Artículo **Revista de Cómputo Aplicado**
Septiembre 2019 Vol.3 No.11 28-35

Criptografía basada en curvas elípticas

Cryptography based on elliptic curves

RAMÍREZ-HERNÁNDEZ, Héctor David†*, CONTRERAS-JUÁREZ, Roberto, ESPINOZA-HERNÁNDEZ, Nelva Betzabel y SÁNCHEZ-MENDOZA, Eduardo

Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación.

ID 1er Autor: *Héctor David, Ramírez-Hernández* / **ORC ID**: 0000-0003-3741-4285

ID 1er Coautor: *Roberto, Contreras-Juárez* / **ORC ID**: 0000-0001-3271-6754

ID 2do Coautor: *Nelva Betzabel, Espinoza-Hernández* / **ORC ID**: 0000-0002-5620-2336

ID 3er Coautor: *Eduardo, Sánchez-Mendoza* / **ORC ID**: 0000-0003-2690-6217

DOI: 10.35429/JCA.2019.11.3.28.35 Recibido Abril 30, 2019; Aceptado Junio 30, 2019

Resumen

La criptografía incorpora las técnicas con las que se busca garantizar protección de la información, frente a personas no autorizadas. Esta manera de resguardar información ha existido desde tiempos remotos, en donde existían elementos que sólo ciertas personas eran capaces de entender e interpretar. En un principio la criptografía fue utilizada para efectos de guerra y poder, pero gracias a los grandes avances tecnológicos desarrollados a finales del siglo pasado, se ha visto la necesidad de resguardar la información que cada individuo maneja y comparte por medio del internet. Es por lo que la criptografía toma una mayor importancia. La criptografía actual se basa en dos tipos de protocolos, uno el de la criptografía simétrica y el otro que corresponde a la criptografía asimétrica. En este trabajo se analizó un protocolo del tipo asimétrico basado en curvas elípticas sobre el campo finito GF(p), proponiendo una librería desarrollada en PHP que permite cifrar y descifrar información, la cual pretende brindar los servicios de seguridad, autenticación, integridad y confidencialidad de la información.

Criptografía, Protocolo, Curvas elípticas

Abstract

Cryptography incorporates the techniques with which it seeks to guarantee protection of information, in front of unauthorized persons. This way of protecting information has existed since ancient times, where there were elements that only certain people were able to understand and interpret. In the beginning, cryptography was used for the purposes of war and power, but thanks to the great technological advances developed at the end of the last century, we have seen the need to safeguard the information that everyone manages and shares through the internet. That is why cryptography takes on greater importance. Current cryptography is based on two types of protocols, one of symmetric cryptography and the other corresponding to asymmetric cryptography. In this paper, an asymmetric type protocol based on elliptic curves on the finite field GF(p) was analyzed, proposing a library developed in PHP that allows to encrypt and decrypt information, which aims to provide security services, authentication, integrity and confidentiality of the information.

Cryptography, Protocol, Elliptic curves

Citación: RAMÍREZ-HERNÁNDEZ, Héctor David, CONTRERAS-JUÁREZ, Roberto, ESPINOZA-HERNÁNDEZ, Nelva Betzabel y SÁNCHEZ-MENDOZA, Eduardo. Criptografía basada en curvas elípticas. Revista de Cómputo Aplicado. 2019, 3-11: 28-35

* Correspondencia al Autor (Correo electrónico: hector.ramirezhe@correo.buap.mx)
† Investigador contribuyendo como primer Autor.

Introducción

La criptografía se puede definir como el arte o la ciencia de cifrar y descifrar información, utilizando técnicas matemáticas que hagan posible el intercambio de mensajes de manera que solo puedan ser leídos por las personas a quienes van dirigidos (Granados, 2006). La finalidad de la criptografía es garantizar que el contenido del mensaje enviado no haya sido modificado en su tránsito, o que si se resguarda esta no pueda ser extraída (Granados, 2006). Los criptosistemas utilizados son los denominados simétricos y los asimétricos. Los criptosistemas simétricos son aquellos que usan un método matemático para cifrar y descifrar un mensaje (Gómez, 2002).

