,Z]$ es una cúbica, es decir, el polinomio F es homogéneo de grado 3, y hay dos puntos racionales en la cúbica, tiene que haber un tercero. Sean $P, Q \in F(\mathbb{K})$, la recta L que los une tiene coeficientes en $\mathbb{K}$, y por el teorema de Bézout corta a la F en un tercer punto $R \in F(\bar{\mathbb{K}})$. Puesto que el cálculo de los puntos de intersección se reduce a resolver un polinomio con coeficientes en $\mathbb{K}$ de grado 3, el hecho de que dos de sus raíces estén en $\mathbb{K}$ implica que la tercera también lo esté. Por lo tanto, $R \in F(\mathbb{K})$.

Sea F una curva proyectiva sobre $\mathbb{K}$ y $P = [x,y,z] \in F(\mathbb{K})$. Diremos que P es un punto singular de F si $\frac{\partial F}{\partial X}(P) = 0$, $\frac{\partial F}{\partial Y}(P) = 0$ y $\frac{\partial F}{\partial Z}(P) = 0$. Una curva es no singular si no tiene ningún punto singular.

Todo polinomio $f \in \mathbb{K}[X,Y]$ de grado n se puede escribir de la forma $f = f_0 + f_1 + ... + f_{n-1} + f_n$ donde $f_i \in \mathbb{K}[X,Y]$ es homogéneo de grado i. Sea f una curva afín sobre $\bar{\mathbb{K}}$ y $0 = (0,0)$, diremos que la multiplicidad del punto 0 en f es el mínimo k tal que $f_k \neq 0$. Notesé que $0 \in f(\mathbb{K})$ si $f_0 = 0$. Se dice que el punto 0 es no singular si tiene multiplicidad 1, es decir, $f_1 \neq 0$. En este caso, $f_1 = ax + by$ es la tangente a f en 0. Si $P = (x,y) \in f(\bar{\mathbb{K}})$, la multiplicidad de P en f es la multiplicidad de $g(X,Y) = f(X+x, Y+y)$ en el punto 0. Sea F es una curva proyectiva y $Q = [x,y,z] \in F(\bar{\mathbb{K}})$, entonces no se pueden anular simultaneamente x, y y z. Supongamos que $z \neq 0$, en este caso, la multiplicidad del punto Q en F es la del punto afín $(\frac{x}{z}, \frac{y}{z})$ en la curva afín $f = F_*$.

Definición 1. *Una curva elíptica E sobre $\mathbb{K}$ se define como una curva proyectiva plana sobre $\mathbb{K}$ no singular de grado 3.*

1.1.1 Forma de Weierstrass

Puesto que toda curva elíptica E es plana y de grado 3, el polinomio $F(X,Y,Z) \in \mathbb{K}[X,Y,Z]$ que define la curva elíptica E debe ser de la forma

$$F(X,Y,Z) = A_0(X,Z)Y^3 + A_1(X,Z)Y^2 + A_2(X,Z)Y + A_3(X,Z)$$

con $A_i(X,Z)$ polinomios homogéneos de grado i en X, Z, $i = 1,2,3$.
Suponemos que el cuerpo $\mathbb{K}$ es algebraicamente cerrado. Un punto no singular $P \in E$ diremos que es de inflexión si la multiplicidad de intersección en P de la curva con su recta tangente es tres. Como consecuencia del teorema de Bézout, toda cúbica proyectiva sobre $\mathbb{K}$ tiene puntos de inflexión.

Mediante un cambio de coordenadas se puede suponer que $O = [0, 1, 0] \in E(\mathbb{K})$ es un punto de inflexión y que su recta tangente es $Z = 0$. Como $O \in E(\mathbb{K})$, se tiene que $F(0, 1, 0) = A_0(0, 0) = 0$. Esto implica que la contante $A_0(X, Z)$ es nula.

Deshomogeneizamos F respecto de Y considerando la carta afín $Y \neq 0$. Se tiene que $O = (0, 0)$ y $f(X, Z) = F(X, 1, Z) = A_1(X, Z) + A_2(X, Z) + A_3(X, Z)$. Puesto que la recta tangente en O a la curva es $Z = 0$, se tiene que $A_1(X, Z) = cZ$ con $c \neq 0$ $c \in \mathbb{K}$. Intersecamos la curva con la recta tangente $Z = 0$ en O, luego $f(X, 0) = A_1(X, 0) + A_2(X, 0) + A_3(X, 0)$. Como O es un punto de inflexión, $A_2(X, 0) = 0$ y $A_3(X, 0) = a_5X^3$ con $a_5 \neq 0$. Entonces $A_2(X, Z) = a_1XZ + a_3Z^2$ y $A_3(X, Z) = a_5X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3$ con $a_1, a_2, ..., a_6 \in \mathbb{K}$, $a_5 \neq 0$.

Luego $F = cZY^2+(a_1XZ+a_3Z^2)Y+a_5X^3+a_2X^2Z+a_4XZ^2+a_6Z^3$ con $c, a_1, a_2, ..., a_6 \in \mathbb{K}$. Puesto que tanto c como a_5 son distintos de 0, mediante otro cambio de coordenadas se tiene que $Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3$ con $a_1, a_2, ..., a_6 \in \mathbb{K}$. Así pues, cualquier curva elíptica se puede expresar en $\mathbb{P}^2(\mathbb{K})$ como

$$Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3$$

con $a_1, a_2, ..., a_6 \in \mathbb{K}$.
En la copia afín $\mathbb{A}^2(\mathbb{K})$ considerando la carta afín $Z \neq 0$, la curva sería de la forma

$$Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6.$$

Si $Char(\mathbb{K}) = 2$ la curva elíptica tiene dos formas diferentes de simplificarse en función del valor de a_1.
Si $a_1 \neq 0$, se puede hacer el cambio $X = a_1^2X' + \frac{a_3}{a_1}$ e $Y = a_1^3Y' + \frac{a_1^2a_4+a_3^2}{a_1^3}$ y obtener la curva elíptica

$$(Y')^2 + X'Y' = (X')^3 + a_2(X')^2 + a_6$$

con $a_2, a_6 \in \mathbb{K}$, $a_6 \neq 0$.
Si en cambio, $a_1 = 0$, se considera el cambio $X = X' + a_2$ y de obtiene la curva

$$Y^2 + a_3Y = (X')^3 + a_4X' + a_6$$

con $a_3, a_4, a_6 \in \mathbb{K}$, $a_6 \neq 0$.

Si $Char(\mathbb{K}) \neq 2$, mediante el cambio $Y = Y' - \frac{1}{2}(a_1X + a_3)$ la ecuación $Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6$ se transforma en

$$(Y')^2 = X^3 + aX^2 + bX + c$$

con $a, b, c \in \mathbb{K}$.

Si $Char(\mathbb{K}) = 3$ la curva elíptica de nuevo se puede simplificar en dos formas diferentes en función del valor de a.
Si $a \neq 0$, se considera el cambio $X = X' + \frac{b}{a}$ y se obtiene la curva

$$Y^2 = (X')^3 + A(X')^2 + B$$

con $A, B \in \mathbb{K}$.
Si se tiene que $a = 0$, se tiene la ecuación

$$Y^2 = X^3 + AX + B$$

con $A, B \in \mathbb{K}$.

Suponemos ahora que $Char(\mathbb{K}) \neq 2$ y $Char(\mathbb{K}) \neq 3$. Realizando el cambio $X = X' - \frac{a}{3}$ en la ecuación $Y^2 = X^3 + aX^2 + bX + c$ se tiene que

$$(Y')^2 = (X' - \frac{a}{3})^3 + a(X' - \frac{a}{3})^2 + b(X' - \frac{a}{3}) + c = (X')^3 + AX' + B$$

donde $A, B \in \mathbb{K}$.

Así pues, podemos suponer que E está definida en la copia afín $Z \neq 0$ por la ecuación

- $Y^2 + XY = X^3 + aX^2 + b$ ó $Y^2 + aY = X^3 + bX + c$ si $Char(\mathbb{K}) = 2$.
- $Y^2 = X^3 + aX^2 + b$ ó $Y^2 = X^3 + aX + b$ si $Char(\mathbb{K}) = 3$.
- $Y^2 = X^3 + aX + b$ si $Char(\mathbb{K}) \neq 2, 3$.

Además, por definición, una curva elíptica E es no singular. Veamos que condición a mayores tiene que cumplir la ecuación $E : Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6$ de una curva elíptica sobre cualquier cuerpo $\mathbb{K}$ para que esto se dé. Se denota el discriminante de E por

$$\Delta = -d_2^2 d_8 - 8d_4^3 - 27d_6^2 + 9d_2d_4d_6$$

donde $d_2 = a_1^2 + 4a_4$, $d_4 = 2a_4 + a_1a_3$, $d_6 = a_3^2 + 4a_6$, $d_8 = a_1^2a_6 + 4a_2a_6 - a_1a_3a_4 + a_2a_3^3 - a_4^2$. Dependiendo de la característica de $\mathbb{K}$ el cálculo del determinante se simplifica

$$\Delta = \begin{cases} b & \text{si} \quad E : Y^2 + XY = X^3 + aX^2 + b \ (Char(\mathbb{K}) = 2) \\ a^4 & \text{si} \quad E : Y^2 + aY = X^3 + bX + c \ (Char(\mathbb{K}) = 2) \\ -a^3b & \text{si} \quad E : Y^2 = X^3 + aX^2 + b \ (Char(\mathbb{K}) = 3) \\ -a^3 & \text{si} \quad E : Y^2 = X^3 + aX + b \ (Char(\mathbb{K}) = 3) \\ -16(4a^3 + 27b^2) & \text{si} \quad E : Y^2 = X^3 + aX + b \ (Char(\mathbb{K}) \neq 2, 3) \end{cases}$$

Veamos que es condición necesaria y suficiente que $\Delta \neq 0$ para que la curva E sea no singular, es decir, $\Delta = 0$ si y solo si $\frac{\partial f}{\partial X}$ y $\frac{\partial f}{\partial Y}$ se anulan simultaneamente siendo f el polinomio afín que define la curva elíptica E.

Si $Char(\mathbb{K}) = 2$ y $f = X^3 + aX^2 + b - Y^2 - XY$, E es singular si y solo si se cumple que $Y = 3X^2$ y que $X = 0$. En este caso, se tiene que $f = b = 0$, es decir $\Delta = 0$.

Si $Char(\mathbb{K}) = 2$ y $f = X^3 + bX + c - Y^2 - aY$, E es singular si y solo si $3X^2 + b = 0$ y $a = 0$. De ser singular, $f = 0$ únicamente en $(X, Y) = (\sqrt{-b}, \sqrt{c})$ y $\Delta = a^4 = 0$.

Si $Char(\mathbb{K}) = 3$ y $f = X^3 + aX^2 + b - Y^2$, E es singular si y solo si se cumple que $2aX = 0$ y que $-2Y = 0$. Si $a = 0$, $\Delta = 0$ y E singular en $(X, Y) = (-\sqrt[3]{b}, 0)$. Si $a \neq 0$, E es singular en $(X, Y) = (0, 0)$ si y solo si $f = b = 0$, es decir, si y solo si $\Delta = -a^3b = 0$.

Si $Char(\mathbb{K}) = 3$ y $f = X^3 + aX + b - Y^2$, E es singular si y solo si $a = 0$ y $-2Y = 0$. Es decir, si y solo si $\Delta = -a^3 = 0$.

Si $Char(\mathbb{K}) \neq 2, 3$, $f = X^3 + aX + b - Y^2$ y E tiene puntos singulares si y solo si $3X^2 + a = 0$ y $-2Y = 0$. Es decir, si $X^2 = \frac{-a}{3}$ y $Y = 0$. Sustituyendo en f, $f = \frac{-a}{3}X + aX + b = 0$, es equivalente a que $X = \frac{-3b}{2a}$. Por tanto, E tiene puntos singulares si y solo si $X^2 = \frac{-a}{3} = \frac{9b^2}{4a^2}$. Es decir, si y solo si $4a^3 + 27b^2 = 0$.

Forma de Weierstrass

Resumiendo, la ecuación de una curva elíptica E definida en la copia afín $Z \neq 0$ se reduce a una de las siguientes:

- Si $Char(\mathbb{K}) = 2$

$$Y^2 + XY = X^3 + aX^2 + b \text{ con } \Delta = b \neq 0$$

$$Y^2 + aY = X^3 + bX + c \text{ con } \Delta = a^4 \neq 0$$

- Si $Char(\mathbb{K}) = 3$

$$Y^2 = X^3 + aX^2 + b \text{ con } \Delta = -a^3 b \neq 0$$

$$Y^2 = X^3 + aX + b \text{ con } \Delta = -a^3 \neq 0$$

- Si $Char(\mathbb{K}) \neq 2, 3$

$$Y^2 = X^3 + aX + b \text{ con } \Delta = -16(4a^3 + 27b^2) \neq 0$$

A partir de ahora para el estudio en criptografía suponemos las curvas elípticas expuestas en estas ecuaciones. En particular, si $Char(\mathbb{K}) \neq 2, 3$, $Y^2 = X^3 + aX + b$ con $\Delta = -16(4a^3 + 27b^2) \neq 0$.

1.2 Estructura de grupo

El uso de las curvas elípticas en criptografía se basa en la estructura de grupo que tienen sus puntos. La construcción de la operación de grupo sobre los puntos de una curva elíptica se fundamenta en el teorema de Bézout y en el teorema siguiente.

Teorema 2. *(Versión teorema de Cayley–Bacharach) Sean C, C_1 y C_2 tres curvas cúbicas. Supongamos que C_1 corta a C en nueve puntos no singulares y que C_2 corta en ocho de ellos a C. Entonces C_2 corta también a C en el noveno punto.*

Demostración. Consultar [29] anexo A página 288. □

Sean P, Q dos puntos pertenecientes a una curva elíptica E sobre un cuerpo $\mathbb{K}$. Definimos el punto $P + Q$ como el tercer punto de intersección de la recta r que pasa por P y Q con la curva elíptica E. Esta operación esta bien definida: por P y Q pasa una única recta r con coeficientes en el cuerpo $\mathbb{K}$ y, por el teorema de Bézout, la recta r corta a la curva en un tercer punto con coordenadas $\mathbb{K}$. En el caso $P = Q$, la recta r es la tangente a la curva en P, por lo que la curva y r se cortan en P con multiplicidad mayor o igual que 2. De nuevo por el teorema de Bézout, r se cortará en otro punto racional de la curva.

Si la multiplicidad es 3 es el propio punto P.

La operación $+$ no tiene elemento neutro. No hay ningún punto $N \in E(\mathbb{K})$ tal que $P + N = P$ para todo $P \in E(\mathbb{K})$. Luego $(E(\mathbb{K}), +)$ no tiene estructura de grupo. Para definir una operación de grupo, fijamos un punto $O \in E(\mathbb{K})$ de inflexión y para $P, Q \in E(\mathbb{K})$ se define el punto $P \oplus Q$ como $O + (P + Q)$. La operación $\oplus$ esta bien definida por estar bien definida $+$. Veamos a continuación que cumple las propiedades de grupo:

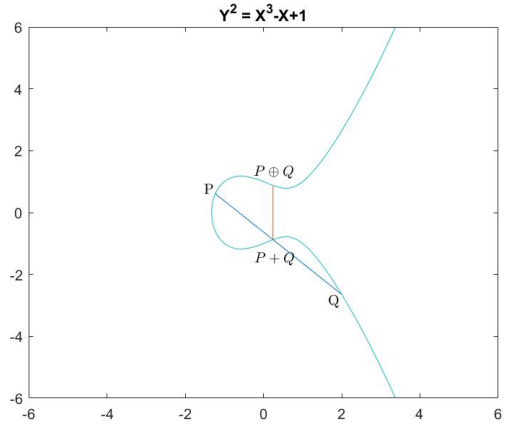

Figura 1.1: Ley de grupo en $\mathbb{R}$

- La operación es conmutativa $P \oplus Q = Q \oplus P$ para todo $P, Q \in E(\mathbb{K})$.

  Demostración. Basta con ver que $P + Q = Q + P$ para todos $P, Q \in E(\mathbb{K})$.

  La recta que une P con Q es la misma que une Q con P ya que en el plano proyectivo dos puntos definen una recta. Por tanto, la intersección de la recta que une P y Q con la curva elíptica $(P + Q)$ y intersección de la recta que une Q y P con la curva elíptica $(Q + P)$ es la misma. □

- El elemento neutro es O.

  Demostración. Veamos que $P \oplus O = P$ para todo $P \in E(\mathbb{K})$.

  Por definición $P \oplus O = O + (P + O)$. Sabemos que $P + O$ es el tercer punto de intersección de la recta que une O y P con la curva. Como dicha recta corta exactamente en tres puntos a la curva (P,O y $P + O$), la recta que pasa por O y por $P + O$ corta a la curva necesariamente en P. Entonces $P \oplus O = O + (P + O) = P$.

  En el caso $P = O$, $O + O$ es la intersección de la recta tangente en O a la curva con la propia curva. Como O es un punto de inflexión, es decir, con orden de contacto 3 con la curva, la intersección de la tangente a la curva en O es el propio punto O. Entonces $O + O = O$. Por tanto, $O \oplus O = O + (O + O) = O$. □

- El opuesto de un punto P es $\ominus P = P + O$.

  Demostración. Veamos que $P \oplus (\ominus P) = O$ para todo $P \in E(\mathbb{K})$. La recta que une P con $\ominus P$ es la recta que une P y O por definición de $\ominus P$. Luego la intersección de la

recta que pasa por P y $\ominus P$ con la curva es necesariamente O, es decir, $P+(\ominus P) = O$. Sabemos por la demostración anterior que $O + O = O$, luego $P \oplus \ominus P = O + (P + (\ominus P)) = O + O = O$. □

- La operación es asociativa $P \oplus (Q \oplus R) = (P \oplus Q) \oplus R$ para todo $P, Q, R \in E(\mathbb{K})$.

  Demostración. Primero veamos que basta con demostrar que $P + (Q \oplus R) = (P \oplus Q) + R$.

  En efecto, si $P + (Q \oplus R) = (P \oplus Q) + R$, la recta que une $P + (Q \oplus R)$ con O y la recta que une $(P \oplus Q) + R$ con O son la misma. Por tanto, $P \oplus (Q \oplus R) = (P \oplus Q) \oplus R$ si $P + (Q \oplus R) = (P \oplus Q) + R$.

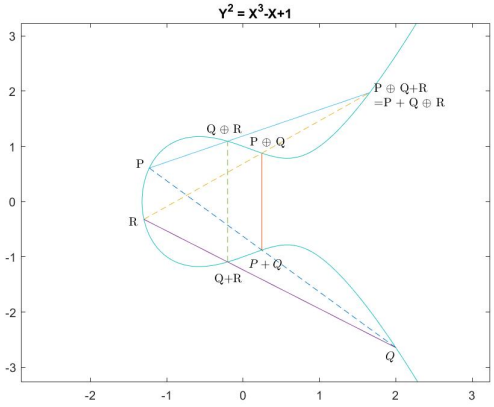

Figura 1.2: Propiedad asociativa en $\mathbb{R}$

Consideramos los ocho puntos: O, P, Q, R, $P + Q$, $P \oplus Q$, $Q + R$, $Q \oplus R$.
Sea l_1 la recta que pasa por P y por Q. Por definición, de $P + Q \in l_2$.
Sea l_2 la recta que pasa por $P + Q$ y O. Es decir, $P \oplus Q \in l_2$ por definición.
Sea l_3 la recta que pasa por $P \oplus Q$ y R. El tercer punto de intersección de la recta con la curva elíptica E es $P \oplus Q + R$.
Sea r_1 la recta que pasa por Q y R. Cono consecuencia, $Q + R \in r_1$.
Sea r_2 la recta que pasa por $Q + R$ y O. Es decir, $Q \oplus R \in r_2$.
Sea r_3 la recta que pasa por $Q \oplus R$ y P. El tercer punto de intersección de la recta con la curva elíptica E es $Q \oplus R + P$.

