

4.- Cifrado en bloque con clave secreta

4.1.- Características generales

4.2.- DES (Data Encryption Standard)

4.3.- AES (Advanced Encryption Standard)

4.4.- Modos de cifrado en bloque

4.5.- Cifrado Múltiple. Triple DES



Criptografía simétrica

- ▶ La criptografía de clave secreta o simétrica se refiere al conjunto de métodos que permiten tener comunicación segura entre las partes **siempre y cuando anteriormente se hayan intercambiado la clave correspondiente.**
- ▶ Ha sido la **más usada en toda la historia** y ha sido implementada en diferentes dispositivos, manuales, mecánicos, eléctricos, hasta los algoritmos actuales que son programables en cualquier ordenador.
- ▶ Todos los sistemas **criptográficos clásicos** se basan en **criptografía simétrica**.



Criptografía simétrica

- ▶ Generalmente el **algoritmo de cifrado es conocido** por lo que la **fortaleza** del mismo dependerá de su complejidad interna y **sobre todo de la longitud de la clave empleada.**
- ▶ Para que un algoritmo de este tipo sea considerado fiable debe cumplir varios requisitos básicos:
 - ▶ **conocido el criptograma** (texto cifrado) **no se pueden obtener de él ni el texto en claro ni la clave.**
 - ▶ **conocidos el texto en claro y el texto cifrado** debe resultar **más caro en tiempo o dinero descifrar la clave** que el valor posible de la información obtenida por terceros.



Criptografía simétrica

- ▶ El **principal problema** para este sistema de cifrado consiste en que para **cada par** de **usuarios** que quieran establecer comunicación se requiere **una clave** diferente, es decir, que un usuario de una red debe almacenar tantas claves como personas con las que quiera mantener una comunicación segura.
- ▶ Al principio (cuando las redes contaban con pocos usuarios) este hecho no constituía ningún problema, pero actualmente, con la cantidad de usuarios que existen en las redes se convierte en impracticable.
- ▶ **Otro problema** que presentan es el hecho de la **distribución** de **claves** y el peligro de que muchas personas deban conocer una misma clave.



4.1 Características generales

- Se **denomina cifrado en bloque** aquel en el que **se cifra** el mensaje original **agrupando los símbolos en grupos** (bloques) de dos o más elementos.
- Algunos sistemas de cifrado clásicos, como el **poligrámico** y el de **transposición**, son **ejemplos** de cifrado en bloque.
- En los **sistemas modernos** de cifrado en bloque
 - Cada **símbolo** se **cifra** de manera **dependiente de los adyacentes**.
 - Cada **bloque** de símbolos se **cifra** siempre de **igual manera**, independientemente del lugar que ocupe en el mensaje.
 - **Dos mensajes originales iguales**, cifrados con la misma clave, **producen siempre mensajes cifrados iguales**.
 - Para **descifrar parte de un mensaje** no es preciso descifrarlo completamente desde el principio, **basta con hacerlo desde el bloque que interese**.



4.1 Características generales

- ▶ Los cifradores en bloque con clave secreta aplican técnicas de **sustitución y transposición**, además de otras operaciones lineales y no lineales.
 - ▶ Se apoyan en los principios de **confusión y difusión** propuestos por Shannon que se combinan para dar lugar a los denominados **cifrados de producto**.
- {
- Recordemos que la **confusión** consiste en tratar de ocultar la relación que existe entre el texto en claro, el texto cifrado y la clave.
 - Por su parte la **difusión** trata de repartir la influencia de cada bit del mensaje original lo más posible entre el mensaje cifrado.



4.1 Características generales

- ▶ La mayoría de los algoritmos se basan en **diferentes capas de sustituciones y permutaciones**, estructura que denominaremos **red de sustitución-permutación**.
- ▶ En muchos casos el criptosistema no es más que **un paso simple de sustitución-permutación repetido n veces**, como ocurre con **DES**.



4.1 Características generales

- ▶ En muchos casos, los cifrados en bloque se componen de cuatro elementos:
 - ▶ Transformación inicial.
 - ▶ Una función criptográficamente débil iterada r veces o «**vuelta**» (« **rondas** »).
 - ▶ Transformación final.
 - ▶ Algoritmo de expansión de clave.



4.1 Características generales

- ▶ La transformación inicial puede tener una o **dos** funciones:
 - ▶ La **primera** consiste simplemente en **distribuir aleatoriamente los datos de entrada** (para ocultar bloques de datos de todo ceros o unos, etc.), **careciendo de significación criptográfica** si no depende de la clave.
 - ▶ La **segunda** función, solamente presente en algunos criptosistemas, **tiene significación criptográfica, dificultando ataques por análisis lineal o diferencial**; en este caso es función de la clave.



4.1 Características generales

- **Las vueltas intermedias:**
 - Consisten en **una función complicada de los datos y la clave**, por lo general no lineal.
 - **No deben formar grupo**, para que el conjunto de varias pasadas sucesivas con sus subclaves correspondientes no sean equivalentes a una pasada única con una subclave diferente.
- La **transformación final** sirve para que las operaciones de cifrado y descifrado sean simétricas.



4.1 Características generales

- ▶ El **algoritmo de expansión de clave** tiene por objeto convertir la clave de usuario en un **conjunto de subclaves** que pueden estar constituidas por varios cientos de bits en total.
- ▶ **Conviene que sea unidireccional** y que el conocimiento de una o varias subclaves intermedias no permita deducir las subclaves anteriores o siguientes.
- ▶ Además, se ha de cuidar que las **subclaves producidas no constituyan un pequeño subconjunto monótono** de todas las posibles.



4.1 Características generales

ALGORITMOS MÁS UTILIZADOS

- ▶ [AES](#)
- ▶ [DES \(Triple DES\)](#)
- ▶ [Serpent](#)
- ▶ [Blowfish, Twofish](#)

OTROS ALGORITMOS

- ▶ [Camellia](#)
- ▶ [CAST-128](#)
- ▶ [IDEA](#)
- ▶ [RC2, RC5, RC6](#)
- ▶ [SEED](#)
- ▶ [ARIA](#)
- ▶ [Skipjack](#)
- ▶ [TEA, XTEA](#)



4.1 Características generales

- **Blowfish** fue creado por Bruce Schneier, autor del libro *Applied Cryptography* (considerado por muchos como la "biblia" en cuestiones de criptografía). Utiliza claves de hasta 448 bits y, hasta el momento, ha resistido con éxito todos los ataques. Por ello y por su estructura se le considera uno de los algoritmos más seguros, a pesar de lo cual no se utiliza masivamente. **Su autor no ha patentado el método para que pueda ser empleado sin limitaciones.**

La versión más actual es **Twofish**, que llegó a la ronda final del concurso AES del NIST (quedó tercero, tras Rijndael y Serpent). Cifra bloques de 128 bits con claves de hasta 256 bits.

- **Serpent** fue diseñado por [Ross Anderson](#), [Eli Biham](#) y [Lars Knudsen](#) y quedó finalista en el concurso Advanced Encryption Standard del [NIST](#), tras [Rijndael](#) que fue el ganador. Cifra bloques de 128 bits con claves de 128, 192 o 256 bits.



4.1 Características generales

- **DES** En enero de 1977 la National Bureau of Standards (NBS) de Los Estados Unidos de América publicó su *Federal Information Processing Standard* con el título de *Data Encryption Standard*, en él se exponía el funcionamiento de un **algoritmo de cifrado estándar** que debía ser utilizado por todas las Agencias Federales para la protección criptográfica de datos informáticos de **naturaleza reservada** pero **no secreta**.
- DES es un algoritmo de cifrado en bloque de los denominados redes Feistel. La longitud de bloque es de 64 bits y la de clave es 56 bits.
- DES ha sido el estándar utilizado mundialmente durante más de 25 años, generalmente en la banca. Hoy presenta signos de envejecimiento y ha sucumbido a los diversos criptoanálisis que contra él se vienen realizando. No obstante se sigue utilizando (TDES).
- **TripleDES** cifrado múltiple, *EDE* $c = E_{k_1} \left(D_{k_2} \left[E_{k_1} (m) \right] \right)$



4.1 Características generales

- **AES.** El 2 de octubre de 2000 el **NIST (National Institute for Standards and Technology)** anunciaba oficialmente la adopción del algoritmo **Rijndael** como nuevo **Estándar Avanzado de Cifrado (AES)** para su empleo en aplicaciones criptográficas no militares, culminando así un proceso de mas de tres años, encaminado a proporcionar a la comunidad internacional un nuevo algoritmo de cifrado potente, eficiente y fácil de implementar.

DES tiene un sucesor

- La palabra **Rijndael** es un acrónimo formado por los nombres de sus dos autores, los belgas
 - *Vincent Rijmen y Joan Daemen.*
- Su interés radica en que todo el proceso de selección, **revisión y estudio** tanto de este algoritmo como de los restantes candidatos, se ha efectuado de **forma pública y abierta**, por lo que, prácticamente por primera vez, toda la comunidad criptográfica mundial ha participado en su análisis, lo cual convierte a **Rijndael** en un algoritmo perfectamente digno de la confianza de todos.



4.1 Características generales

- ▶ Los tres aspectos básicos sobre los que se ha diseñado **AES** son los siguientes:
 - ▶ **Resistencia** contra todo tipo de ataque conocido hasta ese momento.
 - ▶ **Eficiencia** computacional en un amplio abanico de plataformas, tanto hardware como software (optimizado para 32 bits).
 - ▶ **Simplicidad** de diseño.
- ▶ AES es un sistema de cifrado por bloques cuya longitud de **bloque** es de **128** bits, diseñado para manejar longitudes de **clave** variables: **128, 192** y **256** bits.



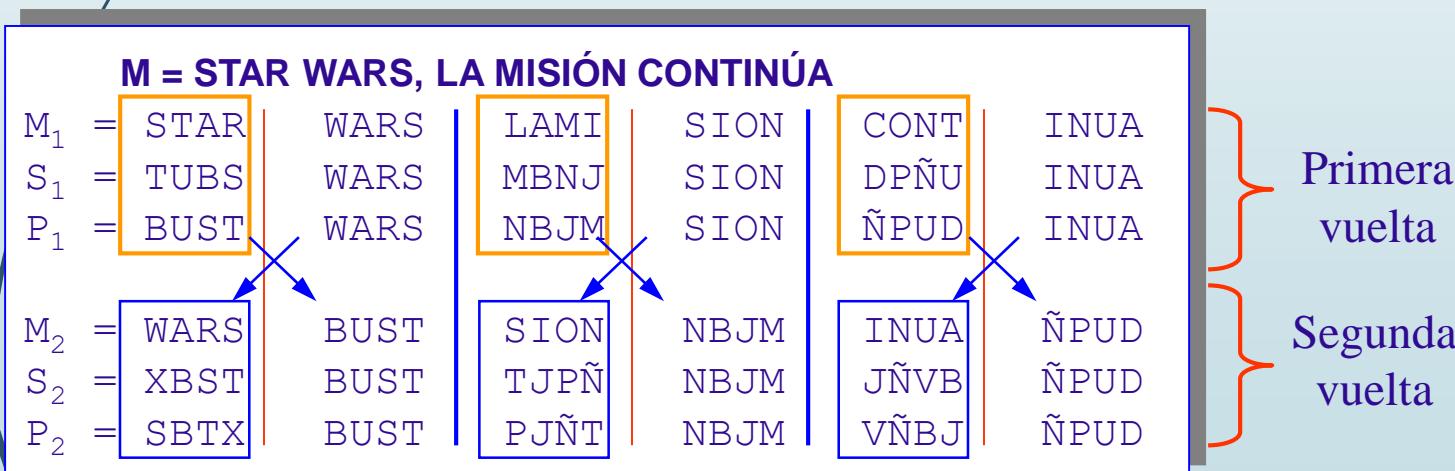
Cifrado tipo Feistel

Se denominan cifradores **tipo Feistel** a aquellos en los que el bloque de datos se divide en dos mitades y en cada vuelta de cifrado se trabaja, alternadamente, con una de las mitades

EJEMPLO: El algoritmo usará bloques de tamaño 8 caracteres. Tendrá dos vueltas y en cada vuelta realizará una operación de sustitución **S** y una permutación **P** sobre la 1^a mitad.

Sustitución S: Desplazamiento +1 mod 27

Permutación P: $\sigma=(3\ 2\ 4\ 1)$



4.2 DES (Data Encryption Standard)

- En enero de 1977 la National Bureau of Standards (NBS) de Los Estados Unidos de América publicó su *Federal Information Processing Standard* con el título de *Data Encryption Standard*, en él se exponía el funcionamiento de un **algoritmo de cifrado estándar** que debía ser utilizado por todas las Agencias Federales para la protección criptográfica de datos informáticos de **naturaleza reservada** pero **no secreta**.
- Este método fue el resultado de la investigación realizada por IBM durante los años 1968 a 1975 inspirada en el sistema de **Horst Feistel** denominado **Lucifer**, consistente en una composición de diferentes transformaciones tal como proponía Shannon en su artículo “*Communication Theory of Secrecy Systems*” (Octubre 1949).



4.2 DES (Data Encryption Standard)

- DES es un algoritmo de cifrado en bloque de los denominados **redes Feistel**. La longitud de bloque es de 64 bits (ocho símbolos ASCII) y la de clave es 56 bits.
- DES ha sido el estándar utilizado mundialmente durante más de 25 años, generalmente en la banca.
 - Hoy no es muy utilizado dado que ha sucumbido a los diversos ataques que contra él se han realizando.
 - No obstante se sigue utilizando (TDES).



4.2.1 Introducción y descripción

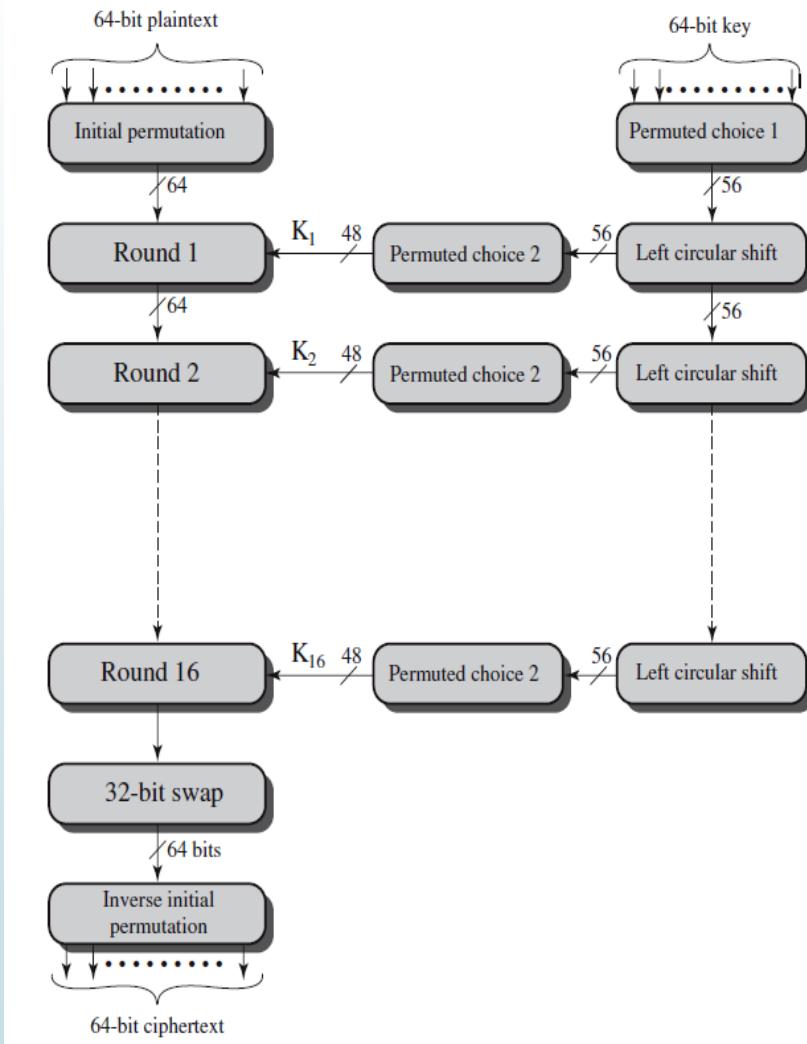
- La NSA fue la que impuso la longitud de clave del DES, que es bastante modesta y que la hace desaconsejable con el actual desarrollo de la informática.
- DES es un algoritmo de cifrado en bloque; la longitud de bloque es de 64 bits (ocho símbolos ASCII); la longitud de la clave es de 56 bits, lo que equivale a que existan

$$2^{56} = 7'2 \cdot 10^{16} \text{ claves diferentes.}$$



4.2.1 Introducción y descripción

- ▶ Cifrador de bloque tipo Feistel
- ▶ Dos entradas
 - ▶ Texto en claro (64bits)
 - ▶ Clave (64bits, sólo 56 útiles)
- ▶ Tres fases
 - ▶ Permutación inicial
 - ▶ 16 rondas de sustitución-permutación, intercambio de mitades
 - ▶ Permutación inicial inversa
- ▶ Generación de subclaves
 - ▶ Desplazamiento circular (56bits)
 - ▶ Selección permutada (48bits)
- ▶ Descifrado
 - ▶ Mismo algoritmo con subclaves en orden inverso



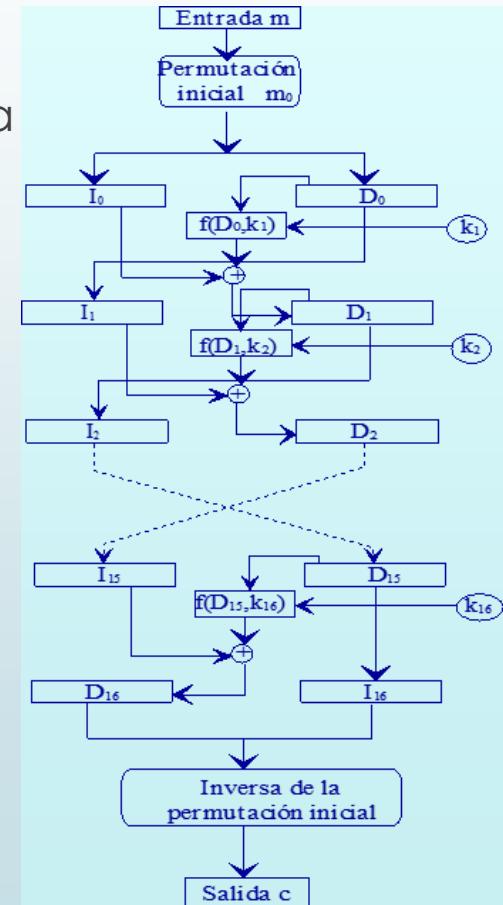
4.2.1 Introducción y descripción

- El algoritmo es válido para el cifrado y descifrado de bloques de 64 bits mediante una clave de 56 a la que se añaden 8 de paridad.
- Si $m=m_1m_2\dots m_{64}$ es un bloque de 64 bits, se realiza una permutación inicial

$$m_0 = m_{58}m_{50}\dots m_7$$

utilizando la siguiente tabla

58	50	42	34	26	18	10	02
60	52	44	36	28	20	12	04
62	54	46	38	30	22	14	06
64	56	48	40	32	24	16	08
57	49	41	33	25	17	09	01
59	51	43	35	27	19	11	03
61	53	45	37	29	21	13	05
63	55	47	39	31	23	15	07

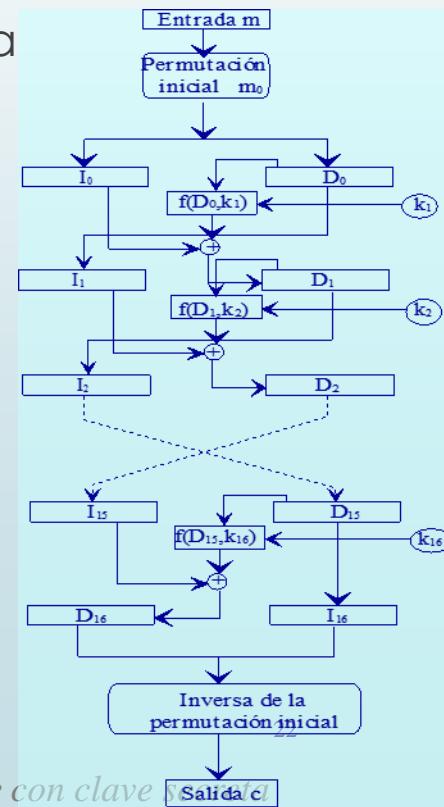


4.2.1 Introducción y descripción

- Este bloque m_0 es dividido en dos subbloques de 32 bits cada uno
 $I_0 = m_{58}m_{50}\dots m_{16}m_8$ y $D_0 = m_{57}m_{49}\dots m_{15}m_7$
- A continuación se realizan sobre estos dos bloques **16 transformaciones** que combinan **sustituciones y transposiciones**.
- Al bloque resultante de concatenar D_{16} e I_{16} se le aplica la permutación inversa a la inicial, dada en la tabla,

40	08	48	16	56	24	64	32
39	07	47	15	55	23	63	31
38	06	46	14	54	22	62	30
37	05	45	13	53	21	61	29
36	04	44	12	52	20	60	28
35	03	43	11	51	19	59	27
34	02	42	10	50	18	58	26
33	01	41	09	49	17	57	25

para obtener el bloque cifrado c , de 64 bits.



