



Universidad Nacional Autónoma de México
Facultad de Ciencias



Criptografía y Seguridad 2025-1

Profesora: Anayanzi Delia Martínez Hernández

Ayudantes:

Cecilia del Carmen Villatoro Ramos

Ivan Daniel Galindo Perez

José Angel Arévalo Avalos

David Armando Silva de Paz

Práctica 10: Curvas elípticas y Protocolo de Diffie-Hellman

BlackRaven



Integrantes:

Carlos Daniel Cortés Jiménez 420004846

Edgar Montiel Ledesma 317317794

Jonathan Martínez Camarillo 318042989

11 de Noviembre 2024

Introducción

En esta práctica veremos el funcionamiento y como implementar curvas elípticas para el cifrado y descifrado de mensaje, el uso del protocolo Diffie-Hellman, se verán ejemplos del funcionamiento para entender el proceso de creación y manejo de llaves públicas y privadas entre las entidades es decir como realizan el intercambio de llaves, la comparativa entre este nuevo como de cifrado en comparación con RSA, y el nivel de seguridad que tiene ante ataques.

Desarrollo.

El archivo diferencial se adjuntara junto con el pdf, a continuación se muestra que el código pasa todas las pruebas:

```
carlos@carlos-G7-7588:~/Descargas/semestre 2025-1/Criptografia y Seguridad/Practicas/BlackRaven-Criptografia/Practicas/Practica10
/p10/elliptic_curves/src$ python3 -m unittest discover -v -s test/
test_cofactor (testEllipticCurve.TestEllipticCurve) ... ok
test_init (testEllipticCurve.TestEllipticCurve) ... ok
test_inv (testEllipticCurve.TestEllipticCurve) ... ok
test_is_in_curve (testEllipticCurve.TestEllipticCurve) ... ok
test_mult (testEllipticCurve.TestEllipticCurve) ... ok
test_order (testEllipticCurve.TestEllipticCurve) ... ok
test_points (testEllipticCurve.TestEllipticCurve) ... ok
test_str (testEllipticCurve.TestEllipticCurve) ... ok
test_sum (testEllipticCurve.TestEllipticCurve) ... ok
test_cifrar_descifrar (testEntity.TestEntity) ... ok
test_final_keys (testEntity.TestEntity) ... ok
test_genera_llaves_publicas (testEntity.TestEntity) ... ok
test_init (testEntity.TestEntity) ... ok
test_recibe_llaves_publicas (testEntity.TestEntity) ... ok
test_str (testEntity.TestEntity) ... ok
test_eq (testPoint.TestPoint) ... ok
test_infinite_point (testPoint.TestPoint) ... ok
test_init (testPoint.TestPoint) ... ok
test_repr (testPoint.TestPoint) ... ok
test_set (testPoint.TestPoint) ... ok
test_str (testPoint.TestPoint) ... ok

-----
Ran 21 tests in 0.014s

OK
```

Preguntas

Contesta las siguiente preguntas, investigando o con tus propias palabras.

1. Cuando sumamos un punto P con $-P$, que nos da como resultado el punto al infinito... ¿Qué quiere decir que un punto sea infinito?

En las curvas elípticas, cuando nos referimos "punto al infinito" (denotado como O) es un concepto matemático abstracto.

Cuando hablamos del punto al infinito en curvas elípticas, no nos referimos a un punto que está "muy lejos".^{en} el sentido usual, es al algo más que eso.

Puede actuar como un elemento neutro en la operación de grupo de la curva elíptica. Esto significa que, para cualquier punto P en la curva:

- $P + O = P$ (para cualquier punto P)
- $O + P = P$
- $P + (-P) = O$
- $O + O = O$

Geométricamente, cuando sumamos P y $-P$, la línea que une estos puntos es vertical, esta línea "inter-seca" la curva en el punto al infinito, es como si continuara hacia arriba indefinidamente, Esto sucede porque $-P$ es el reflejo de P respecto al eje x , y cuando los sumamos, las tangentes a la curva se anulan, proyectándose fuera de cualquier coordenada finita.

2. Cuando sumamos un punto P consigo mismo, ¿qué característica tiene la recta que conecta a estos 2 puntos?

Esta respuesta nos basamos en el link que se puede encontrar en referencias del pdf **Ataques de Tiempo a Rutinas Criptográficas sobre Curvas Elípticas, pag. 18 capítulo 3**

Si P y Q son el mismo punto, la línea recta que los conecta será una tangente a la curva en ese punto. Para obtener el punto de intersección R , simplemente reflejas el punto de intersección sobre el eje x .