Este tipo de criptografía utiliza únicamente una llave para realizar el proceso, así el mensaje únicamente se descifra con la única llave existente. Este tipo de criptografía garantiza confidencialidad, pero al querer compartir el mensaje con otro usuario deberá crearse una nueva llave y el número de llaves aumenta conforme aumenten los usuarios con quienes se comparta el mensaje (Amalraj y Raybin, 2016). En 1976, Whitfield Diffie y Martin Hellman (Diffie y Hellman, 1976) revolucionan la criptografía al introducir el concepto de criptosistema asimétrico, que surge como una solución al problema de intercambiar claves privadas por canales inseguros.

Los sistemas asimétricos son aquellos en los cuales tanto el emisor como el receptor poseen un par de claves: una de tipo pública y la otra de tipo privada y para enviar mensajes el emisor tiene que cifrar el mensaje con la clave pública del receptor para que así este sea el único que pueda descifrar el mensaje usando su clave privada (Hankerson et al. 2004). Este novedoso sistema de encriptación fundamenta su seguridad en problemas matemáticos cuya solución computacional resulta difícil de resolver, ya que, aun conociendo los algoritmos para resolverlos, no es factible su ejecución en un tiempo razonable. Ejemplos de este tipo de criptografía son RSA y criptografía basado en curvas elípticas (ECC) (Amalraj y Raybin, 2016). La teoría de curvas elípticas conforma una herramienta matemática que brinda a la criptografía la oportunidad de optimizar los algoritmos de cifrado, tal es el caso de El Gamal, mejorando la eficiencia y robustez sin utilizar más recursos computacionales.

La criptografía de curva elíptica (ECC), planteada por Koblitz y Miller (Koblitz, 1987), es una variante de la criptografía asimétrica la cual se basa en las propiedades de las curvas elípticas que resulta ser más eficiente que los métodos como el Rivest, Shamir y Adleman (RSA), además de proporcionar un nivel de seguridad equivalente (Gupta et al., 2004).

En la sección de curva elíptica se da un panorama general de las curvas elípticas, en los que se definen los tipos de curvas elípticas utilizadas, así como las operaciones que hacen que conformen un grupo abeliano.

En la sección del protocolo criptográfico, se consideran los parámetros necesarios para implementar la criptografía de curvas elípticas, se menciona un método para calcular la cantidad de puntos que contiene una curva elíptica usando el símbolo de Legendre para el cálculo de los residuos cuadráticos. En la última sección de cifrado de datos, se explica la implementación del algoritmo El Gamal para el cifrado y descifrado de datos, en la que se muestra la aplicación de la librería desarrollada en PHP.

Criptografía de curvas elípticas (ECC)

Como se mencionó anteriormente, la criptografía de curvas elípticas (ECC) pertenece a la criptografía asimétrica, debido a que se utilizan dos tipos de llaves distintas, una pública y una privada, en la que el conocimiento de la llave pública no permite determinar el conocimiento de la clave privada. La criptografía de curvas elípticas fue propuesta en 1985 por Neal Koblitz (Koblitz, 1987) y Víctor Miller (Miller, 1986). Desde entonces una gran cantidad de investigaciones se han realizado para tener implementaciones eficientes y seguras de estos esquemas criptográficos. La Criptografía de Curvas Elípticas ha permitido explorar nuevos criptosistemas, tal como la técnica de emparejamientos bilineales (Kawahara et al., 2006).

Curva Elíptica

Definimos de manera general a las curvas elípticas de la siguiente forma. Sea K un campo. Una curva elíptica sobre K, es la curva plana sobre K definida por la ecuación de Weierstrass:

