Visualización y Estudio de un Ataque en Criptografía de Curva Elíptica

Óscar Cigala-Álvarez
Universidad de La Laguna
La Laguna, Tenerife
ocigalaa@ull.edu.es

Luis Chinea-Rangel
Universidad de La Laguna
La Laguna, Tenerife
alu0101118116@ull.edu.es

Pino Caballero-Gil
Universidad de La Laguna
La Laguna, Tenerife
pcaballe@ull.edu.es

***Resumen*—Este trabajo presenta una implementación visual de algunos conceptos básicos de Criptografía de Curva Elíptica (ECC, *Elliptic Curve Cryptography*), incluyendo un ataque basado en la resolución del Problema del Logaritmo Discreto Elíptico (ECDLP, *Elliptic Curve Discrete Logarithm Problem*). Esta implementación ha sido desarrollada fundamentalmente con un propósito didáctico. El ataque analizado parte de la hipótesis de que se utilizan dispositivos criptográficos de caja negra distribuidos por un fabricante, y aprovecha algunas vulnerabilidades de las implementaciones habituales de la ECC, para permitir que un atacante pueda usar una puerta trasera colocada de forma premeditada por el fabricante del sistema criptográfico para extraer información privada a partir del dispositivo contaminado.**

***Index Terms*—Criptografía, Cleptografía, ECC, SETUP, ECDLP, ECDH, ECEG**

Tipo de contribución: *Investigación original/ Formación e innovación educativa/ Investigación en desarrollo*

I. Introducción

La Criptografía de Curva Elíptica o ECC [1] es criptografía asimétrica basada en curvas elípticas sobre cuerpos finitos [2]. A diferencia de los sistemas criptográficos basados en la factorización de grandes números enteros o en los logaritmos discretos, la seguridad de la ECC se deriva de la dificultad del problema del logaritmo discreto elíptico. Lo que hace especialmente atractiva a la ECC es su capacidad para ofrecer con claves más cortas un nivel de seguridad comparable al de sistemas criptográficos basados en otros problemas matemáticos usando claves más largas. Por ese motivo, la ECC se usa en numerosas tecnologías como Whatsapp, Messenger, navegación segura con TLS, voto electrónico, criptomonedas como Bitcoin, tecnologías *blockchain* para *smart contracts* como Ethereum, y cada vez en más sistemas.

De manera simplificada, se puede definir una curva elíptica como una curva plana sobre un cuerpo finito que consta de los puntos que satisfacen la ecuación: $y^2 = x^3 + ax + b$ junto con el punto del infinito, donde a y b son coeficientes del cuerpo finito que cumplen la condición de que $4a^3 + 27b^2 \neq 0$ para asegurar que la curva no tenga puntos singulares (i.e. vértices o intersecciones con sí misma). La ECC se basa en operaciones con los puntos de curvas elípticas, como la suma, para realizar operaciones criptográficas como el cifrado, el descifrado, la firma digital y la verificación de firma.

Los cuerpos sobre los cuales se definen las curvas elípticas pueden ser de dos tipos principales: cuerpos Fp con p un número primo y cuerpos binarios F_{2^m} de tamaño 2^m. La elección del cuerpo afecta tanto a la seguridad como a la eficiencia del criptosistema. En la herramienta diseñada se utiliza un cuerpo Fp con p un número primo.

La seguridad de ECC se basa en la dificultad de resolver el problema del logaritmo discreto elíptico, que consiste en determinar, dado un punto P y otro punto $Q = kP$, el entero positivo k. Este problema es computacionalmente difícil de resolver, lo que hace que los ataques contra sistemas criptográficos basados en ECC sean impracticables con la tecnología actual, siempre y cuando se elijan adecuadamente los parámetros de la curva, como el tamaño del cuerpo y los coeficientes a y b.

La eficiencia de ECC se traduce en que, para un nivel de seguridad comparable, las claves generadas son significativamente más cortas que en sistemas de clave pública como RSA. Esto implica un menor consumo de recursos, como ancho de banda y almacenamiento, lo cual es particularmente valioso en dispositivos con recursos limitados como tarjetas inteligentes y dispositivos móviles.

Este trabajo se estructura como sigue. La sección II incluye algunos preliminares necesarios sobre ECDLP, aritmética de puntos, cleptografía y ataque SETUP básico. La sección III describe en detalle la herramienta de visualización de ECC implementada, llamada PANDORA. La sección IV da las bases del ataque SETUP contra el ECDLP, incluyendo su implementación. Finalmente, la sección V cierra el trabajo con algunas conclusiones y trabajos futuros.