Si sumamos dos veces P lo que estamos obteniendo es una duplicación de puntos, para este caso la recta es la tangente a la curva elíptica en el punto P .

Esta tangente es única para cada punto P , la recta que "toca" la curva en un solo punto (el punto P), además representa la "pendiente instantánea" de la curva en ese punto.

Para encontrar esta pendiente de la línea tangente se calcula usando la derivada de la ecuación de la curva en el punto P , la recta intersectará la curva en otro punto y el reflejo de este punto de intersección sobre el eje x será $2P$

3. ¿En qué otro cifrado se utiliza el protocolo Diffie-Hellman?

El intercambio de claves Diffie-Hellman se puede utilizar para establecer claves públicas y privadas. Sin embargo, en la práctica, se tiende a utilizar RSA en su lugar. Esto se debe a que el algoritmo RSA también es capaz de firmar certificados de clave pública, mientras que el intercambio de claves Diffie-Hellman no lo es.

El algoritmo ElGamal, que se usó mucho en PGP, se basa en el intercambio de claves Diffie-Hellman, por lo que cualquier protocolo que lo use está implementando efectivamente una especie de Diffie-Hellman.

Como uno de los métodos más comunes para distribuir claves de forma segura, el intercambio de claves Diffie-Hellman se implementa con frecuencia en protocolos de seguridad como:

- SSL (Secure Socket Layer)
- TLS (Transport Layer Security)
- IPsec
- SSH (Secure Shell)
- VPN (Virtual Private Network)
- PGP
- Tor
- Off-the-Record Messaging (OTR)

y muchos otros. Esto lo convierte en una parte integral de nuestras comunicaciones seguras.

4. ¿Cuántos bits de una llave de una curva necesitamos para poder igualar la seguridad en una llave de RSA?

Para poder igualar la seguridad de una llave RSA, las curvas elípticas necesitan de una cantidad de bits relativamente menor esto se debe a su eficiencia en seguridad, entonces tenemos lo siguiente de curva elíptica vs RSA:

- 224 bits \equiv 2048 bits
- 256 bits \equiv 3072 bits
- 384 bits \equiv 7680 bits
- 512 bits \equiv 15360 bits

5. En el ejemplo, en attack tenemos que la letra 't' es cifrada a 2 puntos diferentes de la curva, y sigue siendo recuperable. ¿qué valor hace esto posible?

El carácter t puede ser mapeado a diferentes puntos de la curva, el valor que permite esta variabilidad es el uso de valores aleatorios cuando se cifra por eso el uso de random para Entity, este valor único se incluye en cada operación de cifrado y modifica el resultado sin cambiar el mensaje descifrado, la clave privada puede recuperar el mensaje original independientemente del valor usado, por lo que es aquí que hace que el carácter se pueda cifrar de forma diferente.

6. Cifra el mensaje: Perro salchicha, gordo bachicha usando la curva elíptica $EC_{233}(-2, 8)$ usando los 256 caracteres del código ASCII. Pega la ejecución de la salida del main

Tenemos lo siguiente:

```
carlos@carlos-G7-7588:~/Descargas/semestre 2025-1/Criptografia y Seguridad/Practicas/BlackRaven-Criptografia/Practicas/Practica10
/p10/elliptic_curves/src$ python3 main.py
CREACIÓN DE ENTIDADES CON CURVAS

Alice:
EC: y^2 = x^3 + -2x + 8 mod 233
G: (232, 230)
Private Key: 114
Private Point: (204, 29)

Bob:
EC: y^2 = x^3 + -2x + 8 mod 233
G: (232, 230)
Private Key: 177
Private Point: (158, 180)

INTERCAMBIO DE LLAVES TIPO DIFFIE-HELLMAN
Ronda 1:
Alice PKs: [(180, 103), (142, 52)]
Bob PKs: [(23, 53), (5, 194)]

Ronda2:
Alice PKs: ((180, 103), (142, 52), (14, 188))
Bob PKs: ((23, 53), (5, 194), (206, 157))

CIFRADO DE MENSAJE
Cifrando el mensaje: Perro salchicha, gordo bachicha.

Mensaje cifrado: [(',', ' '), ('s', '6'), ('e', 'i'), ('\x8d', 'ô'), ('o', '\x7f'), ('%', 'r'), ('{', ':'), ('\x93', ' '), ('\x08', ' '), (';', 'L'), ('\x96', 'g'), ('\x8a', 'o'), ('\r', 'B'), ('É', '\x05'), ('E', '\x96'), ('E', 'p'), ('\x8f', 'e'), ('\x94', '/'), ('\x15', '\x98'), (' ', 'h'), ('\x10', 'A'), ('É', 'd'), ('"', '\x8d'), ('V', '>'), ('r', '['), ('j', 'e'), ('\x88', 'o'), ('D', 'i'), ('e', 'Ñ'), ('%', 'o'), ('», '?'), ('n', '\x93')]

Mensaje descifrado: Perro salchicha, gordo bachicha.
El mensaje fue descifrado con éxito
```

