

Curvas elípticas en criptografía

Yabir García Benchakhtir
David Cabezas Berrido
Patricia Córdoba Hidalgo

Contenido

Definición de curva elíptica

Operaciones en el grupo de la curva

Problema del logaritmo discreto

Cifrado y firma con curvas elípticas

Algoritmo de firma digital en curvas elípticas (ECDSA)

Algoritmo de cifrado en curvas elípticas

RSA y ECC

Conceptos previos

El **espacio proyectivo** sobre un cuerpo K , $\mathbb{P}_n(K)$, es el conjunto de puntos en $K^{n+1} - \{0\}$ con la relación de equivalencia \sim que relaciona dos elementos de la siguiente forma

$$(a_0, \dots, a_n) \sim (a'_0, \dots, a'_n) \iff \exists \lambda \in K^* \text{ tal que } (a_0, \dots, a_n) = \lambda(a'_0, \dots, a'_n)$$

En el caso $K = \mathbb{R}$, \mathbb{P}_2 tiene como elementos a las rectas vectoriales de \mathbb{R}^3 . Intuitivamente, este espacio se puede interpretar como un plano y una recta “en el infinito”. Como curiosidad, dos rectas en este espacio siempre se cortan, ya que las rectas paralelas se cortan “en el infinito”.

Definición

Se define una curva elíptica como un par (E, O) , donde E es una curva proyectiva no singular de genus uno y $O \in E$.

Al punto O se le denomina “punto en el infinito”.

Denotaremos la curva como E , sobreentendiendo cual es el punto O .

El **genus** de una curva algebraica proyectiva no singular corresponde al número de agujeros de la superficie orientable compacta obtenida al considerar la curva como una variedad real.

Caracterización

Hay un isomorfismo Φ entre una curva elíptica E y la curva que cumple la ecuación de Weierstrass:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

con $a_1, \dots, a_6 \in K$ y satisfaciendo $\Phi(O) = [0, 1, 0]$ y $\Phi(P) \in \{[x, y, 1]\} \quad \forall P \in E \setminus \{O\}$.

Si la característica de K es distinta de 2 y 3, podemos simplificar la ecuación así:

$$y^2 = x^3 + Ax + B$$

con $A, B \in K$.

La curva quedaría entonces:

$$E = \{(x, y) \in K \times K : y^2 = x^3 + Ax + B\} \cup \{O\}$$

Ejemplos de curvas elípticas

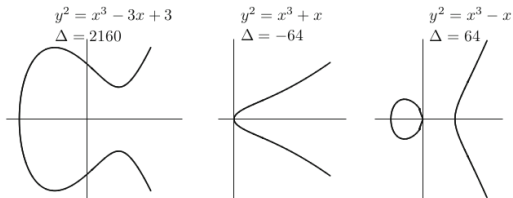


Figure 3.1: Three elliptic curves

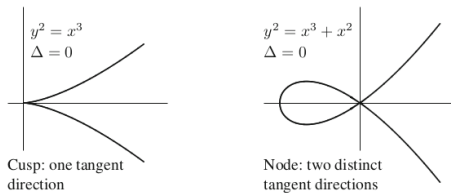
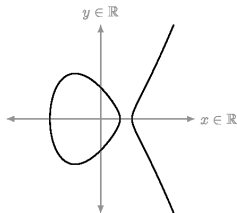


Figure 3.2: Two singular cubic curves.

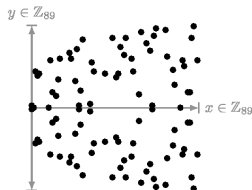
Ejemplos de curvas elípticas ($\Delta = -16(4A^3 + 27B^2)$)

Ejemplos de curvas elípticas

Consideraremos las curvas elípticas sobre grupos finitos, pero ayuda visualizarlas sobre \mathbb{R} para entender las operaciones de grupo sobre ellas. Mostramos un ejemplo de curva elíptica sobre \mathbb{R} y sobre un grupo finito.



$$y^2 = x^3 - 2x + 1 \text{ over } \mathbb{R}$$



$$y^2 = x^3 - 2x + 1 \text{ over } \mathbb{Z}_{89}$$

Curva elíptica sobre \mathbb{R} y sobre \mathbb{Z}_{89}

Contenido

Definición de curva elíptica

Operaciones en el grupo de la curva

Problema del logaritmo discreto

Cifrado y firma con curvas elípticas

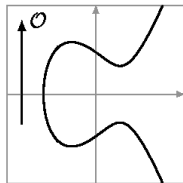
Algoritmo de firma digital en curvas elípticas (ECDSA)