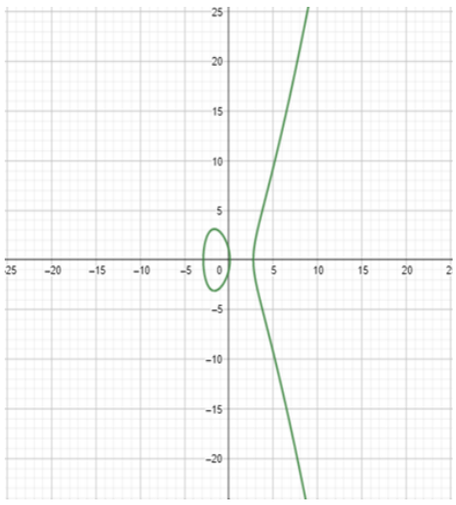
$$y^2 = x^3 + ax + b \quad (1)$$

ISSN-2531-2952

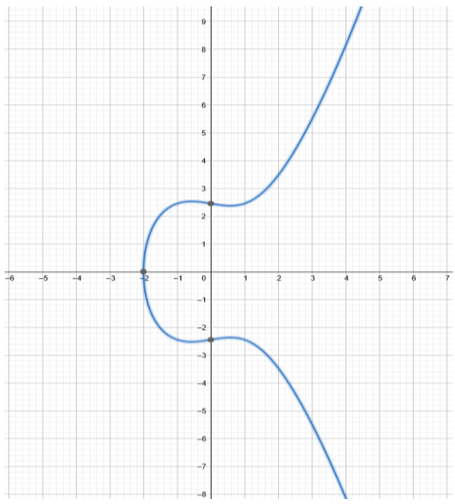
donde $x, y, a, b \in K$.

Para poder definir una estructura algebraica de grupo abeliano es necesario incluir un punto, denotado por ∞ y llamado punto en el infinito, que no se encuentra en la curva. Este punto se encuentra situado por encima del eje de las abscisas a una distancia infinita, y que por lo tanto no tiene un valor en concreto. Si $x^3 + ax + b$ no tiene raíces múltiples, entonces la curva correspondiente, aunado con el punto ∞, más la operación suma definida más adelante, es lo que se denomina grupo de la curva elíptica sobre K, denotado por $E(K)$. Esto es, el conjunto: $E(K) = \{(x, y): x, y \in K, y^2 = x^3 + ax + b\} \cup \{\infty\}$ forma un grupo abeliano, en donde, ∞ es el elemento identidad del grupo de curva elíptica.

Las siguientes gráficas son ejemplos de curvas elípticas definidas sobre $\mathbb{R}$.

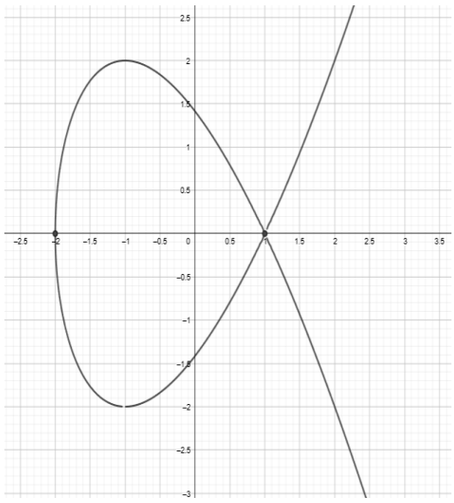


Gráfica 1 *Curva elíptica* $y^2 = x^3 - 8x + 1$

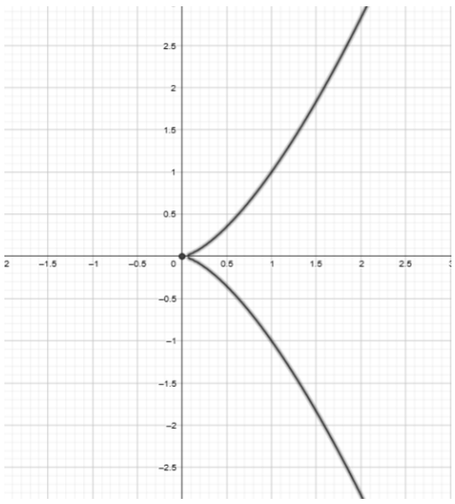


Gráfica 2 *Curva Elíptica* $y^2 = x^3 - x + 6$

A la expresión $\Delta = 4a^3 + 27b^2$ se le conoce como el discriminante de la curva elíptica. Se verifica que para que la curva elíptica no tenga raíces múltiples es necesario que $\Delta \neq 0$. Esta condición permite excluir las curvas elípticas que tengan un punto doble o un pico como lo muestran las Gráficas 3 y 4.



Gráfica 3 Curva Elíptica $y^2 = x^3 - 3x + 2$ sobre $\mathbb{R}$



Gráfica 4 Curva Elíptica $y^2 = x^3$ sobre $\mathbb{R}$

Trabajar criptografía con curvas elípticas sobre los números reales se puede volver lento e inexacto, debido a los errores de redondeo que puedan existir. En la práctica, el trabajo de criptografía se lleva a cabo con curvas elípticas sobre el campo finito $GF(p)$ pertenecientes a los campos primos y $GF(2^m)$ pertenecientes a los campos de binarios. En este trabajo nos centramos en las curvas elípticas sobre el campo finito $GF(p)$, con p un número primo.