Cifrado en bloque con clave k Salida c



4.2.2 La función f y las cajas S_i

- Entre la permutación inicial y la final el algoritmo realiza 16 iteraciones que describiremos a continuación.

- Denotemos por

$$T_i = t_1 \dots t_{32} \quad t_{33} \dots t_{64} = I_i \ D_i$$

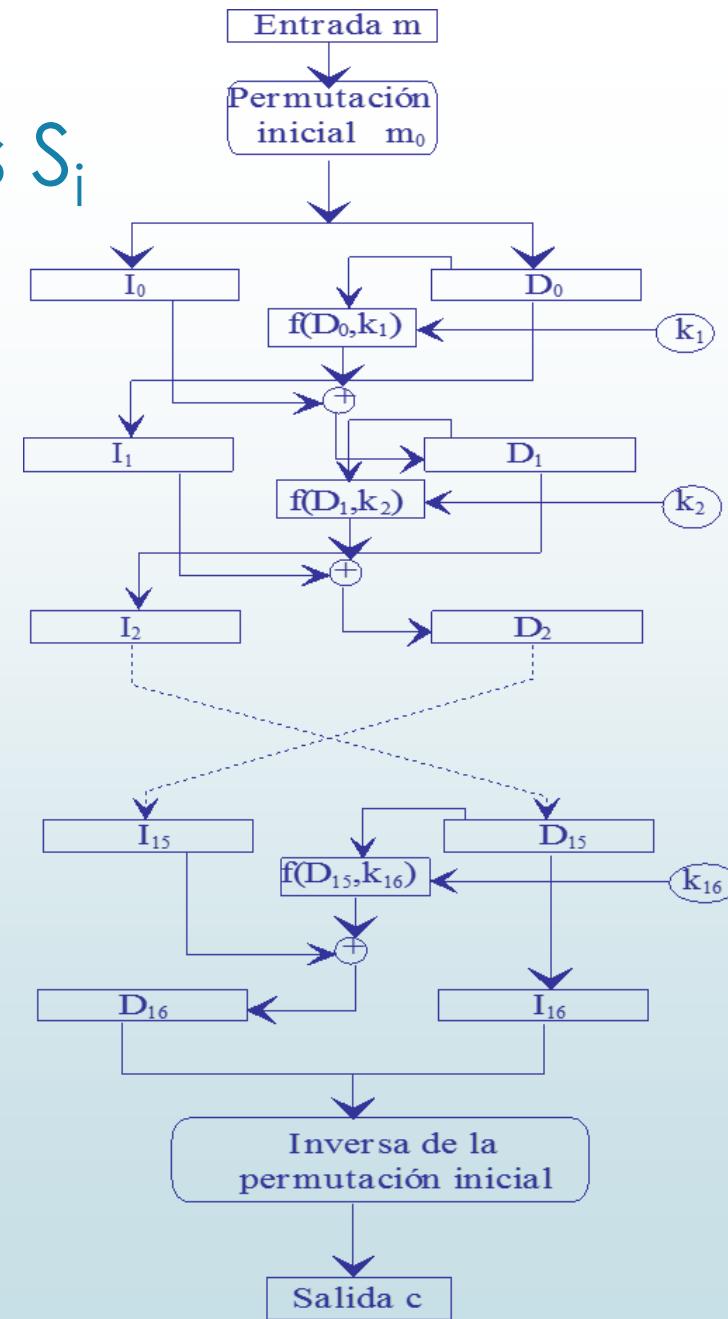
al resultado obtenido tras la iteración i-ésima, donde I_i representa los 32 primeros bits de T_i y D_i los 32 últimos, esto es

$$I_i = t_1 \dots t_{32} \quad D_i = t_{33} \dots t_{64}$$

- Se define

$$I_i = D_{i-1} \quad D_i = I_{i-1} \oplus f(D_{i-1}, k_i) \quad i=1,2,\dots,16$$

donde \oplus es la operación OR-exclusiva y k_i es una subclave de 48 bits que se obtiene a partir de la clave original k .



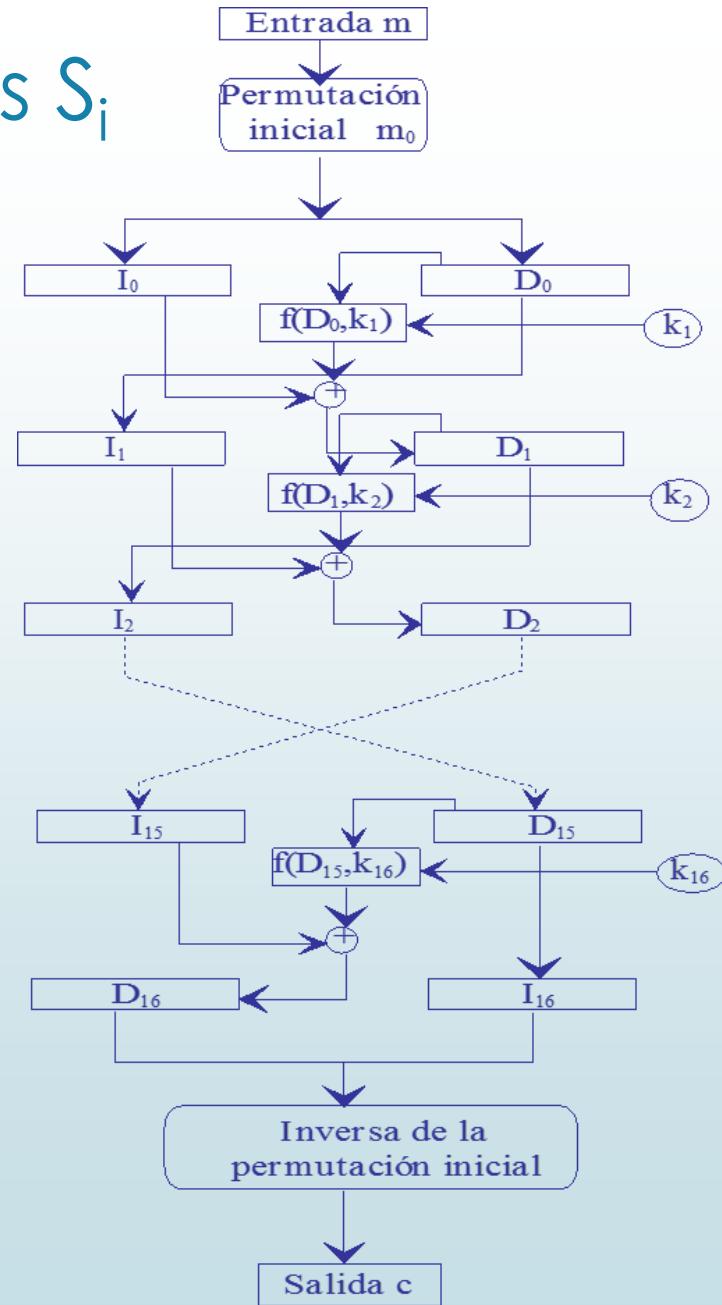
Cifrado en bloque con clave secreta



4.2.2 La función f y las cajas S_i

- Se puede observar que en la **última iteración** las dos mitades, izquierda y derecha, **no son cambiadas**, siendo el bloque $T_{16} = D_{16}, I_{16}$ al que se le aplica la permutación final. Esto es necesario para que el algoritmo sea válido para cifrar y descifrar.
- La función f transforma, mediante la subclave k_i , los bloques D_i en los bloques de 32 bits $f(D_{i-1}, k_i)$.
- Para ello, en primer lugar, se produce una **expansión E** de los **32 bits de D_{i-1} para obtener un bloque de 48 bits**, de tal manera que si numeramos los bits de D_{i-1} de 1 a 32

$$D_{i-1} = d_1 d_2 \dots d_{32}$$



Cifrado en bloque con clave secreta



4.2.2 La función f y las cajas S_i

- La expansión se obtiene permutando dichos bits de acuerdo con el orden de la tabla

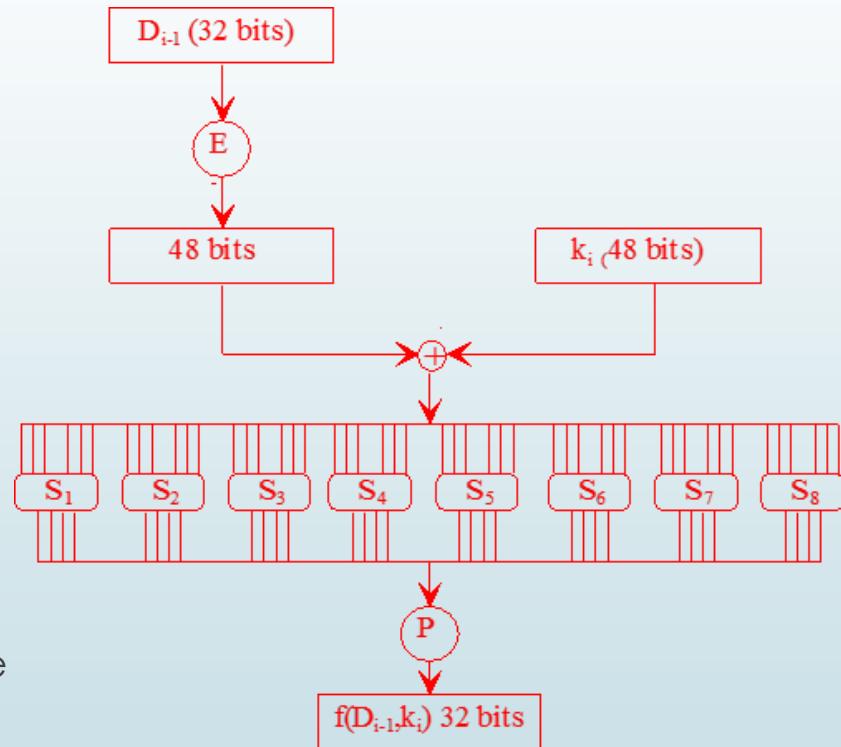
32	01	02	03	04	05
04	05	06	07	08	09
08	09	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	01

esto es

$$E(D_{i-1}) = d_{32}d_1d_2\dots d_{32}d_1$$

- Una vez obtenido el bloque de 48 bits se efectúa la operación \oplus entre $E(D_{i-1})$ y la subclave k_i . El resultado es dividido en 8 bloques B_i de 6 bits cada uno

$$E(D_{i-1}) \oplus k_i = B_1B_2\dots B_8$$



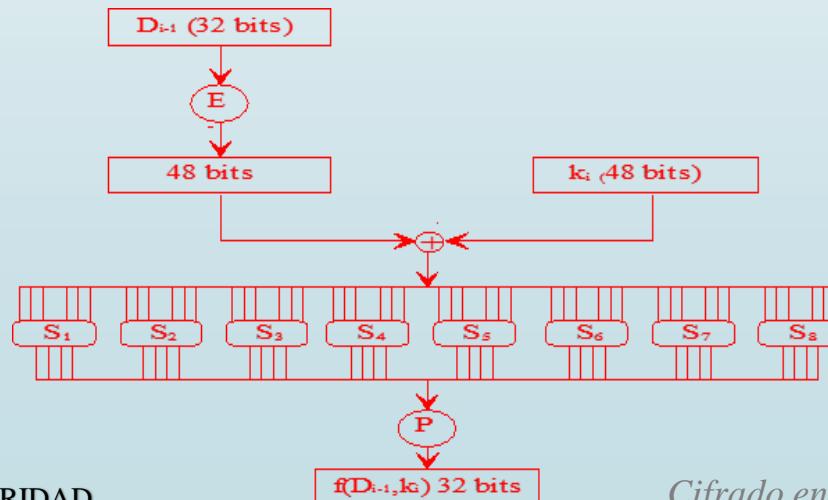
4.2.2 La función f y las cajas S_i

- Cada uno de estos bloques B_i es utilizado como entrada para una **función de selección-sustitución S_i** que tiene como salida un bloque de 4 bits, de manera que si un bloque

$$B_i = b_1 b_2 b_3 b_4 b_5 b_6$$

la salida se obtiene de la siguiente manera:

- $b_2 b_3 b_4 b_5$ representa un número binario cuyo valor decimal está comprendido entre **0 y 15**, este selecciona una **columna en la tabla**.
- $b_1 b_6$ representa un número binario cuyo valor decimal está comprendido entre **0 y 3**. Este número selecciona **una fila de la tabla**.
- La salida de $S_i(B_i)$ es la expresión binaria del **número seleccionado en la tabla**.
- Se obtienen así **8 bloques de 4 bits** que se concatenan.



4.2.2 La función f y las cajas S_i

S1	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	S2	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0	14 4 13 1 2 15 11 8 3 10 6 12 5 9 0 7	0	15 1 8 14 6 11 3 4 9 7 2 13 12 0 5 10
1	0 15 7 4 14 2 13 1 10 6 12 11 9 5 3 8	1	3 13 4 7 15 2 14 12 0 1 10 6 9 11 5
2	4 1 14 8 13 6 2 11 15 12 9 7 3 10 5 0	2	0 14 7 11 10 → 13 1 5 8 12 6 9 3 2 15
3	15 12 8 2 4 9 1 7 5 11 3 14 10 0 6 13	3	13 8 10 1 3 15 4 2 11 6 7 12 0 5 14 9
S3	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	S4	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0	10 0 9 14 6 3 15 5 1 13 12 7 11 4 2 8	0	7 13 14 3 0 9 6 10 1 2 8 5 11 12 4 15
1	13 7 0 9 3 4 6 10 2 8 5 14 12 11 15 1	1	13 8 11 5 6 15 0 3 4 7 2 12 1 10 14 9
2	13 6 4 9 8 15 3 0 11 1 2 12 5 10 14 7	2	10 6 9 0 12 11 7 13 15 1 3 14 5 2 8 4
3	1 10 13 0 6 9 8 7 4 15 14 3 11 5 2 12	3	3 15 0 6 10 1 13 8 9 4 5 11 12 7 2 14
S5	0 1 2 3		15 11 8 6 13 15 7 2 8
S7	0 1 2 3		6 11 13 8 1 4 10 7 9 5 0 15 14 2 3 12 3 2 14 7 4 10 8 13 15 12 9 0 3 5 6 11

Ejemplo:

