. Criptografía. Entregable 2. Curso 2024/25

1. (Matlab, 2.5 p.) **Cifrado de Vernam.**

   a) Escribir una function que implemente el sistema de Vernam similar a la propuesta en clase con las siguientes características:

   - Los textos a cifrar pueden contener los siguientes caracteres: las letras minúsculas y mayúsculas del alfabeto español; las letras con tilde: á, é, í, ó, ú, Á, É, Í, Ó, Ú, ü; los dígitos del 0 al 9; el espacio y todos los signos de puntuación; los símbolos ?, ¿, !, ¡, @, #, $, %, /, (,), [,]. Opcionalmente puede contener más símbolos si se prefiere.
   - La clave debe estar elegida aleatoriamente como una cadena de caracteres de los símbolos del apartado anterior.

   b) Crear dos mensajes m_1 y m_2 de la misma longitud (al menos longitud 30) pero que se diferencien en algún carácter. Encontrar claves k_1 y k_2 de forma que el cifrado de m_1 con k_1 coincida con el cifrado de m_2 con k_2.

2. (1 p.) El **operador XOR**. Probar que para cada a, b, c se cumple:

   i) la propiedad asociativa: $a \oplus (b \oplus c) = (a \oplus b) \oplus c$, lo que permite representar como $a \oplus b \oplus c$ al resultado calculado de cualquiera de las dos formas.

   ii) $\overline{a} \oplus b \oplus c = \overline{a \oplus b \oplus c}$.

3. (1.5 p.) Un sistema Feistel de seis rondas parte de un mensaje original $m = (L_0, R_0)$ y devuelve un cifrado final $c = (L_6, R_6)$. Recordemos que las fórmulas que permiten pasar de una ronda a la siguiente son:

$$\begin{cases} L_i = R_{i-1} \\ R_i = L_{i-1} \oplus f(R_{i-1}, k_i) \end{cases}$$

   donde f es una función dependiente de los R_j y las subclaves K_l. Calcular c a partir de L_0, R_0 y las subclaves $k_1, k_2, k_3, k_4, k_5, k_6$ si definimos la función f como$f(R_{i-1}, k_i) = R_{i-1} \oplus k_i$. ¿Qué ocurre si todas las subclaves k_i son iguales para $1 \leq i \leq 6$?

4. (2 p.) En el sistema DES, denotamos por m al mensaje a cifrar y k a la clave de cifrado (ambas cadenas binarias de 64 bits). Para cada cadena binaria c denotamos por $\overline{c}$ a la cadena que se obtiene cambiando cada bit de c por su complementario. Razonar por qué

$$\overline{\mathrm{DES}(m, k)} = \mathrm{DES}(\overline{m}, \overline{k}),$$

   donde denotamos por $\mathrm{DES}(m, k)$ al proceso de aplicar el algoritmo de cifrado DES al mensaje m con la clave k.

5. (Matlab, 2.5 p.) En este ejercicio se pide crear varias `function` en Matlab que permitan calcular lo siguiente:

   a) Crear una `function` en Matlab de forma que: dada una clave inicial K, se visualicen tanto los C_i, D_i con $0 \leq i \leq 16$ como las subclaves k_i correspondientes para $1 \leq i \leq 16$. Se pide, además:

   - Comprobar en dos ejemplos que se cumplen las igualdades $C_0 = C_{16}$ y $D_0 = D_{16}$.
   - Comprobar que si todos los bits de C_0 son iguales y todos los bits de D_0 son iguales, entonces todas las subclaves k_i con $0 \leq i \leq 16$ son iguales.

   b) Crear una `function` en Matlab de forma que: dado un mensaje inicial de 64 bits y una clave, se visualicen los L_i, R_i con $0 \leq i \leq 16$ que se van generando en las distintas rondas de DES. Probar la `function` con alguna pareja mensaje-clave concretos.

   Utilizar lo anterior para calcular la salida de la primera y segunda rondas de DES cuando el mensaje está formado todo por ceros (64 bits de entrada iguales a 0) y la clave también está formada todo por ceros (descartando los bits de paridad).

