Criptografía de curva elíptica

A. Hernández Estrada[1]
A. Gonzales-Quevedo[1]
I. Cardiel-Alcocer Guillermo[2]
J. E. Rámirez-Navarrete[2]
E. Corona-Organiche[2]

Resumen

La mayor parte de los sistemas de encriptación de clave pública basan su seguridad en la solución de algún problema matemático, ya que las posibilidades de encontrar la respuesta son sumamente bajas, debido a la gran cantidad de números generados. Las curvas elípticas son de nueva generación, usadas en criptografía para formar claves o firmas digitales, especialmente en criptografía asimétrica. Una de sus mayores ventajas es la velocidad que ofrece, debido a que requiere longitudes de clave mucho menor a las de criptosistemas como RSA o Diffie Hallman.

En este documento, estudiaremos inicialmente la formación de una curva elíptica y cómo se están utilizando para construir sistemas de encriptación de clave pública, además de una comparación con otros sistemas criptográficos utilizados en la actualidad.

Palabras clave: Curva elíptica, Criptografía de curva elíptica, Clave pública.

Introducción

Todos los sistemas de encriptación conocidos en la actualidad, basan su seguridad en la resolución de algún problema matemático que por su gran magnitud, es casi imposible de resolver en la práctica en un tiempo menor al esperado.

Como una opción, en 1985 Neil Koblitz y Victor Miller propusieron el Elliptic Curve Criptosystem (ECC) o Criptosistema de Curva Elíptica, basado en los métodos de Diffie-Helman y DSA de clave pública (Belingueres, 2005, 29).

El ECC puede ser usado tanto para encriptar como para firmar digitalmente, y hasta hoy no se ha conocido algún ataque que recorte el tiempo exponencialmente calculado

Acerca de los autores...

[1] Tecnológico de estudios Supriores de Ecatepec (TESE)
[2] División de Ingeniería en Sistemas Computacionales del TESE

para romper el ECC, lo que facilita su uso al crear claves más pequeñas sin necesidad de grandes recursos computacionales.

Las curvas elípticas

En criptografía, se habla de curva elíptica en referencia a una ecuación

$$y^2=x^3+Ax+B \quad (1)$$

que cumple:

$$4A^3+27B^2 \neq 0 \quad (2)$$

Al asignar diferentes valores a A y B, obtenemos un conjunto de curvas que, al ser dibujadas, ofrecen una forma similar. Son ejemplos de curvas elípticas $y^2=x^3-x$ a la izquierda de la Figura 1 y $y^2=x^3-x+1$ a la derecha de la misma.

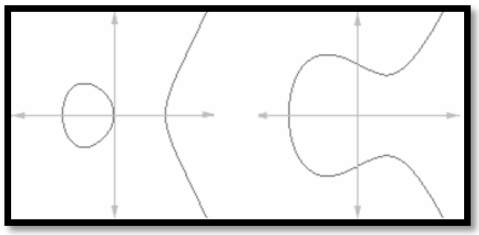

Figura 1
Familia de Curvas Elípticas

Las curvas elípticas tienen ciertas características que las hacen especiales en el mundo de la criptografía. Una de éstas consiste en la posibilidad de generar un punto en una curva, partiendo de dos puntos dados (o incluso de uno). Este concepto es fácil de entender partiendo de la Figura 2, como sigue.

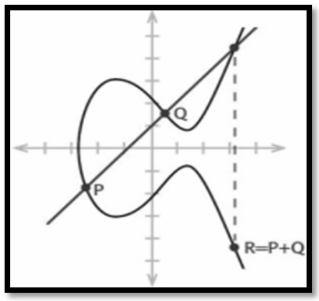

Figura 2
Suma de P y Q.

Usamos como puntos de partida P y Q, dos puntos conocidos. Trazaremos una línea entre P y Q. Si la línea corta la curva en un tercer punto, lo reflejaremos a través del eje, dando lugar a un nuevo punto R. Esta operación se representa como:

$$R = P + Q \quad (3)$$

En caso de que la línea que pasa por P y Q no corte a la curva en ningún otro punto, diremos que corta la curva en un punto O en el infinito y representaremos esta operación como:

$$P + Q = O \quad (4)$$

Partiendo de la suma, no es difícil encontrar un mecanismo que nos permita realizar multiplicaciones de tipo kP, siendo k un escalar. Por ejemplo, imaginemos que que-

remos realizar la operación 13P, es decir, multiplicar 13 por un punto P. Bastaría con realizar la siguiente secuencia de doblado de puntos:

$$P,\ 2P = P + P,\ 4P = 2P + 2P,\ 8P = 4P + 4P,\ 13P = 8P + 4P + P \quad (5)$$

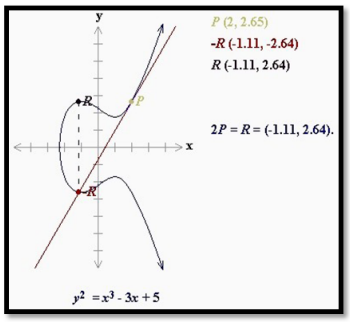

Figura 3
Suma del mismo punto P

Este simple mecanismo para generar nuevos puntos, dota a una curva elíptica (Figura 3) de la posibilidad de realizar operaciones aritméticas sobre ella, que es la base de los criptosistemas mencionados.