Las tres rectas l_1, l_3 y r_2 forman la ecuación de una cúbica C_1. Luego la cúbica corta a la curva elíptica E en los ocho puntos mencionados anteriormente por pertenecer a alguna de las tres rectas. Además, $(P \oplus Q) + R$ es otro punto de intersección de C_1 con E porque $(P \oplus Q) + R \in l_3$.

También, las tres rectas r_1, r_3 y l_2 forman la ecuación de una cúbica, es este caso, C_2. Dado que los ocho puntos mencionados pertenecen a alguna de las tres rectas, la curva E se corta con la cúbica en esos ocho puntos. A mayores C_2 se interseca con E en el punto $P + (Q \oplus R)$ ya que este pertenece a r_3.

Por el teorema de Cayley–Bacharach, C_2 corta también a E en el noveno punto $(P \oplus Q) + R$. Entonces ese punto tiene que coincidir con $P + (Q \oplus R)$. Por tanto, $P + (Q \oplus R) = (P \oplus Q) + R$. □

Una vez definida la operación de grupo $\oplus$, denotaremos por $kP := P \oplus P... \oplus P$ (k sumandos) para $P \in E(\mathbb{K})$ y $k \in \mathbb{N}$. Sea $P \in E(\mathbb{K})$, P es un punto de torsión si existe $k \in \mathbb{N}$, $k > 1$, tal que $kP = O$. Recordemos que el orden de P es el mínimo $k \in \mathbb{N}$ tal que $kP = O$. Si $kP \neq O$ para todo $k \in \mathbb{N}$, diremos que P tiene orden infinito.

1.2.1 Expresión analítica de la operación de grupo

Para toda curva elíptica E sobre el cuerpo $\mathbb{K}$ vamos a considerar la ecuación de Weierstrass y como punto O, el punto del infinito con coordenadas $O = [0, 1, 0]$. Sean $P, Q \in E(\mathbb{K})$ con $P, Q \neq O$. Si $P = O$ entonces sabemos que $P \oplus Q = O \oplus Q = Q$ por ser O el elemento neutro. Análogo para $Q = O$. Luego podemos suponer que tanto P como Q son puntos afines de la siguiente forma: $P = (x_1, y_1)$ y $Q = (x_2, y_2)$, $x_1, x_2, y_1, y_2 \in \mathbb{K}$.

Consideramos la recta vertical afín que pasa por $P = (x_1, y_1)$ y por O, $X = x_1$. Esta recta corta a la curva en P, en O y en un tercer punto punto $P^{'}$. Entonces $P \oplus P^{'} = O$ ya que O está en la recta que une los puntos P y $P^{'}$ y $O + O = O$. Luego necesariamente $P^{'}$ tiene que ser el inverso de P.

Suponemos que $\boldsymbol{Char(\mathbb{K}) \neq 2, 3}$, luego E viene dada por $f = X^3 + aX + b - Y^2$ donde $a, b \in \mathbb{K}$. Como la curva elíptica $E(\mathbb{K})$ es simétrica respecto el eje X, observamos que $P^{'}$ y P tienen que ser simétricos respecto el eje X. Por tanto, si $P = (x_1, y_1)$, $P^{'} = \ominus P = (x_1, -y_1)$. Además, dados $P = (x_1, y_1)$ y $Q = (x_2, y_2)$, si $P = \ominus Q$, es decir, $y_2 = -y_1$ se tiene que $P \oplus Q = O$.

Puesto que $P \oplus Q = O + (P + Q) = \ominus(P + Q)$ con $P, Q \in E(\mathbb{K})$, si $P + Q = (x, y)$, $P \oplus Q = (x_3, y_3)$ con $x_3 = x$ e $y_3 = -y$. Veamos cual es el punto $P + Q = (x, y)$.

Consideramos la recta afín r que une el punto $P = (x_1, y_1)$ con $Q = (x_2, y_2)$,

$$r : Y - y_1 = m(X - x_1)$$

Si $P = Q$, r es la recta tangente en P a la curva, es decir,

$$r : \frac{\partial f}{\partial X}(P)(X - x_1) + \frac{\partial f}{\partial Y}(P)(Y - y_1) = 0$$

Luego m toma el valor $m = \dfrac{3x_1^2 + a}{2y_1}$.

Si $P \neq Q$, es decir, $x_1 \neq x_2$, m toma el valor $m = \dfrac{y_2 - y_1}{x_2 - x_1}$.

Al intersecar la recta r con la curva elíptica tenemos que $(m(X - x_1) + y_1)^2 = X^3 + aX + b$, es decir,

$$X^3 - m^2X^2 + CX + D = 0$$

con $C = a + 2m(mx_1 - 1)$ y $D = b - y_1^2 + mx_1(2 - mx_1)$. Sabemos por el teorema de Bézout que la recta r se corta en tres puntos con la curva, P, Q y $P + Q$. Luego

$$\begin{aligned} X^3 - m^2X^2 + CX + D &= (X - x_1)(X - x_2)(X - x) \\ &= X^3 - (x_1 + x_2 + x)X^2 + (x_1x_2 + x_1x + x_2x)X - x_1x_2x \end{aligned}$$

Por tanto, $x = m^2 - x_1 - x_2$. Sustituyendo el valor de x en la recta r calculamos y, $y = m(x-x_1)+y_1$. Entonces, $P \oplus Q = (x_3, y_3) = (x, -y) = (m^2 - x_1 - x_2, m(x_1 - x_3) - y_1)$.

En resumen, si $Char(\mathbb{K}) \neq 2, 3$, $P = (x_1, y_1)$ y $Q = (x_2, y_2)$, $P \oplus Q$ es de la siguiente forma:

<table>
<tr><td>$P = \ominus Q$</td><td colspan="3">$P \oplus Q = O$</td></tr>
<tr><td rowspan="2">$P \neq \ominus Q$</td><td>$P \neq Q$</td><td>$m = \dfrac{y_2 - y_1}{x_2 - x_1}$</td><td rowspan="2">$P \oplus Q = \begin{cases} x_3 = m^2 - x_1 - x_2 \\ y_3 = m(x_1 - x_3) - y_1 \end{cases}$.</td></tr>
<tr><td>$P = Q$</td><td>$m = \dfrac{3x_1^2 + a}{2y_1}$</td></tr>
</table>

De la misma forma se obtiene una expresión analítica de $P \oplus Q$ con $P = (x_1, y_1)$, $Q = (x_2, y_2)$ y $P \neq \ominus Q$ si $Char(\mathbb{K}) = 2$ o si $Char(\mathbb{K}) = 3$.

Sea $\boldsymbol{Char(\mathbb{K}) = 2}$.

Si la curva elíptica es de la forma $E : Y^2 + XY = X^3 + aX^2 + b$ se tiene que $P \oplus Q = (x_3, y_3)$:

- Si $P \neq Q$, $P \oplus Q = \begin{cases} x_3 = \left(\dfrac{y_1 + y_2}{x_1 + x_2}\right)^2 - \dfrac{y_1 + y_2}{x_1 + x_2} + a + x_1 + x_2 \\ y_3 = \dfrac{y_1 + y_2}{x_1 + x_2}(x_1 + x_3) + x_3 + y_1 \end{cases}$
- Si $P = Q$, $P \oplus Q = \begin{cases} x_3 = x_1^2 + \dfrac{b}{x_1^2} \\ y_3 = x_1^2 + x_3\left(x_1 + \dfrac{y_1}{x_1}\right) + x_3 \end{cases}$

Si la curva elíptica es de la forma $E : Y^2 + aY = X^3 + bX + c$ se tiene que $P \oplus Q = (x_3, y_3)$:

- Si $P \neq Q$, $P \oplus Q = \begin{cases} x_3 = \left(\dfrac{y_1 + y_2}{x_1 + x_2}\right)^2 + x_1 + x_2 \\ y_3 = \dfrac{y_1 + y_2}{x_1 + x_2}(x_1 + x_3) + y_1 + a \end{cases}$
- Si $P = Q$, $P \oplus Q = \begin{cases} x_3 = \dfrac{x_1^4 + b^2}{a^2} \\ y_3 = \dfrac{x_1^2 + b}{a}(x_1 + x_3) + y_1 + a \end{cases}$

Si $\boldsymbol{Char(\mathbb{K}) = 3}$, la curva elíptica es $Y^2 = X^3 + aX^2 + bX + c$ y $P \oplus Q = (x_3, y_3)$:

- Si $P \neq Q$, $P \oplus Q = \begin{cases} x_3 = \left(\dfrac{y_1 + y_2}{x_1 + x_2}\right)^2 - a - x_1 - x_2 \\ y_3 = \dfrac{y_1 + y_2}{x_1 + x_2}(x_1 - x_3) \end{cases}$

- Si $P = Q$, $P \oplus Q = \begin{cases} x_3 = \left(\dfrac{3x_1^2 + 2ax_1 + b}{2y_1}\right)^2 - a - 2x_1 \\ y_3 = -y_1 + \dfrac{3x_1^2 + 2ax_1 + b}{2y_1}(x_1 - x_3) \end{cases}$

Ejemplo 1. Sea $\mathbb{K} = \mathbb{F}_5$. Consideramos $a = 1$ y $b = 4$ y la curva elíptica de la forma $Y^2 = X^3 + aX + b$ sobre $\mathbb{F}_5$. Observemos que $\Delta = -16(4a^3 + 27b^2) = -6976 = 1 \neq 0$.

Simplemente sustituyendo los posibles valores de X en $\mathbb{F}_5$ vemos que $E(\mathbb{F}_5)$ es un grupo abeliano de orden 9. Sus elementos son

$$E(\mathbb{F}_5) = \{[0,1,0], [0,2,1], [0,3,1], [1,1,1], [1,4,1], [2,2,1], [2,3,1], [3,2,1], [3,3,1]\}$$

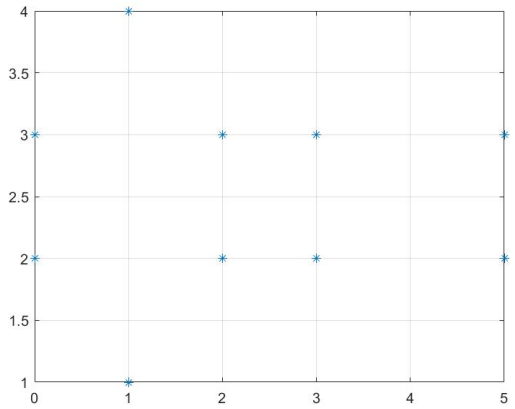

Figura 1.3: Puntos de la curva $Y^2 = X^3 + X + 4 \pmod 5$

Sea $P = [2,3,1]$. Veamos el orden de P.

Calculamos $P \oplus P$. Tenemos que $m = \frac{3x_1^2+a}{2y_1} = \frac{2^2 3+1}{6} = 3$. Por lo tanto,
$x_3 = m^2 - x_1 - x_2 = 3^2 - 2 - 2 = 0$ e $y_3 = m(x_1 - x_3) - y_1 = 3(2-0) - 3 = 3$
Así pues, $P \oplus P = 2P = [0,3,1]$

De forma similar calculamos los siguientes puntos:

$3P = 2P \oplus P = [3,2,1]$ $\quad 4P = 2P \oplus 2P = [1,1,1]$ $\quad 5P = [1,4,1] = \ominus 4P$
$6P = [3,3,1] = \ominus 3P$ $\quad 7P = [0,2,1] = \ominus 2P$ $\quad 8P = [2,2,1] = \ominus P$
$9P = 8P \oplus P = O = [0,1,0]$

Por tanto, el orden de P es 9. Como $E(\mathbb{F}_5)$ tiene 9 elementos, $E(\mathbb{F}_5)$ es un grupo cíclico generado por P.

1.2.2 Implementación ley de grupo

Algoritmo $P \oplus Q$ sobre una curva elíptica $E(\mathbb{F}_p)$ con $p \neq 2, 3$:
Entrada: $P = (x_1, y_1)$, $Q = (x_2, y_2)$ (puntos a sumar) y a,b,p (parámetros de la curva elíptica).
Salida: $R = P \oplus Q$.
Inicio

1. Comprobar que los parámetros a, b y p definen una curva elíptica ($p > 3$ primo y $4a^3 + 27b^2 \neq 0 \pmod p$).
2. Si $P = O$ o $Q = O$ entonces
   (a) Si $P = O$ entonces $R \leftarrow Q$
   (b) Si $Q = O$ entonces $R \leftarrow P$
3. Si $x_1 = x_2$ y $y_1 \neq y_2$ entonces $R \leftarrow O$
4. Sino
   (a) Si $x_1 \neq x_2$ y y $y_1 \neq y_2$ entonces
      - Calcular el inverso de $x_2 - x_1$ módulo p
      - $m \leftarrow (y_2 - y_1)(x_2 - x_1)^{-1} \pmod p$
   (b) Si $x_1 = x_2$ y $y_1 = y_2$ entonces
      - Calcular el inverso de $2y_1$ módulo p
      - $m \leftarrow (3x_1^2 + a)(2y_1) \pmod p$

   $x_3 \leftarrow m^2 - x_1 - x_2 \pmod p$
   $y_3 \leftarrow m(x_1 - x_3) - y_1 \pmod p$
   $R \leftarrow (x_3, y_3)$

Fin

Algoritmo kP sobre una curva elíptica $E(\mathbb{F}_p)$ con $p \neq 2, 3$:
Entrada: $P = (x, y)$ (punto a sumar) y a,b,p (parámetros de la curva elíptica).
Salida: $R = kP$.
Inicio

1. Comprobar que los parámetros a, b y p definen una curva elíptica (p primo mayor que 3 y $4a^3 + 27b^2 \neq 0 \pmod p$) y que $k > 0$.
2. Inicializamos $Q \leftarrow P$ y $R \leftarrow O$
3. Mientras $Q \neq O$ entonces
   (a) Si $k = 1 \pmod 2$ entonces $R \leftarrow R \oplus Q$.

   $Q \leftarrow Q \oplus Q$
   $k \leftarrow \lfloor k/2 \rfloor$
4. $kP \leftarrow R$

Fin

1.3 Curvas elípticas sobre $\mathbb{F}_q$

Para la criptografía se consideran las curvas elípticas definidas sobre un cuerpo finito $\mathbb{F}_q$ con $q = p^n$ donde p es un número primo y $n \in \mathbb{N}$. Para cualquier cuerpo $\mathbb{K}$ extensión finita de $\mathbb{F}_q$, el conjunto de puntos $E(\mathbb{K})$ con coordenadas en $\mathbb{K}$ es un grupo, de hecho un subgrupo de $E(\bar{\mathbb{F}}_q)$. Si $\mathbb{K} \subset \mathbb{K}'$, entonces $E(\mathbb{K})$ es un subgrupo de $E(\mathbb{K}')$. Evidentemente el grupo $E(\mathbb{F}_q)$ es finito por ser $\mathbb{F}_q$ finito, y abeliano. La estructura de grupo viene recogida en el siguiente teorema.

Teorema 3. *(Teorema fundamental) Sea $(E(\mathbb{F}_q),\oplus)$ el grupo de puntos de una curva elíptica sobre $\mathbb{F}_q$ con su operación de suma. Entonces, o bien $(E(\mathbb{F}_q),\oplus)$ es un grupo cíclico, es decir, generado por un solo elemento, o bien se puede descomponer como suma directa de dos subgrupos cíclicos de órdenes n y m, respectivamente, de forma que $E(\mathbb{F}_q) \cong \mathbb{Z}/n\mathbb{Z} \oplus \mathbb{Z}/m\mathbb{Z}$ donde m divide a n, m divide a $q-1$ y $\#E(\mathbb{F}_q) = nm$*

Demostración. La prueba queda lejos de los contenidos razonables de este trabajo. Se puede consultar en [30] sección 3.2 y en [6].

□

1.3.1 Número de puntos de una curva elíptica

A la hora de utilizar en criptografía una curva elíptica E sobre $\mathbb{F}_p$ es necesario que el número de puntos de dicha curva sea lo suficientemente grande para garantizar la seguridad del criptosistema.

El número de puntos de una recta $Y = aX + b$ en el plano afín sobre $\mathbb{F}_p$ viene determinado por el número de valores que puede tomar X, ya que el valor Y está definido a partir del valor de X. En el plano proyectivo la recta tiene un punto adicional, el punto del infinito. Por tanto, la recta proyectiva tiene $p+1$ puntos.

Dada una cónica $C_1 :\ aX^2+bXY+cX+dY^2+eY+f$ con coeficientes en $\mathbb{F}_p$ se puede construir una biyección entre los puntos de la cónica y los puntos de una recta L dada. Considerando el haz de rectas que pasan por un punto fijo $O \in C_1$, a cada punto P de la cónica se le asocia el punto intersección de la recta L con la recta que pasa por P y por O, y a cada punto Q de la recta se le asocia el punto intersección de la cónica con la recta que pasa por Q y por O. Esta biyección está bien definida gracias al teorema de Bézout, la intersección de una recta y una cónica da lugar a dos puntos contando multiplicidades y la intersección de dos rectas distintas a un punto. Entonces, una cónica proyectiva no singular tiene exactamente $p+1$ puntos en $C_1(\mathbb{F}_p)$.

Se considera ahora la curva $C :\ Y^2 = f(X)$ donde $f(X) \in \mathbb{F}_p[X]$. Si $p \neq 2$, la mitad de los elementos del cuerpo $\mathbb{F}_p$ son residuos cuadráticos y la otra mitad no. Entonces si f es una biyección entre los elementos de $\mathbb{F}_p$, se tiene que $f(x)$ es residuo cuadrático para la mitad de valores $x \in \mathbb{F}_p - \{0\}$.

En general f no es una biyección, pero es plausible que proporcione un función pseudoaleatoria $f : \mathbb{F}_p \to \mathbb{F}_p$. Por lo tanto, dado $x \in \mathbb{F}_p^*$ podemos estimar que $f(x)$ es un cuadrado para aproximadamente la mitad de los valores de x. Sustituimos en X los valores de $\mathbb{F}_p$ y observamos la ecuación $Y^2 = f(x)$. Si $f(x) \neq 0$, hay aproximadamente una probabilidad de $\frac{1}{2}$ de que Y tome dos valores y la misma probabilidad de que $Y^2 = f(x)$ no tenga solución. En el caso en el que $f(x) = 0$, Y solo puede tomar el valor, $Y = 0$. Puesto que hay p

posibilidades de escoger X y de cada una de esas posibilidades aproximadamente un 50 % producen dos soluciones y el otro 50 % no produce ninguna, se tienen aproximadamente p puntos en C más el punto del infinito. Entonces el número de puntos de C sobre $\mathbb{F}_p$ es $p+1+error$ donde el $error$ es pequeño comparándolo con p. Observamos que el número de puntos de la curva C no es muy diferente al número de puntos de una recta.

Ahora bien, sea $f(X) = X^3 + aX + b$, es decir, $Y^2 = f(X)$ es una curva elíptica sobre $\mathbb{F}_p$. La estimación aproximada anterior se detalla en el siguiente teorema que permite acotar el '$error$'.

