Compresión y Seguridad

Profesor: ***ANTONIO ZAMORA GÓMEZ***

Compresión y Seguridad

- Grado en Ingeniería Multimedia
- Asignatura de 1r cuatrimestre
- Obligatoria 3r curso

- 6 créditos (3 Teoría 3 Prácticas)

- Guía docente

Compresión y Seguridad

Objetivos

Seguridad

- Introducir el problema de la seguridad de los datos multimedia.
- Conocimiento de las principales herramientas criptográficas para la protección de datos multimedia, así como los niveles de seguridad que se consideran aceptables y las tendencias a corto y medio plazo.
- Introducir el cifrado clásico de la información situándolo en su contexto histórico y favoreciendo el acercamiento a los conceptos y fundamentos de las herramientas de seguridad actuales
- Conocer las especiales características de la criptografía simétrica y asimétrica y sus aplicaciones para gartantizar la confidenicalidad, autenticidad e integridad de los datos multimedia.

Compresión

- Introducir la Teoría de la Información y sus aplicaciones a la compresión de la información.
- Conocer las principales técnicas de compresión de la información, en general y de contenidos multimedia, en particular.

Temario

SEGURIDAD

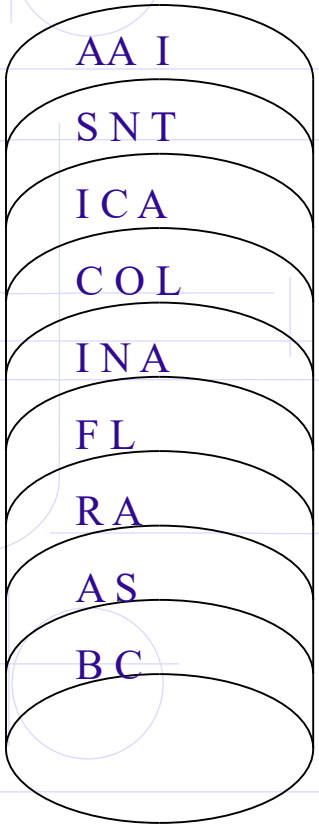
Tema 1.- Introducción a la Seguridad de la Información
Tema 2.- El cifrado de datos clásico
Tema 3.- Criptografía simétrica en flujo
Tema 4.- Criptografía simétrica en bloque
Tema 5.- Criptografía asimétrica
Tema 6.- Autoridades certificadoras

COMPRESIÓN

Tema 7.- Teoría de la Información
Tema 8.- Compresión de datos
Tema 9.- Compresión de contenidos multimedia

Pormenorizado en guía docente

Scítala espartana



Texto en claro

m = ASI CIFRABAN CON LA SCITALA

Texto cifrado

c = AAISNTICACOLINAFLRAASBC

Se trata de un sistema de cifra por transposición

Ejemplo

- Consideremos el alfabeto A={ _ABCDEFGHIJKLMNÑOPQRSTUVWXYZ} y el texto en claro

m=TRANSFERENCIA_CONFORME

- A cada símbolo del alfabeto le asociamos un número

_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27

- Para cifrar m utilizamos la clave k=2 y r=3, obteniendo

$E_k(T)=E_k(21)= 3\cdot 21+2 \bmod 28 = 9 = I$
$E_k(R)=E_k(19)= 3\cdot 19+2 \bmod 28 = 3 = C$
$E_k(E)=E_k(5) = 3\cdot 5+2 \bmod 28 = 17 = P$

o sea

c=ICEOFSPCPOKAEBKUOSUCMP

Ejemplo

- Para descifrar c utilizamos la clave k=2 y r^{-1}=19*, obteniéndose

$$D_k(I) = D_k(9) = (9-2)\ 19 \bmod 28 = 21 = T$$
$$D_k(C) = D_k(3) = (3-2)\ 19 \bmod 28 = 19 = R$$
$$D_k(P) = D_k(17) = (17-2)\ 19 \bmod 28 = 5 = E$$

- Si generamos el alfabeto de cifrado, se simplificará el descifrado de posteriores criptogramas en los que se haya utilizado la misma clave.