Sean los bits 7 al 12 los siguientes: 101100

Los bits corresponderán entonces a la entrada de la caja S₂

Para seleccionar la fila tomamos los bits extremos: 10 = 2

Para seleccionar la columna tomamos los bits centrales: 0110 = 6

La caja S₂ indica una salida igual a 13 = 1101



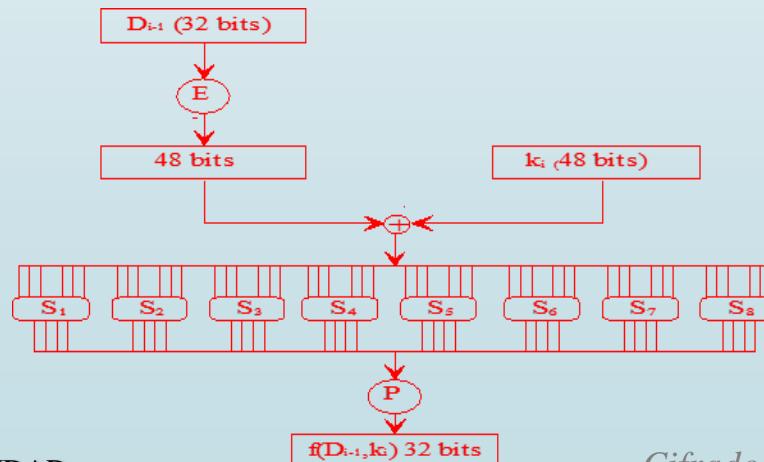
4.2.2 La función f y las cajas S_i

- Finalmente la salida de la función f se obtiene aplicando la permutación P

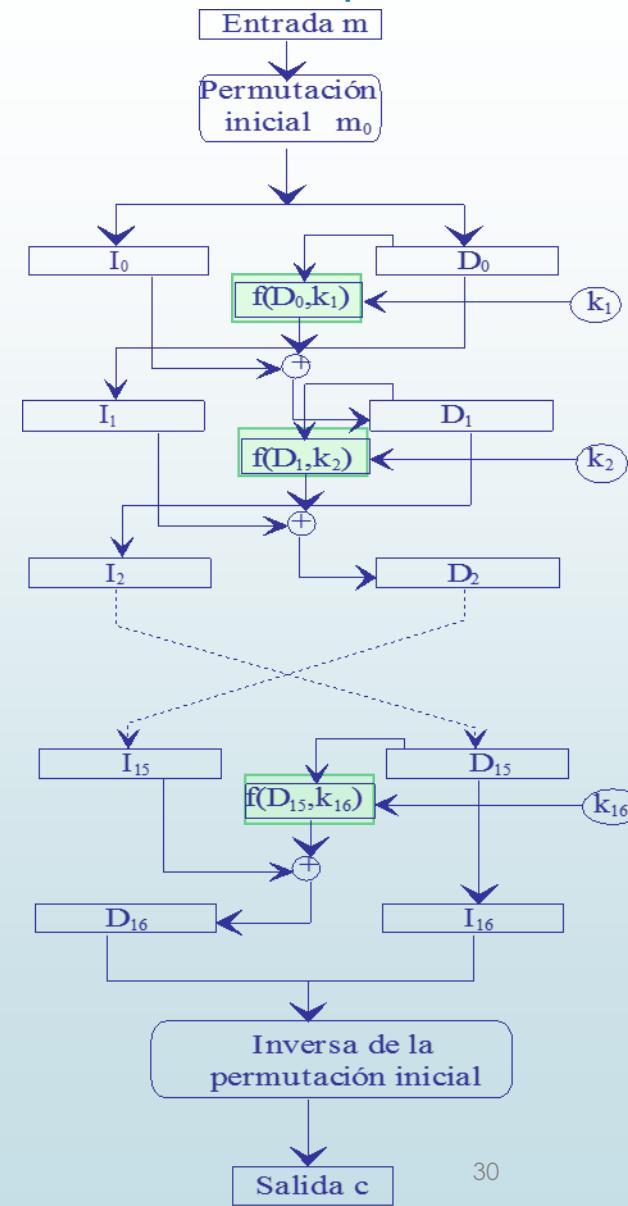
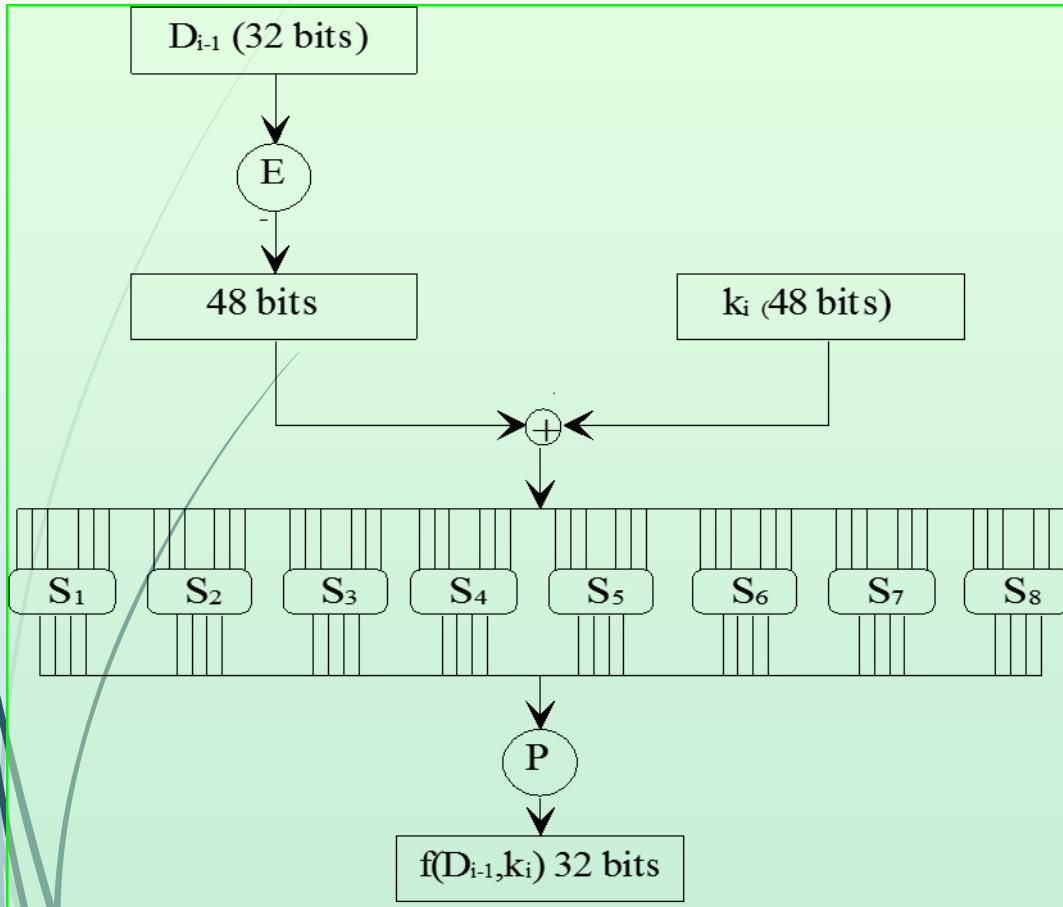
16	07	20	21
29	12	28	17
01	15	23	26
05	18	31	10
02	08	24	14
32	27	03	09
19	13	30	06
22	11	04	25

sobre el bloque de 32 bits obtenido, esto es

$$f(D_{i-1}, k_i) = P[S_1(B_1) \ S_2(B_2) \ S_3(B_3) \ S_4(B_4) \ S_5(B_5) \ S_6(B_6) \ S_7(B_7) \ S_8(B_8)]$$



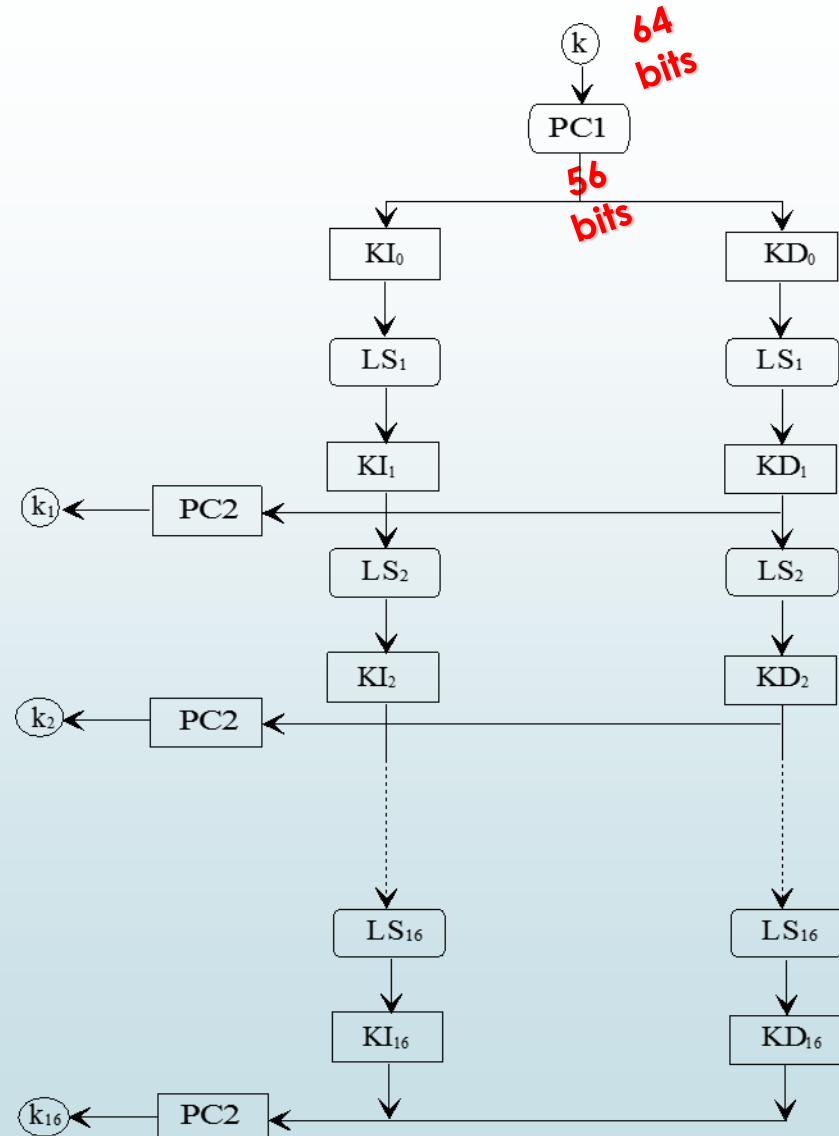
4.2.2 La función f y las cajas S_i



4.2.3 Cálculo de las subclaves k_i

- La clave k tiene inicialmente 64 bits. Los que ocupan las posiciones múltiplo de ocho: 8,16,24,32,40,48,56,64, controlan la paridad de los siete anteriores.
- En cada iteración se obtiene una subclave k_i a partir k .
- En primer lugar se eliminan los ocho bits de paridad de la clave y se aplica la permutación PC1 (permuted choice 1)

57	49	41	33	25	17	09
01	58	50	42	34	26	28
10	02	59	51	43	35	27
19	11	03	60	52	44	36
63	55	47	39	31	23	15
07	62	54	46	38	30	22
14	06	61	53	45	37	29
21	13	05	28	20	12	04



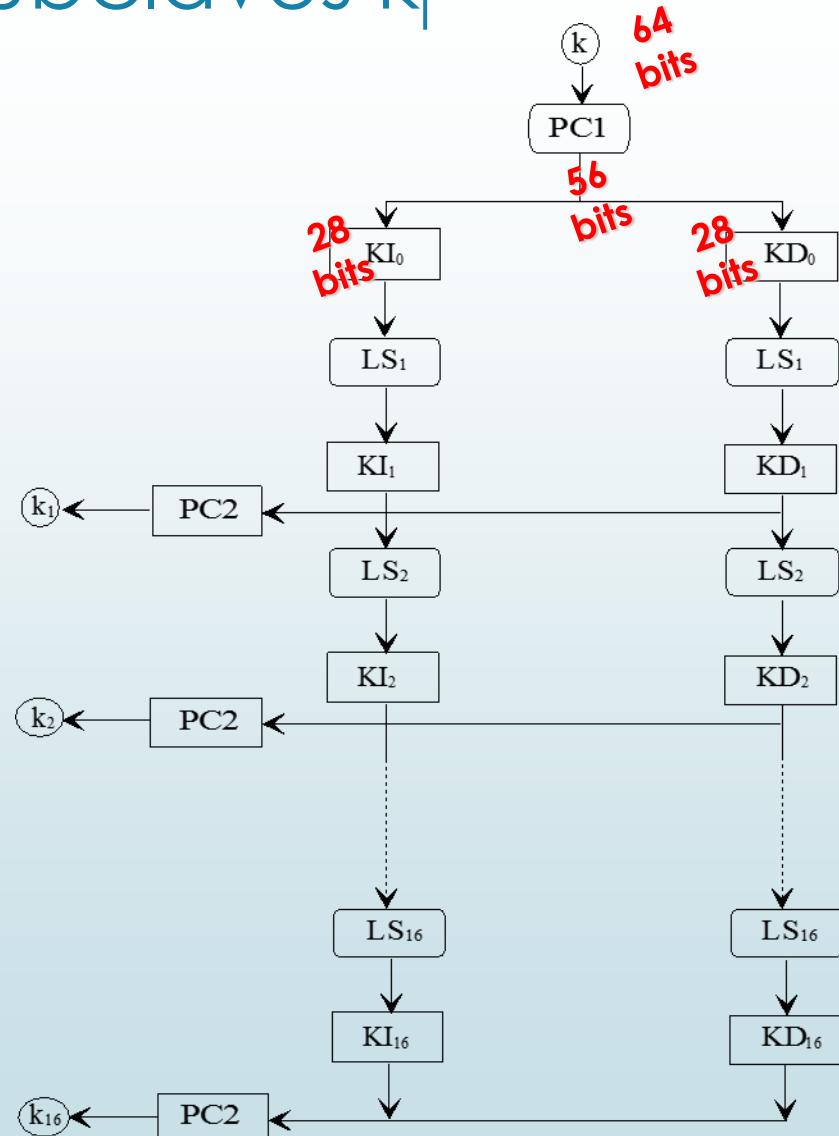
4.2.3 Cálculo de las subclaves k_i

- Una vez efectuada la permutación, los 56 bits se **dividen** en dos bloques de 28 cada uno, KI_0 y KD_0

$$PC1(k) = KI_0 \ KD_0$$

- La obtención de los bloques siguientes, $KI_i \ KD_i$, se hace **siempre a partir del bloque anterior $KI_{i-1} \ KD_{i-1}$** efectuando cada mitad una rotación a la izquierda que depende de cada iteración de acuerdo con

I	bits	I	bits
1	1	9	1
2	1	10	2
3	2	11	2
4	2	12	2
5	2	13	2
6	2	14	2
7	2	15	2
8	2	16	1



4.2.3 Cálculo de las subclaves k_i

- Se tiene

$$KI_i = LS_i(KI_{i-1}) \quad KD_i = LS_i(KD_{i-1})$$

donde LS_i representa el desplazamiento a **izquierda** a efectuar en la iteración i.

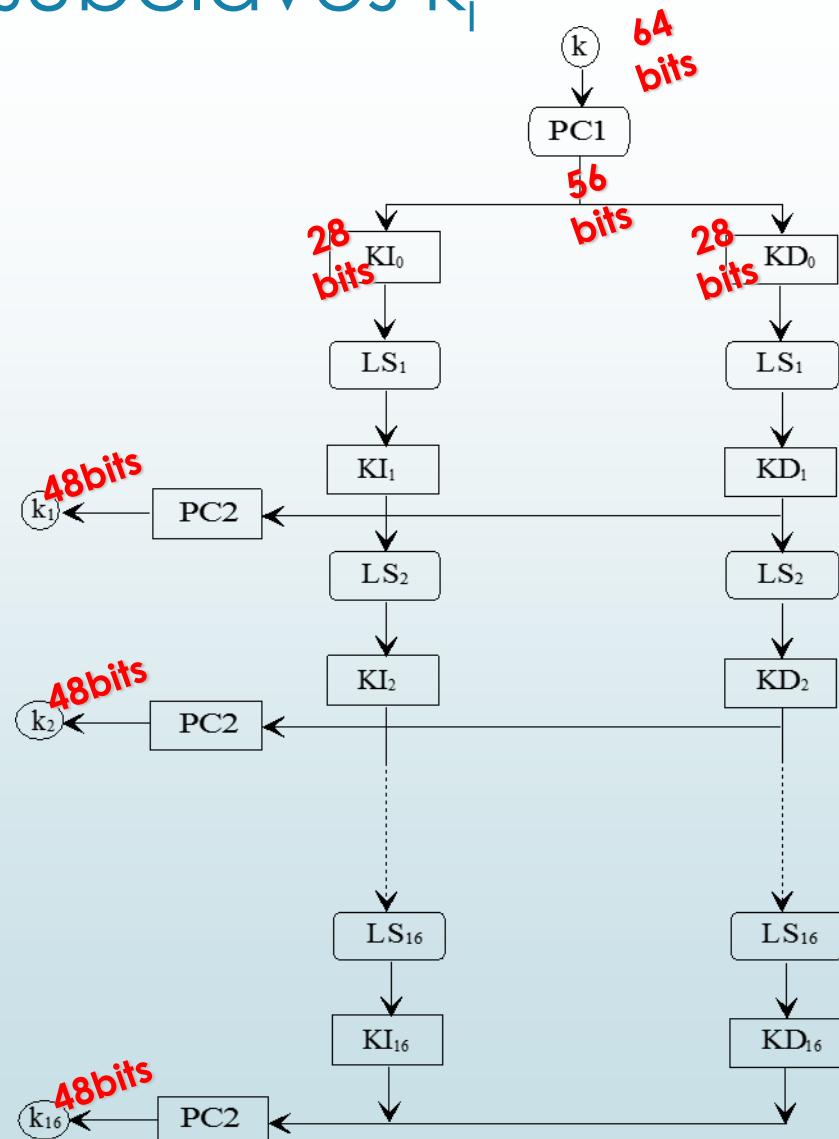
- La subclave k_i viene dada por

$$k_i = PC2(KI_i \mid\mid KD_i)$$

siendo $PC2$ la permutación dada en la tabla

14	17	11	24	01	05
03	28	15	06	21	10
23	19	12	04	26	08
16	07	27	20	13	02
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Se han eliminado los bits 9, 18, 22, 25, 35, 38, 43 y 54



4.2.4 Descifrado de DES

- Este algoritmo, como ya se ha comentado con anterioridad, es válido para descifrar los criptogramas.
- La única variación estriba en la utilización de las subclave k_i que se efectúa en orden inverso, o sea, k_{16} se utiliza en la primera iteración, k_{15} en la segunda y así sucesivamente; esto es así porque la iteración final es la inversa de la inicial. Se tiene además
- Para la obtención de las distintas subclaves se puede utilizar únicamente k_{16} y aplicar las rotaciones en orden inverso, de acuerdo con la tabla

I	bits	I	bits
1	0	9	1
2	1	10	2
3	2	11	2
4	2	12	2
5	2	13	2
6	2	14	2
7	2	15	2
8	2	16	1



4.2.5 Propiedades y características de DES

- ▶ **Dependencia entre símbolos:** Cada bit del texto cifrado es una función compleja de **TODOS** los bits de la clave y **TODOS** los bits del texto original (por bloques).
- ▶ **Cambio de los bits de entrada:** Un cambio de un bit en el mensaje original produce el cambio del 50 %, aproximadamente, de los bits del bloque cifrado.
- ▶ **Cambio de los bits de clave:** Un cambio en un bit de la clave produce, aproximadamente, el cambio de la mitad de los bits del bloque cifrado.



4.2.5 Propiedades y características de DES

- **Claves débiles:** Existen cuatro claves «**débiles**» que producen un mensaje cifrado fácil de descifrar, porque todas las claves parciales k_1 a k_{16} son iguales. Existen 28 claves «**semidébiles**» que producen un mensaje cifrado fácil de descifrar, porque producen sólo dos o cuatro subclaves parciales diferentes. Cuando se elige una clave al azar, es preciso asegurarse de que no se ha producido una de estas claves.

Clave	Clave tras aplicar PC1
0101010101010101	0000000000000000
1F1F1F1F0E0E0E0E	0000000FFFFFFF
E0E0E0E0F1F1F1F1	FFFFFFF00000000
F0F0F0F0F0F0F0F0	FFFFFFFFF00000000

Clave	Clave tras aplicar PC1
01FE01FE01FE01FE	AAAAAAAAAAAAAA
E01FE01FE01FE01	55555555555555
1FE01FE00EF10EF1	AAAAAAA55555555
E01FE01FF10EF10E	5555555AAAAAAA
01E001E001F101F1	AAAAAAA0000000
E001E001F101F101	55555550000000
1FFE1FFE0EFE0EFE	AAAAAAAFFFFFFF
FE1FFE1FFE0EFE0E	5555555FFFFFFF
011F011F010E010E	0000000AAAAAAA
1F011F010E010E01	00000005555555
E0FEE0F0F1F0F1F0	FFFFFFFAAAAAAA
F0FEE0F0F1F0F1F0	FFFFFFF55555555



4.2.6 Seguridad de DES

- En 1976, antes de que el National Bureau of Standards aprobara el proyecto DES se encargaron dos estudios sobre la seguridad del algoritmo. **Diffie, Hellman y otros encontraron ciertas debilidades en el mismo.**
- El tamaño de la clave, 56 bits, hace que el sistema sea vulnerable dado que el conjunto de claves resulta demasiado pequeño:
 2^{56} posibilidades.