Teorema 4. *(Teorema Hasse) Sea E una curva elíptica sobre $\mathbb{F}_q$ con $q = p^n$, $n \in \mathbb{N}$ y $p > 2$ primo. Se cumple que $|\#E(\mathbb{F}_q) - (q+1)| \leq 2\sqrt{q}$.*

Demostración. Consultar [27] capítulo 5. □

Este teorema da una cota del número de puntos que puede tener una curva elíptica sobre un cuerpo finito. No es trivial calcular el número de puntos exactos de $E(\mathbb{F}_q)$. Existen diferentes formas de intentar su cálculo: por fuerza bruta probando con todos los puntos $P \in \mathbb{F}_q^2$, calculando el subgrupo generado por P para un cierto $P \in E(\mathbb{F}_p)$ si el grupo $E(\mathbb{F}_q)$ es cíclico o mediante el algoritmo de Schoof [25] entre otros. Para curvas sobre cuerpos finitos con el número de elementos q elevado, los tres métodos no calculan en un tiempo razonable el número de puntos de $E(\mathbb{F}_p)$. En el primer método porque se tienen que calcular q^2 puntos y comprobar cuales pertenecen a $E(\mathbb{F}_p)$. En el segundo método porque como mínimo para hallar $\langle P \rangle$ se tienen que calcular $2\sqrt{q}-q-1$ sumas de puntos si el grupo $E(\mathbb{F}_p)$ es cíclico, porque si no lo es no funciona el método. Por último, el algoritmo de Schoof tiene una complejidad polinomial de $O(\log_2^8(q))$ que en la práctica no es demasiado útil para q grande. No obstante sí permite precalcular curvas con un número de puntos grande.

Conociendo el número de puntos de una curva elíptica sobre $\mathbb{F}_p$, es sencillo calcular el número de puntos de una curva elíptica sobre la extensión del cuerpo $\mathbb{F}_{p^n}$. Sea $\#E(\mathbb{F}_p) = 1 + p - t$, la conjetura de Weil [31] afirma y demuestra que $\#E(\mathbb{F}_{p^n}) = 1 + p^n - \alpha^n - \beta^n$ donde $\alpha, \beta \in \mathbb{C}$ son las raíces conjugadas de $X^2 + tX + p = 0$.

Por otro lado, para curvas elípticas sobre cuerpos finitos que cumplen ciertas características sí es posible saber de forma exacta el número de puntos de la curva. Además se puede determinar la estructura de grupo. Esto se observa en los siguientes lemas.

Lema 1. *Sea $p > 2$ un primo tal que $p \equiv 2 \pmod 3$. Para $0 < b < p$, sea E la curva elíptica $Y^2 = X^3 + b$ sobre $\mathbb{F}_p$. Entonces $E(\mathbb{F}_p)$ es un grupo cíclico y su orden es $\#E(\mathbb{F}_p) = p + 1$.*

Demostración. Veamos primero que el orden es $p+1$. Para ello es necesario el teorema de Cauchy.

Teorema 5. *(Teorema de Cauchy) Sea G un grupo finito y p un primo. Si p divide al orden de G entonces G tiene un elemento de orden p.*

Puesto que $p \equiv 2 \pmod 3$, la aplicación que envía $x \in \mathbb{F}_p$ a $x^3 \in \mathbb{F}_p$ es una permutación de los elementos de $\mathbb{F}_p$. Veámoslo. Sea

$$\phi : \begin{array}{ccc} (\mathbb{F}_p)^* & \longrightarrow & (\mathbb{F}_p)^* \\ x & \longmapsto & x^3 \end{array}$$

Dado que $\phi(1) = 1$ y $\phi(xy) = \phi(x)\phi(y)$ para $x, y \in (\mathbb{F}_p)^*$, ϕ es un homomorfismo de grupos, y por tanto, $\operatorname{Im}\phi \subset (\mathbb{F}_p)^*$ es un subgrupo. Sea $G = (\mathbb{F}_p)^*/\operatorname{Im}\phi$, si $g \in G$ entonces $g^3 = 1$ en G. Por lo tanto, si $g \neq 1$, el orden de g es 3. Como consecuencia del teorema de Cauchy, necesariamente el orden de G tiene que ser 3^r con $r \in \mathbb{Z}_+$. Esto implica que $3^r \mid |(\mathbb{F}_p)^*| = p - 1$. Luego, $p - 1 \equiv 0 \pmod 3$ si $r > 0$. Es decir, $p \equiv 1 \pmod 3$ si $r > 0$. Esto es absurdo ya que $p \equiv 2 \pmod 3$. Por tanto, $\operatorname{Im}\phi = (\mathbb{F}_p)^*$.

Puesto que ϕ es una permutación, hay $p-1$ elementos de la forma x^3 tal que $x \in (\mathbb{F}_p)^*$. Dado b, $x^3 + b \in \mathbb{F}_p$ es residuo cuadrático para la mitad de valores de x^3. Por tanto, hay exactamente $(p-1)/2$ elementos $x \in \mathbb{F}_p$ tal que $x^3 + b$ sea residuo cuadrático. Además, para cada residuo cuadrático x^3+b, y puede tomar dos valores. Entonces $E(\mathbb{F}_p)$ tiene $p-1$ puntos de la forma $(x, \pm z)$ donde $z^2 \equiv x^3 + b \pmod p$. Por ser ϕ una permutación, existe un elemento b' de $(\mathbb{F}_p)^*$ tal que $(b')^3 = -b$. Entonces $(b', 0)$ pertenece a $E(\mathbb{F}_p)$. Como O también pertenecen a $E(\mathbb{F}_p)$, $E(\mathbb{F}_p)$ tiene exactamente $p + 1$ puntos.

Para ver que $E(\mathbb{F}_p)$ es cíclico suponemos que no lo es, es decir, por el teorema fundamental $E(\mathbb{F}_p) \simeq \mathbb{Z}/n_1\mathbb{Z} \times \mathbb{Z}/n_2\mathbb{Z}$ con $n_1 n_2 = p + 1$, $n_2 \mid n_1$ y $n_2 \mid p - 1$. Puesto que $p + 1 = n_1 n_2$ y $n_2 \mid n_1$, se tiene que $p + 1 = p - 1 + 2 = n_1 n_2$ y existe $k \in \mathbb{N}$ tal que $p-1 = kn_2$. Luego $p-1+2 = kn_2+2 = n_1 n_2$, y por tanto, $n_2(n_1-k) = 2$. Necesariamente $n_2 = 2$. De no ser así, $n_2 = 1$ y $E(\mathbb{F}_p)$ sería cíclico. Además, n_1 tiene que ser par porque $n_2 = 2$ y $n_2 \mid n_1$. El grupo $\mathbb{Z}/n_1\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ con n_1 par tiene que tener cuatro elementos que coincidan con su inverso, es decir, elementos P con $P = \ominus P$. Esto se debe a que el grupo $\mathbb{Z}/n_1\mathbb{Z}$ con n_1 par tiene dos elementos que coinciden con su inverso, 0 y $n_1/2$, y a que en el grupo $\mathbb{Z}/2\mathbb{Z}$ sus dos elementos coinciden con su inverso. Sin embargo, en el grupo $E(\mathbb{F}_p)$ los puntos que coinciden con su inverso tienen que ser de la forma (x, y) con $y = 0$ o ser el propio O. Luego en $E(\mathbb{F}_p)$ los únicos puntos son los de la forma $(b', 0)$ con $(b')^3 \equiv -b \pmod p$ y el punto O. Puesto que b' es único, esto contradice que $E(\mathbb{F}_p) \simeq \mathbb{Z}/n_1\mathbb{Z} \times \mathbb{Z}/n_2\mathbb{Z}$. Por tanto, $E(\mathbb{F}_p)$ es cíclico.

□

Lema 2. *Sea $p > 2$ un primo tal que $p \equiv 3 \pmod 4$. Para $0 < a < p$, sea E la curva elíptica $Y^2 = X^3 + aX$ sobre $\mathbb{F}_p$. Entonces $E(\mathbb{F}_p)$ es un grupo cíclico si a es un residuo cuadrático módulo p o es de la forma $E(\mathbb{F}_p) \simeq \mathbb{Z}/_{(p+1/2)}\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ de no ser así. Además, su orden es $\#E(\mathbb{F}_p) = p + 1$.*

Demostración. Veamos primero que el orden es $p + 1$. para ello necesitamos el siguiente resultado.

Proposición 1. *En $\mathbb{F}_p$, (-1) es residuo cuadrático si y solo si $\frac{p-1}{2}$ es par, es decir, si y solo si $p \equiv 1 \pmod 4$*

Puesto que $p \equiv 3 \pmod 4$, se tiene que si $x \in (\mathbb{F}_p)^*$ es residuo cuadrático módulo p, $-x$ no lo es y viceversa. Esto se debe a que si x e $-x$ son cuadrados, es decir, existen $z_1, z_2 \in (\mathbb{F}_p)^*$ tal que $x \equiv z_1^2 \pmod p$ y $-x = z_2^2 \pmod p$, se tiene que $-x \equiv -z_1^2 \equiv z_2^2$. Luego $(\frac{z_2}{z_1})^2 \equiv -1 \pmod p$ es absurdo ya que -1 no es residuo cuadrático si $p \equiv 3 \pmod 4$. Observemos que $f(X) = X^3 + aX$ es una función impar, es decir $f(-X) = -f(X)$. Consideramos los $(p-1)/2$ pares de la forma $[x, -x]$ con $0 < x \leq (p-1)/2$. Para cada par se pueden dar tres casos: $f(x) = f(-x) = 0$, $f(x)$ residuo cuadrático o $f(-x)$ residuo cuadrático. Para cada caso existen exactamente dos puntos y de $E(\mathbb{F}_p)$ asociado a $[x, -x]$: $(\pm x, 0)$, $(x, \pm y_1)$ con $f(x) \equiv y_1^2 \pmod p$ o $(x, \pm y_2)$ con $f(-x) \equiv y_2^2 \pmod p$ respectivamente. Entonces $E(\mathbb{F}_p)$ tiene exactamente $p - 1$ puntos de esa forma más el punto $(0, 0)$

y el punto O. Es decir, $\#E(\mathbb{F}_p) = p + 1$.

De forma análoga a la demostración del lema 1, vemos que $E(\mathbb{F}_p) \simeq \mathbb{Z}/_{(p+1/2)}\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ si el grupo tienen cuatro puntos que coinciden con su inverso y $E(\mathbb{F}_p)$ es cíclico de no tener esos cuatro puntos. Los puntos $(0, 0)$ y O pertenecen a $E(\mathbb{F}_p)$ y coinciden con su inverso. Si a no es residuo cuadrático módulo p, lo es $-a$. Luego existe $\bar{a} \in \mathbb{F}_p)$ tal que $\bar{a}^2 \equiv -a$ (mód p). Entonces $(\pm\bar{a}, 0) \in E(\mathbb{F}_p)$, y por tanto, $E(\mathbb{F}_p)$ tendría cuatro puntos que coinciden con su inverso. De ser a residuo cuadrático, $E(\mathbb{F}_p)$ tendría solo dos puntos que coinciden con su inverso.

□

Fijado el cuerpo finito $\mathbb{F}_q$, para el uso en criptografía de las curvas elípticas es conveniente que existan bastantes curvas elípticas sobre dicho cuerpo y que el número de puntos de dichas curvas sea suficientemente grande. Veremos en 3.3.3 que esto es clave en el algoritmo de factorización de Lenstra.

Por el teorema de Hasse, el número de puntos de una curva elíptica sobre $\mathbb{F}_q$ varía en el intervalo $[q+1-2\sqrt{q}, q+1-2\sqrt{q}]$. Luego si $q = p$, en el cuerpo $\mathbb{F}_p$, el grupo de puntos de una curva elíptica tiene orden en el intervalo $[p+1-2\sqrt{p}, p+1-2\sqrt{p}]$. En el subintervalo $[p+1-\sqrt{p}, p+1-\sqrt{p}]$ el orden se distribuye uniformemente. Además, existe al menos una curva elíptica sobre $\mathbb{F}_p$ con $\#E(\mathbb{F}_p) = p + 1 + t$ para todo $|t| \leq 2\sqrt{p}$ (Ver [17]). Como consecuencia, el número mínimo de curvas elípticas sobre un mismo cuerpo $\mathbb{F}_p$ es igual al número de enteros t tal que $|t| \leq 2\sqrt{p}$.

En el capítulo 3 veremos que no todas las curvas elípticas sobre $\mathbb{F}_q$ son válidas para que los métodos criptográficos sean seguros. Existen dos grupos de curvas que en particular van a ser criptográficamente débiles: curvas anómalas y supersingulares. Las curvas anómalas son aquellas que $\#E(\mathbb{F}_q) \equiv 0$ (mód p) y las curvas supersingulares las que $\#E(\mathbb{F}_q) = q + 1 \pm t$ donde $p \mid t$.

Capítulo 2

Primeros métodos criptográficos de clave pública

La criptografía consiste en proteger de forma segura información que se va a trasmitir o almacenar, es decir, es el conjunto de métodos y algoritmos capaces de cifrar un texto mediante el uso de claves para transformarlo en otro texto. A lo largo de la historia se ha necesitado el uso de la criptografía para ocultar información a ciertas de personas o grupos, por ejemplo, un secreto guardado entre un grupo de personas o para mantener en secreto datos de un negocio. A día de hoy, estos métodos son esenciales, por ejemplo, las tarjetas de crédito, los mandos de los coches, los mensajes a través de redes sociales y el correo electrónico basan su seguridad en la criptografía. Al igual que surge la necesidad de proteger cierta información, surge el objetivo de descifrar y encontrar esa información oculta. A esto se dedica el cripotanálisis mediante el estudio de los métodos criptográficos y de los textos cifrados. El criptoanálisis junto con la criptografía forman la criptología.

Denotaremos por $C:\ \mathcal{M}\times\mathcal{K}\to\mathcal{C}$ con $(m,k)\mapsto c=C(m,k)$ a la función de cifrado (o función de encriptado) donde $\mathcal{M}$ denota el conjunto de mensajes en claro o información a trasmitir, $\mathcal{C}$ el conjunto de mensajes cifrados (o encriptados) y $\mathcal{K}$ el conjunto de claves necesarias para el proceso de cifrado. La función de descifrado (o función de desencriptado) es de la forma $D:\ \mathcal{C}\times\mathcal{K}\to\mathcal{M}$ con $(c,k^*)\mapsto d=D(c,k^*)$. Estas funciones cumplen que el descifrado de un mensaje previamente cifrado con una clave k es el mensaje original, es decir, para $(m,k)\in\mathcal{M}\times\mathcal{K}$ existe $k'\in\mathcal{K}$ tal que $D(C(m,k),k')=m$. Al conjunto de funciones de cifrado y descifrado lo denominaremos sistema criptográfico (o criptosistema). En el modelado de un criptosistema llamaremos A (Alice) al usuario que envía el mensaje $m\in\mathcal{M}$ cifrado como $c\in\mathcal{C}$, y B (Bob) al usuario que recibe el mensaje cifrado c y lo descifra para obtener en mensaje original m.

Los métodos criptográficos se pueden clasificar en criptosistemas simétricos (o de clave privada) y criptosistemas asimétricos (o de clave pública). En el primero, las claves $k,k'\in\mathcal{K}$ usadas en la transmisión de un mensaje solo son conocidas por el emisor y el receptor del mensaje y dichas claves son las mismas para cifrar que para descifrar, es decir, $k=k'$. Si hay n usuarios en el sistema se necesitarían $\binom{n}{2}$ claves en total y cada usuario tiene que poseer $n-1$ claves. Un hecho relevante es que se necesita un método seguro para concretar la clave por primera ver entre los usuarios. Para garantizar la fortaleza del sistema, es necesario claves muy grades aunque las funciones de cifrado sean operaciones algebraicas sencillas.

En los criptosistemas asimétricos cada usuario tiene dos claves, una pública y otra privada. La clave pública es conocida por todos y se usa para cifrar los mensajes. La privada solo la conoce el usuario y es necesaria para descifrar los mensajes. La seguridad del método reside en la dificultad de encontrar la clave privada a partir de la clave pública. Obsérvese que en un sistema con n usuarios cada usuario tiene una pareja de claves secretas, luego en total son necesarias n claves. Dado que no hay que concertar claves, no hace falta un canal seguro. Sin embargo, a diferencia de los métodos de clave pública, estos criptosistemas se basan en operaciones algebraicas complejas, como por ejemplo, las suma de puntos de curvas elípticas.

2.1 Clave pública

El concepto de los criptosistemas de clave pública se se establece en el artículo "New Directions in Cryptography" de Diffie y Hellman en 1976 [7]. Hoy en día son imprescindibles gracias a la popularización de internet. La criptografía asimétrica no surge para sustituir los métodos de clave privada, si no para resolver algunos de sus problemas que los hacen más vulnerables a ataques criptográficos. Los dos principales son el intercambio seguro de las claves y la autentificación del mensaje y del emisor. Además, los métodos de clave pública deben cumplir las condiciones de Diffie y Hellman:

- Cálculo sencillo de las claves pública y privada.
- Cálculo de la clave privada a partir de la clave pública computacionalmente imposible.
- Funciones de encriptado y desencriptado computacionalmente sencillas, en el caso del desencriptado si se conoce la clave privada.
- Conocer el mensaje original a partir de la cave pública y el mensaje cifrado debe ser computacionalmente imposible.

Los sistemas criptográficos modernos de clave pública usan la aritmética modular (es decir, trabajan en $\mathbb{Z}/n\mathbb{Z}$) o un cuerpo finito $\mathbb{F}_p$ para las operaciones algebraicas. Estos métodos criptográficos basan su seguridad en el problema del logaritmo discreto y en el problema de factorización.

Problema del logaritmo discreto: Sea $p \in \mathbb{Z}$ primo y g una raíz primitiva módulo p, es decir, $g \in \mathbb{F}_p$ tal que $\mathbb{F}_p^* = \mathbb{F}_p - \{0\} = \{1, g, g^2, g^3, ..., g^{p-2}\} = \langle g \rangle$. El problema del logaritmo discreto consiste en dado $a \in \{1, 2, ..., p-1\}$ encontrar un exponente $k \in \{0, 1, ..., p-2\}$ tal que $a \equiv g^k \pmod{p}$. Se dice que k es el logaritmo discreto de a bajo g módulo p. No existe ningún algoritmo conocido, con un tiempo de ejecución razonable, para calcular k, aunque en general no está probado que no exista.

Problema de factorización: Dado un número entero n calcular todos sus factores primos. Como en el caso anterior, no existe ningún algoritmo eficiente que lo resuelva. El problema tiene varios algoritmos eficaces en tiempos razonables si el entero n o sus primos cumplen ciertas condiciones. Veremos alguno de ellos, el método $p-1$ de Pollard que utiliza el grupo $((\mathbb{Z}/n\mathbb{Z})^*, \cdot)$ y el método de Lenstra que opera en el grupo $(E(\mathbb{F}_p), \oplus)$.

Exponemos a continuación diferentes utilidades criptográficas. Se destacan estos métodos ya que son la base de los criptosistemas sobre curvas elípticas que se presentan en el capítulo 3.

2.2 Utilidades basadas en el logaritmo discreto

2.2.1 Intercambio de claves Diffie-Hellman

En un sistema criptográfico es necesario que los usuarios concierten las claves de cifrado y descifrado. Para ello, Diffie y Hellman proponen un protocolo para que los usuarios Alice y Bob intercambien información con el objetivo de concertar las claves a través de un canal inseguro [7].