II. Preliminares

II-A. ECDLP

El Problema del Logaritmo Discreto Elíptico o ECDLP [3] constituye la base esencial de la ECC, ya que permite a dos partes establecer de manera segura una clave compartida sin necesidad de un canal de comunicación seguro. De hecho, la dificultad del ECDLP proporciona el nivel de seguridad necesario tanto para el esquema de intercambio de claves de Diffie-Hellman de Curva Elíptica (ECDH, Elliptic Curve Diffie-Hellman) [4], [5], como para el Cifrado de ElGamal Elíptico (ECEG, Elliptic Curve ElGamal) [6] (ver Figura 1).

Utilizando un punto G sobre una curva elíptica E definida en un cuerpo finito Fp, tras la elección aleatoria de dos números secretos d_A y d_B por las partes A y B que interactúan, se calculan los puntos d_AG y d_BG. El problema que da soporte a ECDH y ECEG, consistente en calcular el punto compartido d_Ad_BG, sin conocer d_A y d_B, resulta ser un desafío computacionalmente inviable, lo que garantiza la seguridad de ambos algoritmos.

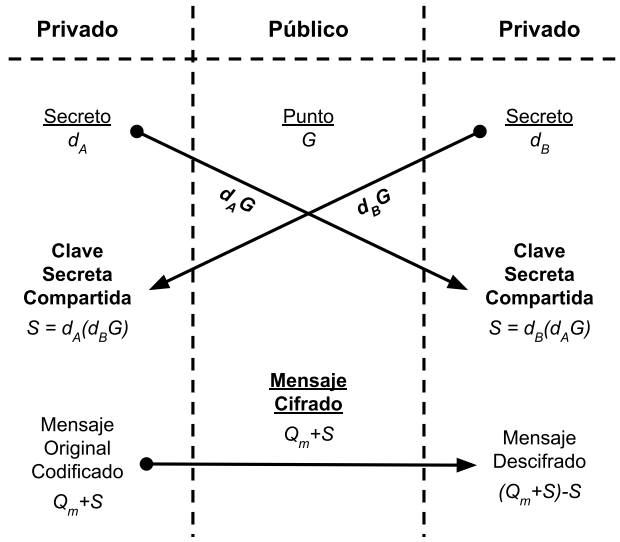

Figura 1. Intercambio de claves ECDH

En ambos esquemas ECDH y ECEG, las partes involucradas eligen de forma conjunta y pública una curva elíptica E sobre un cuerpo finito Fp, y un punto base G sobre E. A partir de ese punto base, cada usuario genera un punto público enviando d_AG o d_BG a la otra parte, siendo d_A y d_B enteros seleccionados al azar que se mantienen secretos. Este intercambio permite a ambas partes, mediante operaciones aritméticas sobre la curva, llegar al mismo punto d_Ad_BG, que es la clave compartida para un cifrado simétrico.

II-B. Aritmética de puntos

Las operaciones fundamentales en ECC son la adición, duplicación y multiplicación escalar de puntos [7], que se rigen por las siguientes reglas:

- Dados dos puntos P y Q en una curva E, la suma de P y Q da como resultado otro punto R de la curva.

$$R = P + Q \quad \text{donde} \quad R, P, Q \in E$$

- Dado un punto P, la duplicación $2P$ es el resultado de sumar P consigo mismo.

$$2P = P + P$$

- Dado un punto P y un escalar k, la multiplicación kP es el resultado de sumar P consigo mismo k veces.

Al sumar dos puntos de una curva elíptica se pueden distinguir dos casos dependiendo de si los puntos son iguales o distintos. Las dos definiciones de suma en cada uno de los casos conducen a distintas fórmulas. Cuando los puntos son distintos, se utiliza una fórmula que involucra las coordenadas x e y de los puntos P y Q y puede expresarse como:

$$\lambda = \frac{y_Q - y_P}{x_Q - x_P}$$

$$x_R = \lambda^2 - x_P - x_Q$$

$$y_R = \lambda(x_P - x_R) - y_P$$

En el caso en que los puntos sean iguales, es decir, cuando se está haciendo la duplicación de un punto pues se está calculando $2P$, se utiliza una fórmula diferente que implica el cálculo de la tangente a la curva en el punto P, dada por:

$$\lambda = \frac{3x_P^2 + a}{2y_P}$$

$$x_R = \lambda^2 - 2x_P$$

$$y_R = \lambda(x_P - x_R) - y_P$$

Cabe destacar que para todas las operaciones de puntos se realiza aritmética modular en base al cuerpo finito F_p.

II-C. Cleptografía

La cleptografía es una rama de la criptografía que se ocupa de la inserción y uso de estructuras ocultas en sistemas criptográficos para filtrar información secreta. Esta disciplina, que surgió del trabajo de Young y Yung [8], se centra en cómo los atacantes pueden diseñar e implementar un mecanismo de caja negra llamado SETUP (Secretly Embedded Trapdoor with Universal Protection) [9] para capturar claves privadas de un usuario y transmitirlas a un atacante sin alterar el funcionamiento aparente del sistema criptográfico en cuestión. En caso de ataque cleptográfico, el sistema criptográfico comprometido puede seguir funcionando con total normalidad, creando una falsa sensación de seguridad, mientras que un atacante puede acceder en secreto a información confidencial. Este enfoque frustra el propósito original del cifrado, el cual es proteger la información.