4.2.6 Seguridad de DES

- Martin Hellman en su artículo "DES will be totally insecure within ten years" (1979), aseguraba que un millón de procesadores trabajando en paralelo, cada uno de ellos procesando un millón de claves por segundo, recorrerían todo el espacio de claves en poco más de veinte horas.
- Como es probable que, en promedio, sólo se tenga que buscar en la mitad del conjunto de claves, un ataque al criptosistema con una máquina de tales características, conociendo un texto en claro y su correspondiente criptograma necesitaría de **poco más de diez horas** para encontrar la clave

$$\frac{2^{56}}{3'6 \cdot 10^{15}} = \frac{7'2 \cdot 10^{16}}{3'6 \cdot 10^{15}} = 20$$



4.2.6 Seguridad de DES

- El **18 de junio de 1997**, un esfuerzo coordinado a través de Internet, en respuesta a un desafío de RSA Data Security, Inc. permitió el descifrado de un mensaje cifrado con una clave DES de 56 bits utilizando la **fuerza bruta**.
- El tiempo invertido desde que comenzó el análisis de las posibles claves hasta que se consiguió descifrar el mensaje fue de **cuatro meses**.
- Sin embargo, se trataba de un ataque no demasiado práctico, puesto que se tardaron cuatro meses en "reventar" un único mensaje.



4.2.6 Seguridad de DES

- RSA Data Security, Inc. lanzó una nueva serie de desafíos, estableciendo premios, de modo que estos sólo se ganaran si el tiempo empleado en reventar una clave era inferior al 25% del empleado en el desafío exitoso anterior.
- El **13 de enero de 1998** se lanzó el **primero de los dos desafíos** conocidos como **DES-II Challenges**.
- La organización distributed.net utilizó el esfuerzo cooperativo de cientos de ordenadores (unos 50.000) a través de Internet, para reventar la clave utilizando la fuerza bruta.
- Tardó **39 días** en conseguir sus propósitos.



4.2.6 Seguridad de DES

- La puntilla llegó el **17 de julio de 1998**. Tras el lanzamiento del **segundo desafío** (el 13 de julio de 1998) y, con el fin de demostrar la inseguridad de DES, la Electronic Frontier Foundation anunció la construcción de una máquina especializada diseñada para reventar mensajes cifrados con DES

Esta máquina, bautizada apropiadamente como **DES Cracker**, fue construida por menos de 250.000 dólares (unos 40 millones de pesetas), y fue capaz de ganar fácilmente el segundo desafío, tardando apenas tres días.

- Los detalles de diseño de esta máquina se encuentran completamente documentados en el libro publicado por la propia EFF y O'Reilly and Associates, titulado "*Cracking DES: Secrets of Encryption Research, Wiretap Politics, and Chip Design*".
 - Curiosamente, la exportación de la propia máquina no está permitida por las leyes de EE.UU. El libro, sin embargo, puede ser comprado libremente y utilizado para la implementación del DES Cracker en cualquier sitio.



4.2.6 Seguridad de DES

19-01-1999 - DES III

- En un esfuerzo conjunto de distributed.net y EFF con el mítico DES Cracker, se consiguió romper el estándar de cifrado en menos de un día:
 - En el **tiempo récord de 22 horas y 15 minutos** el **desafío DES III** sucumbió ante el poder de unos 100.000 PCs y un superordenador especialmente diseñado por la EFF.
 - El equipo combinado comprobó más de **245.000 millones de claves cada segundo**.
- En esta ocasión, el escrito tras la clave ocultaba la siguiente misiva "**See you in Rome (second AES Conference, March 22-23, 1999)**".



4.2.6 Seguridad de DES

- ▶ La resolución en este tiempo récord vino a demostrar muchas cosas en el mundo de la informática y la seguridad.
- ▶ En primer lugar la debilidad del algoritmo DES y la necesidad de buscar un nuevo **sustituto** de forma inmediata y
- ▶ por otro lado la efectividad del uso de la **computación distribuida** para la resolución de problemas que requieran una gran potencia de cálculo.



4.2.6 Seguridad de DES

- ▶ En 1996, el Instituto Nacional de Estándares y Tecnología (*National Institute of Standards and Technology*, NIST) dio los primeros pasos para la consolidación de **un Estándar de Cifrado Avanzado** (*Advanced Encryption Standard*, AES).
- ▶ Su objetivo fue desarrollar una especificación para encontrar un algoritmo de cifrado que sustituyese a DES de manera que el nuevo algoritmo fuese capaz de proteger la información sensible de los ciudadanos y del gobierno hasta bien entrado el siglo XXI.



4.3 Algoritmo AES

- El 2 de octubre de 2000 el **NIST (National Institute for Standards and Technology)** anunciaba oficialmente la adopción del algoritmo **Rijndael** como nuevo **Estándar Avanzado de Cifrado (AES)** para su empleo en aplicaciones criptográficas no militares, culminando así un proceso de mas de tres años, encaminado a proporcionar a la comunidad internacional un nuevo algoritmo de cifrado potente, eficiente y fácil de implementar.

DES tiene un sucesor

- La palabra **Rijndael** es un acrónimo formado por los nombres de sus dos autores, los belgas
 - Vincent Rijmen y Joan Daemen.*
- Su interés radica en que todo el proceso de selección, revisión y estudio tanto de este algoritmo como de los restantes candidatos, se ha efectuado de forma pública y abierta, por lo que, prácticamente por primera vez, toda la comunidad criptográfica mundial ha participado en su análisis, lo cual convierte a **Rijndael** en un algoritmo perfectamente digno de la confianza de todos.



4.3.1 Introducción y descripción

- Los tres aspectos básicos sobre los que se ha diseñado **Rijndael** son los siguientes:
 - **Resistencia** contra todo tipo de ataque conocido hasta ese momento.
 - **Eficiencia** computacional en un amplio abanico de plataformas, tanto hardware como software (optimizado para 32 bits).
 - **Simplicidad** de diseño.



4.3.1 Introducción y descripción

- Rijndael es un sistema de cifrado por bloques, diseñado para manejar longitudes de clave y de bloque variables, ambas comprendidas entre 128 y 256 bits.
- Para el estándar AES se adoptó un tamaño de bloque fijo e igual a 128 bits y tres tamaños de clave:
 - 128 bits (AES128)
 - 192 bits (AES192)
 - 256 bits (AES 256).

Longitud de bloque :

- 128 bits (AES: bloque 128bits)
- ~~192 bits~~
- ~~256 bits~~

Rijndael

Diferentes modos de cifrado

- ECB
- CBC
- OFB
- CFB
- CTR



4.3.1 Introducción y descripción

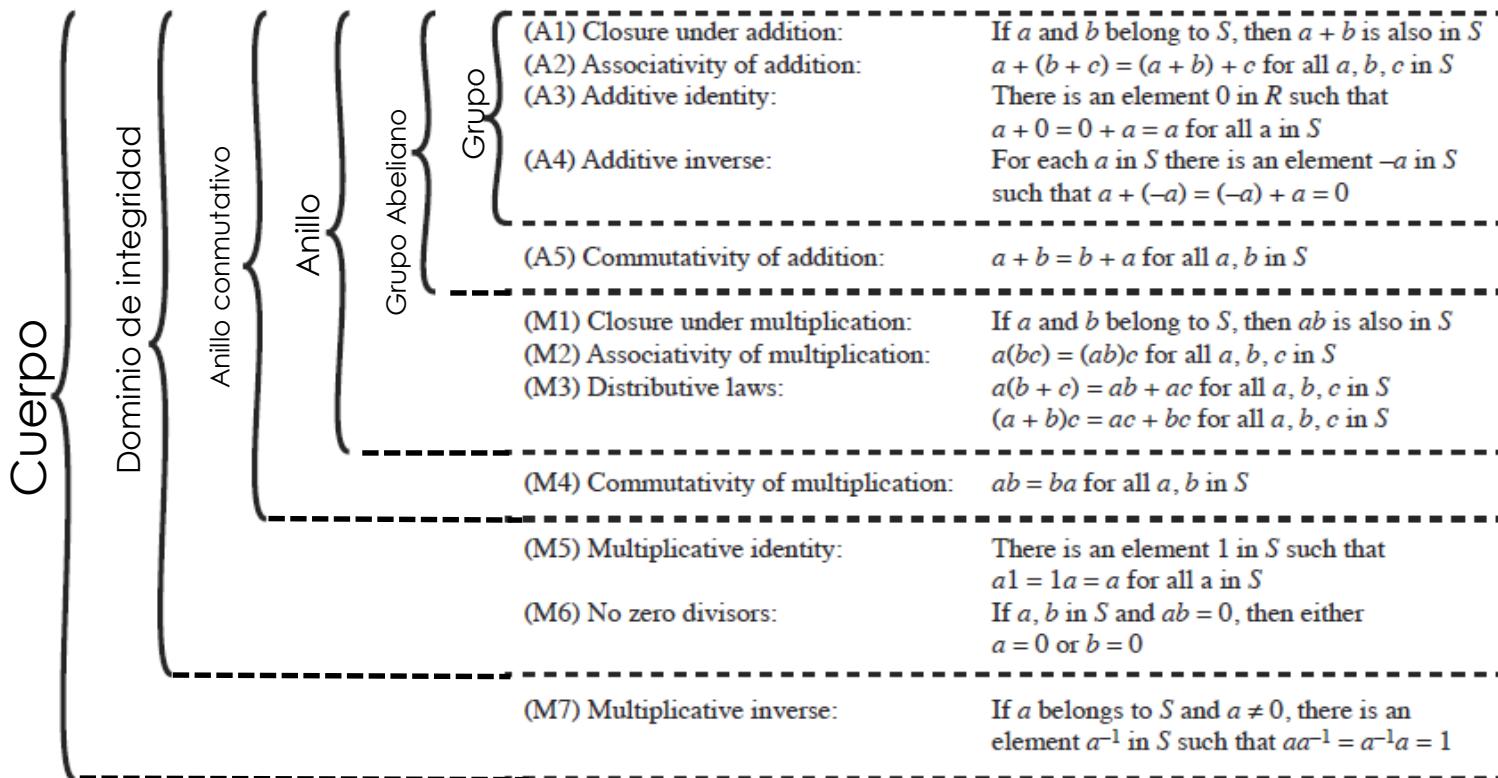
- Realiza varias de sus operaciones internas a nivel de byte, interpretando estos como elementos de un campo de Galois $GF(2^8)$. 
- El resto de operaciones se efectúan en términos de registros de 32 bits.
- Sin embargo, en algunos casos, una secuencia de 32 bits se toma como un polinomio de grado inferior a 4, cuyos coeficientes son a su vez polinomios en $GF(2^8)$. 



4.3.1 Introducción y descripción

La operación suma en el conjunto \mathbb{Z}_n cumple las propiedades asociativa y conmutativa y posee elementos neutro y simétrico por lo que tiene estructura de grupo conmutativo (o abeliano).

Se le denomina **grupo finito inducido por n**.



4.3.1 Introducción y descripción

- Ya se ha comentado que no tiene por qué existir siempre el inverso para el producto.

- **Teorema:**

Si $\text{mcd}(a,n) = 1$, a tiene inverso módulo n.

- **Corolario:**

Si n es primo, el grupo finito que induce tiene estructura de cuerpo (*field*), o sea:

todos sus elementos tienen inverso para el producto excepto el cero.

Nota: Estos cuerpos finitos tienen una gran importancia en la Criptografía, se denominan **Campos de Galois** y se denotan **GF(n)**



4.3.1 Introducción y descripción

- ▶ En el conjunto $\mathbf{Z}_2[x]$ de los polinomios con coeficientes en \mathbf{Z}_2 , todos los coeficientes de los polinomios son 0 ó 1, por lo que un polinomio puede ser representado mediante una secuencia de bits.
- ▶ Por ejemplo, $f(x)=x^3+x+1$ podría representarse mediante la cadena binaria 1011 y $g(x) = x^2 + 1$ vendría dado por la cadena 101.
- ▶ Observemos que $f(x)+g(x)= x^3+x^2+x$, que puede representarse por 1110.
- ▶ Puesto que las operaciones se realizan en \mathbf{Z}_2 , esta suma podría haber sido realizada mediante una simple operación or-exclusiva entre las cadenas binarias que representan a $f(x)$ y $g(x)$.



4.3.1 Introducción y descripción

- Si escogemos un polinomio irreducible en $\mathbf{Z}_2[x]$, podemos generar un cuerpo finito, denominado "**campo de Galois**".
- Dicho conjunto se representa como $GF(2^n)$, siendo n el grado del polinomio irreducible que lo genera.
- En AES se utiliza el polinomio irreducible $m(x) = x^8+x^4+x^3+x+1$ y se trabaja en un campo de Galois $GF(2^8)$.



4.3.1 Introducción y descripción

- » AES Realiza varias de sus operaciones internas a nivel de byte, interpretando estos como elementos de un campo de Galois $GF(2^8)$.

$$\mathbf{Z}_2 = \{0,1\}$$

- » Cada byte $\{b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0\}$ se toma como un polinomio de grado 7 con coeficientes en

$$b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x^1 + b_0x^0$$

- » La ventaja esencial que posee este tipo de conjuntos es que permite llevar a cabo implementaciones muy sencillas y paralelizables de los algoritmos aritméticos.



4.3.1 AES: preliminares matemáticos (suma)

- La suma y la resta de polinomios en GF(2⁸) se corresponde con XOR.

- Ejemplo:**

- Notación polinómica:

$$(x^6 + x^4 + x^2 + x + 1) + (x^7 + x + 1) = x^7 + x^6 + x^4 + x^2$$

- Notación binaria:

$$\{01010111\} \oplus \{10000011\} = \{11010100\}$$

- Notación hexadecimal:

$$\{57_{16}\} \oplus \{83_{16}\} = \{D4_{16}\}$$



4.3.1 AES: prelim. matemáticos (producto)

La operación producto \otimes de AES, se corresponde con el producto de polinomios módulo el polinomio irreducible de grado 8

$$m(x) = x^8 + x^4 + x^3 + x + 1$$

Ejemplo:

► Notación hexadecimal:

$$\{57_{16}\} \otimes \{83_{16}\} = \{C1_{16}\}$$

► Notación binaria:

$$\{01010111\} \otimes \{10000011\} = \{11000001\}$$

Veamos porqué

► Notación polinómica:

$$(x^6 + x^4 + x^2 + x + 1) \otimes (x^7 + x + 1) = x^7 + x^6 + 1$$



4.3.1 AES : prelim. matemáticos (producto)


$$\begin{aligned}(x^6 + x^4 + x^2 + x + 1) (x^7 + x + 1) &= x^{13} + x^{11} + x^9 + x^8 + x^7 + \\&\quad x^7 + x^5 + x^3 + x^2 + x + \\&\quad x^6 + x^4 + x^2 + x + 1 \\&= x^{13} + x^{11} + x^9 + x^8 + \cancel{x^7} + \cancel{x^7} + \\&\quad x^6 + x^5 + x^4 + x^3 + \cancel{x^2} + \cancel{x^2} + \cancel{x} + \cancel{x} + 1 \\&= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1\end{aligned}$$

■ Este producto hay que reducirlo módulo $m(x) = x^8 + x^4 + x^3 + x + 1$

$x^8 + x^4 + x^3 + x + 1 \bmod m(x) = 0$, luego

$$\begin{aligned}x^8 + \cancel{x^4} + \cancel{x^3} + \cancel{x} + 1 &+ \\+ \cancel{x^4} + \cancel{x^3} + \cancel{x} + 1 &\bmod m(x) = 0 + x^4 + x^3 + x + 1\end{aligned}$$

Esto es:

$$x^8 \bmod m(x) = x^4 + x^3 + x + 1$$



4.3.1 AES: prelim. matemáticos (producto)

$$(x^6 + x^4 + x^2 + x + 1) (x^7 + x + 1) \text{ mod } m(x) =$$

$$= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \text{ mod } m(x) =$$

$$= (x^6 + x^3 + x^2 + 1) + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \text{ mod } m(x)$$

$$x^{13} \text{ mod } m(x) = x^5 x^8 \text{ mod } m(x)$$

$$= x^5 (x^4 + x^3 + x + 1) \text{ mod } m(x)$$

$$= x^9 + x^8 + x^6 + x^5 \text{ mod } m(x)$$

$$= x (x^8 + x^8 + x^6 + x^5) \text{ mod } m(x)$$

$$= x (x^4 + x^3 + x + 1) + (x^4 + x^3 + x + 1) + x^6 + x^5 \text{ mod } m(x)$$

$$= (x^5 + x^4 + x^2 + x) + (x^4 + x^3 + x + 1) + x^6 + x^5 \text{ mod } m(x)$$

$$= x^6 + x^5 + x^5 + x^4 + x^4 + x^3 + x^2 + x + x + 1 \text{ mod } m(x)$$

$$= x^6 + x^3 + x^2 + 1$$



4.3.1 AES: prelim. matemáticos (producto)

$$(x^6 + x^4 + x^2 + x + 1) (x^7 + x + 1) \text{ mod } m(x) =$$

$$= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \text{ mod } m(x) =$$

$$= (x^6 + x^3 + x^2 + 1) + (x^7 + x^6 + x^4 + x^3) + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \text{ mod } m(x)$$

$$x^{11} \text{ mod } m(x) = x^3 x^8 \text{ mod } m(x)$$

$$= x^3 (x^4 + x^3 + x + 1) \text{ mod } m(x)$$

$$= x^7 + x^6 + x^4 + x^3$$



4.3.1 AES: prelim. matemáticos (producto)

$$(x^6 + x^4 + x^2 + x + 1) (x^7 + x + 1) \text{ mod } m(x) =$$

$$= x^{13} + x^{11} + \cancel{x^9} + \cancel{x^8} + x^6 + x^5 + x^4 + x^3 + 1 \text{ mod } m(x)$$

$$\begin{aligned} &= (x^6 + x^3 + x^2 + 1) + (x^7 + x^6 + x^4 + x^3) + (\cancel{x^5} + \cancel{x^4} + \cancel{x^2} + x) + (\cancel{x^4} + \cancel{x^3} + x + 1) \\ &\quad + x^6 + x^5 + x^4 + x^3 + 1 \text{ mod } m(x) \end{aligned}$$

$$x^9 \text{ mod } m(x) = x \cancel{x^8} \text{ mod } m(x)$$

$$= x (\cancel{x^4} + \cancel{x^3} + x + 1) \text{ mod } m(x)$$

$$= \cancel{x^5} + \cancel{x^4} + \cancel{x^2} + x$$



4.3.1 AES: polinomios con coeficientes en GF(2⁸)

- En el AES, algunas palabras de 32 bits, se toman como un polinomio de grado menor o igual a 3 cuyos coeficientes son, a su vez, polinomios en GF(2⁸).

$$a(x) = a_3 x^3 + a_2 x^2 + a_1 x + a_0$$

- Para sumar o multiplicar dos polinomios de este tipo se actúa con la operatoria general definida para polinomios, sustituyendo las operaciones de los coeficientes por las operaciones definidas en GF(2⁸), \oplus y \otimes .



4.3.1 AES: estructura

- ▶ AES, a diferencia de algoritmos como DES, no posee estructura de red Feistel (se denomina cifradores **tipo Feistel** a aquellos en los que el bloque de datos se divide en dos mitades y en cada vuelta de cifrado se trabaja, alternadamente, con una de las mitades)
- ▶ En su lugar se ha definido cada **ronda** como una composición de cuatro funciones invertibles diferentes:
 - ▶ **DesplazarFila**
 - ▶ **MezclarColumnas**
 - ▶ **ByteSub**
 - ▶ **XOR**

formando tres capas (layer)

- ▶ **capa de mezcla lineal**
- ▶ **capa no lineal**
- ▶ **capa de adición de clave**

diseñadas para proporcionar resistencia frente a criptoanálisis lineal y diferencial.