El protocolo consiste en lo siguiente: Alice y Bob necesitan concertar una clave secreta a través de un canal inseguro para usar un criptosistema simétrico. Además, suponemos que la clave va a ser un elemento del cuerpo $\mathbb{F}_p$, p primo.

Protocolo:

1. Alice y Bob conciertan un primo p y una raíz primitiva g módulo p.
2. Alice elige aleatoriamente un elemento $a \in \{0, 1, ..., p-2\}$ y envía a Bob $A \equiv g^a \pmod{p}$.
3. Bob escoge $b \in \{0, 1, ..., p-2\}$ aleatoriamente y envía $B \equiv g^b \pmod{p}$.
4. Ambos usuarios calculan la clave $k \equiv g^{ab} \pmod{p}$, Alice de la forma $B^a \equiv (g^b)^a \pmod{p}$ y Bob como $A^b \equiv (g^a)^b \pmod{p}$.

Como el canal es inseguro, A y B se pueden llegar a conocer por todos. Además, p y g son públicos por la misma razón. La clave k será la clave privada para ambos usuarios, solo es conocida por Alice y Bob. La privacidad de la clave se basa en el problema del logaritmo discreto.

En el intercambio de claves de Diffie-Hellman, por el problema del logaritmo discreto, no se puede conocer a ni b a partir de A, B, p, g. Entonces, no se puede conocer k a partir de A, B, p, g. Para que el cambio de claves sea útil es necesario escoger p lo suficientemente grande, de no ser así es sencillo averiguar el valor de a o de b iterando con los elementos de $\mathbb{F}_p$. Existe otro ataque al protocolo, y es el hecho de un atacante intercepte los valores A y B de Alice y Bob y les envíe otras claves. Este problema se soluciona con los sistemas de firma digital que se ven más adelante.

2.2.2 Criptosistema El Gamal

Uno de los criptosistemas más conocidos basado en el problema del logaritmo discreto es el sistema ElGamal [8]. Al igual que en el intercambio de claves de Diffie-Hellman se trabaja en el grupo $(\mathbb{F}_p^*, \cdot)$ donde p es primo.

Los usuarios del sistema generan sus claves de la siguiente forma.

Generación de claves:

1. Bob elige un primo p y una raíz primitiva g módulo p.
2. Bob escoge aleatoriamente $b \in \{0, 1, ..., p-2\}$ y calcula $B \equiv g^b \pmod{p}$. La clave púbica de Bob será (p, g, B) y la clave privada b.

Alice cifra un mensaje $m \in \{0, 1, ..., p-1\}$ para Bob.

Cifrado:

1. Elige aleatoriamente $a \in \{0, 1, ..., p-2\}$.
2. Calcula $A \equiv g^a \pmod{p}$ y $C \equiv B^a m \pmod{p} \equiv g^{ab} m \pmod{p}$.
3. Se cifra m como (A, C) y se lo envía a Bob.

Bob, una vez recibido el mensaje cifrado (A, C), calcula el mensaje original.

Descifrado:

1. Calcula $k \equiv A^b \equiv g^{ab} \pmod{p}$.
2. Recupera el mensaje calculando $m \equiv \frac{C}{k} \pmod{p} \equiv A^{p-1-b} C \pmod{p}$.

Para que el método sea seguro, p tiene que ser suficientemente grande, al menos de longitud binaria 768, para que no se pueda resolver el problema del logaritmo discreto. Además, para evitar ataques de tipo estadístico, a tiene que ser elegido nuevo y de forma aleatoria para cada mensaje diferente m.

Tanto el protocolo de Diffie-Hellman como el criptosistema ElGamal se puede extender a un grupo G. Se expone en el capítulo 3 el cambio de claves Diffie-Hellman y ElGamal con el grupo de puntos una curva elíptica $E(\mathbb{F}_q)$. En este caso, la seguridad se basará en el análogo al problema del logaritmo discreto, el problema del logaritmo elíptico, que tiene una seguridad similar. La ventaja que van a proporcionar las curvas elípticas es que las claves se pueden tomar de longitudes más pequeñas, lo que requiere almacenar menos memoria.

2.2.3 Firma digital DSA

A la hora de trasmitir un mensaje encriptado surge el problema de que dicho mensaje pueda ser modificado por una tercera persona y de que el usuario que envía el mensaje no sea quien dice ser. Estos problemas los resuelve la firma digital. Cuando Alice quiera enviar un mensaje a Bob trasmitirá el mensaje cifrado junto con su firma para que Bob pueda verificar que el mensaje original no ha sido modificado y que se lo ha enviado Alice.

Los esquemas de firma digital suelen ser lentos y la longitud de la firma suele ser similar a la del mensaje que se quiere trasmitir. Por ello, antes de firmar un mensaje, se utilizan

las funciones hash. Denominaremos función hash a una función computable $H : \mathcal{M} \to \mathcal{M}$, $H(m) = m'$ que transforma cada mensaje m de tamaño variable en un resumen $H(m)$ de tamaño fijo del propio mensaje. Normalmente el tamaño de $H(m)$ es menor que el del propio mensaje m. Diremos que la función hash es una función hash unidireccional o función resumen si para cualquier mensaje m' no es posible encontrar el mensaje m tal que $m' = H(m)$, de no ser así, una persona externa al sistema podría sustituir el mensaje. También es conveniente que dos mensajes distintos proporcionen distintos resúmenes. Habitualmente se consideran las funciones hash con llegada en $\mathbb{Z}/N\mathbb{Z}$ en lugar de en $\mathcal{M}$, es decir, $H : \mathcal{M} \to \mathbb{Z}/N\mathbb{Z}$, $H(m) = n$.

Así pues, el problema de la longitud de la firma se resuelve con la función hash. En lugar de firmar el mensaje m se firma su resumen $H(m)$.

Firma:

1. Alice para trasmitir un mensaje a Bob envía el mensaje m cifrado con la clave pública $B \in \mathcal{K}$ de Bob, $c = C(m, B)$, y la firma de $H(m)$, (r, s).
   (a) r se calcula con la clave privada $a \in \mathcal{K}$ de Alice de la forma $r = D(H(m), a)$.
   (b) s se calcula con la clave pública de Bob $s = C(r, B) = C(D(H(m), a), B)$.

Validación:

1. Bob cuando recibe c y (r, s) recupera primero el mensaje m con su clave privada $b \in \mathcal{K}$ como $m = D(c, b) = D(C(m, B), b)$.
2. Bob para validar la firma calcula $D(s, b) = D(C(r, B), b) = r$ y el resumen del mensaje como $H(m) = C(r, A) = C(D(H(m), a), A)$.
3. Bob comprueba que el resumen del mensaje m coincida con el valor de $H(m)$ calculado. Si la comprobación no es correcta Bob debe de rechazar el mensaje ya que puede haber sido manipulado el mensaje.

Es necesario que tanto el algoritmo para firmar digitalmente como la función hash utilizada tengan la misma complejidad debido a que el ataque a la firma digital se puede dar de diferentes formas. Una de ellas es atacar al método de la firma y otra atacar a la función hash utilizada en la firma. La función SHA-1 (Secure Hash Algorithm) [2] es una de las funciones hash más utilizadas en criptografía, el único ataque que se conoce contra ella es la prueba exhaustiva. Uno de los algoritmos estándar para la firma digital es el DSA (Digital Signature Algorithm).

El NIST (National Institute of Standards and Technology) en 1991 propuso el método DSS (Digital Signature Standard) y el algoritmo DSA (Digital Signature Algorithm) [3] como estándar de firma digital. El DSA se basa en el criptosistema ElGamal. Además, existe una versión de este algoritmo para curvas elípticas, ECDSA (Eliptic Curve Digital Signature Algorithm), que se expone en el capítulo 3.

El protocolo DSA consiste en lo siguiente. Primero Alice elige las claves y el grupo donde se opera.

Claves:

1. Se elige un primo q de 160 bits y un primo p de 500 bits con $p \equiv 1 \pmod{q}$.
2. Se escoge un generador g del subgrupo cíclico de orden q del grupo $(\mathbb{F}_p^*, \cdot)$.
3. Alice elige aleatoriamente $0 < a < q$ y calcula $A \equiv g^a \pmod{p}$.
4. La clave privada de Alice es a y la pública A.

Alice firma un mensaje m por (r, s) con la función hash H de SHA-1.

Firma:

1. Escoge aleatoriamente un entero k con $0 < k < q$.
2. Calcula el hash $H(m)$ de m.
3. Calcula $r \equiv (g^k \pmod{p}) \pmod{q}$
4. Calcula s resolviendo la congruencia $H(m) + ar \equiv ks \pmod{q}$.

Bob para verificar la firma de Alice sigue el siguiente procedimiento.

Validación:

1. Descifra el mensaje para tener m y calcula su hash $H(m)$.
2. Calcula $u_1 \equiv s^{-1}H(m) \pmod{q}$ y $u_2 \equiv s^{-1}r \pmod{q}$.
3. Calcula $v \equiv (g^{u_1}A^{u_2} \pmod{p}) \pmod{q}$
4. Valida la firma si $v = r$.

Observemos que la validación funciona:

$$v \equiv (g^{u_1}A^{u_2} \pmod{p}) \pmod{q} \equiv (g^{(H(m)+ar)s^{-1}} \pmod{p}) \pmod{q}$$

Como $H(m) + ar \equiv ks$,

$$(g^{(H(m)+ar)s^{-1}} \pmod{p}) \pmod{q} \equiv (g^k \pmod{p}) \pmod{q} \equiv r$$

La seguridad del método DSS se basa en el problema del logaritmo discreto en el subgrupo cíclico de $(\mathbb{F}_p^*, \cdot)$ de orden q con generador g. Para este problema se requiere calcular logaritmos en $(\mathbb{F}_p^*, \cdot)$, luego se puede decir que su seguridad está basada en el logaritmo discreto en $(\mathbb{F}_p^*, \cdot)$ al igual que el criptosistema ElGamal.

2.3 Utilidades criptográficas basadas en la factorización

2.3.1 Criptosistema RSA

El protocolo criptográfico desarrollado por Rivest, Shamir y Adleman en 1979 [23] se conoce por RSA y es el primer criptosistema construido de clave pública. Este sistema sirve tanto para cifrar como para firmar, es decir, autentificar al usuario que trasmite el mensaje. Este sistema basa su seguridad en la dificultad de factorizar un número entero. Existen muchos métodos de factorización. Destacaremos el método $p-1$ de Pollard ya que es la base del método de factorización Lenstra con curvas elípticas que se expone en la sección 3.3.1. Pero tampoco es eficiente para enteros n con todos sus factores primos grandes.

En el criptosistema RSA, cada usuario construye sus claves de la siguiente forma.

Claves:

1. Se eligen aleatoriamente dos números primos, p y q y calcula $n = pq$.

   El grupo donde se realizan las operaciones algebraicas del método es $((\mathbb{Z}/n\mathbb{Z})^*, \cdot)$, y por tanto, su orden es $\phi(n) = \phi(pq) = (p-1)(q-1)$ donde ϕ denota la función de Euler ($\phi(n) = \#\{k \in \mathbb{Z} | 1 \leq k < n, mcd(k,n) = 1)$.

2. El usuario elige $e \in \mathbb{Z}_+$, $1 \leq e < \phi(n)$ tal que $mcd(e, \phi(n)) = 1$.

3. Se calcula d, el inverso de e módulo $\phi(n)$ mediante el algoritmo de Euclides, $ed \equiv 1 \pmod{\phi(n)}$ y $1 \leq d < \phi(n)$.

4. La clave pública del usuario es la pareja (n, e) y la clave privada es d. Los números $p, q, \phi(n)$ tienen que permanecer en secreto.

Una vez construidas las claves, el cifrado de un mensaje m dirigido a Bob cuyas claves son (n_b, e_b) y d_b se hace de la siguiente manera.

Cifrado:

Alice para enviar el mensaje $m \in \mathbb{Z}/n_b\mathbb{Z}$ a Bob lo encripta con la clave pública de Bob, es decir, le trasmite $c = C(m, e_b) = m^{e_b} \pmod{n_b}$.

Descifrado:

Bob recibe c y para recuperar el mensaje original m utiliza su clave privada, $m = D(c, d_b) = c^{d_b} \pmod{n_b}$.

Ciertamente Bob recibe el mensaje original m de Alice. Puesto que $c \equiv m^{e_b} \pmod{n_b}$ y que d_b es el inverso de e_b módulo $\phi(n_b)$,

$$c^{d_b} \pmod{n_b} \equiv m^{e_b d_b} \pmod{n_b} \equiv m^{1+k\phi(n_b)} \pmod{n_b} \equiv m(m^{\phi(n_b)})^k \pmod{n_b}.$$

Si $mcd(m,n) = 1$, por el teorema de Euler-Fermat, $m^{\phi(n_b)} \equiv 1 \pmod{n_b}$. Lo que implica que

$$D(c, d_b) = c^{d_b} \equiv m(m^{\phi(n_b)})^k \pmod{n_b} \equiv m \pmod{n_b}.$$

Si $mcd(m, n_b) \neq 1$ también se cumple que $D(c, d_b) = m^{e_b d_b} \equiv m \pmod{n_b}$. Veámoslo. Por el teorema chino de los restos (Teorema 7),

$$\begin{aligned} m^{e_b d_b} \equiv m \pmod{n_b} &\Leftrightarrow m^{e_b d_b - 1} \equiv 1 \pmod{n_b} \\ &\Leftrightarrow m^{e_b d_b - 1} \equiv 1 \pmod{p} \text{ y } m^{e_b d_b - 1} \equiv 1 \pmod{q} \end{aligned}$$

Por tanto, es equivalente a que $e_b d_b - 1$ sea múltiplo de $p - 1$ y $q - 1$, es decir, múltiplo de $mcm(p - 1, q - 1)$. Puesto que $e_b d_b \equiv 1 \pmod{\phi(n)}$, $e_b d_b - 1 \equiv 0 \pmod{\phi(n)}$. Luego $e_b d_b - 1$ es múltiplo de $\phi(n) = (p - 1)(q - 1)$, en particular de $mcm(p - 1, q - 1)$. Como consecuencia $m^{e_b d_b} \equiv m \pmod{n_b}$.

La seguridad del método reside en la dificultad de conocer la clave privada d a partir de la clave pública (n, e). Si se conoce $\phi(n)$ es sencillo calcular d a partir de e mediante el algoritmo de Euclides. Conociendo n no es computacionalmente sencillo determinar el valor de $\phi(n)$, a no ser que se conozcan los factores primos de n. El problema de calcular $\phi(n)$ es computacionalmente equivalente al problema de factorizar n. Aunque no se ha probado aún que la única forma de hallar d sea factorizando n. Por ejemplo, si existiera un algoritmo eficiente para calcular las raíces e-ésimas módulo n no sería necesario hallar d para conocer el mensaje m. Si $c = m^e \pmod{n}$, una raíz e-ésima de c módulo n sería m.

Para que sea seguro el método, el usuario tiene que elegir adecuadamente los primos p y q para que n no se pueda descomponer mediante los métodos de factorización conocidos como por ejemplo, el método $p - 1$ de Pollard. Es necesario que verifique las siguientes condiciones para no ser atacado por los algoritmo de factorización: p y q deben diferir en pocos dígitos, $p - 1$ y $q - 1$ deben descomponerse en factores primos grandes (1024 bits) y $mcd(p - 1, q - 1)$ debe ser pequeño. Usualmente se consideran p y q primos de la forma $p = 2r + 1$ y $q = 2s + 1$ con r, s primos grandes de longitudes parecidas. Estos primos cumplen las condiciones para no ser vulnerables a los métodos de factorización.

Un problema que se debe resolver para que este criptosistema sea operativo es encontrar primos grandes p y q de forma eficiente. Este problema tiene dos partes:

1. **Problema de primalidad**. Disponer de un algoritmo eficiente A de manera que, dado un entero t, nos permita saber si es o no primo.

   Durante mucho tiempo se conjeturaba que se trataba de un problema de complejidad polinómica. A día de hoy ya está resuelto. En 2002 Agrawal, Kayal y Saxena [1] probaron con un algoritmo determinista que la complejidad era polinómica $O((\log n)^{12})$ y que bajo ciertas condiciones podía ser de $O((\log n)^6)$. En criptografía se utilizan algoritmos probabilísticos ya que son mucho más rápidos. Estos algoritmos si determinan que el número t no es primo entonces con seguridad t no es primo pero si determina que es primo existe una probabilidad determinada por el algoritmo de que t no sea primo. Existen varios algoritmos probabilísticos que determinan si un número entero t es primo o no de forma eficiente, como por ejemplo, el test de Rabin-Miller [22].

2. **Densidad de los primos**. El método para encontrar primos consiste en repetir el algoritmo A con números aleatorios de un cierto tamaño. Este método funciona si hay muchos primos de ese tamaño, luego necesitamos conocer la densidad del conjunto de números primos en el conjunto de los números naturales.

Nótese que si los usuarios eligen la clave pública e pequeña, los mensajes se cifran de forma rápida, es decir, permite que el cifrado sea más rápido. De esta forma, la clave secreta d será más grande, y por tanto, el descifrado será más difícil. Sin embargo, fijar e pequeño tiene muchos riesgos. Uno de ellos es que si el mensaje m también es pequeño y $m^e < n$, es decir, el cifrado es $c = m^e$, se calcula m fácilmente con un logaritmo. Otro de ellos es que al ser fijo e, es más sencillo calcular el valor de e si se tienen varios mensajes cifrados. Además, veremos en 3.3.2.1 que si e es pequeño el RSA es vulnerable al ataque de exponente bajo de Hastad. Para evitar ataques, uno de los exponentes recomendados es $e = 2^{16} + 1 = 65537$, que es primo y el algoritmo de exponenciación modular es especialmente sencillo para él.

2.3.2 Método p-1 de Pollard

Hemos visto que en el RSA se tienen que elegir correctamente las claves para que no sea vulnerable a los ataques con métodos de factorizacion. Destacamos el método de factorización $p-1$ de Pollard, ya que funciona muy bien si $p-1$ es producto de primos pequeños. Además, existe un método análogo que en lugar de trabajar con el grupo $((\mathbb{Z}/n\mathbb{Z})^*, \cdot)$, lo hace en $(E(\mathbb{F}_p), \oplus)$. Se llama método de factorización de Lenstra.

El método $p-1$ de Pollard consiste encontrar un entero $1 < a < n$ tal que $1 < mcd(a, n) < n$. En este caso, el entero $d = mcd(a, n)$ es un divisor de n. Se fundamenta en el pequeño teorema de Fermat.

Teorema 6. *(Pequeño teorema de Fermat:) Si $p \in \mathbb{Z}+$ primo y $a \in \mathbb{Z}$ con $p \nmid a$ entonces $a^{p-1} \equiv 1 \pmod{p}$. De forma equivalente $a^p \equiv a \pmod{p}$ para todo $a \in \mathbb{Z}$.*

Fijado un entero $C > 0$ denotaremos por $B = \{p_1, p_2, ..., p_r\}$ al conjunto de primos menores o iguales que C. Diremos que un entero n es C-suave si todos sus factores primos son menores o iguales que C. El método $p-1$ de Pollard es eficaz para los enteros n tales que $p-1$ es C-suave siendo p un factor primo de n. Luego para cada $i \in \{1, ..., r\}$ sea k_i el mayor entero tal que $p_i^{k_i} \leq n$. Nótese que $k_i := \lfloor \frac{\log n}{\log p_i} \rfloor$. Sea $M := \prod_i p_i^{k_i}$ un múltiplo de $p-1$.