Recordando que el campo de $GF(p)$ usa los números del 0 al p-1 y en cómputo final se obtiene el módulo de p.

ISSN-2531-2952

Una curva elíptica definida sobre el campo de $GF(p)$, denotada por $E(GF(p))$, esta formada por las variables a y b dentro del campo de $GF(p)$. Las curvas elípticas incluyen todos los puntos de (x, y) que satisface la ecuación de una curva elíptica módulo p. Esto es, una curva elíptica sobre $GF(p)$ tiene por ecuación:

$$y^2 \ mod \ p = (x^3 + ax + b) \ mod \ p, \qquad (2)$$

donde $a, b, x, y \in GF(p)$.

De manera análoga, si $x^3 + ax + b$ contiene factores no repetidos, o equivalentemente si

$$4a^3 + 27b^2 \neq 0 \ mod \ p \qquad (3)$$

entonces la curva elíptica se puede utilizar para la criptografía. Una curva elíptica sobre el campo de $GF(p)$ tiene los puntos correspondientes en la curva elíptica, junto con un punto especial ∞, el cual se le llama punto en infinito o punto cero. La cantidad de puntos de una curva elíptica sobre un campo finito es finita. La cardinalidad de puntos de una curva elíptica se denota por $\#E(GF(p))$. El número de puntos de una curva elíptica es llamado el orden de la curva.

Ejemplo: Consideremos la curva elíptica sobre $GF(11)$. Con $a = 6$ y $b = 10$, la ecuación de la curva elíptica es $y^2 = x^3 + 6x + 10$. Los puntos que pertenecen a esta curva son: (0,3), (0,5), (6,3), (6,8), (8,3), (8,8), (9,1), (9,10), (10,5), (10,6) incluyendo a ∞. Esto es, $\#E(GF(p)) = 11$.

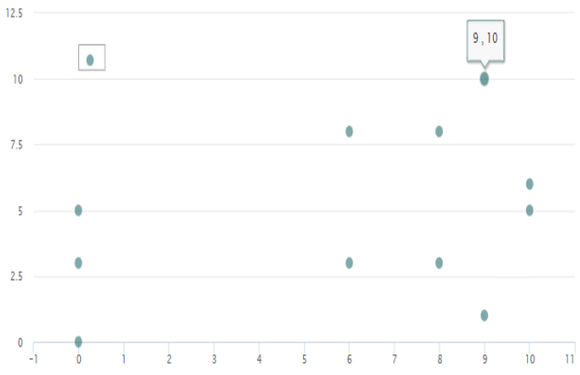


Figura 1 Número de puntos de la curva

Desde el punto de vista algebraico la ley de grupo para una Curva Elíptica representada por la ecuación de *Weierstrass* (1), se define de acuerdo con las siguientes propiedades:

1. $P_1 + \infty = P_1$
2. Si $P_1 = (x_1, y_1)$, entonces $-P_1 = (x_1, -y_1)$.
3. Sean $P_1 = (x_1, y_1)$ y $P_2 = (x_2, y_2)$ puntos de la curva elíptica con $P_1, P_2 \neq \infty$. Entonces si $x_1 = x_2$ pero $y_1 \neq y_2$ o $P_1 = P_2$ y $y_1 = 0$ entonces $P_1 + P_2 = \infty$. En otro caso
   $P_1 + P_2 = P_3 = (x_3, y_3)$ con

$$x_3 = m^2 - x_1 - x_2,$$
$$y_3 = m(x_1 - x_3) - y_1$$
$$m = \begin{cases} \dfrac{3x_1^2 + a}{2y_1} & x_1 = x_2 \\ \dfrac{y_2 - y_1}{x_2 - x_1} & x_1 \neq x_2 \end{cases}$$

La curva elíptica $E(GF(p))$ dotada de la operación suma definida anteriormente forma un grupo abeliano. Sea P un punto de la curva elíptica $E(GF(p))$. Entonces, kP significa la suma del punto P consigo mismo k-veces, con la suma definida anteriormente.