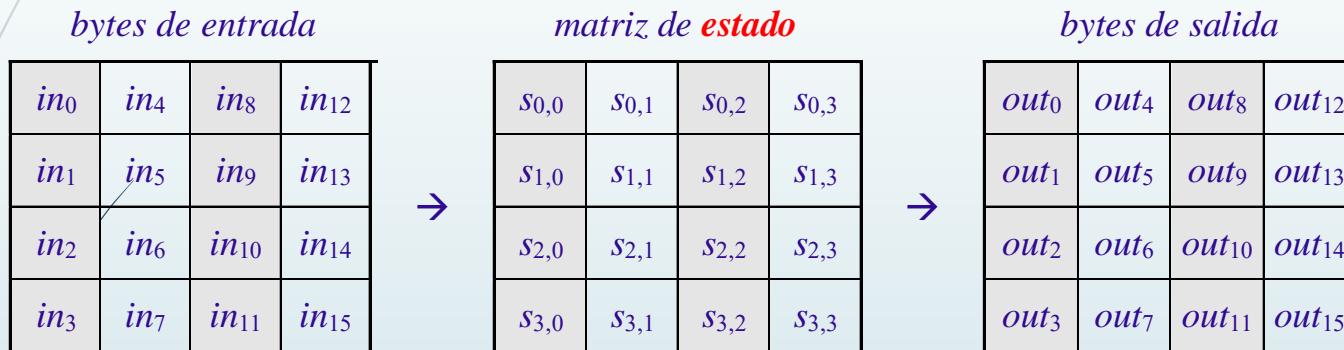
4.3.1 AES: estructura

- Cada una de las funciones tiene un propósito preciso:
 - La **capa de mezcla lineal** (funciones **DesplazarFila** y **MezclarColumnas**) permite obtener un alto nivel de difusión a lo largo de varias rondas.
 - La **capa no lineal** (función **ByteSub**) consiste en la aplicación paralela de s-cajas con propiedades óptimas de no linealidad.
 - La **capa de adición de clave** es un simple **XOR** entre el estado intermedio y la subclave correspondiente a cada ronda.



4.3.1 AES: elementos

- AES es un algoritmo que se basa en aplicar un número determinado de rondas a un valor intermedio denominado **estado** que puede representarse mediante una matriz rectangular de bytes, que posee 4 filas y 4 columnas



NOTA:

- En el algoritmo ganador del concurso AES, **Rijndael**, el tamaño de bloque que se puede elegir es 128, 192 o 256 bits. En el mismo se define N_b como el tamaño del bloque a cifrar dividido por 32 .
- Finalmente se adoptó como estándar AES el algoritmo Rijndael con tamaño de bloque 128 bits, por lo que $N_b = \frac{128}{32} = 4$.



4.3.1 AES: elementos (estado)

Bloque de Entrada

in0	in1	in2	in3	in4	in5	in6	in7	in8	in9	in10	in11	in12	in13	in14	in15
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------

Entrada

in0	in4	in8	in12
in1	in5	in9	in13
in2	in6	in10	in14
in3	in7	in11	in15

Matriz de Estado

S _{0,0}	S _{0,1}	S _{0,2}	S _{0,3}
S _{1,0}	S _{1,1}	S _{1,2}	S _{1,3}
S _{2,0}	S _{2,1}	S _{2,2}	S _{2,3}
S _{3,0}	S _{3,1}	S _{3,2}	S _{3,3}

Salida

out0	out4	out8	out12
out1	out5	out9	out13
out2	out6	out10	out14
out3	out7	out11	out15

Bloque de Salida

out0	out1	out2	out3	out4	out5	out6	out7	out8	out9	out10	out11	out12	out13	out14	out15
------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------



4.3.1 AES: elementos (estado)

EJEMPLO:

Para la entrada:

10101000										01100011
----------	---	------	-----	-----	----	-----	--	--	--	--	--	--	--	--	--	----------

A8	F2	45	3D	59	6A	00	32	12	BA	8C	37	6C	E4	23	63
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

se tiene la siguiente matriz de estado

A8	59	12	6C
F2	6A	BA	E4
45	00	8C	23
3D	32	37	63



4.3.1 AES : estado

- La clave tiene una estructura análoga a la del estado, y se representa mediante una matriz de bytes con 4 filas y N_k columnas, este valor depende del tamaño de la clave que puede ser 128, 192 ó 256

$N_k = 4$ (128 bits)
$N_k = 6$ (192 bits)
$N_k = 8$ (256 bits)

Clave de 128 bits

$k_{0,0}$	$k_{0,1}$	$k_{0,2}$	$k_{0,3}$
$k_{1,0}$	$k_{1,1}$	$k_{1,2}$	$k_{1,3}$
$k_{2,0}$	$k_{2,1}$	$k_{2,2}$	$k_{2,3}$
$k_{3,0}$	$k_{3,1}$	$k_{3,2}$	$k_{3,3}$

- En algunos casos, tanto el estado como la clave se consideran como **vectores de registros de 32 bits**, estando cada registro constituido por los bytes de la columna correspondiente, ordenados de arriba a abajo.



4.3.1 AES: algoritmo de cifrado

- Si B es el bloque a cifrar y S la matriz de estado, el algoritmo AES con N_r rondas queda como sigue:

- 1.- Calcular las subclaves K_0, K_1, \dots, K_{N_r} a partir de la clave K .
- 2.- $S \leftarrow B \oplus K_0$
- 3.- Para $i=1,2,\dots, N_r$, aplicar la ronda i -ésima con la subclave K_i

- Como cada ronda es una sucesión de funciones invertibles, el **algoritmo de descifrado** consiste en
 - aplicar las inversas de cada una de las funciones en el orden contrario, utilizando las mismas subclaves K_i que en el cifrado, en orden inverso.



4.3.1 AES: rondas

- El número de rondas, N_r , depende de la longitud de la clave.
- Para el algoritmo estándar del AES el tamaño del bloque de entrada, del estado y del bloque de salida es de 128 bits.
- El tamaño de la clave, N_k puede variar: 128, 192 ó 256.
- El número de rondas N_r es el siguiente:

Longitud de clave (N_k palabras de 32 bits)	Número de rondas N_r
AES-128	4
AES-192	6
AES-256	8



4.3.1 AES: rondas

- 1.- Calcular las subclaves K_0, K_1, \dots, K_{Nr} a partir de la clave K .
- 2.- $S \leftarrow B \oplus K_0$
- 3.- Para $i=1,2,\dots, N_r$, aplicar la ronda i -ésima con la subclave K_i

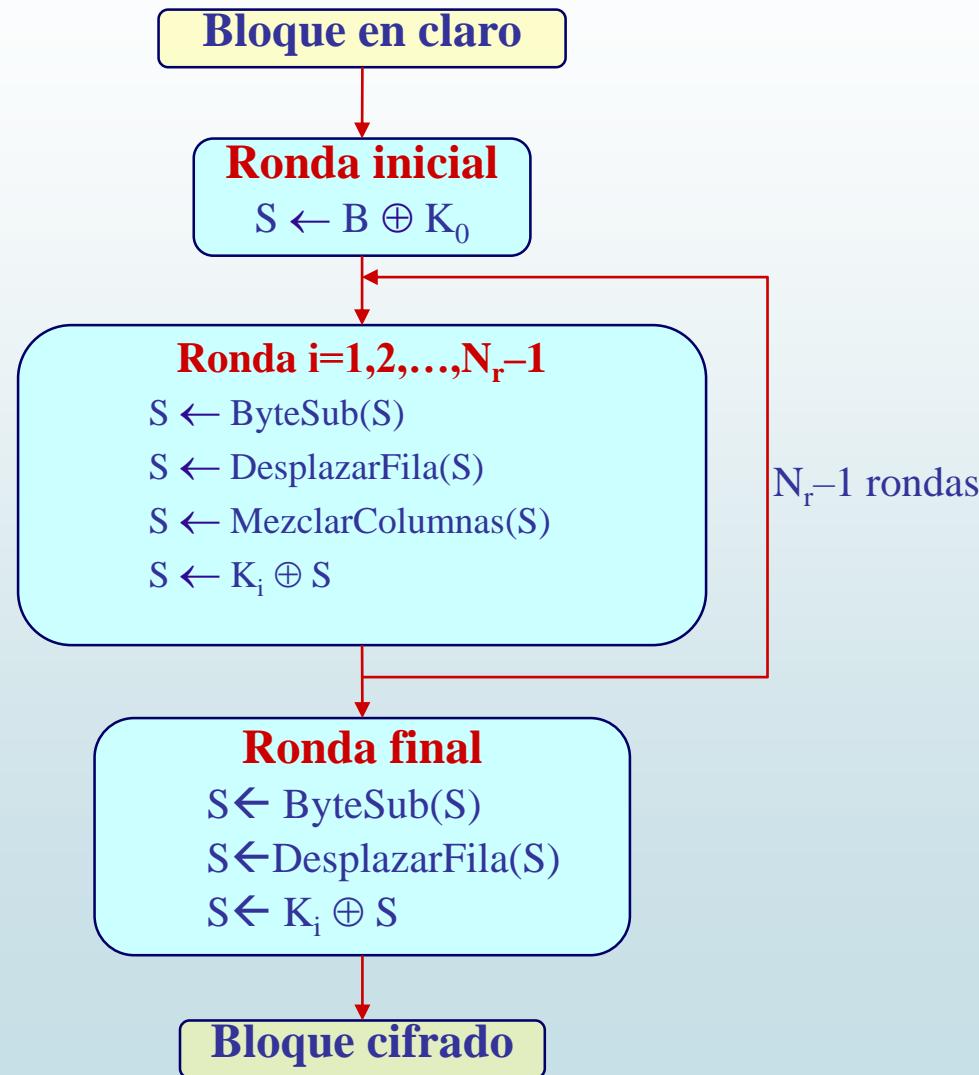
» 3.- Ronda i -ésima

- » 3.1.- $S \leftarrow \text{ByteSub}(S)$
- » 3.2.- DesplazarFila(S)
- » 3.3.- MezclarColumnas(S)
- » 3.4.- $S \leftarrow S \oplus K_i$

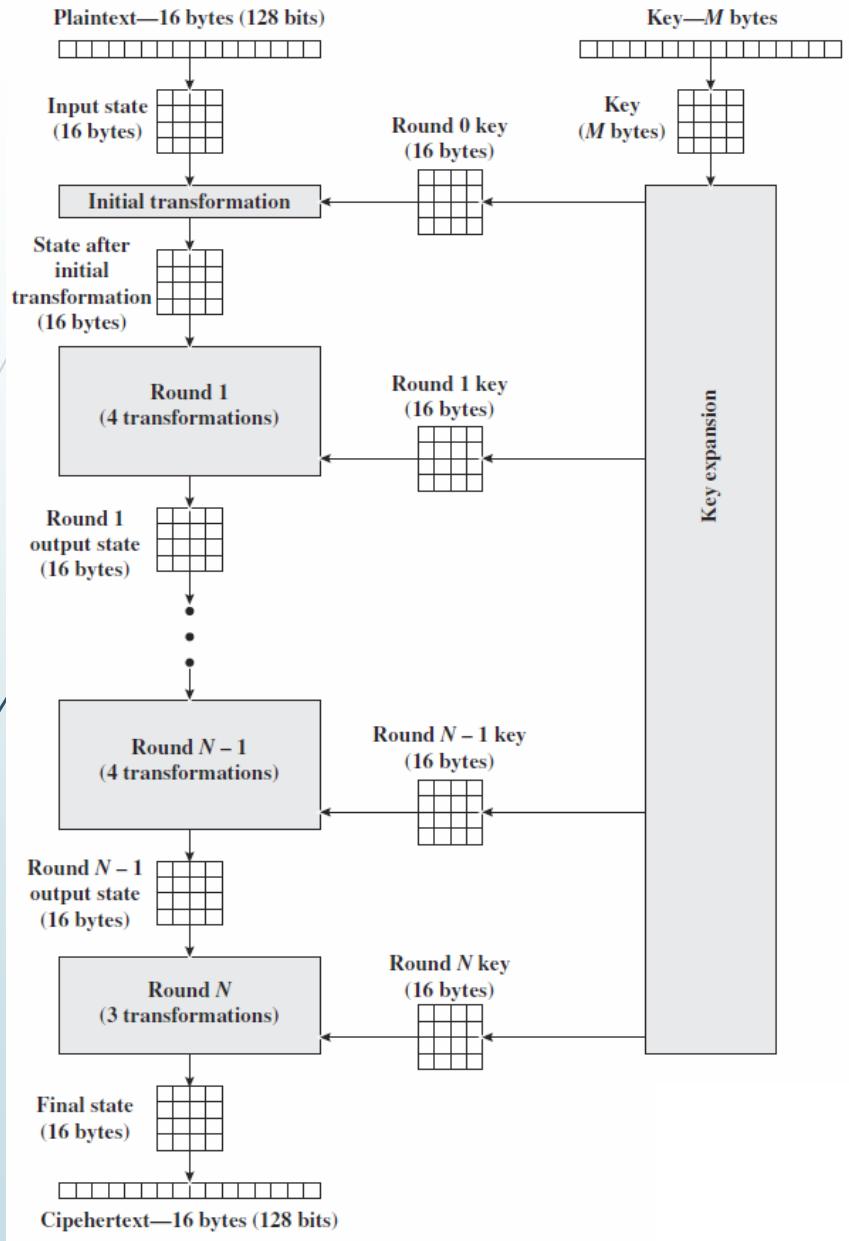
- En la ronda final, no se aplica la función MezclarColumnas(S)



AES: esquema de cifrado

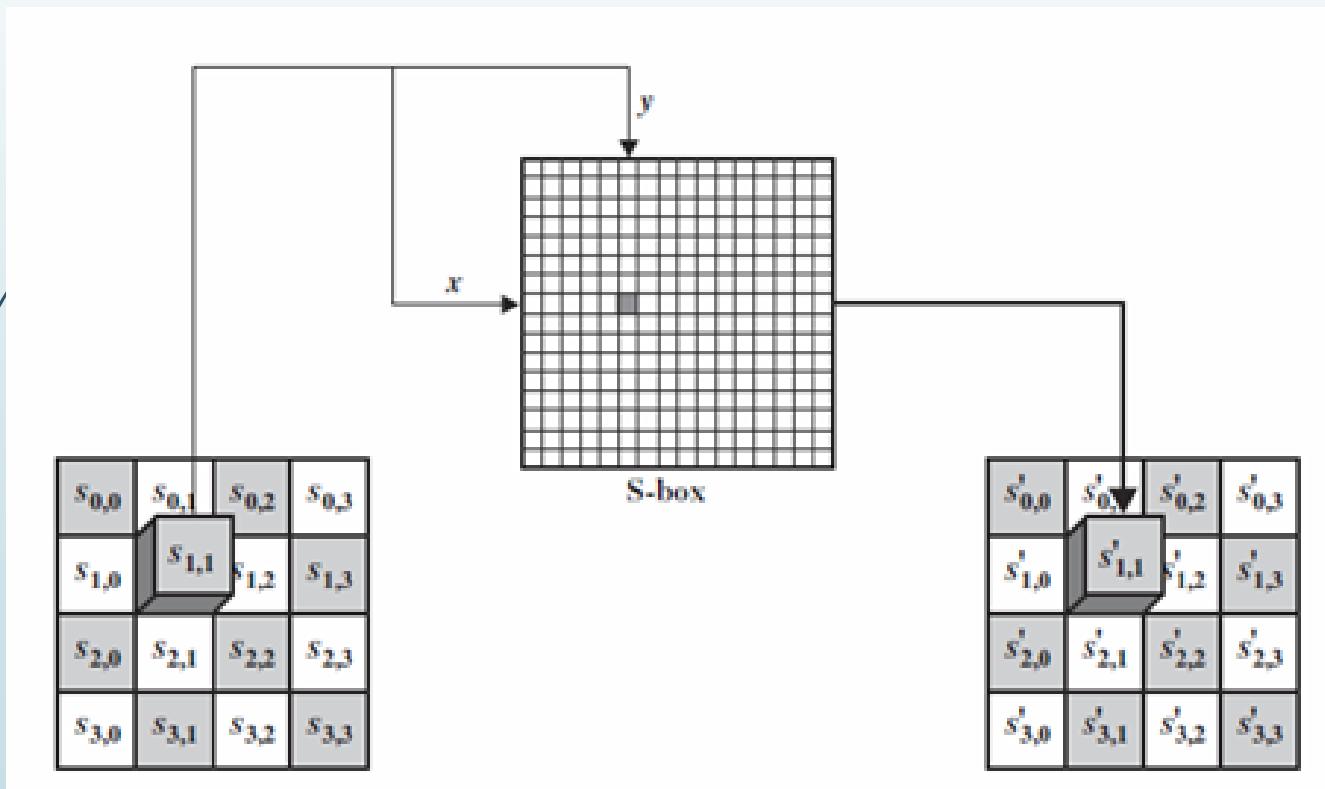


4.3.1 AES: esquema de cifrado



4.3.2 La función ByteSub

- La función ByteSub es una *sustitución no lineal* que se aplica a cada byte de la matriz de estado, mediante una s-caja invertible, que se obtiene componiendo dos transformaciones.



4.3.2 La función ByteSub

Transformaciones

- 1.- Cada byte es considerado como un elemento del GF(2⁸) que genera el polinomio irreducible $m(x)=x^8+x^4+x^3+x+1$, y sustituido por su inversa multiplicativa. El valor cero queda inalterado.

$$g(x) = x^{-1} \bmod m(x), \quad x \in GF(2^8), \quad x \neq 0$$

$$g(0) = 0$$

- 2.-Después se aplica la transformación afín en GF(2), definida por
 $y=f(x)$.

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

siendo $x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7$, los bits del byte correspondiente, e $y_0, y_1, y_2, y_3, y_4, y_5, y_6, y_7$ los del resultado.