El método se basa en lo siguiente: si $a \in \mathbb{Z}$ con $1 < a < n$ y $mcd(a, n) = 1$, por el pequeño teorema de Fermat, $a^{p-1} \equiv 1 \pmod{p}$. Como M es múltiplo de $p-1$, se tiene que $(p-1) \mid M$. Por tanto, $a^M \equiv 1 \pmod{p}$, es decir, $p \mid (a^M - 1)$. Entonces $p \mid mcd(a^M - 1, n)$. Luego se tiene que $mcd(a^M - 1, n)$ es un divisor de n.

Algoritmo p-1 de Pollard:

1. Se escoge aleatoriamente $a \in \{2, ..., n-1\}$ y se calcula $d = mcd(a, n)$.
   - Si $d > 1$, entonces d es factor de n y se termina.
2. Se calcula $d = mcd(a^M - 1, n)$ estratégicamente.
   Para $i = 1, ..r$ hacemos
   (a) Calculamos $k = \lfloor \frac{\log n}{\log p_i} \rfloor$.
   (b) Calculamos $a = a^{p_i^k} \pmod{n}$.
   (c) Calculamos $d = mcd(a - 1, n)$.

- Si $a = 1$ el algoritmo da fallo.
- Si $d \neq 1$ entonces d es factor de n y se termina.

3. Si $d = 1$ el algoritmo no da solución.

Si $p - 1$ es C-suave y existe un factor primo p de n, el algoritmo solo falla si $a = 1$ (mód n) en alguna etapa, luego la probabilidad de encontrar un factor de n es mayor del 50 %. De ser así, el tiempo de ejecución del algoritmo $p-1$ de Pollard es de $O(C \log_C((n))$ multiplicaciones modulares. La cota C no se escoge muy grande para limitar la complejidad del algoritmo y el conjuntos de primos B se calculan previamente. Veremos en 3.3.3 que el método de factorización de Lenstra mejora este tiempo y la probabilidad de encontrar un divisor de n.

Capítulo 3

Métodos criptográficos con curvas elípticas

Las curvas elípticas han sido objeto de estudio en el campo de la teoría de números y la geometría algebraica desde 1908 pero hasta casi 80 años después no se usaron para la criptología. Los primeros métodos criptográficos con curvas elípticas los propusieron de forma independiente Victor Miller [19] y Neal Koblitz [13] en 1985.

Los métodos criptográficos con curvas elípticas surge como una posibilidad de implementar criptosistemas de clave pública basados en el logaritmo discreto con la misma seguridad pero con la ventaja de utilizar claves más pequeñas. Esto requiere utilizar menos memoria del ordenador y elementos hardware. Posteriormente también se han desarrollado sistemas criptográficos basados en la factorización con curvas elípticas.

La criptografía con curvas elípticas tiene más seguridad y eficiencia que los métodos criptográficos sobre cuerpos finitos $\mathbb{F}_q$. A pesar de esto, la criptografía con curvas elípticas también se enfrenta a los ordenadores cuánticos que son capaces de reducir el tiempo de computación exponencial a un tiempo polinómico. A día de hoy no se ha podido construir un ordenador cuántico con capacidad de cálculo suficiente para resolver el problema del logaritmo elíptico. Además, se ha comenzado a investigar el uso de curvas hiperelípticas, una generalización de las curvas elípticas, para aplicarlas a sistemas criptográficos de clave pública. Sin embargo, estos sistemas son menos seguros o son menos eficientes que los sistemas que usan curvas elípticas.
.

3.1 Asignación mensaje-punto de una curva elíptica

En el caso de criptosistemas sobre el grupo $(\mathbb{F}_p^*, \cdot)$, cada mensaje m en un cierto alfabeto se corresponde con un entero n con $0 \leq n < p$, es decir, un elemento de $\mathbb{F}_p$ para poder cifrarlo. En los métodos criptográficos con curvas elípticas se necesita una relación entre el mensaje m y un punto P perteneciente a una curva elíptica. Además, este proceso debe ser rápido y sistemático para que se puedan trasmitir el mensaje a una velocidad razonable.

No existe a día de hoy un algoritmo determinístico en tiempo polinómico en el tamaño del cuerpo $\mathbb{F}_p$ que realice esta asignación para un número grande de puntos de una curva elíptica E sobre $\mathbb{F}_p$. De modo que se utiliza un método probabilístico, es decir, que existe una probabilidad de que a un mensaje m no se le pueda asignar ningún punto de la curva

elíptica.

Datos iniciales:

Suponemos que el conjunto de mensajes es el conjunto de enteros m con $0 \leq m < M$ para cierto entero positivo M. Sea E una curva elíptica sobre $\mathbb{F}_p$, con p primo grande en proporción a M, definida por la ecuación de Weierstrass $Y^2 = X^3 + aX + b$. El objetivo es asignar a cada mensaje m un punto $P = (x, y) \in E(\mathbb{F}_p)$.

Para ello, tomamos $k := \lfloor \frac{p}{M} \rfloor \in \mathbb{N}$, de manera que $p > Mk$, y partimos del siguiente hecho:

Para todo entero z, $0 \leq z \leq Mk$ existen l, j enteros tal que $z = lk + j$ y $0 \leq j < k$.

Asignación mensaje→ punto:

Dado un mensaje m con $0 \leq m < M$ se le asocia un punto $P_m = (x, y)$ de la siguiente forma:

1. Se calcula $x = mk$
2. Para $j = 0, ..., k - 1$,
   - Se calcula $z = f(x) \pmod p$.
   - Si z es un residuo cuadrático en $\mathbb{F}_p$, es decir, existe $y \in \mathbb{F}_p$ tal que $y^2 \equiv f(x) \pmod p$, se asocia $P_m = (x, y)$ a m y termina.
   - Si z no es un residuo cuadrático en $\mathbb{F}_p$, se considera $x = x + 1$ y se vuelve a calcular z.
3. Si $j = k - 1$ y a m no se le ha asociado ningún punto P, el algoritmo da error y termina.

Nótese que en cada etapa de la parte (2), $f(x)$ es un cuadrado módulo p con una probabilidad estimada de $\frac{1}{2}$, ya que la mitad de los elementos de $\mathbb{F}_p$ son cuadrados. Por lo tanto, la probabilidad de que el algoritmo de un error como salida y no haya asignado al mensaje m ningún punto es menor o igual que 2^{-k}. En este caso, cambiaremos el primo p y/o la curva elíptica seleccionada.

Recuperación del mensaje:

Dado un punto $P = (x, y) \in E(\mathbb{F}_p)$ el entero $m = \lfloor \frac{x}{k} \rfloor$ permite recuperar el mensaje inicial. En efecto, si $m < M$ y $P_m = (x, y)$, sabemos que x es de la forma $x \equiv mk + j \pmod p$ con $0 \leq j < k$. Por lo tanto $mk \leq x \leq mk + (k - 1)$ y $m \leq \frac{x}{k} \leq m + \frac{k-1}{k} < m + 1$. Es decir, $\lfloor \frac{x}{k} \rfloor = m$.

Nótese que si $P = (x, y)$ es un punto arbitrario de $E(\mathbb{F}_p)$, el entero $m = \lfloor \frac{x}{k} \rfloor$ puede ser mayor o igual que M, y por tanto, no corresponder a ningún mensaje.

Observación:

En el esquema anterior hemos supuesto inicialmente fijos P y E. Si queremos asegurar una probabilidad de error acotada se puede fijar k a priori de manera que $2^{-k} < \epsilon$, elegir un primo $p > Mk$ y una curva elíptica sobre $\mathbb{F}_p$.

Dada una curva elíptica E sobre $\mathbb{F}_q$ con $q = p^n$, se tiene que $E(\mathbb{F}_p) \subset E(\mathbb{F}_q)$. Por tanto, la asignación definida en $E(\mathbb{F}_p)$ es también una asignación en $E(\mathbb{F}_q)$. Otra posibilidad es adaptar el método de asignación descrito al cuerpo F_q.

Ejemplo 2. Sea $k = 20$, $M = 35$ y $p = 727$ primo. Consideramos la curva elíptica $Y^2 = X^3 + X + 54$ sobre $\mathbb{F}_p$.

Consideramos el mensaje $m_1 = 23$. Luego $x_1 = m_1 k = 460$ y $z_1 = f(x_1) \equiv 665 \pmod{p}$. Como $238^2 \equiv 665 \pmod{p}$, asociamos al mensaje m_1 el punto $P = (460, 238)$. Si en cambio tenemos el punto $P = (460, 238)$, el mensaje asociado debe ser $m = \lfloor \frac{x_1}{k} \rfloor = \lfloor \frac{460}{20} \rfloor = 23$.

Sea ahora $Q = (x_2, y_2) = (42, 311)$ un punto de la curva elíptica. Entonces su mensaje asociado es $m_2 = \lfloor \frac{x}{k} \rfloor = \lfloor \frac{42}{20} \rfloor = 2$. Veamos que si tenemos el mensaje $m_2 = 2$, Q es el punto de la curva asociado a m_2. Sea $x_2 = m_2 k = 40$ y $z_2 = f(x_2) \equiv 118 \pmod{p}$. Como 118 no es residuo cuadrático módulo p, consideramos ahora $x_2 = m_2 k + 1 = 41$. Entonces $z_2 = f(x_2) \equiv 678 \pmod{p}$ que tampoco es residuo cuadrático. Sea $x_2 = 42$ y $z_2 = f(x_2) \equiv 30 \pmod{p}$. Puesto que $311^2 \equiv 30 \pmod{p}$, el punto asociado a $m_2 = 2$ es efectivamente $Q = (42, 311)$.

3.2 Utilidades basadas en el logaritmo elíptico

En esta sección veremos una adaptación con curvas elípticas del intercambio de claves de Diffie-Helman, del criptosistema El Gamal y de la firma digitas DSA. Para ello, se fija un cuerpo $\mathbb{F}_q$ con $q = p^n$ y p primo, una curva elíptica E sobre ese cuerpo y un punto $G \in E(\mathbb{F}_q)$. Generalmente, la curva E y el punto G son preseleccionados y publicados por un organismo de normalización del protocolo a utilizar.

Antes de ello veremos como se define el problema del logaritmo elíptico, que ventajas tiene comparado con el logaritmo discreto y un algoritmo que resuelve ambos problemas en un tiempo exponencial.

3.2.1 Problema del logaritmo elíptico

Problema del logaritmo elíptico (PLE): Sea E una curva elíptica sobre un cuerpo $\mathbb{F}_q$ con $q = p^n$ y p primo y $P \in E(\mathbb{F}_q)$ de orden N, el problema del logaritmo elíptico es el siguiente: dado $Q \in E(\mathbb{F}_q)$ punto del subgrupo generado por P ($Q \in \langle P \rangle$), encontrar $k \in \mathbb{N}$ tal que $Q = kP$.

Es un problema análogo al logaritmo discreto donde se intercambia el grupo multiplicativo $((\mathbb{F}_p)^*, \cdot)$ por el grupo de puntos de una curva elíptica sobre un cuerpo finito $E(\mathbb{F}_q)$

y la operación multiplicativa por la operación aditiva $\oplus$. Este problema se puede generalizar a cualquier grupo finito G: sea $g \in G$, dado $a \in \langle g \rangle \subset G$ encontrar $x \in \mathbb{N}$ tal que $g^x = a$.

Para resolver este problema en un grupo finito G existen diversos algoritmos aunque con tiempo de computación de orden exponencial $O(p^{1/2})$ siendo p el mayor factor primo del orden del grupo G. Destacamos el algoritmo de Pohlig-Hellman [21] ya que es capaz de reducir ese tiempo a $O(\log^2 n)$ si n, el orden del grupo, tiene todos sus factores primos pequeños. Además, para el problema del logaritmo discreto sobre $\mathbb{F}_p$ existe un algoritmo que lo resuelven en tiempo subexponencial $O(e^{\sqrt{\log p \log\log p}})$. Este método se denomina Index-Calculus [28] y está basado en expresar un elemento del grupo como producto de elementos de una cierta base. No entraremos en detalle de este método ya que tampoco resuelve en un tiempo razonable el problema del logaritmo elíptico.

Una de las ventajas más significativas de utilizar curvas elípticas en criptografía es que sobre un mismo cuerpo $\mathbb{F}_q$ se pueden considerar diferentes curvas elípticas como hemos visto en 1.3. Por tanto, cada usuario del método puede elegir una curva sobre $\mathbb{F}_q$ y cambiar periódicamente la curva para aumentar la seguridad.

Para los criptosistemas que basan su seguridad en el problema de logaritmo discreto, si se utiliza una curva elíptica sobre $\mathbb{F}_q$ con q siendo aproximadamente 2^{160}, se consigue la misma seguridad que utilizando el grupo multiplicativo $\mathbb{F}_p$ con p siendo aproximadamente 2^{1024}. Por esto conviene utilizar curvas elípticas en criptografía.

En estos métodos criptográficos es esencial para su seguridad que el orden del subgrupo generado por P sea grande. De no cumplirse esta condición, el problema del logaritmo elíptico se resuelve por fuerza bruta en tiempo polinómico. Además, el orden de dicho subgrupo debe tener factores primos grandes para ser resistente al algoritmo de Pohlig-Hellman. Sin embargo, dado una curva elíptica E sobre $\mathbb{F}_q$ y un punto $P \in E(\mathbb{F}_q)$ no es sencillo calcular el orden de P. Para resolver este problema y asegurar la seguridad de estos criptosistemas, el NIST (National Institute of Standards and Technology) [3] (Apéndice D) publicó una lista estandarizada de quince cuerpos, curvas elípticas y puntos que cumplen estándares de seguridad suficientes. Las curvas se clasifican en tres tipos: curvas sobre $\mathbb{F}_p$ con $p \neq 2, 3$, curvas sobre $\mathbb{F}_{2^m}$ y curvas de Koblitz (caso particular de curvas sobre $\mathbb{F}_{2^m}$).

3.2.1.1 Algoritmo de Pohlig-Hellman

El algoritmo consiste en reducir el problema del logaritmo discreto de un grupo G a grupos más pequeños cuyos órdenes son factores del orden de G. Para ello se basa en el teorema chino de los restos.

Teorema 7. *(Teorema chino de los restos) Sean $n_1, ...n_k$ enteros positivos primos entre sí y $a_1, ..., a_k \in \mathbb{Z}$. Dado el sistema de congruencias $x \equiv a_i \pmod{n_i}$ con $i = 1, ..., k$, existe un entero x que las resuelve de forma única módulo $N = \prod_{i=1}^{k} n_i$.*

El objetivo es hallar x tal que $g^x = a$, dado $g \in G$ y $a \in \langle g \rangle$. Sea n el orden del grupo G y $n = \prod_{i=1}^{k} n_i$ la descomposición en factores primos. Conociendo los valores $x_i \equiv x \pmod{p_i^{n_i}}$ para cada $i = 1, ..., r$, mediante el teorema chino de los restos, se puede calcular $x \pmod{n}$ con $0 \leq x < n$. Luego el propósito es conocer los enteros x_i.

Puesto que $x_i \equiv x \pmod{p_i^{n_i}}$ para cada $i = 1, ..., r$, cada entero x_i es puede descomponer de la forma

$$x_i = \sum_{j=0}^{n_i-1} b_j p_i^j$$

donde $0 \leq b_j \leq p_i - 1$. Entonces necesitamos conocer los valores de b_i para conocer x_i. Calculamos las raíces p_i-ésimas de la unidad,

$$\gamma_{i,k} = g^{k(n/p_i)}$$

para $k = 0, .., p_i - 1$ y se determina el coeficiente b_0 por la potencia n/p_i de a. Puesto que $a^n \equiv 1 \pmod{n}$ se obtiene una raíz y

$$a^{n/p_i} \equiv g^{x_i(n/p_i)} \equiv g^{b_0(n/p_i)} g^{(b_1+...+b_{j-1}p^{j-2})n} \equiv g^{b_0(n/p_i)} \equiv \gamma_{i,k_0} \pmod{n}$$

Luego $b_0 = j$ tal que j cumple que $a^{n/p_i} = \gamma_{i,k_j}$.
Para calcular b_1 consideramos la potencia n/p_i^2 de $\bar{a} = a/g^{b_0}$. Como $\bar{a} = a/g^{b_0} = g^{x_i - b_0}$ y entonces

$$\bar{a}^{n/p_i^2} \equiv g^{(x_i-b_0)(n/p_i^2)} \equiv g^{(b_1+...+b_{j-1}p^{j-2})n/p_i} \equiv g^{b_1 n/p_i} \equiv \gamma_{i,b_1} \pmod{n}$$

Al igual que para hallar b_0, se calcula b_1 comparando con las raíces p_i-ésimas de la unidad. $b_1 = j$ donde $\bar{a}^{n/p_i^2} = \gamma_{i,k_j}$.

Recursivamente se van calculando los valores $b_2, ..., b_{n_i-1}$. Para el valor b_j se considera la potencia n/p_i^{j+1} de $\bar{a} = a/g^{b_0+b_1 p_i+...+b_{j-1}p_i^{j-1}}$ y se tiene $\bar{a}^{n/p_i^{j+1}} = \gamma_{i,b_k}$.

Una vez obtenidos $b_0, ..., b_{n_i-1}$ para cada $i = 1, ..., r$ se calcula $x_i = \sum_{j=0}^{n_i-1} b_j p_i^j \pmod{p_i^{n_i}}$ y mediante el teorema chino de los restos obtenemos x.

Este algoritmo es eficiente si los factores p_i primos de n son pequeños. De no ser así, el cálculo de las raíces p_i-ésimas de la unidad tendía un costo computacional muy elevado. Además, no será sencillo conocer la factorización de n.

Veamos ahora como se pueden definir diferentes protocolos y criptosistemas con curvas elípticas que basan su seguridad en el problema del logaritmo elíptico.

3.2.2 Claves Diffie-Helman con curvas elípticas

En este protocolo, Alice y Bob concertarán una clave utilizando una curva elíptica $E(\mathbb{F}_q)$. El punto $G \in E(\mathbb{F}_q)$ jugará mismo papel que el generador g del grupo $(\mathbb{F}_p^*, \cdot)$ en el intercambio de claves de Diffie-Helman sobre dicho grupo finito. El punto G no tiene por que ser un generador de $E(\mathbb{F}_q)$, vale con que tenga un orden grande. Veamos la forma de determinar la clave.

Protocolo:

1. Se fija una curva elíptica E sobre $\mathbb{F}_q$ y un punto $G \in E(\mathbb{F}_q)$ con orden $|G| = N$. Tanto $\mathbb{F}_q$ como $E(\mathbb{F}_q)$ y G son públicos.
2. Alice elige aleatoriamente a con $1 < a < N$ y envía a Bob el punto $A = aG$.

3. Bob escoge b aleatoriamente con $1 < b < N$ y envía el punto $B = bG$ a Alice.

4. Ambos calculan la clave $P = abG$, Alice de la forma aB y Bob como bA.

Puesto que la curva elíptica E es de la forma $Y^2 = X^3 + aX + b$, no es necesario que los usuarios envíen las coordenadas (x, y) de los puntos A y B. Basta con enviar la coordenada x y un bit indicando que raíz y se tiene que escoger. A esto se le llama compresión de puntos. El intercambio de claves quedaría de esta forma:

1. Se fija una curva elíptica E sobre $\mathbb{F}_q$ y un punto $G \in E(\mathbb{F}_q)$ con orden $|G| = N$. Tanto $\mathbb{F}_q$ como $E(\mathbb{F}_q)$ y G son públicos.