Los ataques SETUP [10] son particularmente sigilosos debido a que no pueden ser detectados por los usuarios y diseñadores del sistema, a menos que se conozca la existencia específica del ataque y se busque activamente. Además, la sofisticación de estos ataques los hace particularmente peligrosos, ya que no requieren una violación explícita de la seguridad del sistema para tener éxito. En cambio, utiliza las sutilezas matemáticas de los algoritmos de cifrado, como el problema del logaritmo discreto, para filtrar paquetes o datos de manera que el atacante pueda interpretar y utilizar la información filtrada.

III. Propuesta

III-A. Software Pandora

En el contexto de la seguridad informática, el estudio y comprensión de las propiedades inherentes a las curvas elípticas es fundamental para la implementación y evaluación de sistemas de seguridad. Para abordar esta necesidad, se ha desarrollado Pandora, una aplicación web innovadora que no solo facilita la investigación en criptografía de curvas elípticas, sino que también tiene aplicaciones educativas significativas.

La herramienta, construida utilizando el framework Flask y escrita en Python, se presenta como una plataforma accesible disponible públicamente en [11].

III-B. *Funciones disponibles*

En la actualidad el software cuenta con las siguientes funcionalidades:

- Configuración y visualización de curvas elípticas con parámetros variables.
- Generación de claves públicas y compartidas a partir de claves secretas a elección del usuario.
- Codificación y cifrado de mensajes de longitud variable.
- Desencriptado y decodificación de mensajes de longitud variable.
- Configuración del alfabeto a utilizar en el encriptado, contando con opciones por defecto como la elección de idiomas y sistemas numéricos, entre otros.
- Ataque SETUP a curva elíptica.

III-C. *Características del software*

Pandora proporciona una simulación en tiempo real del intercambio de mensajes ECDH entre dos entidades, con una curva elíptica personalizable. Una de sus características principales es la posibilidad de representar gráficamente la curva elíptica y los puntos (ver Figuras 2 y 3), lo que facilita la comprensión visual de los conceptos básicos de curvas elípticas.

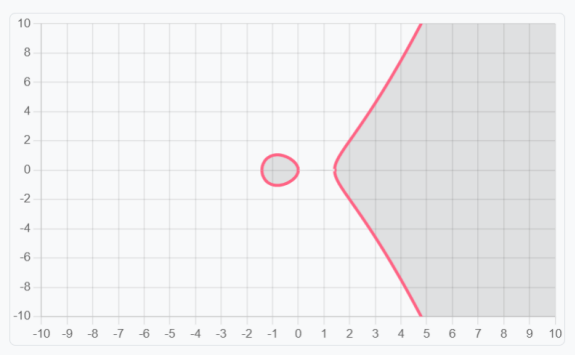

Figura 2. Visualización de la curva $E : y^2 = x^3 - 2x$ en Pandora

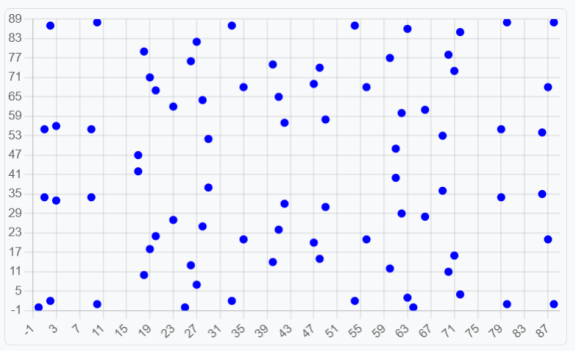

Figura 3. Puntos de la curva E en el cuerpo finito F_{89} usando Pandora

La interfaz gráfica interactiva de Pandora permite explorar diversas configuraciones y escenarios, lo que la convierte en una útil herramienta didáctica. Puede emplearse para demostrar visualmente algunas características y operaciones fundamentales de las curvas elípticas, así que tiene potencial para ser utilizada en entornos académicos en cursos de criptografía y seguridad informática.

III-D. *ECDH en Pandora*

Para ilustrar el funcionamiento de Pandora se considera a continuación un ejemplo de intercambio de claves utilizando el algoritmo ECDH. En este ejemplo se supone que ambos participantes acuerdan como parámetros públicos comunes: la curva $E : y^2 = x^3 - 2x$, el cuerpo finito F_{89} y el punto base $G = (29, 37)$ (ver Figura 4). A partir de ahí se ejecuta lo siguiente (ver Figura 5):