Podemos ver que el código funciona y cifra y descifra el mensaje correctamente.

Conclusión

1. **Carlos:** El curso me gustó mucho siento que es imprescindible conocer, aprender y saber como poder defenderse y como mantenerse seguro en el día a día en cuanto a la ciberseguridad, los temas y la estructura que tiene el curso me parece adecuada, pero a mi parecer falta tener más tiempo, ya que a veces por entregar las tareas o prácticas uno le interesa más entregar los trabajos, sin embargo considero que en cuanto a lo que se enseñó está muy bien, también cabe recalcar de que al haber aprendido todo lo del curso debemos tener la ética y responsabilidad de no causar daño a otros con los conocimientos adquiridos.

Otro punto a considerar diría es que se podría dar mas ejemplos o tutoriales como nuestro compañero comenta abajo, como lo fue el caso de la práctica 9 de servidores en la nube, que en mi caso si nos costó mucho trabajo y nos sentíamos perdidos continuamente, con respecto a lo demás no tengo más que añadir, diría que está muy bien.

En general dire que es un curso completo, que resulto ser de mi agrado pero muy estresante debido a los tiempos, muchas gracias por haber impartido esta materia.

2. **Jonathan:** El curso me gustó y es interesante todo lo que vimos, desde los algoritmos hasta los temas de ciberseguridad, de igual manera todo lo que se podía hacer simplemente buscando un poco en internet o preguntándole a la IA (en el caso de los malware), lo cuál nos deja con una responsabilidad de no usar lo aprendido para hacer algún daño. Quizá algo a mejorar podría ser un mejor tutorial al usar tecnologías que no se hayan usado antes, como fue el caso de los servicios en la nube. Si bien la ciberseguridad es importante y día a día se hacen avances respecto a esta, no creo profundizar más allá de leer por mera curiosidad.

3. **Edgar:** El curso me pareció pesado pero entretenido, me gustaría haberla cursado con una menor cantidad de materias para poder dedicarle más tiempo, en lo personal no creo dedicarme a la ciberseguridad, aunque me pareció interesante pero adentrarse más a fondo requiere de mucha concentración, en lo que respecta al curso, me gustaría que dejen material complementario cuando se realizan las prácticas o tarea, compartió la idea de mis otros compañeros refiriendonos a la práctica 9 de servidores en la nube, me hubiera gustado tener una guía mas completa para poder haber abordado mejor ese tema.
-

Referencias

- Buurstra Parmo, G. (2023). Criptografía de curvas elípticas [Trabajo de Fin de Grado, Universidad Politécnica de Madrid]. Archivo Digital UPM.
https://oa.upm.es/75495/1/TFG_GABRIELA_BUURSTRA_PARMO.pdf
 - Ciberseguridad.com. (s.f.). Intercambio de claves Diffie-Hellman.
<https://ciberseguridad.com/guias/recursos/intercambio-claves-diffie-hellman/>
 - Diffie-Hellman key exchange. (2023). En Wikipedia.
https://es.wikipedia.org/wiki/Intercambio_de_claves_de_Diffie-Hellman
 - Elliptic curve. (2023). En Wikipedia.
https://es.wikipedia.org/wiki/Curva_el%C3%ADptica
 - Elliptic-curve cryptography. (2023). En Wikipedia.
https://en.wikipedia.org/wiki/Elliptic-curve_cryptography
 - Kolokium. (2023). Criptografía de curva elíptica.
<https://kolokium.com/blog/criptografia-de-curva-eliptica/>
 - Elliptic Curves. (s.f.). En Brilliant.
<https://brilliant.org/wiki/elliptic-curves/>
-