2. Alice elige aleatoriamente a con $1 < a < N$ y calcula el punto $A = aG = (A_x, A_y)$.

3. Considera los representantes $0 \leq A_x, A_y < q$.

   - Si $q/2 < A_y < q$ envía a Bob $(A_x, 1)$
   - Si $0 \leq A_y < q/2$ envía a Bob $(A_x, 0)$

4. Bob escoge b aleatoriamente con $1 < b < N$ y calcula el punto $B = bG = (B_x, B_y)$.

5. Considera los representantes $0 \leq B_x, B_y < q$

   - Si $q/2 < B_y < q$ envía a Alice $(B_x, 1)$
   - Si $0 \leq A_y < q/2$ envía a Alice $(B_x, 0)$

6. Ambos calculan la clave $P = abG$:

   - Alice calcula los valores y_0, y_1 con $0 \leq y_0 < q/2$ e $q/2 < y_1 < q$ tal que $y^2 = f(B_x)$ donde $f(X) = X^3 + aX + b$. Toma como $B_y = y_1$ si recibe $(B_x, 1)$ de Bob y $B_y = y_0$ si recibe $(B_x, 0)$.
     Alice calcula $P = aB = a(B_x, B_y)$.
   - Bob calcula los valores y_0, y_1 con $0 \leq y_0 < q/2$ e $q/2 < y_1 < q$ tal que $y^2 = f(A_x)$ donde $f(X) = X^3 + aX + b$. Toma $A_y = y_1$ si recibe $(A_x, 1)$ de Bob y $A_y = y_0$ si recibe $(A_x, 0)$.
     Bob calcula $P = bA = b(A_x, A_y)$

Como el canal es inseguro, tanto A y B como $E(\mathbb{F}_q)$ y G pueden ser conocidos por todos. En cambio, la clave P y los enteros a y b deben ser secretos. La privacidad de la clave P reposa en la fortaleza del problema del logaritmo elíptico. No se pueden conocer a y b a partir de conocer A, B, G y $E(\mathbb{F}_q)$, luego no se puede calcular P sabiendo A, B, G y $E(\mathbb{F}_q)$. Observemos que el cálculo del logaritmo elíptico no tiene porque ser la única forma para calcular P. Si a partir de aG y bG se pudiese conocer $abG = P$ se rompe la seguridad del protocolo. A día de hoy tampoco se conoce ninguna forma de calcularlo.

Para la seguridad del protocolo es necesario que el orden del punto G sea relativamente grande, lo ideal es que N sea el orden del grupo $E(\mathbb{F}_q)$ o un divisor grande suyo. De no ser así, sería fácil calcular a a partir de A probando con todos valores enteros menores que N.

3.2.3 Criptosistema El Gamal con curvas elípticas

Al igual que en el intercambio de claves de Diffie-Helman con curvas elípticas, en el método criptográfico El Gamal con curvas elípticas se opera en el grupo $(E(\mathbb{F}_q), \oplus)$ y su seguridad depende del problema del logaritmo elíptico.

Los usuarios del sistema generan sus claves privada y pública de la siguiente forma.

Claves:

1. Se elige una curva elíptica E sobre $\mathbb{F}_q$ y un punto $G \in E(\mathbb{F}_q)$ con orden $|G| = N$. Tanto $\mathbb{F}_q$ como G son públicos.
2. Bob escoge b aleatoriamente con $1 < b < N$ y calcula $B = bG$. La clave pública de Bob será (E, G, B) y la privada b.

Alice envía un mensaje m a Bob siguiendo este proceso.

Cifrado:

1. Calcula el punto $P \in E(\mathbb{F}_q)$ asociado al mensaje m.
2. Elige aleatoriamente $1 \leq a < N$.
3. Calcula $A = aG$ y $C = P \oplus aB$.
4. Se cifra m como (A, C).

Bob recibe el mensaje cifrado (A, C) y calcula el mensaje m.

Descifrado:

1. Calcula $bA = baG$.
2. Recupera el punto P de la forma $P = C \oplus (\ominus bA) = P \oplus aB \oplus (\ominus abG) = P \oplus aB \oplus (\ominus aB)$.

Como se ha mencionado anteriormente, la curva E y el punto G se eligen normalmente de una lista que estandariza el criptosistema. Entonces, fijado previamente E y G Alice puede elegir su clave privada a y calcular el punto A antes de saber a quién le va a enviar el mensaje ya que el punto $G \in E(\mathbb{F}_q)$ no va a cambiar.

De la misma forma que en el intercambio de claves, se pueden enviar las coordenadas x de los puntos A y C más dos bits, en lugar de los propios puntos. Aún así, en este criptosistema para enviar un mensaje en claro $P \in E(\mathbb{F}_q)$ se necesitan dos puntos $A, C \in E(\mathbb{F}_q)$ para cifrarlo. Lo que duplica la cantidad de información a trasmitir. Veremos después que esto no ocurre en el RSA.

Para garantizar la seguridad, como en el intercambio de claves de Diffie-Helman, el orden N del punto G debe ser grande, lo deseable es que G genere $E(\mathbb{F}_q)$. Es esencial que N sea divisible por un primo grande porque de no ser así el algoritmo de Pohlig-Hellman puede romper el problema del logaritmo elíptico ya que este solo depende del primo más

grande que divide a N.

El criptosistema El Gamal, como muchos otros, no te asegura que el mensaje que envía Alice a Bob no haya sido modificado por una tercera persona o que directamente a Bob le llegue un mensaje que no sea de Alice creyendo él que sí. Para resolver esto se usa la firma digital, protocolo que se implementa también utilizando curvas elípticas.

3.2.4 Firma digital con curvas elípticas (ECDSA)

El protocolo de firma digital con curvas elípticas ECDSA (Eliptic Curve Digital Signature Algorithm) está basado en el protocolo DSA. Se cambia el grupo $(\mathbb{F}_p^*, \cdot)$ por $E(\mathbb{F}_p)$ y el papel que toma el generador g de un subgrupo de $(\mathbb{F}_p^*, \cdot)$, lo toma ahora un punto $G \in E(\mathbb{F}_p)$ con orden grande.

En primer lugar, Alice, antes de firmar un mensaje, elige las claves que se van a utilizar para ello.

Claves:

1. Se fija un primo p y una curva elíptica E sobre $\mathbb{F}_p$.
2. Se toma un punto $G \in E(\mathbb{F}_p)$ con orden q primo.
3. Alice escoge aleatoriamente $a \in [1, q-1]$ y calcula $A = aG$.
4. La clave privada de Alice es a y la pública A.

Una vez tomadas las claves, Alice firma el mensaje m por (r, s) con la función hash de SHA-1.

Firma:

1. Escoge aleatoriamente un entero k con $0 < k < q$.
2. Calcula el hash de m, $H(m)$.
3. Calcula $kG = (x_1, y_1)$ y considera $r = x_1 \pmod q$. Si $r = 0$ se escoge otro valor de k.
4. Calcula $s = k^{-1}(H(m) + ar) \pmod q$. Si $s = 0$ se escoge otro valor de k.

Bob recibe el mensaje m cifrado y la firma (r, s) asociada. Para verificar la firma realiza lo siguiente.

Validación:

1. Descifra el mensaje m y calcula su hash $H(m)$.
2. Verifica que $0 < r, s < q$
3. Calcula $u_1 \equiv s^{-1}H(m) \pmod q$ y $u_2 \equiv s^{-1}r \pmod q$.
4. Calcula $(x, y) = u_1G + u_2A$ y toma $v = x \pmod q$

5. Valida la firma si $v = r$

Observemos que la validación de la firma es coherente ya que

$$(x, y) = u_1G + u_2A = (s^{-1}H(m))G + (s^{-1}r)aG = s^{-1}(H(m) + ar)G = kG = (x_1, y_1)$$

Por tanto, $v = x = x_1 = r \pmod q$.

Nótese que, en el ECDSA, q denota el orden de G y tiene que tener aproximadamente el mismo tamaño que p para que sea un protocolo seguro. En cambio, en el DSA, el papel de q lo toma un número primo diferente mucho más pequeño de p. Pero la diferencia principal entre ECDSA y DSA es la generación de r. El DSA considera $r \equiv g^k \pmod p$. Por otro lado, el ECDSA genera $r \in [1, q-1]$ tomando la coordenada x de la $kG \pmod q$. Recordemos que en el DSA q debe ser un primo de longitud 160 bits para garantizar la seguridad, luego en el ECDSA, q también debe tener 160 bits para tener una seguridad similar a la del DSA. En este caso ambas firmas tienen longitud 320 bits.

Una ventaja que tiene este protocolo con curvas elípticas sobre el DSA es que, sobre el cuerpo $\mathbb{F}_p$ se pueden considerar varias curvas elípticas. En lugar de fijar una curva elíptica E sobre $\mathbb{F}_p$ y un punto $P \in E(\mathbb{F}_p)$ se puede fijar solamente el cuerpo $\mathbb{F}_p$ y que cada usuario elija una curva elíptica E diferente sobre $\mathbb{F}_p$, un punto G de la misma y añada a su clave pública E y G. Esto es una gran ventaja para optimizar los cálculos ya que se puede construir un entorno para operar con curvas elípticas sobre el cuerpo fijo $\mathbb{F}_p$.

La seguridad de este protocolo reside en la dificultad de calcular a a partir de $A = aP$, es decir, en el logaritmo elíptico. Además, para asegurar dicha seguridad, el NIST recomienda que para la firma se escoja una curva elíptica con orden aq donde q es primo y a un entero pequeño. De no ser así el protocolo puede ser vulnerable al ataque de Pohlig-Hellman. Para asegurar esto, dada una curva elíptica se debe calcular su cardinal y factorizarlo cosa que no es trivial. Es recomendable usar las curvas elípticas y los puntos estandarizados por el NIST (National Institute of Standards and Technology) para usar de forma segura la firma digital ECDSA.

3.3 Utilidades criptográficas basadas en la factorización

En este apartado mostraremos un versión del criptosistema RSA con curvas elípticas llamado KMOV y un método de factorización con curvas elípticas que ataca a dicho criptosistema entre otros. Este método de factorización se denomina método de factorización de Lenstra. Ambos protocolos utilizan análogos de las curvas elípticas pero ahora sobre un anillo $\mathbb{Z}/n\mathbb{Z}$ con n un entero no primo. Veamos como se definen y que operación se considera sobre los puntos de dichas curvas para realizar las operaciones oportunas.

3.3.1 Curvas pseudo-elípticas sobre $\mathbb{Z}/n\mathbb{Z}$

La definición de curva pseudo-elíptica sobre $\mathbb{Z}/n\mathbb{Z}$ es análoga a la de curva elíptica sobre un cuerpo finito $\mathbb{F}_p$. Por simplicidad eliminaremos el prefijo 'pseudo' y nos referiremos a

ellas simplemente como curvas elípticas sobre $\mathbb{Z}/n\mathbb{Z}$.

Sea n un entero, una curva curva elíptica E sobre $\mathbb{Z}/n\mathbb{Z}$ es un polinomio de la forma $Y^2 = X^3+aX+b$ con coeficientes en $\mathbb{Z}/n\mathbb{Z}$ y tal que $\Delta = -16(4a^3+27b^2) \neq 0$. Recordamos que para un cuerpo $\mathbb{K}$ con $Char(\mathbb{K}) \neq 2, 3$, los puntos de una curva elíptica sobre $\mathbb{K}$ se definen como $E(\mathbb{K}) = \{(x, y) \in \mathbb{K}^2 \mid y^2 = x^3 + ax + b\} \cup O$. Entonces los puntos de la curva elíptica E sobre el anillo $\mathbb{Z}/n\mathbb{Z}$ se definen como

$$E(\mathbb{Z}/n\mathbb{Z}) = \{(x, y) \in (\mathbb{Z}/n\mathbb{Z})^2 \mid y^2 = x^3 + ax + b\} \cup O$$

Evidentemente, si p es un divisor primo de n, la ecuación de la curva sobre $\mathbb{Z}/n\mathbb{Z}$ define una curva sobre $\mathbb{F}_p$ siempre que $\Delta \neq 0 \pmod p$. Por tanto, si imponemos que $mcd(4a^3 + 27b^2, n) = 1$ se obtiene una curva elíptica en $\mathbb{F}_p$ para todo divisor primo p de n. Dado un punto $P = (x, y) \in (\mathbb{Z}/n\mathbb{Z})^2$, entonces se tiene que $P_p \in (\mathbb{F}_p)^2$ es el punto P reducido módulo p. Además, se define $O_p = O$, es decir, $P_p = O_p$ si y solo si $P = O$. Luego dado $P \in E(\mathbb{Z}/n\mathbb{Z})$ y un primo p con $p \mid n$, se tiene entonces que $P_p \in E(\mathbb{F}_p)$.

A este conjunto de puntos $E(\mathbb{Z}/n\mathbb{Z})$ se le asocia una operación aditiva $\oplus$ análoga a la definida en 1.2 para el cuerpo $\mathbb{F}_p$ remplazando las operaciones en $\mathbb{F}_p$ por las operaciones en $\mathbb{Z}/n\mathbb{Z}$.

Sean $P, Q \in E(\mathbb{Z}/n\mathbb{Z})$, $R = P \oplus Q$

- Si $P = O$ o $Q = O$, entonces $R = Q$, o $R = P$ respectivamente.
- Si $P = (x_1, y_1)$, $Q = (x_2, y_2)$ y $x_1 = x_2$, $y_1 = -y_2$, entonces $R = O$.
- En caso contrario, sea m el siguiente entero
  - Si $x_1 \neq x_2$, $m = (y_2 - y_1)(x_2 - x_1)^{-1}$.
  - Si $x_1 = x_2$, $m = (3x_1^2 + a)(2y_1)^{-1}$

  Se define $R = (x_3, y_3) = (m^2 - x_1 - x_2, m(x_1 - x_3) - y_1)$

Esto tiene algunos inconvenientes. El primero es que la suma $P \oplus Q$ de dos puntos $P, Q \in E(\mathbb{Z}/n\mathbb{Z})$ no está siempre definida. Para obtener $P \oplus Q$ es necesario calcular la pendiente m que es de la forma $m = (y_2-y_1)(x_2-x_1)^{-1}$ si $P \neq Q$ o $m = (3x_1^2+a)(2y_1)^{-1}$ si $P = Q$. Esto requiere dividir en el anillo $\mathbb{Z}/n\mathbb{Z}$, operación que solo está definida si $x_2 - x_1$ es unidad en el caso $P \neq Q$ o si $2y_1$ es unidad en el caso $P = Q$. Otro problema es que $E(\mathbb{Z}/n\mathbb{Z})$ no es un grupo. Aunque esto no imposibilita que se pueda construir un criptosistema en $E(\mathbb{Z}/n\mathbb{Z})$.

3.3.2 Criptosistema RSA con curvas elípticas

En 1991 Koyama, Maurer, Okamoto y Vastone exponen una versión al criptosistema RSA con curvas elípticas llamado KMOV [15]. En este esquema se usa el conjunto de puntos de una curva elíptica sobre $\mathbb{Z}/n\mathbb{Z}$ en lugar de $((\mathbb{Z}/n\mathbb{Z})^*, \cdot)$. El esquema KMOV es más flexible que el RSA, la curva elíptica no es fija, varía con cada mensaje nuevo. Por otro lado, este sistema es menos eficiente que el RSA aunque más seguro para ataques que no se basan en la factorización, como por ejemplo, el ataque de bajo exponente.

Fijamos n que es el producto de dos primos p y q, es decir, $n = pq$. La idea es relacionar la operación $\oplus$ en $E(\mathbb{Z}/n\mathbb{Z})$ con la operación de grupo $E(\mathbb{F}_p) \times E(\mathbb{F}_q)$ usando el teorema chino de los restos.

Debido al teorema chino de los restos, todo elemento $c \in \mathbb{Z}/n\mathbb{Z}$ se puede representar por el par $[c_p, c_q]$ donde $c_p \equiv c \pmod{p}$ y $c_q \equiv c \pmod{q}$. Luego cada punto $P = (x, y) \in E(\mathbb{Z}/n\mathbb{Z})$ está relacionado con el par $[P_p, P_q] = ((x_p, y_p), (x_q, y_q))$ con $P_p \in E(\mathbb{F}_p)$ y $P_q \in E(\mathbb{F}_q)$. Por convenio al punto del infinito $O \in E(\mathbb{Z}/n\mathbb{Z})$ se le asocia el par $[O_p, O_q]$ donde O_p es el punto del infinito de $E(\mathbb{F}_p)$ y O_q el de $E(\mathbb{F}_q)$.

$$\begin{array}{ccc} E(\mathbb{Z}/n\mathbb{Z}) & \longrightarrow & E(\mathbb{F}_p) \times E(\mathbb{F}_q) \\ P & \longmapsto & [P_p, P_q] \end{array}$$

Todos los elementos $[P_p, P_q]$ de $E(\mathbb{F}_p) \times E(\mathbb{F}_q)$ están definidos por un punto $P \in E(\mathbb{Z}/n\mathbb{Z})$ salvo si P_p o P_q (solo uno) es el punto del infinito. Luego la operación de suma, $\oplus$, en $E(\mathbb{Z}/n\mathbb{Z})$ no está definida si y solo si alguno de los puntos P_p o P_q es el punto del infinito en su respectiva curva $E(\mathbb{F}_p)$ o $E(\mathbb{F}_q)$. Pero esto es poco frecuente que ocurra: si los primos p y q son lo suficientemente grandes, los ordenes de los grupos $E(\mathbb{F}_p)$ y $E(\mathbb{F}_q)$ serían elevados y es poco probable que alguno de los puntos sea el punto del infinito. Además, de no ser p y q grandes sería muy fácil factorizar n con algún método de factorización, lo cual descarta esta posibilidad para un criptosistema. Por lo tanto, la operación aditiva en $E(\mathbb{Z}/n\mathbb{Z})$ está definida 'casi siempre'.

Sabemos que $(E(\mathbb{Z}/n\mathbb{Z}), \oplus)$ no es un grupo. Para resolver este problema se utiliza el siguiente lema.

Lema 3. *Sea $E : Y^2 = X^3 + aX + b$ una curva elíptica sobre $\mathbb{Z}/n\mathbb{Z}$ con $mcd(4a^3 + 27b^2, n) = 1$ y $n = pq$ con p y q primos. Sea $N = mcm(\#E(\mathbb{F}_p), \#E(\mathbb{F}_q))$. Entonces para todo $P \in E(\mathbb{Z}/n\mathbb{Z})$ y todo $k \in \mathbb{N}$*

$$(kN + 1)P = P$$

sobre $E(\mathbb{Z}/n\mathbb{Z})$.

Demostración. Por definición, existen n_p, n_q tal que $N = n_p \#E(\mathbb{F}_p) = n_q \#E(\mathbb{F}_q)$. Luego
$(kN + 1)P = (kn_p \#E(\mathbb{F}_p))P \oplus P = O_p \oplus P = P$ en $E(\mathbb{F}_p)$ y
$(kN + 1)P = (kn_p \#E(\mathbb{F}_q))P \oplus P = O_q \oplus P = P$ en $E(\mathbb{F}_q)$.
El lema se tiene como consecuencia del teorema chino de los restos. □

Por el lema, se podría decir que $N = mcm(\#E(\mathbb{F}_p, \#E(\mathbb{F}_q))$ juega el papel del 'orden' de $E(\mathbb{Z}/n\mathbb{Z})$.