4.3.2 La función ByteSub

- ▶ La s-caja se obtiene con la composición $f \circ g$.
- ▶ La función inversa de ByteSub se obtiene de

$$(f \circ g)^{-1} = g^{-1} \circ f^{-1} = g \circ f^{-1}$$

- ▶ La s-caja utilizada, en formato hexadecimal es la que se muestra:

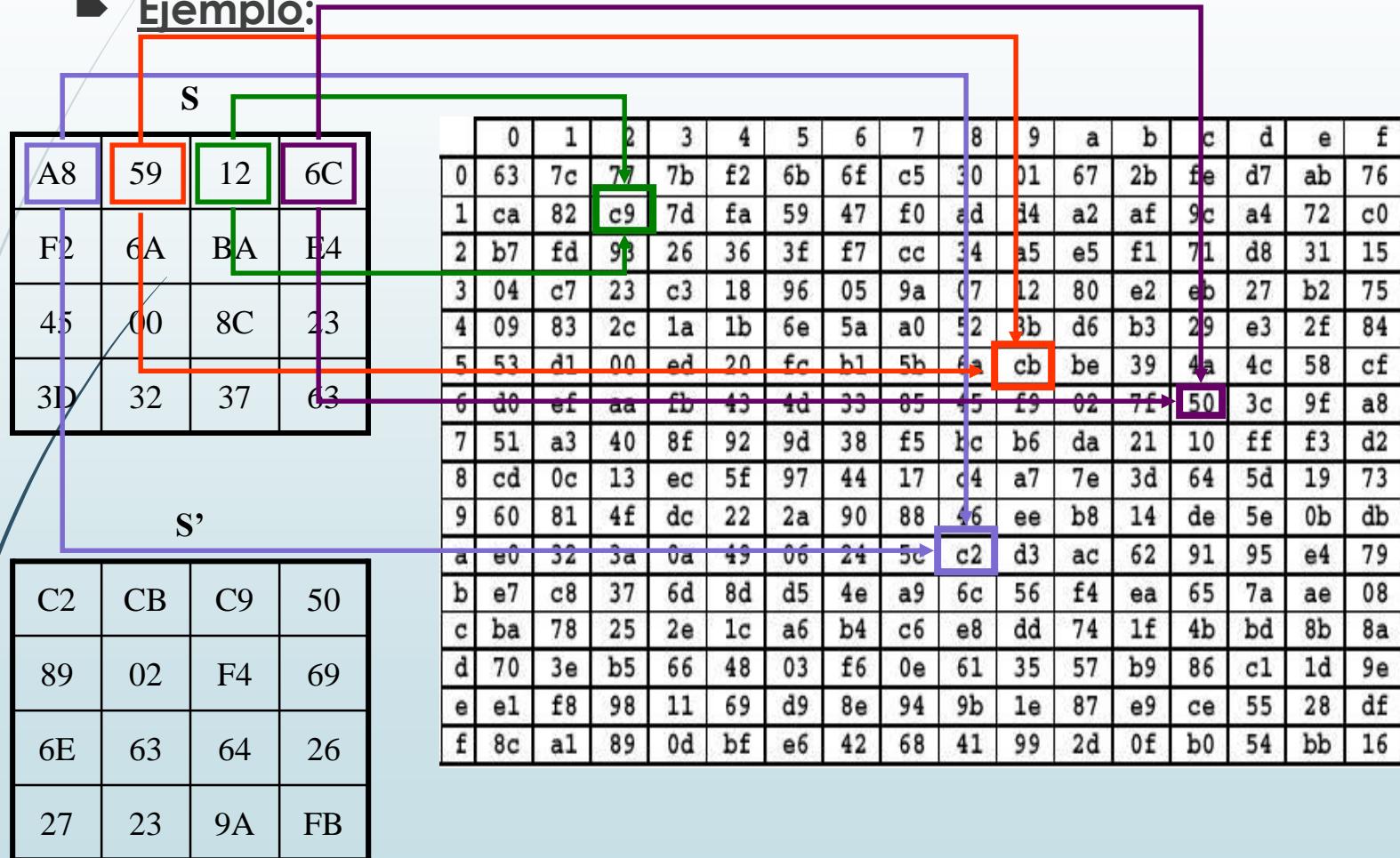
	y															
x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16



4.3.2 La función ByteSub

- Para buscar en esta tabla, se divide el byte en 2 bloques xy de 4 bits cada uno. Se busca x en la fila e y en la columna, y el valor que obtengamos es el resultado de realizar las dos transformaciones definidas anteriormente.

- Ejemplo:**



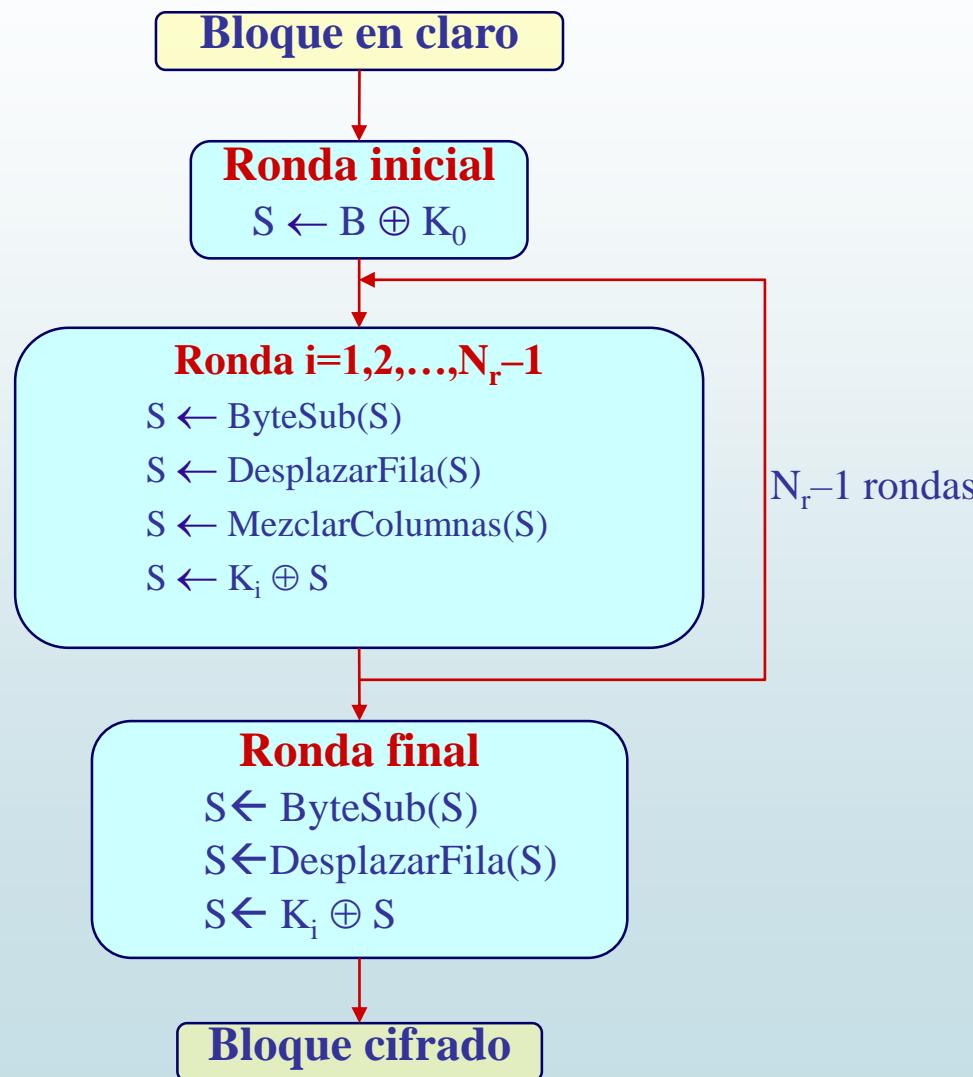
4.3.2 La función ByteSub

- Para el proceso de descifrado es necesario calcular la función inversa de ByteSub. Se recoge en la siguiente S-Box.

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	52	09	6a	d5	30	36	a5	38	bf	40	a3	9e	81	f3	d7	fb
	1	7c	e3	39	82	9b	2f	ff	87	34	8e	43	44	c4	de	e9	cb
	2	54	7b	94	32	a6	c2	23	3d	ee	4c	95	0b	42	fa	c3	4e
	3	08	2e	a1	66	28	d9	24	b2	76	5b	a2	49	6d	8b	d1	25
	4	72	f8	f6	64	86	68	98	16	d4	a4	5c	cc	5d	65	b6	92
	5	6c	70	48	50	fd	ed	b9	da	5e	15	46	57	a7	8d	9d	84
	6	90	d8	ab	00	8c	bc	d3	0a	f7	e4	58	05	b8	b3	45	06
	7	d0	2c	1e	8f	ca	3f	0f	02	c1	af	bd	03	01	13	8a	6b
	8	3a	91	11	41	4f	67	dc	ea	97	f2	cf	ce	f0	b4	e6	73
	9	96	ac	74	22	e7	ad	35	85	e2	f9	37	e8	1c	75	df	6e
	a	47	f1	1a	71	1d	29	c5	89	6f	b7	62	0e	aa	18	be	1b
	b	fc	56	3e	4b	c6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
	c	1f	dd	a8	33	88	07	c7	31	b1	12	10	59	27	80	ec	5f
	d	60	51	7f	a9	19	b5	4a	0d	2d	e5	7a	9f	93	c9	9c	ef
	e	a0	e0	3b	4d	ae	2a	f5	b0	c8	eb	bb	3c	83	53	99	61
	f	17	2b	04	7e	ba	77	d6	26	e1	69	14	63	55	21	0c	7d



AES: esquema de cifrado



4.3.3 DesplazarFila (ShiftRows)

- ▶ Esta transformación consiste en desplazar a la izquierda cíclicamente las filas de la matriz de estado.
- ▶ Cada la fila f_i se desplaza un número de posiciones c_i diferente.
- ▶ Mientras que c_0 siempre es igual a cero (esta fila siempre permanece inalterada), el resto de valores se refleja en la tabla

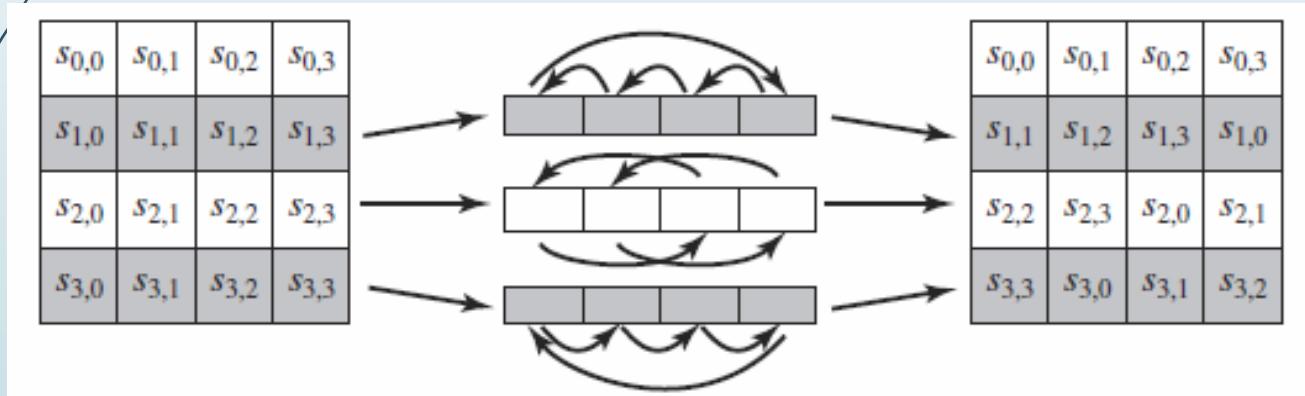
c_1	c_2	c_3
1	2	3

- ▶ La función inversa de DesplazarFila es un desplazamiento de las filas de la matriz de estado el mismo número de posiciones que en la tabla pero a la derecha



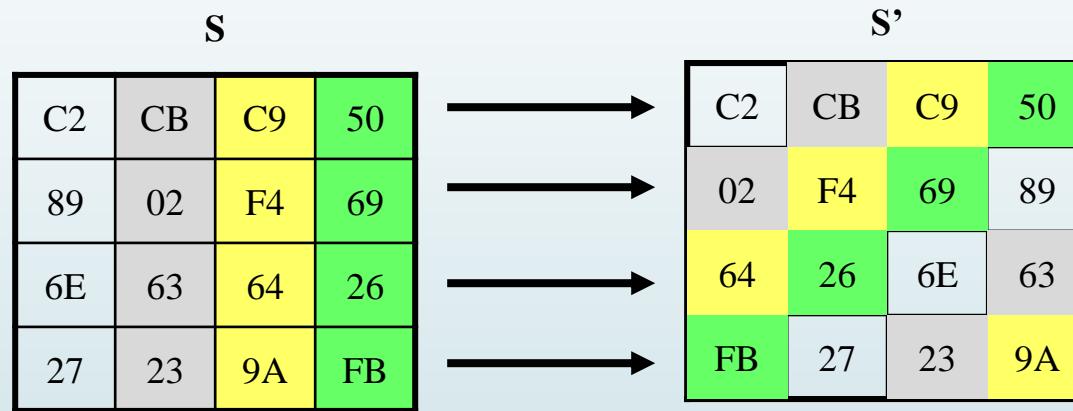
4.3.3 DesplazarFila (ShiftRows)

- Para el algoritmo estándar del AES, $N_b = 4$.
 - En este caso, para cada fila del estado se efectúa una rotación a izquierda tantos bytes como indique el numero de fila, empezando a contar de 0.

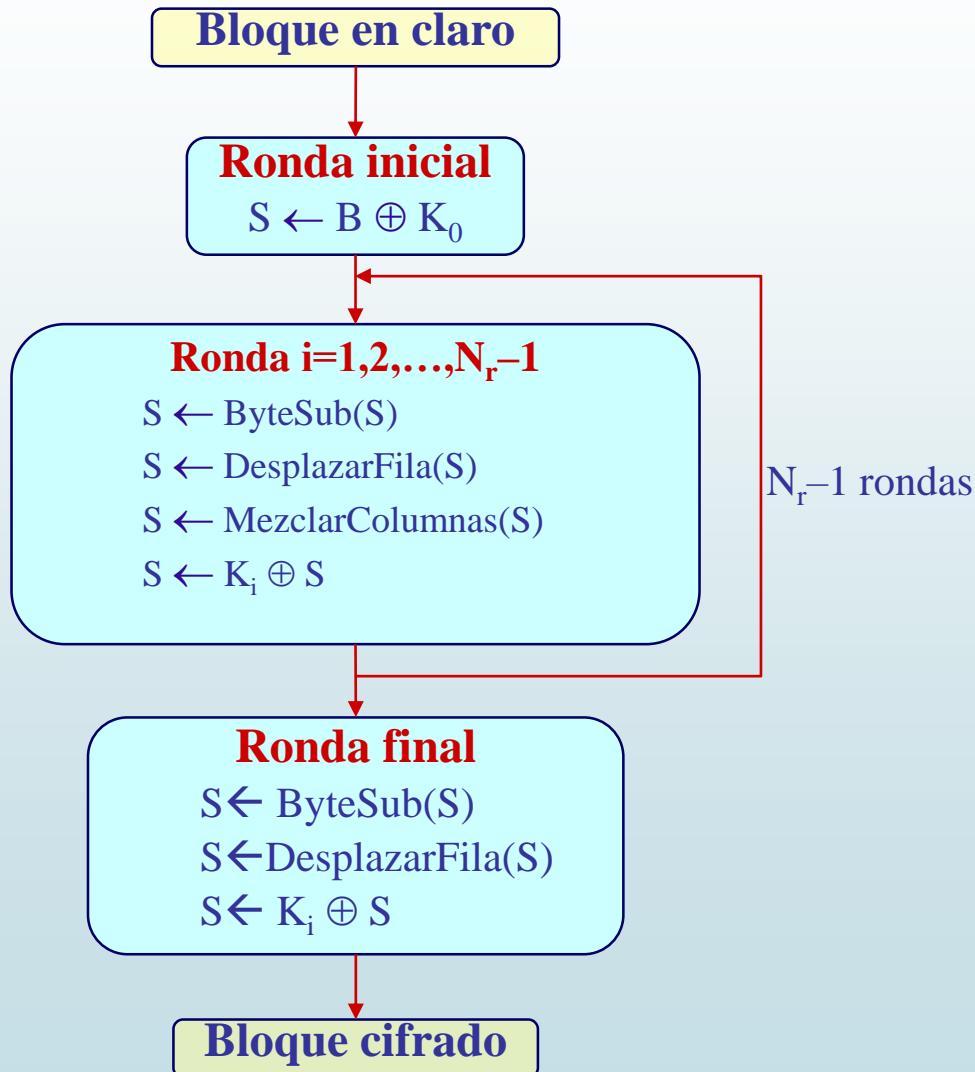


4.3.3 DesplazarFila (ShiftRows)

► Ejemplo:



AES: esquema de cifrado



4.3.4 MezclarColumnas (MixColumns)

- Recordemos que en AES, algunas palabras de 32 bits, se toman como un polinomio de grado menor o igual a 3 cuyos coeficientes son, a su vez, polinomios en $GF(2^8)$:

$$a(x) = a_3 x^3 + a_2 x^2 + a_1 x + a_0$$

- Para sumar o multiplicar dos polinomios de este tipo se actúa con la operatoria general definida para polinomios, sustituyendo las operaciones de los coeficientes por las operaciones definidas en $GF(2^8)$, \oplus y \otimes .

- Para esta función, cada columna de la matriz de estado se interpreta como un palabra de 4 bytes (32 bits) y cada palabra se considera como un polinomio de grado 3 con coeficientes en $GF(2^8)$.

- Dicho polinomio se multiplica módulo $x^4 + \{01_{16}\}$ por

$$a(x) = \{03_{16}\}x^3 + \{01_{16}\}x^2 + \{01_{16}\}x + \{02_{16}\}$$

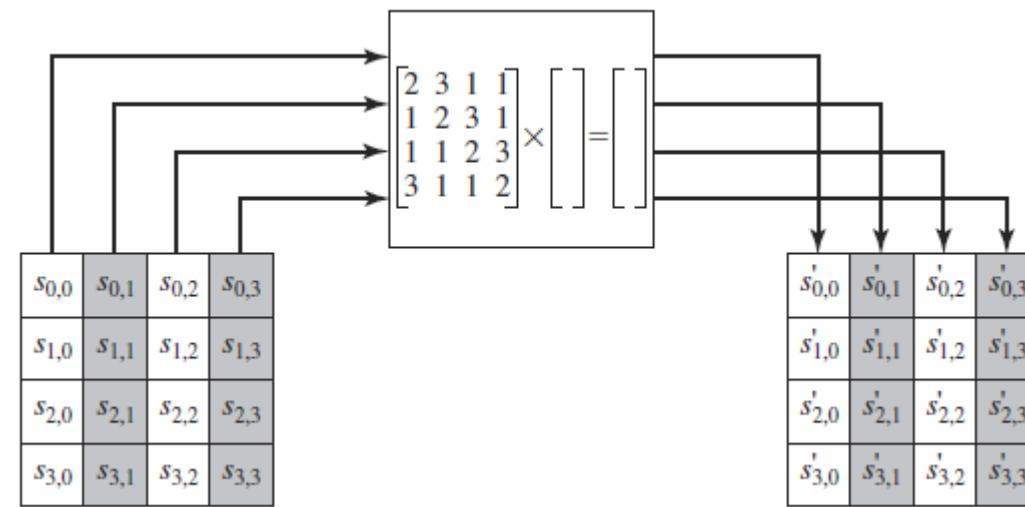
- Este producto queda expresado matricialmente de la siguiente manera

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} \text{ para } 0 \leq c < 4$$

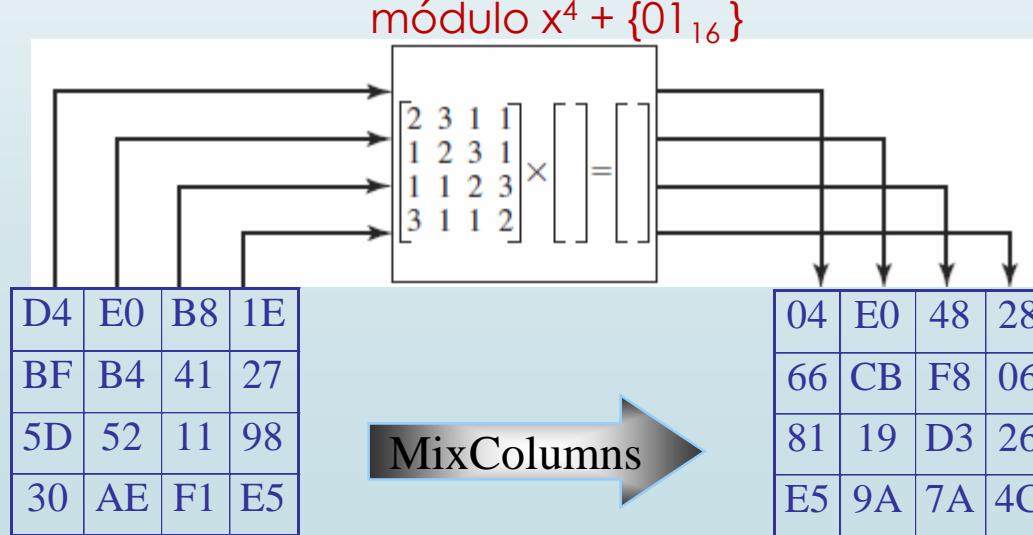
$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$



4.3.4 MezclarColumnas (MixColumns)



Ejemplo



4.3.4 MezclarColumnas (MixColumns)

- La inversa de MezclarColumnas se obtiene multiplicando módulo $x^4 + \{01_{16}\}$ cada columna de la matriz de estado por el polinomio inverso de a (x) módulo $x^4 + \{01_{16}\}$.