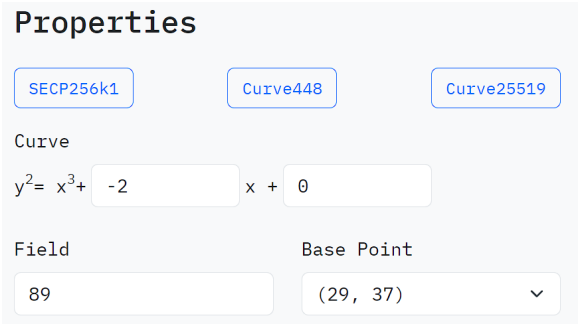

Figura 4. Introducción de datos en Pandora

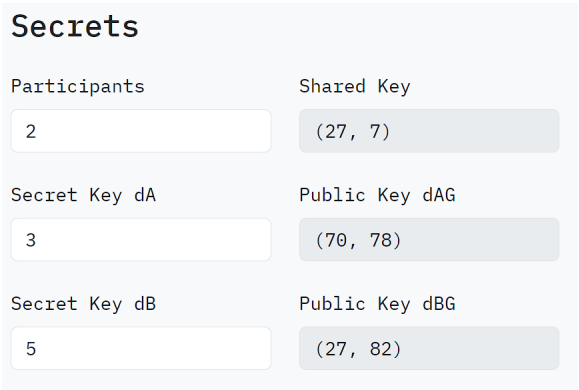

Figura 5. Claves privadas y públicas de A y B en Pandora

- *Paso 1:* Alice escoge su clave privada $d_A = 3$ como número entero elegido aleatoriamente. Luego, calcula su clave pública multiplicando su clave privada por el punto base G:

$$d_A \cdot G = 3 \cdot (29, 37) = (70, 78)$$

y comparte este valor con Bob.
- *Paso 2:* Bob selecciona de la misma forma su clave privada $d_B = 5$. Luego, calcula su clave pública multiplicando su clave privada por el punto base G:

$$d_B \cdot G = 5 \cdot (29, 37) = (27, 82)$$

y comparte este valor con Alice.
- *Paso 3:* Una vez ambos han intercambiado sus claves públicas, ambos pueden calcular de forma independiente la clave secreta compartida S. Para hacerlo, cada uno

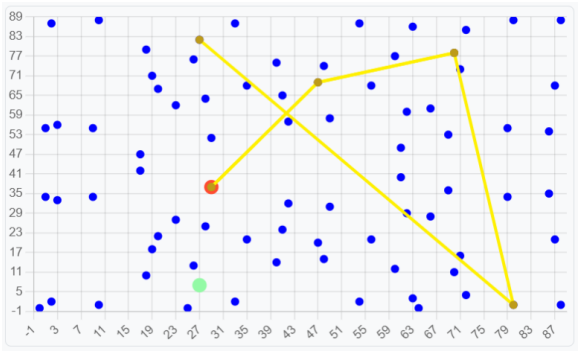

Figura 6. Punto base (en rojo) y clave secreta compartida (en verde)

multiplica su clave privada por la clave pública del otro. Así, Alice calcula:

$$S = d_A \cdot (27, 82) = 3 \cdot (27, 82) = (27, 7)$$

mientras Bob calcula:

$$S = d_B \cdot (70, 78) = 5 \cdot (70, 78) = (27, 7)$$

Como se muestra en la Figura 6, se puede observar el proceso mediante el cual se van generando en este ejemplo las claves públicas y la clave secreta compartida utilizando la herramienta Pandora. El punto G (marcado en rojo) se utiliza como base para obtener los múltiplos necesarios, y la clave secreta compartida S generada se marca en verde.

III-E. ECEG en Pandora

Para enviar un mensaje utilizando ECEG, primero es necesario codificar el mensaje antes de cifrarlo. A continuación, se muestra un ejemplo didáctico de cómo se puede realizar este proceso con Pandora, utilizando parámetros concretos:

- Se elige una curva elíptica E definida por la ecuación $y^2 = x^3+2x-1$ sobre el cuerpo finito F_{89}, donde $p = 89$ es un número primo, y con punto base $G = (6, 82)$.
- Se establecen los parámetros para los usuarios involucrados de manera que cada uno elige una clave privada, $d_A = 4$ para Alice y $d_B = 3$ para Bob.
- Se define el alfabeto del mensaje a transmitir. En el ejemplo mostrado en la Figura 7 se utiliza el alfabeto español representado como $A = \langle$A, B, C, ..., Ñ, ..., Z$\rangle$. En este ejemplo, cada letra del mensaje es codificada como un punto en la curva elíptica E.

Una vez se han establecido los parámetros, se procede a codificar y cifrar el mensaje. Para codificar cada carácter del mensaje, se calcula el punto en la curva elíptica que corresponda al carácter en el alfabeto definido. Esto se logra por ejemplo utilizando un esquema de codificación que asigna a cada carácter que ocupa la posición m en el alfabeto A, un punto único Q_m de la curva E, de forma que Q_m tiene como coordenada x el menor entero positivo tal que $x = mh + j \in E$, siendo $h = \left\lfloor \frac{p}{|A|} \right\rfloor$ y j un entero positivo que se incrementa partiendo de 0 hasta obtener la coordenada de un punto válido. Una vez calculada la coordenada x, se toma como coordenada y la menor posible. Una vez que se ha codificado el carácter, se procede a cifrarlo.