_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
2	5	8	11	14	17	20	23	26	1	4	7	10	13	16	19	22	25	0	3	6	9	12	15	18	21	24	27
B	E	H	K	N	P	S	V	Y	A	D	G	J	M	O	R	U	X	_	C	F	I	L	Ñ	Q	T	W	Z

3.4 Algoritmo RC4

Key Scheduling Algorithm (KSA)

Para calcular los valores iniciales de la S-Caja, se hace lo siguiente:

1. $S(i) = i \quad \forall i \in \{0,1,\ldots,255\}$
2. Rellenar el array K(0) a K(255) repitiendo la clave tantas veces como sea necesario.
3. $j = 0$
4. Para i = 0 hasta 255 hacer:

$$j = [j + S(i) + K(i)] \bmod 256$$

Intercambiar S(i) y S(j) .

Pseudo-Random Generation Algorithm (PRGA)

Dos contadores i y j se ponen a cero.

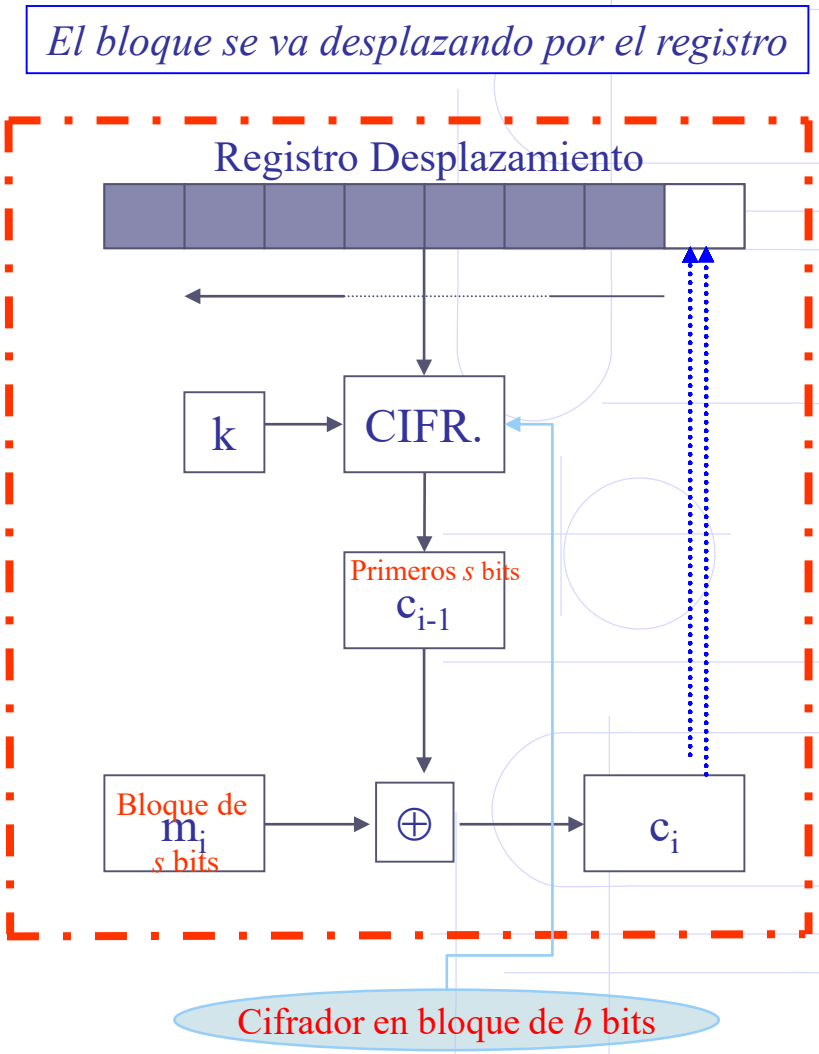
En la iteración r, cada byte, O_r, de la secuencia cifrante se calcula como sigue:

1. $i = (i + 1) \bmod 256$
2. $j = [j + S(i)] \bmod 256$
3. Intercambiar los valores de S(i) y S(j)
4. $t = [S(i) + S(j)] \bmod 256$
5. $O_r = S(t)_{(2)}$
3. Mientras se necesite secuencia cifrante volver a 1