Una vez que tenemos una operación definida en $E(\mathbb{Z}/n\mathbb{Z})$, exponemos el criptosistema. En este método criptográfico en lugar de considerar cualquier curva elíptica de la forma $Y^2 = X^3 + aX + b$ sobre $\mathbb{F}_p$, se consideran las curvas elípticas con $a = 0$ y $0 < b < p$, es decir, curvas elípticas con una ecuación del tipo $Y^2 = X^3 + b$. De esta forma se puede aplicar el Lema 1 expuesto en la sección 1.3 y es conocido que $\#E(\mathbb{F}_p) = p + 1$.

En el criptosistema un usuario construye sus claves tanto pública como privada de la siguiente manera:

Claves:

1. Se escogen dos primos grandes p y q tal que $p \equiv q \equiv 2 \pmod 3$ y se calcula $n = pq$.
2. Se calcula $N = mcm(\#E(\mathbb{F}_p), \#E(\mathbb{F}_q)) = mcm(p+1, q+1)$.
3. Se elige un entero e con $1 \leq e < N$ tal que $mcd(e, N) = 1$.
4. Se calcula el inverso d de e módulo N, mediante el algoritmo de Euclides, es decir, $ed \equiv 1 \pmod N$.
5. La clave pública es (n, e) y la clave privada es $(p, q, \#E(\mathbb{F}_p), \#E(\mathbb{F}_q), N, d)$.

Notemos que para el cálculo de N no es necesario fijar una curva elíptica $Y^2 = X^3 + b$ ya que el número de puntos de la curva sobre $\mathbb{F}_p$ o sobre $\mathbb{F}_q$, no va a cambiar con el valor que tome b, siempre que $mcd(b, n) = 1$ para $\Delta \neq 0$. Por el Lema 1, sabemos que $\#E(\mathbb{F}_p) = p+1$ para cualquier valor de b (análogo para $\mathbb{F}_q$). Además, puesto que $p \equiv q \equiv 2 \pmod 3$, entonces $2 \mid N$ y $3 \mid N$. Luego el mínimo valor que puede tomar e para que $mcd(e, N) = 1$ es $e = 5$.

Una vez construidas las claves de los usuarios, Alice envía un mensaje M a Bob. Ese mensaje tiene que ser un punto de una curva elíptica de la forma $Y^2 = X^3 + b$ sobre $\mathbb{Z}/n_b\mathbb{Z}$ donde (n_b, e_b) es la clave pública de Bob. Luego se va a construir la curva elíptica en función del punto M para que éste pertenezca a la curva.

Alice envía a Bob un mensaje $M = (m_x, m_y)$ con $m_x, m_y \in \mathbb{Z}/n_b\mathbb{Z}$ de la siguiente forma:

Cifrado:

1. Alice calcula $b = m_y^2 - m_x^3 \pmod{n_b}$ para construir la curva $E : Y^2 = X^3 + b$ sobre $\mathbb{Z}/n_b\mathbb{Z}$.
2. Alice cifra M por $C = (c_x, c_y) = e_bM$ sobre $E(\mathbb{Z}/n_b\mathbb{Z})$.

Descifrado:

1. Bob recibe C y calcula $b = c_y^2 - c_x^3 \pmod{n_b}$ para construir la curva $E : Y^2 = X^3 + b$ sobre $\mathbb{Z}/n_b\mathbb{Z}$.
2. Bob descifra C usando su clave privada d_b, $M = d_bC$ sobre $E(\mathbb{Z}/n_b\mathbb{Z})$.

La comprobación de que Bob realmente recupera M se basa en el Lema 3 y en que $ed \equiv 1 \pmod{N_b}$.

$$d_bC = d_be_bM = (kN + 1)M = M$$

Notemos que el cálculo que hace Alice de $C = e_bM$ y el cálculo de Bob de $M = d_bC$ sobre $E(\mathbb{Z}/n_b\mathbb{Z})$ se realizan de forma diferente. Bob como conoce la clave privada $(p, q, \#E(\mathbb{F}_p), \#E(\mathbb{F}_q), N_b, d_b)$ puede calcular d_bC mediante la relación entre $E(\mathbb{Z}/n_b\mathbb{Z})$ y $E(\mathbb{F}_p) \times E(\mathbb{F}_q)$. Primero calcula $M_p = d_bC \pmod p$ y $M_q = d_bC \pmod q$. Después obtiene $M = d_bC$ sobre $E(\mathbb{Z}/n_b\mathbb{Z})$ mediante el teorema chino de los restos. Por otro lado, Alice no puede calcular $C = e_bM$ de esta forma ya que no conoce los valores p y q. Si los conociera, el método no sería seguro ya que obtendría la clave privada d fácilmente. Por tanto, Alice

calcula C mediante la operación de suma sobre $E(\mathbb{Z}/n_b\mathbb{Z})$ descrita en 3.3.1. Hemos visto que esta operación no funciona si $2y_1$ no es unidad en $\mathbb{Z}/n_b\mathbb{Z}$ siendo $M = (x_1, y_1)$. Pero es muy poco probable que surja este problema. En el caso en el que $2y_1$ no sea unidad, se tiene que $D = mcd(2y_1, n) \neq 1$. Luego $D = mcd(2y_1, n)$ sería divisor de n y se encontrarían p y q de forma sencilla. Dado que en el criptosistema se impone que tanto p como q tiene que ser relativamente grandes, es muy difícil que encontrar $D = mcd(2y_1, n) \neq 1$. Entonces, Alice va a poder calcular $C = e_bM$ sin problema.

Una observación que cabe señalar entre el RSA y el KMOV es que el papel que toma $\phi(n)$ en el RSA, lo toma N en el KMOV. Así pues, $\phi(n)$ es el orden del grupo $\mathbb{Z}/n_b\mathbb{Z}$, en cambio, N no denota el orden de $E(\mathbb{Z}/n_b\mathbb{Z})$ porque este no es un grupo, pero tiene la misma función. Además, el KMOV tiene una propiedad significativa. Esta es que el criptosistema está definido para cualquier curva elíptica sobre $\mathbb{Z}/n\mathbb{Z}$ con N elementos y no para un grupo específico. Una desventaja ante el RSA es que las operaciones algebraicas requieren más tiempo computacional, lo que resulta menos eficiente.

Para este criptosistema existe una versión en la que en lugar de considerar las curva elípticas $Y^2 = X^3 + b$ se consideran las curvas $Y^2 = X^3 + aX$. Para ello se escogen primos $p \equiv q \equiv 3 \pmod 4$ y se aplica el Lema 2 para calcular N. En esta versión, el valor mínimo que puede tomar la clave e es $e = 3$ ya que $2 \mid N$. La única diferencia a la hora de cifrar un mensaje $M = (m_x, m_y)$ o descifrar $C = (c_x, c_y)$ es el cálculo de la curva elíptica. Para calcular $Y^2 = X^3 + aX$ se toma $a = \frac{m_y^2 - m_x^3}{m_x} \pmod n$ en el caso de cifrar y $a = \frac{c_y^2 - c_x^3}{c_x} \pmod n$ en el caso de descifrar.

La seguridad de este método se basa en la dificultad de factorizar n. Si no se conocen los factores p y q de n no es posible calcular el valor de N, y por tanto, no se puede hallar la clave privada d a partir de la clave pública (n, e). El problema de encontrar N a partir de n es computacionalmente equivalente al problema de factorización. De ser conocidos p y q, se tiene $N = mcm(p + 1, q + 1)$ y es sencillo calcular d mediante el algoritmo de Euclides. De esta forma cualquier persona, no solo el dueño de la clave privada d, puede descifrar un mensaje M con solo realizar la operación dM sobre $E(\mathbb{Z}/n_b\mathbb{Z})$. Al igual que en el RSA, no está probado que la única forma de romper el método sea factorizando n. Pero para evitar posibles ataques al criptosistema, los factores primos de n se tienen que escoger de forma adecuada como se ha descrito en el RSA.

Para los ataques basados en métodos de factorización, el esquema KMOV es igual de vulnerable que el RSA. En cambio, para ataques diferentes es más seguro que el RSA. A continuación veremos el ataque de bajo exponente de Hastad que solo es efectivo para el RSA. También exponemos el método de factorización de Lenstra que es capaz de romper, para factores primos pequeños, los criptosistemas que basan su seguridad en la factorización, en particular, el RSA y el KMOV.

3.3.2.1 Ataque de bajo exponente

En 1985, Johan Hastad [11] publica un ataque a criptosistemas de clave pública donde se conoce un mensaje cifrado con módulos diferentes. El problema que consideró fue el siguiente: sean $n_1, ..., n_k$ enteros primos entre sí, $P_1, .., P_k$ polinomios del mismo grado e. Supongamos que sabemos que existe un entero $m < min\{n_1, ..., n_k\}$ tal que $P_i(m) \equiv 0$

(mód n_i), $i = 1, ...k$. Hastad logró demostrar que es posible calcular m resolviendo las congruencias en tiempo polinomial si $k > \frac{e(e+1)}{2}$.

Veamos que esto implica que no es seguro cifrar el mismo mensaje m con el mismo exponente de cifrado e para k usuarios, si $k > \frac{e(e+1)}{2}$. Supongamos conocidos $c_1, ..., c_k$ tales que $c_i < n_i$, $c_i \equiv m^e \pmod{n_i}$ para $i = 1, ...k$. En este caso m es solución módulo n_i del polinomio $P_i(x) = x^e - c_i$ ya que $P_i(m) \equiv 0 \pmod{n_i}$. Por lo tanto, si $k > \frac{e(e+1)}{2}$ el resultado de Hastad permite recuperar el mensaje original m siempre que $m^e < n = \prod_{i=1}^{k} n_i$.

Luego para que el RSA sea seguro usando el mismo exponente de cifrado e, este no debería ser muy pequeño ya que si se conocen los suficientes mensajes cifrados de un mismo mensaje en claro sería muy fácil conocer el mensaje original. Una forma segura de evitar este ataque es modificar el mensaje m con información diferente para cada usuario, de esta forma se evita el cifrado del mismo mensaje m.

Este ataque no se puede adaptar a la criptografía con curvas elípticas. En el KMOV, sea M el mensaje cifrado como $C_1, .., C_k$ sobre una misma curva E sobre los anillos $\mathbb{Z}/n_1\mathbb{Z}, ..., \mathbb{Z}/n_k\mathbb{Z}$ respectivamente. Luego $C_i = eM$ sobre $E(\mathbb{Z}/n_i\mathbb{Z})$ para $i = 1, ...k$ y $C = eM$ sobre $E(\mathbb{Z}/n\mathbb{Z})$ con $n = \prod_{i=1}^{k} n_i$. Obtener $C = (C_x, C_y)$ a partir de $C_i = (C_{i_x}, C_{i_y})$ con $i = 1, ...k$ es sencillo. Basta aplicar el teorema chino de los restos a las ecuaciones C_{i_x} (mód n_i) para calcular C_x (mód n) y a las ecuaciones C_{i_y} (mód n_i) para calcular C_y (mód n). En cambio, no es fácil calcular $M = (M_x, M_y)$ a partir de $C = (C_x, C_y)$. Puesto que $C = eM$ sobre $E(\mathbb{Z}/n\mathbb{Z})$, los valores C_x y C_y se expresan mediante ecuaciones racionales en función de los valores M_x y M_y, es decir, $C_x \equiv P_1(M_x, M_y) \pmod{n}$ y $C_y \equiv P_2(M_x, M_y) \pmod{n}$ con P_1 y P_2 funciones que incluyen sumas, restas, multiplicaciones y divisiones módulo n. Si se transforma la relación de forma racional a forma polinomial, el orden del tamaño del coeficiente aumentaría hasta el orden de n. Luego no es factible resolver las ecuaciones despreciando el módulo de n. Por tanto, si e es pequeño un ataque como el de Hastad no rompería el KMOV.

3.3.3 Método de factorización de Lenstra

En el capítulo anterior hemos visto que el método $p - 1$ de Pollard es un buen algoritmo para factorizar números enteros con un factor primo p tal que los factores primos de $p - 1$ sean pequeños. Hendrik Lenstra propone en 1987 otro nuevo método de factorización con curvas elípticas basado en el ya conocido $p - 1$ de Pollard [16].

En este método se intercambia el grupo multiplicativo $\mathbb{F}_p$ por el grupo aditivo $E(\mathbb{F}_p)$ y la condición $a^k \equiv 1 \pmod{p}$ por $kP = O$. Otra diferencia significativa es que, el orden del grupo $((\mathbb{F}_p)^*, \cdot)$ es $p - 1$. Sin embargo, por el teorema de Hasse, el orden de $(E(\mathbb{F}_p)), \oplus)$ es $p + 1 - t$ donde t depende de la curva E y $|t| \leq 2\sqrt{p}$. Dado un número primo p, existe un único grupo $((\mathbb{F}_p)^*, \cdot)$ asociado, en cambio, para p existen varias curvas elípticas donde $(E(\mathbb{F}_p), \oplus)$ es un grupo abeliano (ver apartado 1.3). Esto es una gran ventaja para el método de Lenstra ya que si el algoritmo falla, se puede repetir el intento con una curva diferente que dará un nuevo valor de $\#E(\mathbb{F}_p) = p + 1 - t$. Notemos que en el algoritmo $p - 1$ de Pollard no se puede hacer algo análogo si el algortimo falla.

A día de hoy es uno de los métodos más rápidos de factorización junto con la cri-

ba cuadrática de múltiples polinomios y el método de grupos de clases. Además, más adelante se muestra que el tiempo de factorización depende solo del factor primo p más pequeño del número n a factorizar. Esto no es así en otros métodos de factorizacion como el $p-1$ de Pollard que tienen un tiempo dependiente del tamaño del entero a factorizar.

El método consiste en lo siguiente: dado un entero n (número que se quiere factorizar) elegimos una curva elíptica E sobre $\mathbb{Z}/n\mathbb{Z}$, un punto P perteneciente a dicha curva y un entero k mayor que 1. Se va calculando kP de forma eficiente hasta encontrar un divisor de n. Describamos en primer lugar la base del método.

El conjunto de puntos $E(\mathbb{Z}/n\mathbb{Z})$ necesita una operación aditiva para calcular kP. Como hemos visto antes en 3.3.1, la operación aditiva $P \oplus Q$ con $P = (x_1, y_1)$ y $Q = (x_2, y_2)$ está definida si $x_2 - x_1$ es unidad en $\mathbb{Z}/n\mathbb{Z}$ en el caso $P \neq Q$ o si $2y_1$ es unidad en el caso $P = Q$.

Supongamos que $P = (x_1, y_1) \in E(\mathbb{Z}/n\mathbb{Z})$ y $Q = (x_2, y_2) \in E(\mathbb{Z}/n\mathbb{Z})$ tales que $P \oplus Q$ no está definido en $\mathbb{Z}/n\mathbb{Z}$. Entonces se tiene que

$$\begin{aligned} d = mcd(x_2 - x_1, n) \neq 1 \quad &\text{si} \quad P \neq Q \\ d = mcd(2y_1, n) \neq 1 \quad &\text{si} \quad P = Q \end{aligned}$$

Puesto que $P = (x_1, y_1), Q = (x_2, y_2) \in E(\mathbb{Z}/n\mathbb{Z})$ con $0 \leq x_1, x_2, y_1, y_2 < n$, se tiene que d es un divisor propio de n y hemos resuelto el problema.

En resumen, dado un entero n, una curva elíptica E sobre $\mathbb{Z}/n\mathbb{Z}$ y dos puntos $P, Q \in E(\mathbb{Z}/n\mathbb{Z})$ entonces, o bien se genera un punto $R = P \oplus Q$, o bien, se encuentra un divisor d de n. Así pues, la clave del método es:

- Si $P = 0$ o $Q = 0$, entonces $R = Q$, o $R = P$ respectivamente.
- Si $P = (x_1, y_1)$, $Q = (x_2, y_2)$ y $x_1 = x_2$, $y_1 = -y_2$, entonces $R = O$.
- Si $P = (x_1, y_1)$, $Q = (x_2, y_2)$ con $x_1 \neq x_2$ y $d = mcd(x_2 - x_1, n)$
  - Si $d \neq 1$, d es un divisor no trivial de n
  - Si $d = 1$, $x_2 - x_1$ tiene inverso multiplicativo.
    $m = (y_2 - y_1)(x_2 - x_1)^{-1}$, $x_3 = m^2 - x_1 - x_2$, $y_3 = m(x_3 - x_1) + y_1$
    $\Rightarrow R = (x_3, -y_3)$
- Si $P = (x_1, y_1)$, $Q = (x_2, y_2)$ con $x_1 = x_2$ $y_1 \neq -y_2$ y $d = mcd(y_1 + y_2, n)$
  - Si $d = n$, $y_1 = -y_2$ y $R = 0$
  - Si $d \neq 1$, d es un divisor no trivial de n
  - Si $d = 1$, $x_2 - x_1$ tiene inverso multiplicativo.
    $m = (3x_1^2 + a)(y_1 + y_2)^{-1}$, $x_3 = m^2 - x_1 - x_2$, $y_3 = m(x_3 - x_1) + y_1$
    $\Rightarrow R = (x_3, -y_3)$

A la vista de la observación, el método para encontrar un factor del entero n consiste en lo siguiente.

Algoritmo de Lenstra (1)

1. Se elige una curva $Y^2 = X^3 + aX + b$ sobre $\mathbb{Z}/n\mathbb{Z}$ y un punto $P \in E(\mathbb{Z}/n\mathbb{Z})$

2. Para $j = 2, 3, 4, .., B$ se calcula iteradamente $Q = jP$ donde $Q = P$ en el primer paso
   (a) Si falla la operación aditiva, se encuentra d divisor de n y se para.
   (b) Si calcula Q hasta $j = B$ se vuelve al paso 1 y se elije una nueva curva elíptica y un nuevo punto en ella.

A la hora de elegir una curva elíptica y un punto que pertenece ella se escogen tres enteros pertenecientes a $\mathbb{Z}/n\mathbb{Z}$ y se construye la curva elíptica y el punto en ella a la vez. Dado $n \in \mathbb{Z}$, $n > 1$, para construir una curva elíptica sobre $\mathbb{Z}/n\mathbb{Z}$ se consideran $x, y, a \in \mathbb{Z}/n\mathbb{Z}$. Se calcula $b = y^2 - x^3 - ax \pmod{n}$ y se comprueba que $\Delta = -16(4a^3 + 27b^2) \neq 0$. Entonces $E: Y^2 = X^3 + aX + b$ es una curva elíptica sobre $\mathbb{Z}/n\mathbb{Z}$ y $P = (x, y) \in E(\mathbb{Z}/n\mathbb{Z})$.