Supóngase que Alice desea enviar a Bob cifrada con ECEG el mensaje original COMA, que contiene la letra C del alfabeto español, correspondiente a $m = 2$.

$$h = \left\lfloor \frac{89}{|27|} \right\rfloor = 3$$

$$Q_m = 2 * 3 + 0 = 6$$

$$(x, y) = (6, y) \in E$$

Dado que existe un punto con $x = 6$ en la curva E, se escoge la menor coordenada $y = 7$ como coordenada del punto correspondiente a m.

$$(6, 7) \in E$$

Una vez codificado m como punto, el cifrado implica calcular $Q_m + d_A(d_B G)$, a partir de la clave pública de Bob $d_B G$ y la clave privada de Alice d_A.

Este proceso se repite para cada uno de los caracteres del mensaje COMA, tal como se muestra en la Figura 8.

En resumen, la herramienta Pandora desarrollada ofrece una solución útil para la formación básica en ECC, permitiendo tanto la experimentación práctica como la comprensión teórica de algunos conceptos fundamentales en este campo.

IV. Ataque analizado

IV-A. Ataque SETUP al ECDLP

El ataque SETUP en el contexto del ECDLP implica la implementación de un mecanismo oculto dentro de un dispositivo criptográfico que utiliza curvas elípticas para la seguridad de las comunicaciones [10], [12]. Este mecanismo secreto está diseñado para aprovechar la estructura matemática de las curvas elípticas y las operaciones criptográficas utilizadas en ECC para filtrar información secreta, específicamente las claves privadas de los usuarios, de forma que el atacante se hace con ellas de manera encubierta. La eficacia del ataque se basa en la dificultad de resolver ECDLP, consistente en encontrar el número entero x, dado un punto P y otro punto $Q = xP$ de una curva E conocida sobre un cuerpo finito F_p, que, como ya se ha mencionado, es un problema computacionalmente intratable, es decir, no se conocen algoritmos rápidos que permitan resolverlo con los ordenadores actuales.

El ataque SETUP al ECDLP se basa en que el dispositivo comprometido permite realizar operaciones criptográficas normales, como la generación de claves o el intercambio de claves, utilizando un punto base G y generando puntos en la curva como dG, donde d es la clave privada. Sin embargo, el mecanismo de trampa modifica estas operaciones para incluir un paso adicional que calcula un valor especial Z usando la clave privada del usuario junto a la clave pública del atacante $V = vG$ definida a partir de la clave privada del atacante v. Este valor Z se utiliza para generar un nuevo punto en la curva, que se envía como parte del mensaje o de la operación de intercambio de claves. Al observar estos valores modificados, el atacante puede aplicar un algoritmo específico para recuperar la clave privada del usuario a partir de la información transmitida, sin que el usuario sea consciente del robo de dicha información.