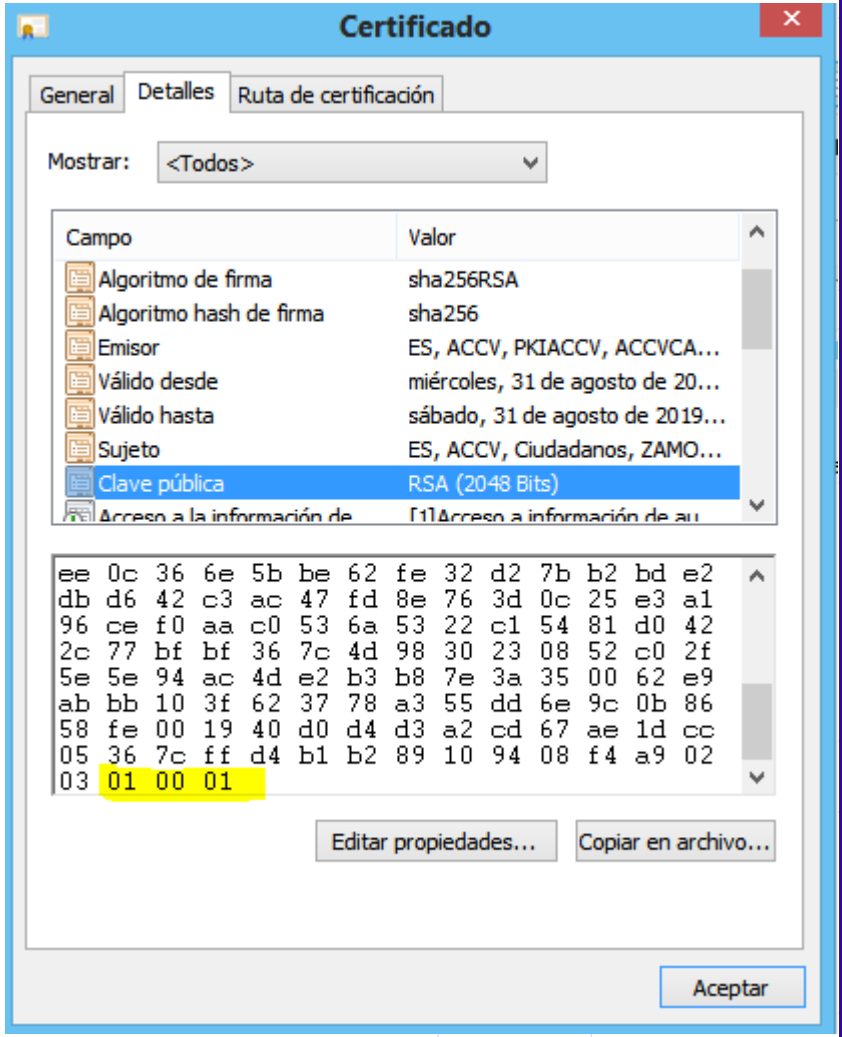
4.3.3 Modo CFB (Cipher-Feedback)

- El vector inicial VI del Registro de Desplazamiento RD se carga, al igual que en el modo CBC, con un valor aleatorio de *b* bits.
- El mensaje se divide en bloques de *s* bits (normalmente un byte) que se suma or-exclusivo con los *s* bits más significativos que resultan de aplicar el algoritmo en bloque a los *s* bits del anterior registro con la clave k.
- En cada operación, se realimenta el bloque de *s* bits del criptograma al extremo derecho de dicho registro, produciendo un desplazamiento de *s* bits a la izquierda.
- Si $S_s(x)$ representa los *s* bits más significativos de x, se tiene

$$\mathbf{c_i = m_i \oplus S_s[E_k(RD)]},\ i = 1,2,\cdots$$



5.4 Algoritmo RSA



ESQUEMA DE FIRMA DIGITAL CON CIFRADO EN RSA

A n_A, e_A, d_A | **B** n_B, e_B, d_B

m

$c = m^{e_B} \bmod n_B$ →

c

$m = c^{d_B} \bmod n_B$

m

$r = m^{d_A} \bmod n_A$

$s = r^{e_B} \bmod n_B$

$s = r^{e_B} \bmod n_B$ →

c

m

s

$c; m; s;$

$s^{d_B} \bmod n_B = r^{d_B e_B} \bmod n_B = r$

$c; m; s; r;$

$r^{e_A} \bmod n_A \equiv m$?