En criptografía, las curvas elípticas se usan sobre campos finitos ($\mathbb{F}q$) con q muy grande. Un ejemplo de campo finito podría ser $\mathbb{F}5=\{0,1,2,3,4\}$. De manera que el número 7 representado en el campo finito correspondería a

$$7 \bmod 5 = 2 \quad (6)$$

Cuando se usan campos finitos, el número de puntos que hay en una curva también es finito. Este número se llama orden de la curva y se representa como $\#\mathbb{E}$. Debemos diferenciarlo del orden de un punto, que se refiere al valor k más pequeño (diferente de 0) que multiplicado por P da O.

El problema del logaritmo discreto (DPL)

La criptografía de clave pública basa su fuerza en la dificultad de resolver ciertos problemas matemáticos. Uno de los más usados es el logaritmo discreto (Discrete Logarithm Problem DLP). Este problema se basa en la dificultad que representa resolver una ecuación de tipo

$$x = ay \bmod n \quad (7)$$

donde x, a y n son conocidas y e es la variable que se busca. De hecho, para valores de n e y suficientemente grandes, es computacionalmente imposible resolver el problema, al menos con los algoritmos y ordenadores actuales.
Otros criptosistemas han sido propuestos, cuya seguridad se basa en el DLP, entre ellos están:

- Los esquemas de acuerdo de claves, derivados del Diffie-Hellman, como ElGamal, la familia de protocolos MTI y el protocolo STS (Station-to-Station).

- El esquema de firma digital ElGamal y sus variantes, como DSA (Digital Signature Algorithm), el esquema de firma de Schnorr, y el esquema ElGamal con recuperación de mensaje de Nyberg-Rueppel.

El algoritmo más rápido conocido para resolver este problema, es el Index Calculus, que permite resolverlo en tiempo subexponencial.

El problema del logaritmo discreto en curvas elípticas (ECDLP)

Existe un problema similar al del logaritmo discreto que puede usarse con curvas elípticas. Anteriormente, hemos visto cómo realizar una operación tipo Q = kP de una forma sencilla. Sin embargo, obtener k y P partiendo sólo de Q, es computacionalmente difícil. De hecho, el algoritmo más rápido que permite encontrar una solución es el Rho de Pollard, pero éste es de tiempo exponencial, mucho más lento que en el caso del ataque a DLP mediante el Index Calculus.

Este hecho es muy importante, pues la dificultad de resolver ECDLP frente a DLP permite que los criptosistemas que se basan en el primero, usen claves mucho más cortas. Así, los sistemas que usan ECDLP requieren mucha menos memoria y capacidad de proceso.

Una clave RSA de 4,096 bits, ofrece la misma seguridad que la de un criptosistema de Curva Elíptica de 313 bits.

La similitud de las definiciones, hace que todos los criptosistemas basados en el DLP puedan ser adaptados utilizando curvas elípticas. De esta forma, existen variantes en los protocolos y esquemas anteriores convertidos a curvas elípticas, y entonces tenemos ECDSA, EC Diffie-Hellman, encriptación EC ElGamal, etcétera.

Algunas leves modificaciones técnicas son necesarias para adaptarlos al grupo de las curvas elípticas, pero los principios subyacentes son los mismos que para los otros sistemas basados en el DLP.

Comparación con otros criptosistemas

Seguridad: el nivel de seguridad está determinado por el valor de la información, el tiempo que debe ser protegida, el tamaño de los parámetros que serán usados, entre otros. Los sistemas basados en ECC brindan la misma protección que los ya tradicionales, como se puede observar en la Tabla 1, donde los niveles de seguridad se dan en bits según la longitud de la clave.

Nivel de Seguridad	Esquema Simétrico (tamaño de clave)	Esquema Basado en ECC (tamaño de n)	DSA/RSA (tamaño del módulo)
56	56	112	512
80	80	160	1024
112	112	224	2048
128	128	256	3072
192	192	384	7680
256	256	512	15360

Tabla 1
Tamaños de clave comparables (en bits)

Eficiencia: se deben considerar los siguientes factores:

- Sobrecarga en Cálculos: Cuánto tiempo de ejecución se requiere para transformar las claves privadas y públicas.