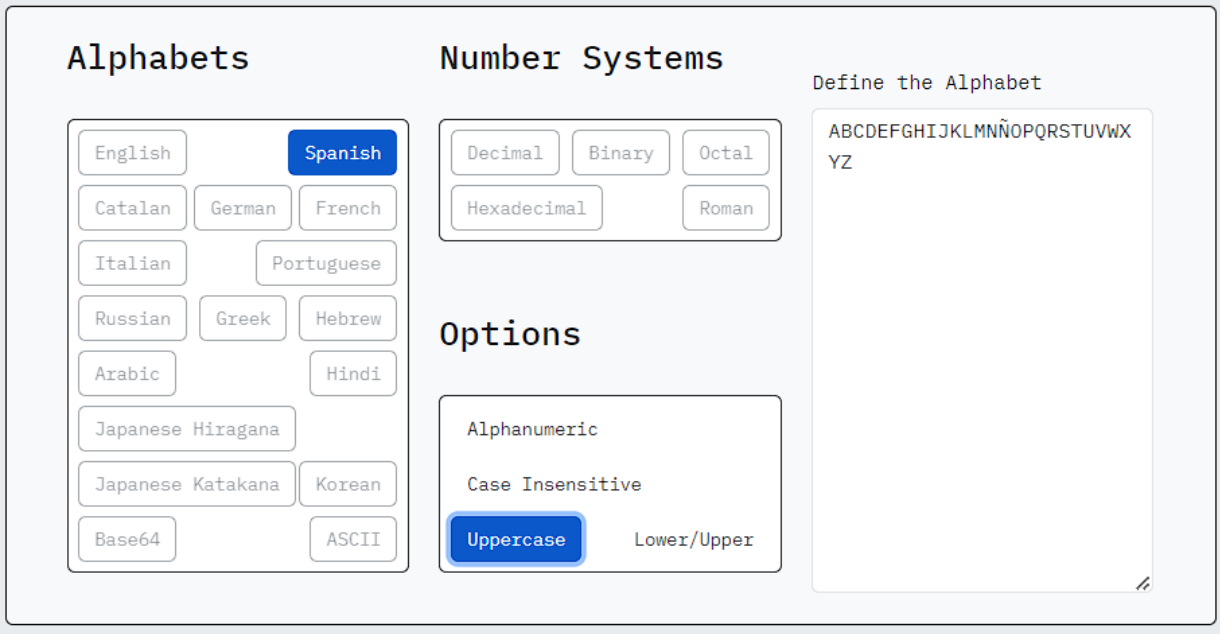

Figura 7. Ajustes de alfabeto en Pandora

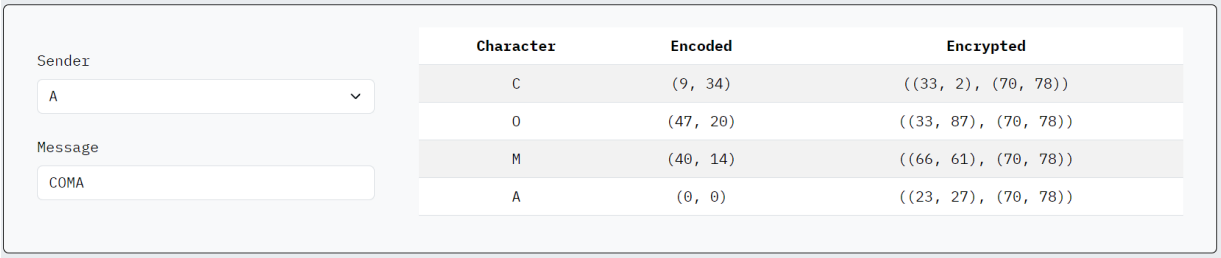

Figura 8. Codificación y Encriptación de Mensajes

IV-B. Implementación del ataque SETUP en Pandora

En la implementación realizada del ataque SETUP dentro de Pandora, se ha representado el funcionamiento y el flujo de datos mostrados en la Figura 9. A continuación, se detalla el ataque en base a diferentes algoritmos:

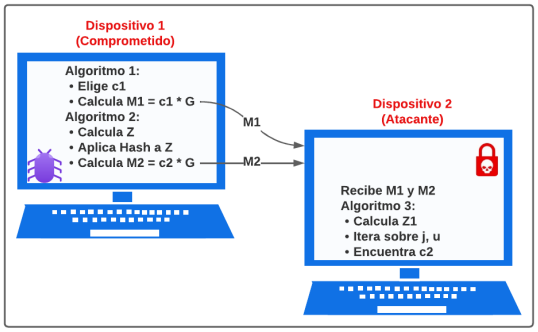

Figura 9. Esquema del ataque SETUP

- *Algoritmo 1:* Cuando el dispositivo se ejecuta por primera vez, se selecciona un número secreto c_1 de manera aleatoria dentro del rango permitido y se almacena en memoria.

$$2 \leq c_1 \leq n-1 \tag{1}$$

Este número c_1 se utiliza para generar el punto $M_1 = c_1 G$, donde G es un punto base de la curva elíptica. Este punto M_1 se envía luego como salida del dispositivo. Este paso inicial es crucial porque establece el valor inicial que será utilizado en futuras operaciones y filtraciones de información.

$$M_1 = c_1 G \tag{2}$$

- *Algoritmo 2:* Después de la primera ejecución, el dispositivo pasa a un modo de operación diferente para las ejecuciones futuras. En este modo, a partir de los valores aleatorios a, b, h, e, j y u:

$$a, b, h, e < n \tag{3}$$

$$j, u \in 0, 1 \tag{4}$$

El dispositivo calcula un nuevo punto Z utilizando una combinación de operaciones criptográficas que incluyen el punto base G, el punto $V = vG$ (donde v es la clave privada del atacante, V su clave pública correspondiente) y el valor previamente almacenado c_1, tal que:

$$Z = ac_1 G + bc_1 V + hjG + euV \tag{5}$$

El resultado de estas operaciones es Z, que luego se pasa a través de una función hash criptográficamente segura H para generar un nuevo valor secreto c_2. Este valor c_2 se almacena para uso futuro y se utiliza para generar un nuevo punto $M_2 = c_2G$, que se envía como la salida del dispositivo.

$$c_2 = H(Z) \tag{6}$$

$$M_2 = c_2G \tag{7}$$

- *Algoritmo 3:* El atacante, monitorizando la comunicación y teniendo acceso a M_1 y M_2, emplea este algoritmo para recuperar el valor secreto c_2. Este proceso implica calcular un conjunto de posibles valores de Z_1 utilizando M_1, la clave privada del atacante v, la clave pública del atacante V y variaciones de los valores aleatorios j y u, para finalmente aplicar la función hash H a estos valores hasta encontrar uno que coincida con M_2 cuando se multiplica por G. Esto permite al atacante descifrar el valor c_2 sin que el usuario se dé cuenta de la filtración de su clave secreta.

$$Z_1 = aM_1 + bvM_1 = aM_1 + bvc_1G = ac_1G + bc_1V \tag{8}$$

Para cada valor posible de j, u:

$$Z_2 = Z_1 + hjG + euV \tag{9}$$

$$c_2 = H(Z_2) \tag{10}$$

Si $c_2G = M_2$, se devuelve como salida c_2.