Algoritmo de cifrado en curvas elípticas

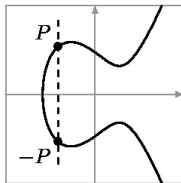
RSA y ECC

Estructura de grupo de la curva E

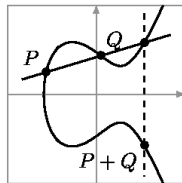
$$E = \{(x, y) \in K \times K : y^2 = x^3 + Ax + B\} \cup \{O\}$$



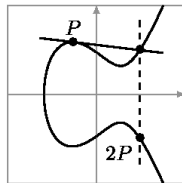
Neutral element O



Inverse element $-P$



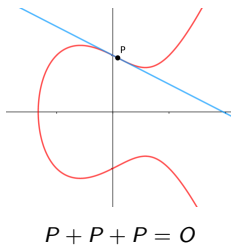
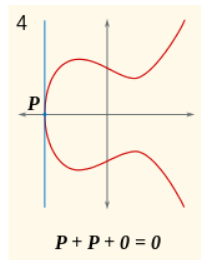
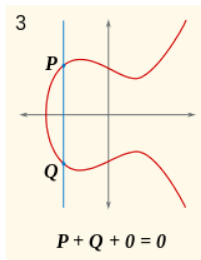
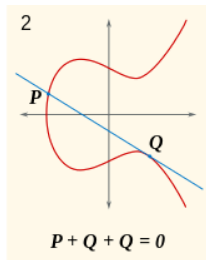
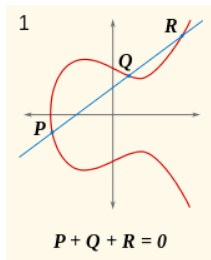
Addition $P + Q$
"Chord rule"



Doubling $P + P$
"Tangent rule"

Definimos también $O + O = O$.

Estructura de grupo de la curva E



Producto por escalares

A partir de la suma de puntos definimos el producto de un punto P por un escalar n como:

$$nP = \underbrace{P + P + \dots + P}_n$$

Esta operación puede calcularse con eficiencia $O(\log n)$ escribiendo n en base 2 y realizando duplicaciones sucesivas.

Encontrar subgrupo cíclico $\langle G \rangle \subset E(\mathbb{F}_p)$

1. Calculamos el número de puntos de la curva elíptica, $N = \#E(\mathbb{F}_p)$. Esto se puede lograr mediante el [algoritmo de Schoof](#).
2. Elegimos el factor primo mayor de N , al que llamaremos n .
3. Tomamos $h = N/n$. Para que una curva sea segura, el cofactor ha de ser pequeño.
4. Escogemos un punto cualquiera de la curva $P \in E(\mathbb{F}_p)$ y sea $G = hP$.
5. Si G es el punto en el infinito, cogemos otro punto P . De esta manera, el orden de G es n .

Podemos utilizar una curva de la [lista de curvas seguras](#) ya conocidas con cofactor pequeño.

Contenido

Definición de curva elíptica

Operaciones en el grupo de la curva

Problema del logaritmo discreto

Cifrado y firma con curvas elípticas

Algoritmo de firma digital en curvas elípticas (ECDSA)

Algoritmo de cifrado en curvas elípticas

RSA y ECC

Problema del logaritmo discreto

Sea $\langle G \rangle$ un subgrupo aditivo de $E(K)$, el **problema del logaritmo discreto** para curvas elípticas es el problema de encontrar k de manera que $kG = P$, para un punto dado $P \in \langle G \rangle$.

La seguridad de las curvas elípticas en criptografía, descansa en la dificultad de resolver este problema.

Contenido

Definición de curva elíptica

Operaciones en el grupo de la curva

Problema del logaritmo discreto

Cifrado y firma con curvas elípticas

Algoritmo de firma digital en curvas elípticas (ECDSA)

Algoritmo de cifrado en curvas elípticas

RSA y ECC

Parámetros compartidos

Alice y Bob intercambian los siguientes parámetros por un canal potencialmente inseguro:

- ▶ Una curva $E(\mathbb{F}_p)$ segura.
- ▶ G un punto de la curva de orden primo.
- ▶ Ambos deben conocer el valor n que es el orden del grupo $\langle G \rangle \subset E(\mathbb{F}_p)$.

Claves pública y privada

Las claves son:

- ▶ Clave privada: Un entero $d_X \in [1, n - 1]$ elegido de manera aleatoria.
- ▶ Clave pública: $Q_X = d_X G$.

Con $X = A$ para las claves de Alice y $X = B$ para las de Bob.

El cálculo de Q_X se puede realizar en tiempo $O(\log d_X)$.

Algoritmo de firma digital en curvas elípticas (ECDSA)

Para firmar un mensaje, Alice sigue los siguientes pasos:

1. Calcula $e = \text{HASH}(m)$.
2. Si definimos L_n como el número de bits de n , toma z los L_n bits menos significativos de e .
3. Elige de manera aleatoria un entero secreto $k \in [1, n - 1]$.
4. Calcula el punto de la curva $(x_1, y_1) = kG$.
5. Toma $r = x_1 \bmod n$. En el caso de que r sea 0, vuelve al paso 3.
6. Calcula $s = k^{-1}(z + rd_A) \bmod n$. Si s es 0, vuelve al paso 3.