- Tamaño de Clave: Cantidad de bits requeridos para almacenar el par de claves y otros parámetros (Tabla 2 y 3).

	Parámetros del Sistema	Clave Pública	Clave Privada
RSA	n / a	1088	2048
DSA	2208	1024	160
ECC	481	161	160

Tabla 2
Tamaño de los parámetros del sistema y par de claves (en bits)

	Tamaño de Firma
RSA	1024
DSA	320
ECC	320

Tabla 3
Tamaño de firma (en bits)

- Ancho de banda: Cantidad de bits que deben ser comunicados para transferir un mensaje (Tabla 4).

	Tamaño del mensaje encriptado
RSA	1024
DSA	2048
ECC	321

Tabla 4
Tamaños de mensajes encriptados (en bits)

Por lo tanto todos estos ahorros provocan una sobre carga en cálculos más eficiente en una proporción considerablemente menor de tiempo mejorando el consumo y reduciendo el tamaño del código.

Intercambio de claves usando curva eliptica con Diffie-Hellman (ECDH)

El intercambio de claves de Diffie-Hellman es un protocolo que hace posible un intercambio secreto y seguro de claves entre dos partes que no han tenido un contacto previo. Se usa ampliamente en criptografía y se basa en el problema del logaritmo discreto (DLP). Por lo tanto, puede usarse el mismo algoritmo a través del problema ECDLP.

Al algoritmo puede resumirse en los siguientes pasos:

1. Alice y Bob eligen una curva elíptica E sobre un campo finito $\mathbb{F}q$, de manera que el ECDLP sea computacionalmente difícil. También eligen un punto P en dicha curva de modo que su orden sea un número primo grande.

2. Alice elige un entero grande a, calcula PA = aP y envía PA a Bob.

3. Bob elige un entero grande b, calcula PB = bP y envía PB a Alice.

4. Alice calcula aPB = abP

5. Bob calcula bPA = abP

Al finalizar el algoritmo, tanto Alice como Bob disponen de abP, pero un usuario que escuche el canal, sólo habrá podido obtener PA y PB, los cuales no le permiten calcular abP a menos que resuelva el ECDLP. Alice y Bob únicamente necesitarán extraer una clave a partir de abP y usarla para enviar datos cifrados. Para tal propósito, podrán usar cualquier algoritmo simétrico como DES, AES, etcétera.

Algoritmo de curva elíptica para firma digital (ECDSA)

El algoritmo de firma digital para curvas elípticas está basado en el estándar de firma digital DSA. Este algoritmo ofrece un esquema que permite firmar documentos y verificar las firmas. Los pasos a seguir para generar claves (firmar y verificar la firma), se muestran a continuación:

Alice genera un par de claves:

1. Alice elige una curva E con orden $\#\mathbb{E}$ = fr, de manera que r sea un primo grande.
2. Alice busca un punto en la curva de orden r.
3. Alice elige un número aleatorio d situado en el intervalo [2, r-2] y calcula Q = dP.
4. La clave pública corresponde a (E,P,r,Q) y la clave privada a d.

Alice firma un documento M. (h(M) corresponde al hash de M)

1. Alice elige un número aleatorio k en el intervalo [2, r-2].
2. Se calcula el punto (x, y) = kP
3. R=x mod r
4. s = k-1 (h(M) +Rd) mod r, si s es igual cero, empezamos de nuevo.
5. La firma de Alice es (R,s) y se transmite junto con el mensaje M.

Bob verifica la firma de Alice.

1. Bob obtiene la clave pública de Alice.
2. w = s-1 mod r
3. u1 = h(M) w mod r
4. u2 = Rw mod r
5. (x, y) = u1P + u2P
6. v = x mod r
7. Si v es igual a R, la firma es válida.

Conclusiones

Los sistemas basados en curva elíptica ofrecen un mejor rendimiento, hablando en términos computacionales, a pesar de que su implementación no ha llegado a un alto porcentaje como un método de encriptación aceptado, puesto que aún no posee el nivel de confianza que se gana a través de los años, sin embargo, tiene un futuro prometedor, ya que su implementación no requiere de grandes equipos de computo.

Referencias...

[1] Belingueres, Gabriel. Introducción a los Criptosistemas de Curva Elíptica, Chucabuco Buenos Aires, Argentina, 2005.

[2] Certicom, Certicom ECC tutorials, http://www.certicom.com/index.php

[3] De Win,E. and Prencel, B. Elliptic Curve public key Cryptosystems - an introduction,

[4] Hankerson, Darrel, Menezes, Alfred and Vanstone, Scott. Guide to Elliptic Curve Cryptography, 2003.

[5] NIST, Digital Signature Standar, FIPS PUB 186-2, http://csrg.ncsl.nist.gov/fips/, Enero 2000.