Es conocido el hecho (Katz y Lindell, 2015) de que si $\#E(GF(p))$ es un número primo entonces el grupo $E(GF(p))$ es un grupo cíclico. Este proceso facilita el hecho de que, si una curva elíptica cumple con esta condición, entonces cualquiera de sus puntos diferente del ∞ será un punto generador del grupo.

Protocolo criptográfico

Hoy en día se vive en un ambiente donde la interacción con las nuevas tecnologías es tan cotidiana como tener que ir al colegio. No hay cosa que no manejemos a través de un ordenador, ya sea para mandar un mensaje a un ser querido, hacer un trabajo escolar o hasta nuestro mismo trabajo necesita de un ordenador, es claro que la mayoría de las veces es a través de diferentes páginas web.

Estas tecnologías han permitido un sin fin de cosas, como las aplicaciones móviles y aplicaciones web. Es más que un hecho que cualquier documento importante, transferencia bancaria, correos confidenciales hasta la documentación de un caso penal no sea enviada por internet, es por ello por lo que es importante proteger esa información que en términos computacionales llamamos encriptar y la herramienta que nos permite trabajar en el lado del servidor para proteger este tipo de datos es PHP.

ISSN-2531-2952

Para llevar a cabo la encriptación y desencriptación usando curvas elípticas, es necesario realizar múltiples operaciones que nos permiten establecer los mecanismos de seguridad que se necesitan en el resguardo de la información. Para ello detallamos las operaciones utilizadas para llevar a cabo el objetivo de este trabajo. El desarrollo de este sistema fue utilizando PHP.

El primer paso consiste en proporcionar un número primo p y los coeficientes a y b, para formar la curva elíptica de la forma **¡Error! No se encuentra el origen de la referencia.** cumpliendo la condición **¡Error! No se encuentra el origen de la referencia.**. Para mostrar este paso, consideremos los coeficientes a = 9, b = 13 y el número primo p = 19. Con estos parámetros se obtiene la curva $y^2 = (x^3 + 9x + 13)\ mod\ 19$, cumpliendo con la condición *(3)*.
Para el siguiente paso se necesita determinar el número de puntos con los que cuenta la curva elíptica, $\#E(GF(p))$. Para calcularlo se puede implementar el algoritmo de Schoff que es de un tiempo polinómico.

Sin embargo, su teoría e implementación queda fuera del alcance de este artículo, para obtener detalles de este algoritmo se puede consultar (Schoff, 1995). Para nuestro propósito, utilizamos el cálculo de los residuos cuadráticos para determinar $\#E(GF(p))$. Para ello, el número de Legendre nos apoya para determinar los puntos que contienen residuos cuadráticos. Dados un número primo p y un entero cualquiera x, el símbolo de Legendre está definido de la siguiente manera:

$$\left(\frac{x}{p}\right) = \begin{cases} 1\ si\ x\ es\ un\ residuo\ cuadrático \\ -1\ si\ x\ no\ es\ residuo\ cuadrático \\ 0\ si\ x\ es\ múltiplo\ de\ p \end{cases} \quad (4)$$

Una alternativa para el cálculo del símbolo de Legendre es mediante el criterio de Euler utilizando la siguiente fórmula:

$$\left(\frac{x}{p}\right) = x^{\frac{p-1}{2}} mod\ p, \quad (5)$$

En el que, al resolver **¡Error! No se encuentra el origen de la referencia.** se obtienen los valores 1, -1 y 0, y para determinar si es o no un residuo cuadrático se utiliza **¡Error! No se encuentra el origen de la referencia.**. A continuación, se muestra el algoritmo, del número de Legendre. Tomando a x como un número entero positivo cualquiera menor a p, z representa el resultado de evaluar en la ecuación $y^2 = (x^3 + 9x + 13)\ mod\ 19$ y r es el residuo cuadrático aplicando el símbolo de Legendre, se obtiene la tabla 1.