- Esto es, por

$$\{0B_{16}\}x^3 + \{0D_{16}\}x^2 + \{09_{16}\}x + \{0E_{16}\}$$

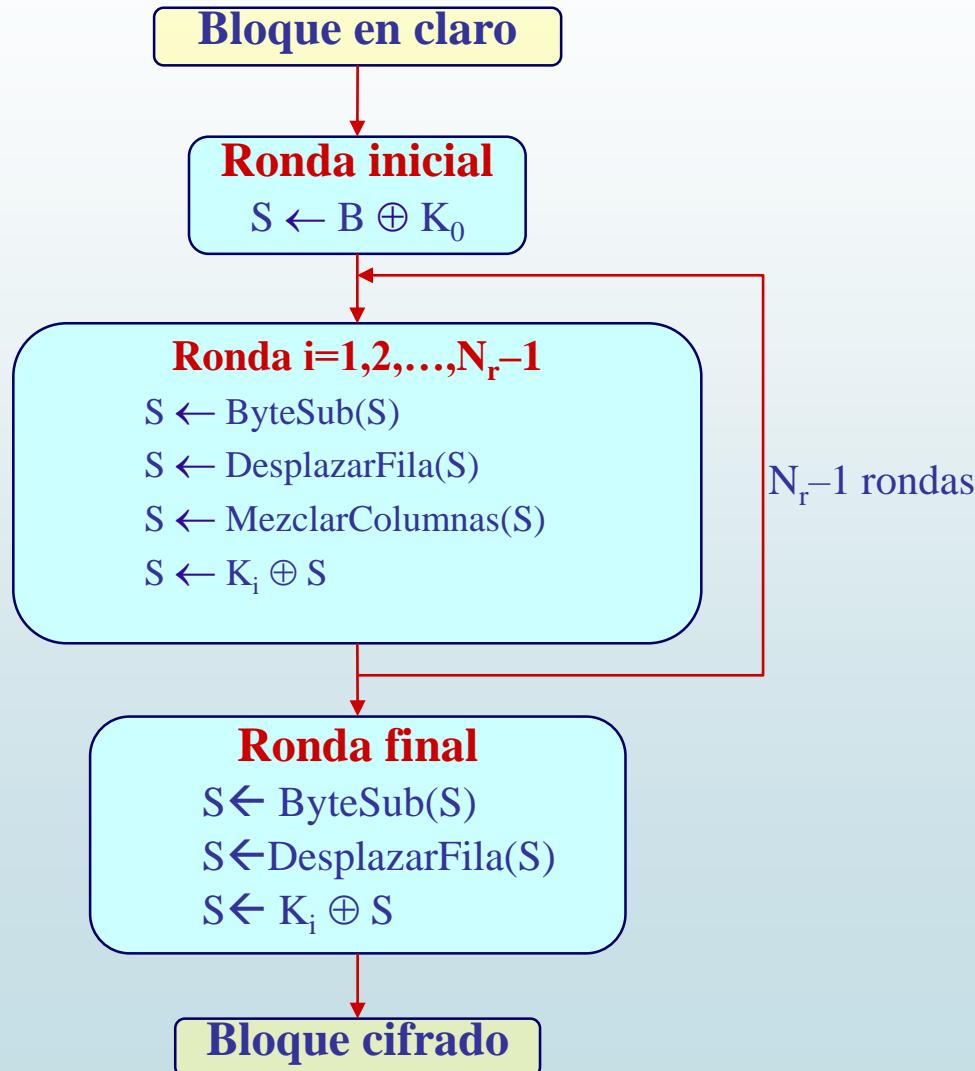
- Este producto queda expresado matricialmente de la siguiente manera

$$\begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} = \begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0d & 09 & 0E \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} \quad \text{para } 0 \leq c < 4$$

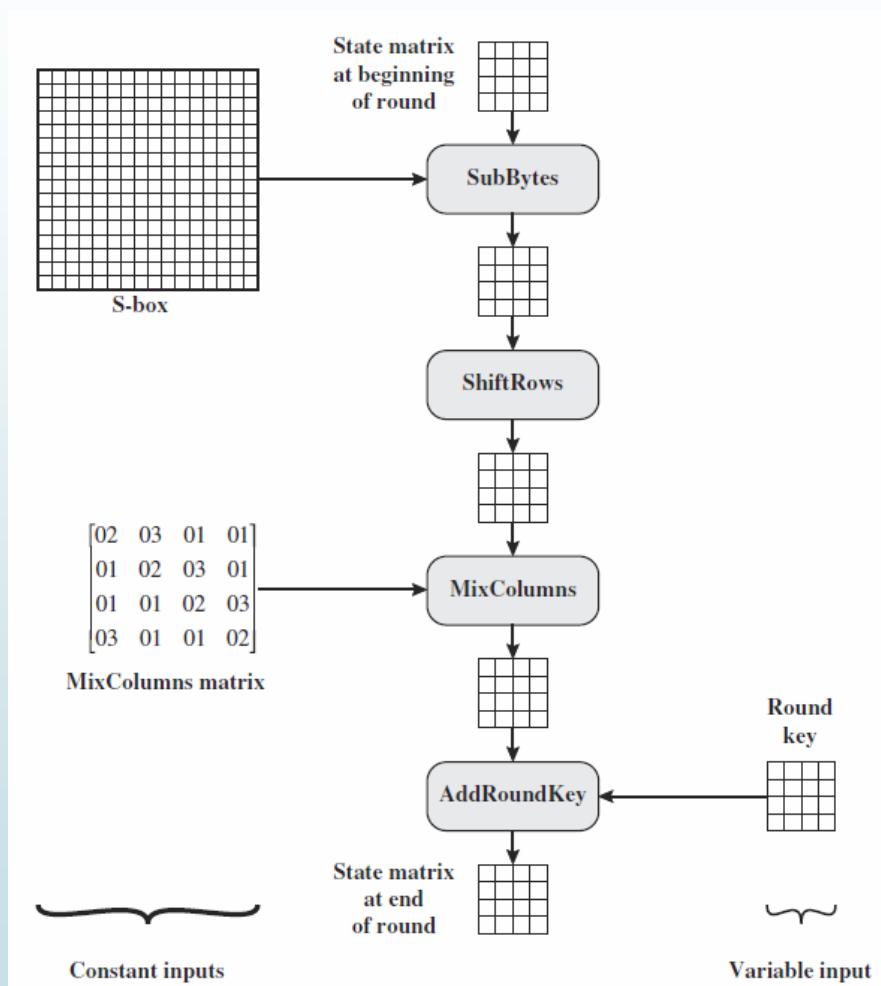
$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$



AES: esquema de cifrado



4.3.5 AddRoundKey



4.3.6 Cálculo de las subclaves

- ▶ Las diferentes subclaves K_i derivan de la clave principal K mediante el uso de dos funciones: una de **expansión** y otra de **selección**.
- ▶ La función de **expansión** permite obtener, a partir del valor de K , una secuencia de $16(n+1)$ bytes (donde n es el número de rondas que se aplican).
- ▶ La función de **selección** toma (consecutivamente), de la secuencia obtenida en la expansión, bloques del mismo tamaño que la matriz de estado y los va asignando a cada K_i .
- ▶ La función de expansión tiene dos versiones, según el valor de N_k .



4.3.6 Cálculo de las subclaves

N_k ≤ 6

AES128

AES192

1. Para i desde 0 hasta $N_k - 1$ hacer
2. $W(i) \leftarrow (K(4 \cdot i), K(4 \cdot i + 1), K(4 \cdot i + 2), K(4 \cdot i + 3))$
3. Para i desde N_k hasta $N_b \cdot (n + 1)$ hacer
4. $tmp \leftarrow W(i - 1)$
5. Si $i \bmod N_k = 0$
6. $tmp \leftarrow Sub(Rot(tmp)) \oplus R(i/N_k)$
7. $W(i) \leftarrow W(i - N_k) \oplus tmp$

- $K(i)$ es un vector de bytes de tamaño $4N_k$ conteniendo la clave.
- $W(i)$ es un vector de $4(n + 1)$ registros de 4 bytes
- n es el número de rondas.



4.3.6 Cálculo de las subclaves

N_k > 6

AES256

1. Para i desde 0 hasta $N_k - 1$ hacer
2. $W(i) \leftarrow (K(4 \cdot i), K(4 \cdot i + 1), K(4 \cdot i + 2), K(4 \cdot i + 3))$
3. Para i desde N_k hasta $N_b \cdot (n + 1)$ hacer
4. $tmp \leftarrow W(i - 1)$
5. Si $i \bmod N_k = 0$
6. $tmp \leftarrow Sub(Rot(tmp)) \oplus Rc(i/N_k)$
7. Si $i \bmod N_k = 4$
8. $tmp \leftarrow Sub(tmp)$
9. $W(i) \leftarrow W(i - N_k) \oplus tmp$

- $K(i)$ es un vector de bytes de tamaño $4N_k$ conteniendo la clave.
- $W(i)$ es un vector de $4(n + 1)$ registros de 4 bytes
- n es el número de rondas.



4.3.6 Cálculo de las subclaves

- ▶ En los algoritmos anteriores,
- ▶ **Sub** devuelve el resultado de aplicar la s-caja de AES a cada uno de los bytes del registro de cuatro que se le pasa como parámetro.
- ▶ **Rot** desplaza a la izquierda una posición los bytes del registro,
 - ▶ de la entrada (a; b; c; d) devuelve (b; c; d; a).
- ▶ **Rc(j)** es una constante definida como: $Rc(j) = (R(j); 0; 0; 0)$
- ▶ **R(i)** es el elemento de $GF(2^8)$ correspondiente al valor $x^{(i-1)}$.



4.3.6 Cálculo de las subclaves

i (dec)	temp	Después de Rot()	Después de SubByte()	Rc (i/Nk)	Después de XOR con Rcon	W(i-Nk)	W(i)= temp XOR W(i-Nk)
4	09CF4F3C	CF4F3C09	8A84EB01	01000000	8B84EB01	2B7E1516	A0FAFE17
5	A0FAFE17					28AED2A6	88542CB1
6	88542CB1					ABF71588	23A33939
7	23A33939					09CF4F3C	2A6C7605
8	2A6C7605	6C76052A	50386BE5	02000000	52386BE5	A0FAFE17	F2C295F2
9	F2C295F2					88542CB1	7A96B943
10	7A96B943					23A33939	5935807A
11	5935807A					2A6C7605	7359F67F
12	7359F67F	59F67F73	CB42D28F	04000000	CF42D28F	F2C295F2	3D80477D
13	3D80477D					7A96B943	4716FE3E
14	4716FE3E					5935807A	1E237E44
15	1E237E44					7359F67F	6D7A883B
16	6D7A883B	7A883B6D	DAC4E23C	08000000	D2C4E23C	3D80477D	EF44A541
17	EF44A541					4716FE3E	A8525B7F
18	A8525B7F					1E237E44	B671253B
19	B671253B					6D7A883B	DB0BAD00
20	DB0BAD00	0BAD00DB	2B9563B9	10000000	3B9563B9	EF44A541	D4D1C6F8
21	D4D1C6F8					A8525B7F	7C839D87
22	7C839D87					B671253B	CAF2B8BC
23	CAF2B8BC					DB0BAD00	11F915BC



4.3.6 Cálculo de las subclaves

i (dec)	temp	Después de Rot ()	Después de SubByte ()	Rc (i/Nk)	Después de XOR con Rcon	W (i-Nk)	W (i) = temp XOR W (i-Nk)
24	11F915BC	F915BC11	99596582	20000000	B9596582	D4D1C6F8	6D88A37A
25	6D88A37A					7C839D87	110B3EFD
26	110B3EFD					CAF2B8BC	DBF98641
27	DBF98641					11F915BC	CA0093FD
28	CA0093FD	0093FDCA	63DC5474	40000000	23DC5474	6D88A37A	4E54F70E
29	4E54F70E					110B3EFD	5F5FC9F3
30	5F5FC9F3					DBF98641	84A64FB2
31	84A64FB2					CA0093FD	4EA6DC4F
32	4EA6DC4F	A6DC4F4E	2486842F	80000000	A486842F	4E54F70E	EAD27321
33	EAD27321					5F5FC9F3	B58DBAD2
34	B58DBAD2					84A64FB2	312BF560
35	312BF560					4EA6DC4F	7F8D292F
36	7F8D292F	8D292F7F	5DA515D2	1B000000	46A515D2	EAD27321	AC7766F3
37	AC7766F3					B58DBAD2	19FADC21
38	19FADC21					312BF560	28D12941
39	28D12941					7F8D292F	575C006E
40	575C006E	5C006E57	4A639F5B	36000000	7C639F5B	AC7766F3	D014F9A8
41	D014F9A8					19FADC21	C9EE2589
42	C9EE2589					28D12941	E13F0CC8
43	E13F0CC8					575C006E	B6630CA6



AES: referencias

- ▶ Página principal AES (historical purposes):
<http://csrc.nist.gov/archive/aes/index.html>
Federal Information Processing Standards (FIPS)
J. Daemen and V. Rijmen, *AES Proposal: Rijndael*, AES Algorithm Submission,
- ▶ Primera vulnerabilidad (2011) en el algoritmo de cifrado AES, que reduce la longitud efectiva de clave en 2 bits.
 - ▶ Esto significa que las longitudes usuales de clave de 128, 192 y 256 bits son reducidas a 126, 190 y 254 bits.

<https://www.ccn-cert.cni.es/seguridad-al-dia/noticias-seguridad/954-primer-vulnerabilidad-en-el-algoritmo-de-cifrado-aes.html>



4.3.7 Ejemplo de aplicación del algoritmo

Ronda	Entrada =	Clave =	Inicio de la ronda	Después de SubBytes	Después de ShiftRows	Después de MixColumns	Subclave
Entrada	32 43 F6 A8 88 5A 30 8D 31 31 98 A2 E0 37 07 34	2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C					
1							
	32 88 31 E0 43 5A 31 37 F6 30 98 07 A8 8D A2 34						
	19 A0 9A E9 3D F4 C6 F8 E3 E2 8D 48 BE 2B 2A 08	D4 E0 B8 1E 27 BF B4 41 11 98 5D 52 AE F1 E5 30	D4 E0 B8 1E BF B4 41 27 5D 52 11 98 30 AE F1 E5	04 E0 48 28 66 CB F8 06 81 19 D3 26 E5 9A 7A 4C	A0 88 23 2A FA 54 A3 6C FE 2C 39 76 17 B1 39 05	\oplus	
2							
	A4 68 6B 02 9C 9F 5B 6A 7F 35 EA 50 F2 2B 43 49	49 45 7F 77 DE DB 39 02 D2 96 87 53 89 F1 1A 3B	49 45 7F 77 DB 39 02 DE 87 53 D2 96 3B 89 F1 1A	58 1B DB 1B 4D 4B E7 6B CA 5A CA B0 F1 AC A8 E5	F2 7A 59 73 C2 96 35 59 95 B9 80 F6 F2 43 7A 7F	\oplus	=