La firma es el par (r, s) .

Verificación de la firma

Para verificar que el emisor es Alice, Bob seguirá los pasos siguientes:

1. Comprueba que $Q_A \neq O$.
2. Se debe cumplir que $nQ_A = O$

Si las comprobaciones anteriores son satisfactorias, Bob deberá entonces proceder de la siguiente manera:

1. Comprueba que $r, s \in [1, n - 1]$, en otro caso, la firma es inválida.
2. Calcula e usando la misma función de hashing que uso Alice.
3. Toma de nuevo z los L_n bits menos significativos de e .
4. Obtiene $u_1 = zs^{-1} \bmod n$ y $u_2 = rs^{-1} \bmod n$.
5. Calcula el punto $C = (x_1, y_1) = u_1G + u_2Q_A$. Si $(x_1, y_1) = O$ entonces la firma no es válida.

Finalmente la firma será válida si $r = x_1 \bmod n$. En caso contrario, no lo será.

Comprobación de los pasos de Bob

Veremos por qué con $C = u_1 G + u_2 Q_A$ obtenemos el resultado que queremos. Para ello notamos en primer lugar que $Q_A = d_A G$ por lo que

$$C = u_1 G + u_2 d_A G$$

Ahora usamos la propiedad distributiva:

$$C = (u_1 + u_2 d_A) G$$

desarrollamos las expresiones de u_1 y u_2

$$C = (zs^{-1} + rd_A s^{-1}) G$$

y aplicamos la propiedad distributiva de nuevo con lo que

$$C = (z + rd_A) s^{-1} G$$

Sustituimos s por su expresión tal y como se calculó en el algoritmo:

$$C = (z + rd_A)(z + rd_A)^{-1}(k^{-1})^{-1} G$$

con lo que obtenemos $C = kG$

Problema

Bajo ningún concepto debemos elegir el mismo número k para calcular dos firmas (r, s) , (r, s') de dos mensajes distintos, m y m' , ya que si eso ocurre podría averiguarse el valor de d_A .

Como curiosidad, esta debilidad fue explotada para extraer la clave usada en los sistemas PlayStation 3.

Problema

El receptor podría seguir los siguientes pasos para averiguar la clave privada d_A :

1. Calcular z y z' , dado que la función *HASH* es pública.
2. Puesto que $s - s' = k^{-1}(z - z')$ (por el paso 6 de la creación de la firma) podría averiguar $k = \frac{z - z'}{s - s'}$.
3. Como $s = k^{-1}(z + d_A)$, podría conseguir la clave privada $d_A = \frac{sk - z}{r}$.

Algoritmo de cifrado en curvas elípticas

Sea Φ una función pública invertible que transforme el mensaje un punto de la curva $E(\mathbb{F}_p)$. Para cifrar un mensaje m , Alice sigue los siguientes pasos:

1. Alice elige un número $k \in [1, n - 1]$.
2. El texto cifrado será $(kG, kQ_B + P_m)$, donde $P_m = \Phi(m)$.

Algoritmo de cifrado en curvas elípticas

Bob recibe la tupla (C, D) y realiza los siguientes pasos para descifrar el mensaje:

1. Bob obtiene P_m así: $P_m = D - d_B C = kQ_B + P_m - d_B kG$, puesto que $kQ_B = kd_B G = d_B kG$.

El mensaje m es $\Phi^{-1}(P_m)$.

Algoritmo de cifrado en curvas elípticas

Se puede usar cifrado simétrico, dado que ambos comparten el secreto

$$d_A Q_B = d_A d_B G = d_B d_A G = d_B Q_A$$

En lugar de un número k , Alice puede utilizar d_A en el cifrado. En este caso, C sería Q_A , por lo que no es necesario transmitirla junto al mensaje.

Contenido

Definición de curva elíptica

Operaciones en el grupo de la curva

Problema del logaritmo discreto

Cifrado y firma con curvas elípticas

Algoritmo de firma digital en curvas elípticas (ECDSA)

Algoritmo de cifrado en curvas elípticas

RSA y ECC

RSA vs. ECC

En esta tabla mostramos los tamaños de las claves de estos dos algoritmos, denotados k y f , al mismo nivel de seguridad.

Security Strength	RSA	ECDSA
≤ 80	$k = 1024$	$f = 160 - 223$
112	$k = 2048$	$f = 224 - 255$
128	$k = 3072$	$f = 256 - 383$
192	$k = 7680$	$f = 384 - 511$
256	$k = 15360$	$f = 512+$