Con estos algoritmos se logra implementar un ataque SETUP al ECDLP que permite a un atacante extraer claves secretas de un sistema criptográfico sin levantar sospechas. Así, utilizando una combinación de técnicas de ECC y funciones hash seguras, es posible crear un canal oculto que permite filtrar sistemáticamente información secreta del atacado, remitiéndola hacia el atacante.

IV-C. Pruebas realizadas al ataque SETUP en Pandora

Para probar la eficacia del algoritmo, se han realizado pruebas con diferentes curvas elípticas. Una de las curvas más relevantes en la actualidad es SECP256k1 (ver Figura 10), ampliamente utilizada en criptomonedas como Bitcoin. Se ha escogido esta curva por su relevancia y el impacto que tendrían posibles brechas de seguridad, además de para usarla como punto de comparación con el resto de curvas probadas.

Curva	Intentos	Éxitos	% Éxito
SECP256k1	50	50	100.0 %
Curva 2	50	31	62.0 %
Curva 3	50	24	48.0 %
Curva 4	50	50	100.0 %
Curva 5	50	31	62.0 %

Tabla I
PRUEBA DE RENDIMIENTO DEL ATAQUE SETUP CON PANDORA

Como se puede observar en la Tabla I, la tasa de éxito del ataque es significativamente alta, alcanzando un 100 % de efectividad en la curva SECP256k1. Sin embargo, es importante considerar alternativas para mejorar la seguridad en sistemas que dependen de estas curvas elípticas.

V. CONCLUSIONES Y TRABAJOS FUTUROS

En este trabajo se ha presentado una nueva herramienta de fácil uso, desarrollada con un objetivo didáctico para reducir la curva de aprendizaje de ECC al crear una aplicación que permite entender los conceptos de forma muy visual. Por otro lado, se ha demostrado que se puede montar un ataque SETUP contra el intercambio de claves ECDH y el cifrado ECEG si un fabricante malintencionado de criptosistemas de caja negra implementa una puerta trasera de este tipo en sus dispositivos.

Este ataque destaca por su sofisticación y potencial sigiloso al poder ser implementado incluso dentro de sistemas que, en apariencia, funcionan de manera segura y conforme a las especificaciones. Esto refuerza la importancia de la investigación y el desarrollo continuo en materia de criptografía e implementación segura de los criptoalgoritmos, para mitigar en la medida de los posible este tipo de ataques.

Respecto a trabajos futuros, actualmente se continúa trabajando en la herramienta visual con el objetivo de añadir, entre otras, las siguientes funciones y características:

- Implementación de otros tipos de ataques a ECC como Pollig-Hellman, Baby-step Giant-step, Pollard's Rho Attack.
- Operaciones de firma y validación de mensajes.
- Persistencia de datos y configuraciones para los usuarios de Pandora.
- Adición de curvas elípticas preestablecidas ampliamente usadas y nuevas opciones de envío de mensajes.
- Mejorar la GUI de Pandora pensando en la usabilidad y accesibilidad.

AGRADECIMIENTOS

Este trabajo ha sido posible gracias a las Cátedras de Ciberseguridad de la Universidad de La Laguna patrocinadas por Binter, y por INCIBE en el marco de los fondos del Plan de Recuperación, Transformación y Resiliencia, financiada por la Unión Europea (Next Generation). Además forma parte del proyecto PID2022-138933OB-I00 financiado por MCIN/AEI/ 10.13039/501100011033/FEDER, UE.

REFERENCIAS

[1] Koblitz, N., Menezes, A., Vanstone, S.: "The state of elliptic curve cryptography", *Designs, codes and cryptography*, vol. 19, no. 6, pp. 173-193, 2000.
[2] Koblitz, N.: "Elliptic curve cryptosystems", *Mathematics of computation*, vol. 48, no. 177, pp. 203-209, 1987.
[3] Menezes, A., Menezes, A.: "The Discrete Logarithm Problem", *Elliptic Curve Public Key Cryptosystems*, pp. 49-59, 2005.
[4] Diffie, W., Hellman, M.E.: "New Directions in Cryptography", *IEEE transactions on Information Theory*, vol. 22, no. 6, pp. 644-654, 1976.
[5] Hwang, R.J., Lai, C.H., Su, F.F.: "An efficient signcryption scheme with forward secrecy based on elliptic curve", *Applied Mathematics and computation*, vol. 167, pp. 870-881, 2005.
[6] Rabah, K.: "Elliptic curve elgamal encryption and signature schemes", *Information Technology Journal*, vol. 4, no 3, p. 299-306, 2005.
[7] Roy, M., Deb, N., Kumar, A.J.: "Point generation and base point selection in ECC: An overview", *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 3, no. 5, pp. 6711-6713, 2014.
[8] Young, A., Yung, M.: "Kleptography: Using cryptography against cryptography", *Advances in Cryptology—EUROCRYPT'97: International Conference on the Theory and Application of Cryptographic Techniques*, pp. 62-74, 1997.