Una vez calculado el número de Legendre, se conoce la cantidad de puntos que tiene la curva, que en nuestro caso es #E(GF(p)) = 21. Ahora para calcular todos los puntos de la curva se toman los valores mostrados en la tabla 1 y se utiliza la formula $H = (x - p)^2$ dando como resultado la tabla 2.

x	z	residuo
0	13	-1
1	4	1
2	1	1
3	10	-1
4	18	-1
5	12	-1
6	17	1
7	1	1
8	8	-1
9	6	1
10	1	1
11	18	-1
12	6	1
13	9	1
14	14	-1
15	8	-1
16	16	1
17	6	1
18	3	-1

Tabla 1 Residuos Cuadráticos

y	H
0	0
1	1
2	4
3	9
4	16
5	6
6	17
7	11
8	7
9	5
10	5
11	7
12	11
13	17
14	6
15	16
16	9
17	4
18	1

Tabla 2 coordenadas y

Cuando obtenemos los valores de $x, z,$ y H en las tablas (1) y (2) respectivamente, se aplica el algoritmo de la Figura 2 para encontrar todos los puntos que pertenecen a la curva $E(GF(p))$. Para esto, se reciben dos arreglos donde tenemos almacenados los datos de las tablas anteriores, primeramente, se toma el primer valor de la tabla 1, es decir, el valor z**,** y si algún valor de z coincide con un valor de H en la tabla 2**,** entonces el punto está formado por el valor de x de la tabla 1 y el valor de y de la tabla 2.

Entrada: $arrayXZ$, $arrayYH$
Salida: Puntos de la curva E(GF(p))
1: $count \leftarrow 0$
2: **for** ($i = 0$ to length arrayXZ) **do**
3: $z \leftarrow arrayXZ[i] -> getZ()$ //función para obtener el valor de z
4: **for** ($j = 1$ to length arrayYH) **do**
5: **if** ($z = \boldsymbol{arrayYH[j] -> getH()}$) **then**
6: $arrayPoint[count] \leftarrow (arrayXZ[i] -> getX(), arrayYH[j] -> getY())$
7: $count \leftarrow +1$
8: **if** (isGen($arrayPoint[count]$)) **then**
9: return $arrayPoint[count]$

```
10:                    EndIf
11:          Endif
12:             Endfor
13: Endfor
```

Figura 2 Algoritmo para encontrar los puntos de la curva elíptica

Al hacer una modificación al algoritmo anterior se puede calcular el orden de todos los puntos de la curva. Para fines de ilustración en nuestro ejemplo se calcularon los órdenes de todos los puntos de la curva obteniendo la tabla 3. En la práctica esto no es necesario, puesto que sería demasiado costoso, ya que al ser un método que necesita calcular el símbolo de Legendre desde 0 hasta $p-1$ este se vuelve inviable a medida que el p crece.

x	y	Orden
1	2	21
1	17	21
2	1	7
2	18	7
6	6	21
6	13	21
7	1	21
7	18	21
9	5	21
9	14	21
10	1	3
10	18	3
12	5	7
12	14	7
13	3	21
13	16	21
16	4	7
16	15	7
17	5	21
17	14	21

Tabla 3 Orden de cada punto de la curva elíptica

Cifrado de datos

Para realizar el cifrado de los datos, se debe contar con los siguientes parámetros: el número primo p, los coeficientes a y b, la cardinalidad de puntos de la curva elíptica $\#E(GF(p))$, un punto generador G de la curva, valores M y h con $Mh < p$, $llaveSA$ (llave secreta del usuario A), $llaveSB$ (llave secreta del usuario B) en donde $mcd(llaveSA, llaveSB) = 1$**.** Estos parámetros son utilizados por el cifrado de El Gamal. Mostramos con un ejemplo el uso de la librería desarrollada en PHP. Para ello consideremos:

$p = 500009$
$a = 15567$
$b = 7896$
$\#E(GF(p)) = 499879$
$G = (241479, 71146)$
$M = 456$
$h = 123$
$LlaveSA = 24528$
$LlaveSB = 11923$
$y^2 = (x^3 + 15567x + 7896) \ mod\ 500009.$

Supongamos que tenemos un usuario que se va a registrar en una plataforma y lo que interesa cifrar es la contraseña. Para nuestro caso la contraseña es carlos123 e ingresa sus datos en un formulario, Figura 3.

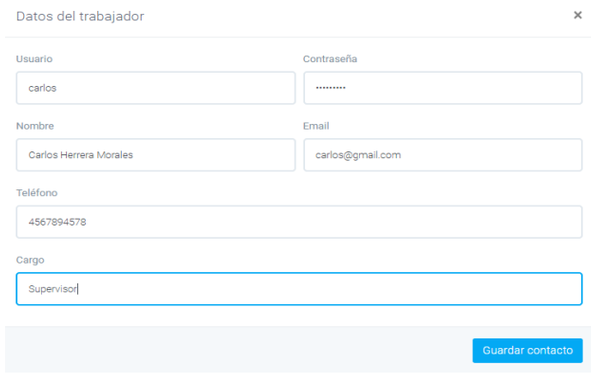


Figura 3 Formulario de contacto

Procedemos a cifrar la contraseña del usuario de la siguiente manera:

Paso 1. Codificar el mensaje, para ello usamos el código ascii que comprende del 0 al 255.