   Repetir lo anterior cuando en el mensaje cada uno de los 64 bits valen 1 y la clave está formada todo por ceros.

   c) Crear una `function` en Matlab de forma que: dada una cadena de 48 bits devuelva la salida generada cuando se utilizan las 8 S-boxes (cajas de sustitución) correspondientes al método DES.

   Comprobar la salida en el caso en que la entrada es:

   cadena = '000000111111000000111111000000111111000000111111'

6. (1.5 p., Matlab) Triple DES. Dadas dos claves k_1 y k_2 de DES, se trata de cifrar un mensaje m según el esquema

$$\mathrm{E}_{k_1}(\mathrm{D}_{k_2}(\mathrm{E}_{k_1}(m)))$$

   es decir, se cifra con la clave k_1, después se descifra con la clave k_2 y finalmente se cifra de nuevo con la clave k_1.

   a) Implementar este sistema usando DES.m como cifrador-descifrador. Probarlo con un ejemplo.

   b) Dar un ejemplo con un mensaje de prueba m de 8 caracteres Unicode y claves iguales $k_1 = k_2$ donde se comprueba que se obtiene el mismo resultado que cifrando una vez con DES.

7. (3 p., Matlab) Usando el código DES.m, disponible en Campus Virtual, implementarlo para que funcione en modo CBC. En concreto, se pide:

   a) generar un vector de inicialización IV de 64 bits (8 caracteres si se prefiere) o bien de forma aleatoria o eligiendo una palabra;

   b) utilizar DES en modo CBC con el mensaje de 24 caracteres `m='Prueba CBC con 3 bloques'`

y el vector IV generado anteriormente. Se trata de calcular los tres bloques de cifrado, para descifrar después y comprobar que el sistema funciona correctamente.

8. (1.5 p., Matlab) Bob utiliza un sistema RSA con clave pública $n = 42421$ y exponente de cifrado $e = 7$. Supongamos que nadie salvo Bob puede factorizar n ni encontrar el exponente de descifrado d. Alice quiere enviar un mensaje cifrado a Bob asignando a las letras del alfabeto inglés a, b, $\dots$, z los valores 1, 2, $\dots$, 26, y cifrando cada letra del mensaje individualmente: por ejemplo, si cifra la letra que se corresponde al número 9 envía el valor de $9^e \pmod n$. El mensaje cifrado que envía a Bob es el vector

$$c = (160, 18848, 1, 7858, 1, 28484, 20144, 18848, 1, 16384, 30008)$$

donde cada componente del vector se corresponde con una letra cifrada del mensaje original. Hallar dicho mensaje.

9. (2.5 p., Matlab) RSA con un módulo de 80 dígitos.

   a) Generar aleatoriamente dos números primos distintos de 40 dígitos cada uno. Hallar su producto, n y elegir un exponente de cifrado e. Hallar también el exponente de descifrado d.

   b) Elegir un mensaje m tal que `tonumberp(m)` sea mayor que n pero de forma que podamos dividir el mensaje m en dos trozos tal que cada uno de ellos pueda ser cifrado por nuestro sistema.

   c) Cifrar y descifrar ambos trozos comprobando que el sistema funciona correctamente.

10. (2 p., Matlab) Se sabe que el número $p = 601388945245846505101048134456739817602$1 es primo y $e = 3$ es una raíz primitiva módulo p. Alicia y Bob usan estos valores para un intercambio de clave de Diffie-Hellman.

    a) Probar que si los números que se intercambian Alicia y Bob son:

    $$\text{Alicia2Bob} = 1485121808391515125049604614485735928033$$
    $$\text{Bob2Alicia} = 3413383369016012953225518871319720279142$$

    entonces un adversario puede hallar la clave de uno de ellos y por lo tanto obtener la clave común. Hallar dicha clave.

    b) Elegir ahora dos números secretos $xAlicia$ y $xBob$ que proporcionen más seguridad al sistema. Hallar los números que se intercambiarían Alicia y Bob así como la clave común compartida.