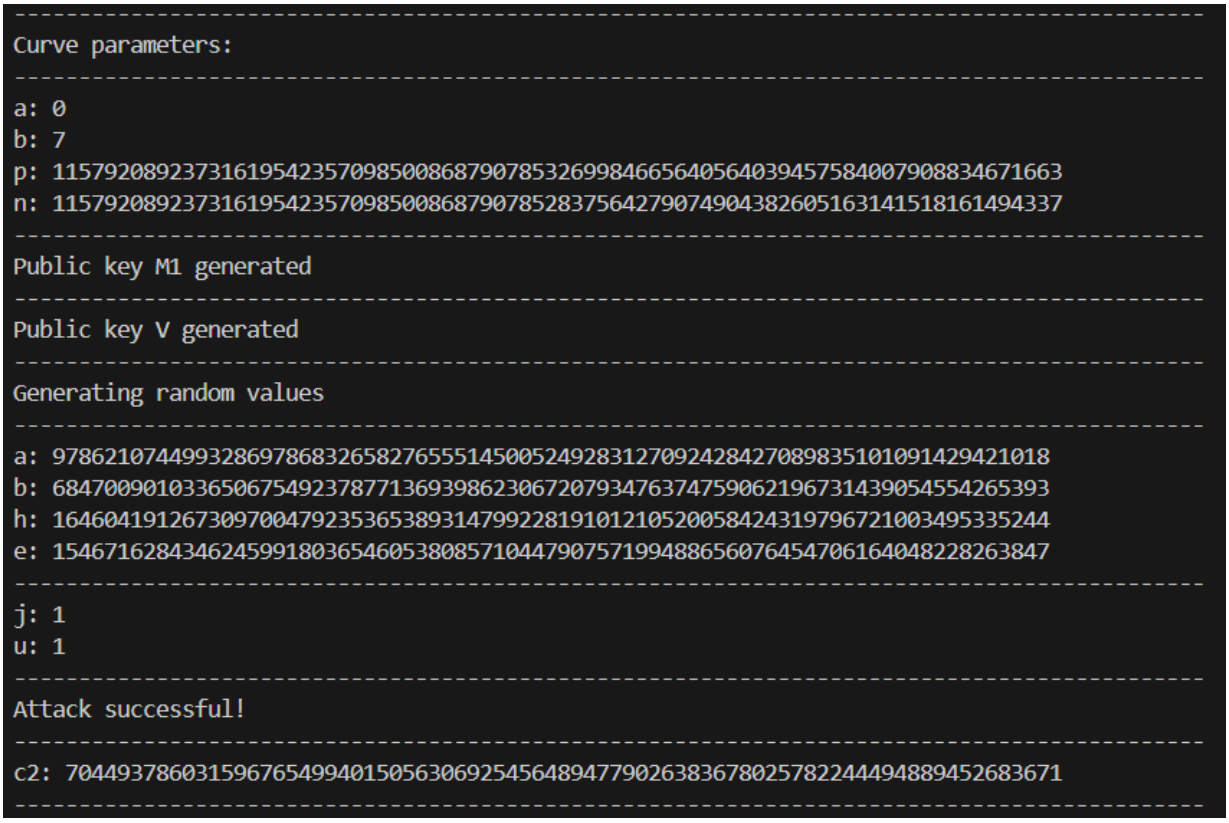

Figura 10. Prueba de ataque SETUP con Pandora

[9] Young, A., Yung, M.: “The Dark Side of Black-Box Cryptography or Should We Trust Capstone?”, *IEEE Transactions on Information Theory*, pp. 89-103, 1996.
[10] Mohamed, E., Elkamchouchi, H.: “Elliptic Curve Kleptography”, *Elliptic Curve Kleptography*, vol. 10, no. 6, pp. 183-185, 2010.
[11] Cigala-Álvarez, Ó., Chinea-Rangel, L.: https://github.com/iluzioDev/pandora
[12] Sajjad, A., Afzal, M., Iqbal, M.M.W., Abbas, H., Latif, R., Raza, R.A.: “Kleptographic attack on elliptic curve based cryptographic protocols”, *IEEE Access*, no. 8, pp. 139903-139917, 2020.