Inicializamos con $j = 1$. Sabiendo que el ascii de la letra c es 99 se realizan los siguientes pasos:

Paso 1. 1 Calculamos $x = ascii(c)(h) + j = 99\,(123) + 1 = 12178$, y se sustituye este resultado en la ecuación de la curva elíptica, esto es, $y^2 = (x^3 + 15567x + 7896) = 12178^3 + 15567(12178) + 7896 = 334992$**,** dado que no existe valor de y que cumpla con esta ecuación, se aumenta a $j = 2$ y se reinicia el proceso.

Paso 1.2 $x = ascii(c)\,h + j = 99\,(123) + 2 = 12179$**,** y se sustituye este resultado en la ecuación de la curva elíptica, esto es, $y^2 = (x^3 + 15567x + 7896) = 12179^3 + 15567(12179) + 7896 = 290136$ este número si tiene raíz cuadrada que es igual a 161435. Por tanto, la codificación de la letra c es $(12179, 161435)$. Repetimos este proceso para cada una de las letras y números obtenemos:

$c = (12179, 161435)$
$a = (11936, 218659)$
$r = (14023, 101746)$
$l = (13285, 115296)$

$o = (13656,29398)$
$s = (14147,176240)$
$1 = (6031,183341)$
$2 = (6152,153375)$
$3 = (6277,52620)$

Paso 2. Se calcula la llave pública de A: $llavePA = LlaveSA * G$ 0 24528 (241479 , 71146) (253513 , 78497), entonces la pareja es igual $A = (24528, (253513, 78497))$.

Paso 3. Se calcula la llave pública de B: $llavePB = LlaveSB * G =$ 11923 (241479 , 71146) = (339894 , 358573), entonces la pareja es igual $B = (4562, (339894,358573))$.

Paso 4. Ciframos la letra c eligiendo un entero aleatorio $k = 205887$ y multiplicamos ese número por el punto G, esto es, $kG = 205887(241479,71146) = (45235,155942)$.

Paso 5. Sumando la codificación de la letra c $+ k(llavePB)$ se obtiene:
$(12179,161435) + 205887(339894,358573)$
$= (12179,161435) + (237547,189319)$
$= (493092,311065)$.

Paso 6. Se unen los resultados obtenidos en los pasos 4 y 5 para que la pareja de coordenadas quede como: $((45235,155942), (493092,311065))$.
Repetimos estos pasos para cada uno de los caracteres restantes, se obtuvieron los siguientes resultados:

	Coordenadas Asignadas	k
c	(45235,155942,493092,311065)	205887
a	(464947,454208,261541,436656)	425387
r	(1424,194779,211919,45374)	294652
l	(91384,133847,124363,495034)	258306
o	(22940,216821,391383,182458)	99447
s	(30055,298941,323275,69773)	340580
1	(220188,341581,147194,45165)	162281
2	(254288,121557,358833,451256)	470133
3	(310143,142529,222099,59930)	92723

Al tener todos los puntos cifrados solo bastará con mandarlos a una base de datos, como se muestra en la Figura 4.

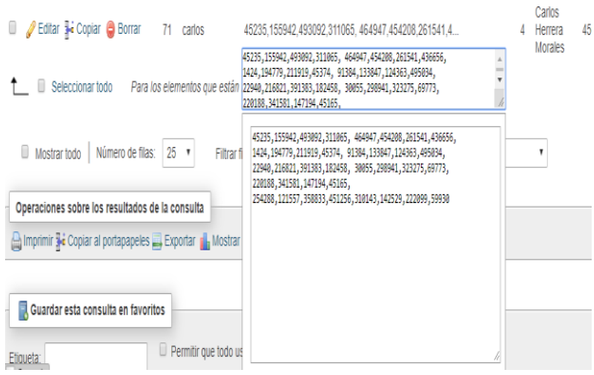


Figura 4 Puntos almacenados en una base de datos

El usuario al ser registrado puede ingresar a su información mediante su usuario y contraseña.