MÉTODO DE HUFFMAN

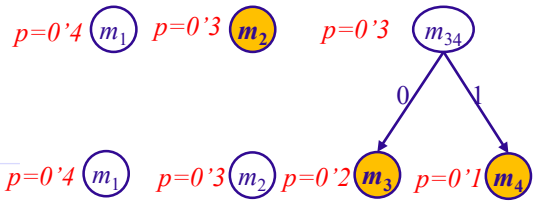
Ejemplo

- Se puede obtener el código de Huffman construyendo un árbol en sentido ascendente teniendo en cuenta el teorema que fundamenta el método.

MÉTODO DE HUFFMAN

Ejemplo

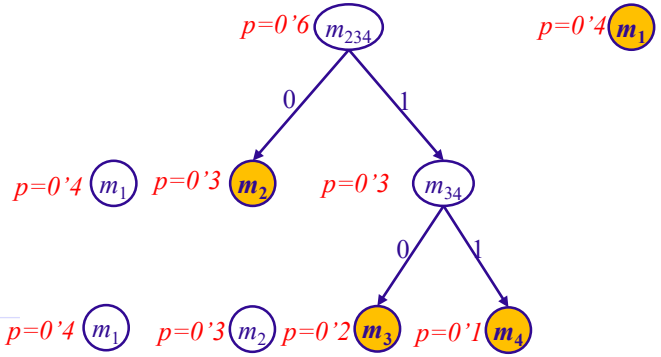
- Se puede obtener el código de Huffman construyendo un árbol en sentido ascendente teniendo en cuenta el teorema que fundamenta el método.



MÉTODO DE HUFFMAN

Ejemplo

- Se puede obtener el código de Huffman construyendo un árbol en sentido ascendente teniendo en cuenta el teorema que fundamenta el método.

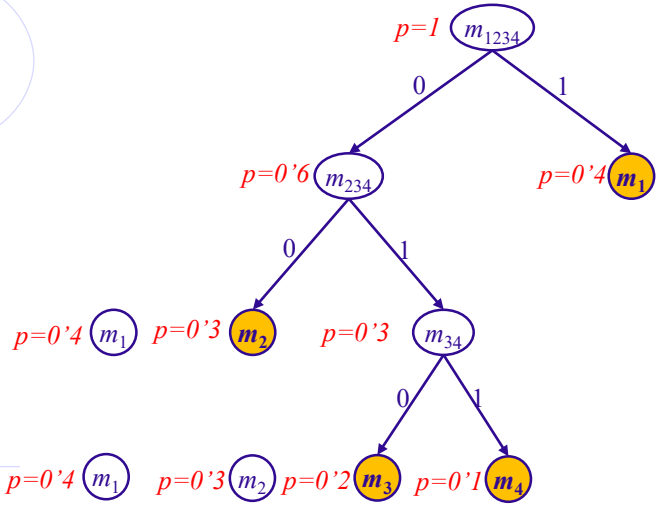


MÉTODO DE HUFFMAN

Ejemplo

- Se puede obtener el código de Huffman construyendo un árbol en sentido ascendente teniendo en cuenta el teorema que fundamenta el método.

m_1=1
m_2=00
m_3=010
m_4=011



8.2 Compresión sin pérdida

- Los métodos de compresión sin pérdida de datos pueden ser clasificados de acuerdo a los tipos de datos para los que fueron diseñados.

- Los tres tipos principales de datos para comprimir son: texto, imágenes y sonido.

- Algunos de los algoritmos de propósito general más conocidos para la compresión sin pérdida de datos son:
  - La transformada Burrows-Wheeler.
  - LZ77
  - LZW
  - Huffman
  - Codificación Aritmética
  - RLE
  - Deflate

8.3.1 El estándar JPEG

- Las técnicas de compresión de imagen, actualmente en desarrollo, logran razones de compresión desde 10:1 a 50:1 sin perdida perceptible de calidad.