Para encontrar una factorización de n con este método es necesario que la curva que elijamos tenga un número de puntos sobre $\mathbb{Z}/n\mathbb{Z}$ elevado y que el orden del punto P escogido inicialmente tenga un orden alto, es decir que $kP \neq O$ para k pequeño. De no ser así, es muy fácil que el algoritmo falle ya que se hacen pocas sumas en $E(\mathbb{Z}/n\mathbb{Z})$. Además, la cota de parada B del algoritmo debe ser lo suficientemente grande para realizar las operaciones aditivas necesarias para encontrar d pero no demasiado grande ya que disminuiría la eficiencia del método. Lo mismo ocurre con el número de curvas elípticas con las que probar. Hemos visto en 1.3.1 que sobre un mismo cuerpo se pueden considerar diferentes curvas elípticas, luego el algoritmo se va a poder repetir para diferentes curvas sobre $E(\mathbb{Z}/n\mathbb{Z})$ si falla para la primera curva elíptica escogida. Como el número puntos de un curva elíptica sobre un cuerpo fijo $\mathbb{F}_p$ varía en el intervalo $[p + 1 - 2\sqrt{p}, p + 1 + 2\sqrt{p}]$, no es difícil encontrar una curva elíptica E sobre $\mathbb{F}_p$ tal que $E(\mathbb{F}_p)$ sea producto de primos pequeños. De ser así, dado $P \in E(\mathbb{F}_p)$, $kP = O$ en $E(\mathbb{F}_p)$ para k pequeño con $\#E(\mathbb{F}_p) \mid k$. El hecho de que $kP = O$ nos permite generalmente encontrar mediante el algoritmo de Lenstra un divisor p no trivial de n.

3.3.3.1 Algoritmo de Lenstra

El método concreto consiste en materializar la idea anterior con un cálculo inteligente y adecuado de las suma iteradas kP junto con el establecimiento de la cota para k.

Datos iniciales: un entero $n > 1$ (número a factorizar) y las cotas $v, w \in \mathbb{Z}$ $v, w > 1$.

Algoritmo:

1. Calculamos $d = mcd(n, 6)$. Si $d \neq 1$ terminamos con d divisor de n.
2. Elegimos aleatoriamente $x, y, a \in \mathbb{Z}/n\mathbb{Z}$ y calculamos $b := y^2 - x^3 - ax \pmod{n}$.
   - Fijamos la curva elíptica $E: Y^2 = X^3 + aX + b$ en $\mathbb{Z}/n\mathbb{Z}$.
   - Fijamos $P := (x, y) \in E(\mathbb{Z}/n\mathbb{Z})$.
3. Para $r = 2, ..., w$ hacemos
   - $e(r) := \max\{m \mid r^m \leq v + 2\sqrt{v} + 1\}$.
   - Para $m = 1, .., e(r)$ hacemos
   $$P := rP \tag{3.1}$$
4. Devolvemos P

Análisis del algoritmo:

- La operación $P = rP$ se ejecuta con el algoritmo específico de sumas iteradas. En particular, dicho algoritmo debe contener un control de salida. De manera que si no se puede calcular rP termine el procedimiento dando como salida un divisor propio de n. Así pues, se encuentra con éxito un factor de n si el algoritmo anterior no termina.

- Si el algoritmo realiza toda las sumas devuelve kP siendo $k = \prod_{r=1}^{w} r^{e(r)}$.
  Al completar el paso 3 para $r = 2$ hemos obtenido $2^{e(r)}P$. En efecto,
  Para $m = 1$ obtenemos $P \oplus P = 2P$.
  Para $m = 2$ obtenemos $2(2P) = 2^2P$.
  ...
  Para $m = e(r)$ obtenemos $2(2^{e(r)-1})P = 2^{e(r)}P$.
  De la misma forma, en la etapa r partimos de $Q = \prod_{s=1}^{r-1} s^{e(s)}P$.
  Para $m = 1$, calculamos calculamos rQ.
  Para $m = 2$, $r(rQ) = r^2Q$
  ...
  Para $m = e(r)$ obtenemos $r(r^{e(r)-1})Q = r^{e(r)}Q = \prod_{s=1}^{r} s^{e(s)}P$.
  Por tanto, si no ha habido paradas el algoritmo devuelve al final $kP = \prod_{r=1}^{w} r^{e(r)}P$.
  Nótese que k actúa como una cota del número de iteraciones. Es conveniente que esta cota k sea lo suficientemente grande para realizar las sumas necesarias hasta encontrar un divisor pero no es práctico que sea demasiado grande ya que aumentaría considerablemente el tiempo de ejecución del algoritmo.

- En el caso en el que el algoritmo termine, es decir, no hayamos conseguido un divisor propio de n, podemos ejecutarlo de nuevo tantas veces como queramos. De esta forma se realizaran las sumas en diferentes curvas elípticas.

Proposición 2. *Dado $n, v, w \in \mathbb{Z}$ con $n, v, w > 1$ y $a, x, y \in \mathbb{Z}/n\mathbb{Z}$. Sea $b = y^2 - x^3 - ax \in \mathbb{Z}/n\mathbb{Z}$ y $P = (x, y) \in (\mathbb{Z}/n\mathbb{Z})^2$. Suponemos que n tiene dos divisores primos p y q que cumplen:*

1. $p \leq v$
2. $6(4\bar{a}^3 + 27\bar{b}^2) \neq 0$ *donde* $\bar{a} = a \mod p$ *y* $\bar{b} = b \mod p$
3. $6(4\hat{a}^3 + 27\hat{b}^2) \neq 0$ *donde* $\hat{a} = a \mod q$ *y* $\hat{b} = b \mod q$
4. *Todo primo r que divide a $\#E_{\bar{a},\bar{b}}(\mathbb{F}_p)$ es menor o igual que w*
5. $\#E_{\hat{a},\hat{b}}(\mathbb{F}_q)$ *no es divisible por el mayor primo que divide al orden del punto P_p*

Entonces el algoritmo descrito anteriormente logra encontrar un factor no trivial de n.

Demostración. Por (2) y (3) tanto $E_{\bar{a},\bar{b}}(\mathbb{F}_p)$ como $E_{\hat{a},\hat{b}}(\mathbb{F}_q)$ tienen una estructura de grupo.

Por (5), $\#E_{\hat{a},\hat{b}}(\mathbb{F}_q)$ no es divisible por el mayor primo que divide al orden del punto P_p, luego sabemos que dicho primo existe. Entonces el orden del punto P_p, g, tiene que ser mayor o igual que 2. Evidentemente g divide a $\#E_{\bar{a},\bar{b}}(\mathbb{F}_p)$, en particular, el mayor primo de g, l, divide a $\#E_{\bar{a},\bar{b}}(\mathbb{F}_p)$. Por (4), tenemos que l es menor o igual que w. Por tanto, el

orden de P_p no es infinito, lo que lleva a que $P_p \neq O_p$.

Por (5), $p \neq q$. Si $p = q$ entonces $E_{\hat{a},\hat{b}}(\mathbb{F}_q) = E_{\bar{a},\bar{b}}(\mathbb{F}_p)$ y el orden de P_p sería el mismo que el orden de P_q. Luego $\#E_{\hat{a},\hat{b}}(\mathbb{F}_q)$ es divisible por el mayor primo del orden de P_p. Esto va en contra de la condición (5).

Por (1), $p \leq v$, y por el teorema de Hasse , $\#E_{\hat{a},\hat{b}}(\mathbb{F}_p) \leq p+1+2\sqrt{p}$. Luego se tiene que $\#E_{\hat{a},\hat{b}}(\mathbb{F}_p) \leq v+1+2\sqrt{v}$. Esto implica que para cada primo r, su exponente en $\#E_{\hat{a},\hat{b}}(\mathbb{F}_p)$ es como máximo $e(r)$. Lo mismo ocurre para el exponente de r en el orden g de P_p ya que hemos visto que $g > 1$.

Sea l el mayor primo que divide al orden g de P_p y sea m el exponente de l en g $(1 \leq m \leq e(l))$. Sea

$$k_0 := (\prod_{r=2}^{l-1} r^{e(r)}) l^{m-1} \tag{3.2}$$

Por la hipótesis (4), $l \leq w$. Entonces k_0 y $k_0 l$ son divisores de $k = \prod_{r=2}^{w} r^{e(r)}$.

Puesto que l es el mayor primo que divide a g, $g = \prod_{r=2}^{l} r^s$ con $s \leq e(r)$ para cada $r \in \{2, .., l\}$ y $s = m$ para $r = l$. Entonces $k_0 \not\equiv 0 \bmod g$, y por tanto, en $E_{\bar{a},\bar{b}}(\mathbb{F}_p)$ se tiene que

$$k_0 P_p \neq O_p \tag{3.3}$$

Además, $k_0 l \equiv 0 \bmod g$. Luego en $E_{\bar{a},\bar{b}}(\mathbb{F}_p)$

$$k_0 l P_p = O_p \tag{3.4}$$

Si el algoritmo calcula con éxito hasta kP, previamente calcula $k_0 P$ y $k_0 l P$. Para probar que el algoritmo encuentra un divisor no trivial basta con demostrar que $k_0 P$ y $k_0 l P$ no pueden definirse a la vez.

Notemos que por definición $O_p = O$ para todo primo p divisor de n. Entonces, $P = O$ si y solo si $P_p = O_p$. Además, para $P, Q \in E(\mathbb{Z}/n\mathbb{Z})$ y $p > 3$ primo con $p \mid n$ se tiene que $(P \oplus Q)_p = (P)_p \oplus (Q)_p$ si está definida $P \oplus Q$. Veámoslo.

Si $Q = O$ resulta que $(P \oplus O)_p = P_p = P_p \oplus O_p$,.

Si $Q = \ominus P$ entonces $\ominus P_p = (\ominus P)_p$ y $(P \oplus (\ominus P))_p = O_p = P_p \oplus (\ominus P_p) = P_p \oplus (\ominus P)_p$.

Si $P \neq Q$ con $P = (x_1, y_1)$ y $Q = (x_2, y_2)$ entonces

$$\begin{aligned} (P \oplus Q)_p &= (m^2 - x_1 - x_2, m(x_3 - x_1) + y_1)_p \\ &= ((m)_p^2 - (x_1)_p - (x_2)_p, (m)_p((x_3)_p - (x_1)_p) + (y_1)_p) \\ &= (P)_p \oplus (Q)_p \end{aligned}$$

Como consecuencia, $kP_p = (kP)_p$ con $P \in E(\mathbb{Z}/n\mathbb{Z})$, $k \in \mathbb{Z}$ y $p > 3$ primo con $p \mid n$.

Si $k_0 l P \in (\mathbb{Z}/n\mathbb{Z})^2$ existe, por (3.4) se tiene la cadena de implicaciones

$$\begin{aligned} k_0 l P_p = (k_0 l P)_p = O_p \text{ en } E_{\bar{a},\bar{b}}(\mathbb{F}_p) &\Rightarrow k_0 l P = O \\ &\Rightarrow (k_0 l P)_q = k_0 l P_q = O_q \text{ en } E_{\hat{a},\hat{b}}(\mathbb{F}_q) \end{aligned}$$

Como l divide al orden de P_p, por (5), se tiene que $k_0 P_q = O_q$

Si $k_0 P \in (\mathbb{Z}/n\mathbb{Z})^2$ está también definido, se tiene también la cadena de implicaciones

$$\begin{aligned} k_0 P_q = (k_0 P)_p = O_q \text{ en } E_{\hat{a},\hat{b}}(\mathbb{F}_q) &\Rightarrow \quad k_0 P = O \\ &\Rightarrow \quad (k_0 P)_q = k_0 P_p = O_p \text{ en } E_{\bar{a},\bar{b}}(\mathbb{F}_p) \end{aligned}$$

Esto contradice (3.3).

□

3.3.3.2 Efiencia

Corolario 8. *Sean $n, v \in \mathbb{Z}$, $n, v > 1$ donde n tiene al menos dos primos distintos mayores que 3 y el menor primo p de n cumple que $p \leq v$. Sea $w \in \mathbb{Z}$, $w > 1$,*

$$u = \#\{s \in \mathbb{Z} \mid |s - (p+1)| < \sqrt{p} \text{ y } q \leq w \ \forall q \text{ primo} \mid s\}$$

que satisface $u \geq 3$ y $f(w) = \frac{u}{2\sqrt{p}+1}$ la probabilidad de que un entero aleatorio del intervalo $(p+1-\sqrt{p}, p+1+\sqrt{p})$ tenga todos sus factores primos menores o iguales que w. Entonces existe una constante c tal que para todo $h \in \mathbb{Z}$, $h > 1$ la probabilidad de éxito del algoritmo del apartado 3.3.3.1 con los valores n, v, w, h, donde h denota el número de curvas elípticas, es al menos $1 - c^{\frac{-hf(w)}{\log v}}$

Demostración. Consultar [16] □

Por tanto, para tener una probabilidad razonable de éxito, el número de curva elípticas con las que se prueba el algoritmo de Lenstra, h, tiene que ser del mismo orden que $\frac{\log v}{f(w)}$.

Sea $M(n)$ el tiempo computacional, es decir, el máximo número de operaciones bit necesarias, del algoritmo de la suma en $\mathbb{Z}/n\mathbb{Z}$. Se tiene que $M(n) = O((\log n)^2)$ si se usa el algoritmo de Euclides clásico para calcular el inverso multiplicativo necesario para el algoritmo de suma. Entonces el tiempo del algoritmo de Lenstra es de $O(hw(\log v)M(n))$ donde $k = \prod_{r=2}^{w} r^{e(r)}$ satisface $\log k = O(w \log v)$. Luego el orden de la complejidad del algoritmo es $O(\frac{w}{f(w)}(\log v)^2 M(n))$ si se quiere tener una probabilidad de éxito elevada.

Para minimizar dicho tiempo se debe elegir w tal que $\frac{w}{f(w)}$ sea mínimo. Lenstra demuestra en [16] a partir del teorema de Canfield, Erdös y Pomerace [5](corolario al teorema 3.1) que el valor óptimo para w es de orden de $L(p)^{\frac{-1}{\sqrt{2}+o(1)}}$ donde $L(x) = e^{\sqrt{\log x \log\log x}}$. Para ese w, $\frac{w}{f(w)} = L(p)^{\sqrt{2}+o(1)}$ para $p \to \infty$.

A la hora de implementarlo, no se conoce el valor de p, el factor primo más pequeño de n. Por lo que se intercambia el valor de p por el de v y se trabaja con el valor $w = L(v)^{\frac{-1}{\sqrt{2}+o(1)}}$ donde v va tomando diferentes valores de forma creciente para el algoritmo. Además, los factores $\log v$ del tiempo computacional del algoritmo tienen un tiempo estimado de $L(v)^{o(1)}$. Recapitulando, el tiempo del algoritmo es $O(\frac{w}{f(w)}(\log v)^2 M(n))$, es decir,

$$O(cL(p)^{\sqrt{2}+o(1)})$$

con una probabilidad de éxito de

$$1 - e^{-c}$$

donde c es un entero positivo.

La gran ventaja de este método de factorización es que el tiempo de ejecución del algoritmo depende solo del factor primo más pequeño de n. Aunque el algoritmo no garantiza que se encuentre el menor divisor de n. De ser así, se podría repetir el algoritmo hasta encontrar todos los factores de n. El peor caso es que del segundo mayor primo de n no sea mucho más pequeño que $\sqrt{n}$. En este caso, n tendría dos primos grandes del mismo orden de magnitud.

Bibliografía

[1] Agrawal, M., Kayal, N., and Saxena, N. Primes is in p. *Annals of Mathematics 160*, 2 (2004), 781–793.

[2] Barker, E. Secure hash standard (SHS). *NIST FIPS Pub 180–1* (1995).

[3] Barker, E. Digital signature standard (DSS). *NIST FIPS Pub 186–4* (2013).

[4] Blake, I., Seroussi, G., and Smart, N. *Elliptic Curves in Cryptography*. London Mathematical Society Lecture Note Series. Cambridge University Press, 1999.

[5] Canfield, E. R., Erdös, P., and Pomerance, C. On a problem of oppenheim concerning "factorisatio numerorum". *Journal of number theory 17*, 1 (1983), 1–28.

[6] Cassels, J. W. S. Diophantine Equations with Special Reference To Elliptic Curves. *Journal of the London Mathematical Society s1-41* (1966), 193–291.

[7] Diffie, W., and Hellman, M. E. New directions in cryptography. *IEEE transactions on Information Theory 22*, 6 (1976), 644–654.

[8] ElGamal, T. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory 31*, 4 (1985), 469–472.

[9] Fulton, W. *Curvas algebraicas*. Reverté, 2008.

[10] Fúster, A., de la Guía, D., Hernández, L., Montoya, F., and Muñoz, J. Técnicas criptográficas de protección de datos. *Alfaomega, Grupo Editor* (2001).

[11] Hastad, J. N using rsa with low exponent in a public key network. In *Advances in Cryptology — CRYPTO '85 Proceedings* (1986), H. C. Williams, Ed., Springer Berlin Heidelberg, pp. 403–408.

[12] Hoffstein, J., Pipher, J., Silverman, J. H., and Silverman, J. H. *An introduction to mathematical cryptography*, vol. 1. Springer, 2008.

[13] Koblitz, N. Elliptic curve cryptosystems. *Mathematics of computation 48*, 177 (1987), 203–209.

[14] Koblitz, N. *Algebraic Aspects of Cryptography*. Springer, 1998.

[15] Koyama, K., Maurer, U. M., Okamoto, T., and Vanstone, S. A. New public-key schemes based on elliptic curves over the ring zn. In *Advances in Cryptology — CRYPTO '91* (1992), J. Feigenbaum, Ed., Springer Berlin Heidelberg, pp. 252–266.

[16] Lenstra Jr, H. W. Factoring integers with elliptic curves. *Annals of mathematics* (1987), 649–673.

[17] Menezes, A. J. *Elliptic curve public key cryptosystems*, vol. 234. Springer Science & Business Media, 1993.

[18] Menezes, A. J., and Vanstone, S. A. Elliptic curve cryptosystems and their implementation. *Journal of cryptology 6*, 4 (1993), 209–224.

[19] Miller, V. S. Use of elliptic curves in cryptography. In *Conference on the theory and application of cryptographic techniques* (1985), Springer, pp. 417–426.

[20] Odlyzko, A. M. Discrete logarithms in finite fields and their cryptographic significance. In *Advances in Cryptology* (1984), Springer, pp. 224–314.

[21] Pohlig, S., and Hellman, M. An improved algorithm for computing logarithms over $gf(p)$ and its cryptographic significance (corresp.). *IEEE Transactions on Information Theory 24*, 1 (1978), 106–110.

[22] Rabin, M. O. Probabilistic algorithm for testing primality. *Journal of Number Theory 12*, 1 (1980), 128–138.

[23] Rivest, R. L., Shamir, A., and Adleman, L. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM 26*, 1 (1983), 96–99.

[24] Rotger, L. H., Coma, J. R., and Tena-Ayuso, J. Criptografía con curvas elípticas. *Universitat Oberta de Catalunya. España* (2012).

[25] Schoof, R. Elliptic curves over finite fields and the computation of square roots mod p. *Mathematics of Computation 44*, 170 (1985), 483–494.

[26] Shemanske, T. R. *Modern Cryptography and Elliptic Curves*, vol. 83. American Mathematical Soc., 2017.

[27] Silverman, J. H. *The arithmetic of elliptic curves*, vol. 106. Springer, 2009.

[28] Silverman, J. H., and Suzuki, J. Elliptic curve discrete logarithms and the index calculus. In *Advances in Cryptology — ASIACRYPT'98* (1998), K. Ohta and D. Pei, Eds., Springer, pp. 110–125.

[29] Silverman, J. H., and Tate, J. T. *Rational points on elliptic curves*, vol. 9. Springer, 1992.

[30] Washington, L. C. *Elliptic curves: number theory and cryptography*. Chapman and Hall/CRC, 2008.

[31] Weil, A. Numbers of solutions of equations in finite fields. *Bulletin of the American Mathematical Society 55*, 5 (1949), 497–508.