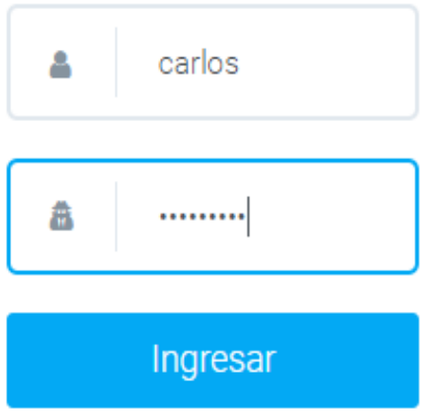


Figura 5 Datos del usuario

Ahora debemos verificar que el usuario ingreso la contraseña correcta para ello debemos descifrar esa información.

Paso 1. Descifraremos el primer punto que son $(45235,155942,493092,311065)$, tomando la primera pareja y multiplicándola por la llave secreta de B, esto es, $11923(45235,155942) = (237547,189319)$.

Paso 2. Se resta el resultado del paso 1 con la segunda pareja, recordando que $(x1, y1) - (x2, y2) = (x1, y1) + (x2, -y2)$. Por lo que se obtiene: $(493092,311065) + (237547, -189319) = (12179,161435)$.

Paso 3. Se realiza la decodificación usando la fórmula $\frac{x-1}{h}$. Sustituyendo los valores tenemos: $\frac{12179-1}{123} = 99.00813$. Redondeando el resultado para que este quede solo en 99, y como sabemos el número 99 en el código ascii corresponde a la letra c, se obtiene el primer descifrado. Procedemos así para cada uno de los puntos.

Conclusiones

Se muestran los conceptos matemáticos para la realización de los algoritmos de cifrado y descifrado usando criptografía en curvas elípticas.

Dado que actualmente las plataformas móviles o web recaban mucha información confidencial, se observó la necesidad de aportar un método de cifrado de esta información para que no pueda ser utilizada de manera incorrecta. Al hacer uso de una programación en PHP, se propone una librería que puede ser implementada para estos propósitos.

La dificultad que se presenta es la de calcular el número de puntos que contiene una curva elíptica, para ello será necesario establecer la programación adecuada para que logremos contar con este dato de una manera eficiente y poder trabajar con valores de números primos aún más grandes. Es por ello que, como trabajo a futuro se pueda realizar la implementación del algoritmo de Schoff para este propósito.

Referencias

Amalraj, J., Raybin, J. (2016). A survey paper on cryptography techniques. IJCSMC, Vol. 5, Issue 8, 55 – 59.

Diffie, W. & Hellman, M. (1976). New directions in cryptography. IEEE transactions on Information Theory, 22(6), 644-654.

ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE transactions on information Theory, 31, 469-472

Gómez, J. (2002). Criptografía y curvas elípticas. La Gaceta de la RSME, Vol. 5, 737-777.

Granados, G. (2006). Introducción a la criptografía. Revista Digital Universitaria, Vol. 7, No. 7, 1-17.

Gupta, V., Stebila, D. y Chang, S. (2004). Integrating Elliptic Curve Cryptography into the Web's Security Infrastructure. Sun Microsystems, Inc., 402-403.

Hankerson, D., Menezes, A. and Vanstone, S. (2004). Guide to Elliptic Curve Cryptography. Springer-Verlag New York, Inc.

Koblitz, N. (1987). Elliptic curve in cryptography. American Mathematical Society J. Comput. Math., 207–209.

Miller, V. S. (1986). Use of elliptic curves in cryptography. In Lecture notes in computer sciences; 218 on Advances in cryptology. CRYPTO 85, USA. Springer-Verlag New York, Inc., 417– 426,

Kawahara, Y. Takagi, T. and Okamoto E. (2006). Efficient Implementation of Tate Pairing on a Mobile Phone Using Java. In Computational Intelligence and Security, vol. 2, 1247 - 1252, Berlin.

Katz, J., Lindell, Y. (2015) Introduction to Modern Cryptography. NY: Chapmann & Hall Book/CRC.

Schoff, N. (1995). Counting points on elliptic curves over _finite _fields, Journal de Théorie des Nombres de Bordeaux 7, 219-254.

ISSN-2531-2952