- Sin embargo, disponer de esta tecnología no es suficiente.

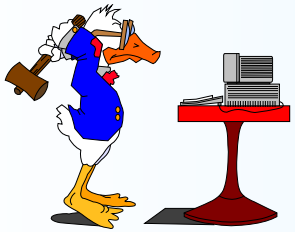
- Para que una aplicación del mercado que emplee almacenamiento y transmisión de imágenes digitales sea utilizable, es necesario disponer de un estándar de compresión que permita la interoperabilidad entre equipos de diferentes fabricantes

- Durante las últimas décadas expertos fotográficos trabajando juntos bajo los auspicios de ITU, ISO e IEC. han desarrollado el estándar JPEG (*Joint Photographic Experts Group*) que pretende ser el estándar internacional de compresión de imagen digital para ajustarse a las necesidades de la mayor parte de las aplicaciones que utilizan imágenes digitales.

Sistema de evaluación

- **El 50% de la nota final corresponde a la del examen final y el otro 50% a la de un trabajo práctico sobre seguridad.**
- **Nota mínima 4 sobre 10 en las dos partes.**
- **Las notas del trabajo práctico se mantienen para la convocatoria de julio.**
- **En las convocatorias extraordinarias se realizará un examen de teoría y otro de prácticas, aplicándose los mismos criterios de ponderación y mínimos. Siendo, por tanto, todos los componentes recuperables.**

Pormenorizado en guía docente



Trabajo práctico

- Grupos de 4 alumnos
- Se dejará información en UACloud (Materiales docentes)

- Grupos 2 (lunes 11:00 a 13:00) y 3 (lunes 09:00 a 11:00) profesor Antonio Zamora Gómez (responsable de la asignatura)

- Grupos 1 (martes 13:00 a 15:00) y 4 (martes 11:00 a 13:00) profesor José Vicente Aguirre Pastor

GRADO EN INGENIERÍA MULTIMEDIA
COMPRESIÓN Y SEGURIDAD
Planificación curso 2021-2022

Nº	SEMANA	DÍAS LECTIVOS	TEORÍA	PRÁCTICAS
1	13 sep - 17 sep	5	Presentación 1.- Introducción a la Seguridad de los datos multimedia	Presentación. Nociones generales sobre seguridad.
2	20 sep - 24 sep	5	2.- El cifrado de datos clásico	Nociones generales sobre seguridad. Creación grupos y elección trabajo práctico
3	27 sep - 01 oct	5	3.- Criptografía simétrica en flujo	
4	04 oct - 08 oct	5	4.- Criptografía simétrica en bloque	
5	11 oct - 15 oct	0	11 de octubre, lunes, festivo por Día de la Comunidad Valenciana	11 de octubre, lunes, festivo por Día de la Comunidad Valenciana 12 de octubre, martes, festivo por Fiesta Nacional de España
6	18 oct - 22 oct	5	5.- Criptografía asimétrica	Entrega 1ª memoria antes de 23:59 h. del 23 oct
7	25 oct - 29 oct	5	5.- Criptografía asimétrica 6.- Autoridades certificadoras	
8	01 nov - 05 nov	4 (M,X,J,V)	1 de noviembre, lunes, festivo por Todos los Santos	
9	08 nov - 12 nov	5	7.- Teoría de la Información	Entrega 2ª memoria antes de 23:59 h. del 13 nov
10	15 nov - 19 nov	5	7.- Teoría de la Información 8.- Compresión de datos	
11	22 nov - 26 nov	5	8.- Compresión de datos 9.- Compresión de contenidos multimedia	
12	29 nov - 03 dic	5	9.- Compresión de contenidos multimedia	
13	06 dic - 10 dic	3 (M,J,V)	6 de diciembre, lunes, festivo por Día de la Constitución 8 de diciembre, miércoles, festivo por La Inmaculada Concepción	Entrega prácticas antes de 23:59 h. del 11 dic
14	13 dic - 17 dic	5	Ejercicios	Evaluación de prácticas 13, 14 dic
15	20 dic - 23 dic	4 (L,M,X,J)	Ejercicios	Evaluación de prácticas 20, 21 dic
	TOTAL DÍAS ->	69		