4.3.7 Ejemplo de aplicación del algoritmo

Entrada = 32 43 F6 A8 88 5A 30 8D 31 31 98 A2 E0 37 07 34

Clave = 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C

Ronda	Inicio de la ronda	Después de SubBytes	Después de ShiftRows	Después de MixColumns	Subclave																																																																																	
3	<table border="1"> <tr><td>AA</td><td>61</td><td>82</td><td>68</td></tr> <tr><td>8F</td><td>DD</td><td>D2</td><td>32</td></tr> <tr><td>5F</td><td>E3</td><td>4A</td><td>46</td></tr> <tr><td>03</td><td>EF</td><td>D2</td><td>9A</td></tr> </table>	AA	61	82	68	8F	DD	D2	32	5F	E3	4A	46	03	EF	D2	9A	<table border="1"> <tr><td>AC</td><td>EF</td><td>13</td><td>45</td></tr> <tr><td>73</td><td>C1</td><td>B5</td><td>23</td></tr> <tr><td>CF</td><td>11</td><td>D6</td><td>5A</td></tr> <tr><td>7B</td><td>DF</td><td>B5</td><td>B8</td></tr> </table>	AC	EF	13	45	73	C1	B5	23	CF	11	D6	5A	7B	DF	B5	B8	<table border="1"> <tr><td>AC</td><td>EF</td><td>13</td><td>45</td></tr> <tr><td>C1</td><td>B5</td><td>23</td><td>73</td></tr> <tr><td>D6</td><td>5A</td><td>CF</td><td>11</td></tr> <tr><td>B8</td><td>7B</td><td>DF</td><td>B5</td></tr> </table>	AC	EF	13	45	C1	B5	23	73	D6	5A	CF	11	B8	7B	DF	B5	<table border="1"> <tr><td>75</td><td>20</td><td>53</td><td>BB</td></tr> <tr><td>EC</td><td>0B</td><td>C0</td><td>25</td></tr> <tr><td>09</td><td>63</td><td>CF</td><td>D0</td></tr> <tr><td>93</td><td>33</td><td>7C</td><td>DC</td></tr> </table>	75	20	53	BB	EC	0B	C0	25	09	63	CF	D0	93	33	7C	DC	<table border="1"> <tr><td>3D</td><td>47</td><td>1E</td><td>6D</td></tr> <tr><td>80</td><td>16</td><td>23</td><td>7A</td></tr> <tr><td>47</td><td>FE</td><td>7E</td><td>88</td></tr> <tr><td>7D</td><td>3E</td><td>44</td><td>3B</td></tr> </table>	3D	47	1E	6D	80	16	23	7A	47	FE	7E	88	7D	3E	44	3B	=
AA	61	82	68																																																																																			
8F	DD	D2	32																																																																																			
5F	E3	4A	46																																																																																			
03	EF	D2	9A																																																																																			
AC	EF	13	45																																																																																			
73	C1	B5	23																																																																																			
CF	11	D6	5A																																																																																			
7B	DF	B5	B8																																																																																			
AC	EF	13	45																																																																																			
C1	B5	23	73																																																																																			
D6	5A	CF	11																																																																																			
B8	7B	DF	B5																																																																																			
75	20	53	BB																																																																																			
EC	0B	C0	25																																																																																			
09	63	CF	D0																																																																																			
93	33	7C	DC																																																																																			
3D	47	1E	6D																																																																																			
80	16	23	7A																																																																																			
47	FE	7E	88																																																																																			
7D	3E	44	3B																																																																																			
4	<table border="1"> <tr><td>48</td><td>67</td><td>4D</td><td>D6</td></tr> <tr><td>6C</td><td>1D</td><td>E3</td><td>5F</td></tr> <tr><td>4E</td><td>9D</td><td>B1</td><td>58</td></tr> <tr><td>EE</td><td>0D</td><td>38</td><td>E7</td></tr> </table>	48	67	4D	D6	6C	1D	E3	5F	4E	9D	B1	58	EE	0D	38	E7	<table border="1"> <tr><td>52</td><td>85</td><td>E3</td><td>F6</td></tr> <tr><td>50</td><td>A4</td><td>11</td><td>CF</td></tr> <tr><td>2F</td><td>5E</td><td>C8</td><td>6A</td></tr> <tr><td>28</td><td>D7</td><td>07</td><td>94</td></tr> </table>	52	85	E3	F6	50	A4	11	CF	2F	5E	C8	6A	28	D7	07	94	<table border="1"> <tr><td>52</td><td>85</td><td>E3</td><td>F6</td></tr> <tr><td>A4</td><td>11</td><td>CF</td><td>50</td></tr> <tr><td>C8</td><td>6A</td><td>2F</td><td>5E</td></tr> <tr><td>94</td><td>28</td><td>D7</td><td>07</td></tr> </table>	52	85	E3	F6	A4	11	CF	50	C8	6A	2F	5E	94	28	D7	07	<table border="1"> <tr><td>0F</td><td>60</td><td>6F</td><td>5E</td></tr> <tr><td>D6</td><td>31</td><td>C0</td><td>B3</td></tr> <tr><td>DA</td><td>38</td><td>10</td><td>13</td></tr> <tr><td>A9</td><td>BF</td><td>6B</td><td>01</td></tr> </table>	0F	60	6F	5E	D6	31	C0	B3	DA	38	10	13	A9	BF	6B	01	<table border="1"> <tr><td>EF</td><td>A8</td><td>B6</td><td>DB</td></tr> <tr><td>44</td><td>52</td><td>71</td><td>0B</td></tr> <tr><td>A5</td><td>5B</td><td>25</td><td>AD</td></tr> <tr><td>41</td><td>7F</td><td>3B</td><td>00</td></tr> </table>	EF	A8	B6	DB	44	52	71	0B	A5	5B	25	AD	41	7F	3B	00	=
48	67	4D	D6																																																																																			
6C	1D	E3	5F																																																																																			
4E	9D	B1	58																																																																																			
EE	0D	38	E7																																																																																			
52	85	E3	F6																																																																																			
50	A4	11	CF																																																																																			
2F	5E	C8	6A																																																																																			
28	D7	07	94																																																																																			
52	85	E3	F6																																																																																			
A4	11	CF	50																																																																																			
C8	6A	2F	5E																																																																																			
94	28	D7	07																																																																																			
0F	60	6F	5E																																																																																			
D6	31	C0	B3																																																																																			
DA	38	10	13																																																																																			
A9	BF	6B	01																																																																																			
EF	A8	B6	DB																																																																																			
44	52	71	0B																																																																																			
A5	5B	25	AD																																																																																			
41	7F	3B	00																																																																																			
5	<table border="1"> <tr><td>E0</td><td>C8</td><td>D9</td><td>85</td></tr> <tr><td>92</td><td>63</td><td>B1</td><td>B8</td></tr> <tr><td>7F</td><td>63</td><td>35</td><td>BE</td></tr> <tr><td>E8</td><td>C0</td><td>50</td><td>01</td></tr> </table>	E0	C8	D9	85	92	63	B1	B8	7F	63	35	BE	E8	C0	50	01	<table border="1"> <tr><td>E1</td><td>E8</td><td>35</td><td>97</td></tr> <tr><td>4F</td><td>FB</td><td>C8</td><td>6C</td></tr> <tr><td>D2</td><td>FB</td><td>96</td><td>AE</td></tr> <tr><td>9B</td><td>BA</td><td>53</td><td>7C</td></tr> </table>	E1	E8	35	97	4F	FB	C8	6C	D2	FB	96	AE	9B	BA	53	7C	<table border="1"> <tr><td>E1</td><td>E8</td><td>35</td><td>97</td></tr> <tr><td>FB</td><td>C8</td><td>6C</td><td>4F</td></tr> <tr><td>96</td><td>AE</td><td>D2</td><td>FB</td></tr> <tr><td>7C</td><td>9B</td><td>BA</td><td>53</td></tr> </table>	E1	E8	35	97	FB	C8	6C	4F	96	AE	D2	FB	7C	9B	BA	53	<table border="1"> <tr><td>25</td><td>BD</td><td>B6</td><td>4C</td></tr> <tr><td>D1</td><td>11</td><td>3A</td><td>4C</td></tr> <tr><td>A9</td><td>D1</td><td>33</td><td>C0</td></tr> <tr><td>AD</td><td>68</td><td>8E</td><td>B0</td></tr> </table>	25	BD	B6	4C	D1	11	3A	4C	A9	D1	33	C0	AD	68	8E	B0	<table border="1"> <tr><td>D4</td><td>7C</td><td>CA</td><td>11</td></tr> <tr><td>D1</td><td>83</td><td>F2</td><td>F9</td></tr> <tr><td>C6</td><td>9D</td><td>B8</td><td>15</td></tr> <tr><td>F8</td><td>87</td><td>BC</td><td>BC</td></tr> </table>	D4	7C	CA	11	D1	83	F2	F9	C6	9D	B8	15	F8	87	BC	BC	=
E0	C8	D9	85																																																																																			
92	63	B1	B8																																																																																			
7F	63	35	BE																																																																																			
E8	C0	50	01																																																																																			
E1	E8	35	97																																																																																			
4F	FB	C8	6C																																																																																			
D2	FB	96	AE																																																																																			
9B	BA	53	7C																																																																																			
E1	E8	35	97																																																																																			
FB	C8	6C	4F																																																																																			
96	AE	D2	FB																																																																																			
7C	9B	BA	53																																																																																			
25	BD	B6	4C																																																																																			
D1	11	3A	4C																																																																																			
A9	D1	33	C0																																																																																			
AD	68	8E	B0																																																																																			
D4	7C	CA	11																																																																																			
D1	83	F2	F9																																																																																			
C6	9D	B8	15																																																																																			
F8	87	BC	BC																																																																																			



4.3.7 Ejemplo de aplicación del algoritmo

Entrada = 32 43 F6 A8 88 5A 30 8D 31 31 98 A2 E0 37 07 34

Clave = 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C

Ronda	Inicio de la ronda	Después de SubBytes	Después de ShiftRows	Después de MixColumns	Subclave																																																																																	
6	<table border="1"> <tr><td>F1</td><td>C1</td><td>7C</td><td>5D</td></tr> <tr><td>00</td><td>92</td><td>C8</td><td>B5</td></tr> <tr><td>6F</td><td>4C</td><td>8B</td><td>D5</td></tr> <tr><td>55</td><td>EF</td><td>32</td><td>0C</td></tr> </table>	F1	C1	7C	5D	00	92	C8	B5	6F	4C	8B	D5	55	EF	32	0C	<table border="1"> <tr><td>A1</td><td>78</td><td>10</td><td>4C</td></tr> <tr><td>63</td><td>4F</td><td>E8</td><td>D5</td></tr> <tr><td>A8</td><td>29</td><td>3D</td><td>03</td></tr> <tr><td>FC</td><td>DF</td><td>23</td><td>FE</td></tr> </table>	A1	78	10	4C	63	4F	E8	D5	A8	29	3D	03	FC	DF	23	FE	<table border="1"> <tr><td>A1</td><td>78</td><td>10</td><td>4C</td></tr> <tr><td>4F</td><td>E8</td><td>D5</td><td>63</td></tr> <tr><td>3D</td><td>03</td><td>A8</td><td>29</td></tr> <tr><td>FE</td><td>FC</td><td>DF</td><td>23</td></tr> </table>	A1	78	10	4C	4F	E8	D5	63	3D	03	A8	29	FE	FC	DF	23	<table border="1"> <tr><td>4B</td><td>2C</td><td>33</td><td>37</td></tr> <tr><td>86</td><td>4A</td><td>9D</td><td>D2</td></tr> <tr><td>8D</td><td>89</td><td>F4</td><td>18</td></tr> <tr><td>6D</td><td>80</td><td>E8</td><td>D8</td></tr> </table>	4B	2C	33	37	86	4A	9D	D2	8D	89	F4	18	6D	80	E8	D8	<table border="1"> <tr><td>6D</td><td>11</td><td>DB</td><td>CA</td></tr> <tr><td>88</td><td>0B</td><td>F9</td><td>00</td></tr> <tr><td>A3</td><td>3E</td><td>86</td><td>93</td></tr> <tr><td>7A</td><td>FD</td><td>41</td><td>FD</td></tr> </table>	6D	11	DB	CA	88	0B	F9	00	A3	3E	86	93	7A	FD	41	FD	=
F1	C1	7C	5D																																																																																			
00	92	C8	B5																																																																																			
6F	4C	8B	D5																																																																																			
55	EF	32	0C																																																																																			
A1	78	10	4C																																																																																			
63	4F	E8	D5																																																																																			
A8	29	3D	03																																																																																			
FC	DF	23	FE																																																																																			
A1	78	10	4C																																																																																			
4F	E8	D5	63																																																																																			
3D	03	A8	29																																																																																			
FE	FC	DF	23																																																																																			
4B	2C	33	37																																																																																			
86	4A	9D	D2																																																																																			
8D	89	F4	18																																																																																			
6D	80	E8	D8																																																																																			
6D	11	DB	CA																																																																																			
88	0B	F9	00																																																																																			
A3	3E	86	93																																																																																			
7A	FD	41	FD																																																																																			
7	<table border="1"> <tr><td>26</td><td>3D</td><td>E8</td><td>FD</td></tr> <tr><td>0E</td><td>41</td><td>64</td><td>D2</td></tr> <tr><td>2E</td><td>B7</td><td>72</td><td>8B</td></tr> <tr><td>17</td><td>7D</td><td>A9</td><td>25</td></tr> </table>	26	3D	E8	FD	0E	41	64	D2	2E	B7	72	8B	17	7D	A9	25	<table border="1"> <tr><td>F7</td><td>27</td><td>9B</td><td>54</td></tr> <tr><td>AB</td><td>83</td><td>43</td><td>B5</td></tr> <tr><td>31</td><td>A9</td><td>40</td><td>3D</td></tr> <tr><td>F0</td><td>FF</td><td>D3</td><td>3F</td></tr> </table>	F7	27	9B	54	AB	83	43	B5	31	A9	40	3D	F0	FF	D3	3F	<table border="1"> <tr><td>F7</td><td>27</td><td>9B</td><td>54</td></tr> <tr><td>83</td><td>43</td><td>B5</td><td>AB</td></tr> <tr><td>40</td><td>3D</td><td>31</td><td>A9</td></tr> <tr><td>3F</td><td>F0</td><td>FF</td><td>D3</td></tr> </table>	F7	27	9B	54	83	43	B5	AB	40	3D	31	A9	3F	F0	FF	D3	<table border="1"> <tr><td>14</td><td>46</td><td>27</td><td>34</td></tr> <tr><td>15</td><td>16</td><td>46</td><td>2A</td></tr> <tr><td>B5</td><td>15</td><td>56</td><td>D8</td></tr> <tr><td>BF</td><td>EC</td><td>D7</td><td>43</td></tr> </table>	14	46	27	34	15	16	46	2A	B5	15	56	D8	BF	EC	D7	43	<table border="1"> <tr><td>4E</td><td>5F</td><td>84</td><td>4E</td></tr> <tr><td>54</td><td>5F</td><td>A6</td><td>A6</td></tr> <tr><td>F7</td><td>C9</td><td>4F</td><td>DC</td></tr> <tr><td>0E</td><td>F3</td><td>B2</td><td>4F</td></tr> </table>	4E	5F	84	4E	54	5F	A6	A6	F7	C9	4F	DC	0E	F3	B2	4F	=
26	3D	E8	FD																																																																																			
0E	41	64	D2																																																																																			
2E	B7	72	8B																																																																																			
17	7D	A9	25																																																																																			
F7	27	9B	54																																																																																			
AB	83	43	B5																																																																																			
31	A9	40	3D																																																																																			
F0	FF	D3	3F																																																																																			
F7	27	9B	54																																																																																			
83	43	B5	AB																																																																																			
40	3D	31	A9																																																																																			
3F	F0	FF	D3																																																																																			
14	46	27	34																																																																																			
15	16	46	2A																																																																																			
B5	15	56	D8																																																																																			
BF	EC	D7	43																																																																																			
4E	5F	84	4E																																																																																			
54	5F	A6	A6																																																																																			
F7	C9	4F	DC																																																																																			
0E	F3	B2	4F																																																																																			
8	<table border="1"> <tr><td>5A</td><td>19</td><td>A3</td><td>7A</td></tr> <tr><td>41</td><td>49</td><td>E0</td><td>8C</td></tr> <tr><td>42</td><td>DC</td><td>19</td><td>04</td></tr> <tr><td>B1</td><td>1F</td><td>65</td><td>0C</td></tr> </table>	5A	19	A3	7A	41	49	E0	8C	42	DC	19	04	B1	1F	65	0C	<table border="1"> <tr><td>BE</td><td>D4</td><td>0A</td><td>DA</td></tr> <tr><td>83</td><td>3B</td><td>E1</td><td>64</td></tr> <tr><td>2C</td><td>86</td><td>D4</td><td>F2</td></tr> <tr><td>C8</td><td>C0</td><td>4D</td><td>FE</td></tr> </table>	BE	D4	0A	DA	83	3B	E1	64	2C	86	D4	F2	C8	C0	4D	FE	<table border="1"> <tr><td>BE</td><td>D4</td><td>0A</td><td>DA</td></tr> <tr><td>3B</td><td>E1</td><td>64</td><td>83</td></tr> <tr><td>D4</td><td>F2</td><td>2C</td><td>86</td></tr> <tr><td>FE</td><td>C8</td><td>C0</td><td>4D</td></tr> </table>	BE	D4	0A	DA	3B	E1	64	83	D4	F2	2C	86	FE	C8	C0	4D	<table border="1"> <tr><td>00</td><td>B1</td><td>54</td><td>FA</td></tr> <tr><td>51</td><td>C8</td><td>76</td><td>1B</td></tr> <tr><td>2F</td><td>89</td><td>6D</td><td>99</td></tr> <tr><td>D1</td><td>FF</td><td>CD</td><td>EA</td></tr> </table>	00	B1	54	FA	51	C8	76	1B	2F	89	6D	99	D1	FF	CD	EA	<table border="1"> <tr><td>EA</td><td>B5</td><td>31</td><td>7F</td></tr> <tr><td>D2</td><td>8D</td><td>2B</td><td>8D</td></tr> <tr><td>73</td><td>BA</td><td>F5</td><td>29</td></tr> <tr><td>21</td><td>D2</td><td>60</td><td>2F</td></tr> </table>	EA	B5	31	7F	D2	8D	2B	8D	73	BA	F5	29	21	D2	60	2F	=
5A	19	A3	7A																																																																																			
41	49	E0	8C																																																																																			
42	DC	19	04																																																																																			
B1	1F	65	0C																																																																																			
BE	D4	0A	DA																																																																																			
83	3B	E1	64																																																																																			
2C	86	D4	F2																																																																																			
C8	C0	4D	FE																																																																																			
BE	D4	0A	DA																																																																																			
3B	E1	64	83																																																																																			
D4	F2	2C	86																																																																																			
FE	C8	C0	4D																																																																																			
00	B1	54	FA																																																																																			
51	C8	76	1B																																																																																			
2F	89	6D	99																																																																																			
D1	FF	CD	EA																																																																																			
EA	B5	31	7F																																																																																			
D2	8D	2B	8D																																																																																			
73	BA	F5	29																																																																																			
21	D2	60	2F																																																																																			



4.3.7 Ejemplo de aplicación del algoritmo

Entrada = 32 43 F6 A8 88 5A 30 8D 31 31 98 A2 E0 37 07 34

Clave = 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C

Ronda	Inicio de la ronda	Después de SubBytes	Después de ShiftRows	Después de MixColumns	Subclave																																																																																
9	<table border="1"> <tr><td>EA</td><td>04</td><td>65</td><td>85</td></tr> <tr><td>83</td><td>45</td><td>5D</td><td>96</td></tr> <tr><td>5C</td><td>33</td><td>98</td><td>B0</td></tr> <tr><td>F0</td><td>2D</td><td>AD</td><td>C5</td></tr> </table>	EA	04	65	85	83	45	5D	96	5C	33	98	B0	F0	2D	AD	C5	<table border="1"> <tr><td>87</td><td>F2</td><td>4D</td><td>97</td></tr> <tr><td>EC</td><td>6E</td><td>4C</td><td>90</td></tr> <tr><td>4A</td><td>C3</td><td>46</td><td>E7</td></tr> <tr><td>8C</td><td>D8</td><td>95</td><td>A6</td></tr> </table>	87	F2	4D	97	EC	6E	4C	90	4A	C3	46	E7	8C	D8	95	A6	<table border="1"> <tr><td>87</td><td>F2</td><td>4D</td><td>97</td></tr> <tr><td>6E</td><td>4C</td><td>90</td><td>EC</td></tr> <tr><td>46</td><td>E7</td><td>4A</td><td>C3</td></tr> <tr><td>A6</td><td>8C</td><td>D8</td><td>95</td></tr> </table>	87	F2	4D	97	6E	4C	90	EC	46	E7	4A	C3	A6	8C	D8	95	<table border="1"> <tr><td>47</td><td>40</td><td>A3</td><td>4C</td></tr> <tr><td>37</td><td>D4</td><td>70</td><td>9F</td></tr> <tr><td>94</td><td>E4</td><td>3A</td><td>42</td></tr> <tr><td>ED</td><td>A5</td><td>A6</td><td>BC</td></tr> </table>	47	40	A3	4C	37	D4	70	9F	94	E4	3A	42	ED	A5	A6	BC	<table border="1"> <tr><td>AC</td><td>19</td><td>28</td><td>57</td></tr> <tr><td>77</td><td>FA</td><td>D1</td><td>5C</td></tr> <tr><td>66</td><td>DC</td><td>29</td><td>00</td></tr> <tr><td>F3</td><td>21</td><td>41</td><td>6E</td></tr> </table>	AC	19	28	57	77	FA	D1	5C	66	DC	29	00	F3	21	41	6E
EA	04	65	85																																																																																		
83	45	5D	96																																																																																		
5C	33	98	B0																																																																																		
F0	2D	AD	C5																																																																																		
87	F2	4D	97																																																																																		
EC	6E	4C	90																																																																																		
4A	C3	46	E7																																																																																		
8C	D8	95	A6																																																																																		
87	F2	4D	97																																																																																		
6E	4C	90	EC																																																																																		
46	E7	4A	C3																																																																																		
A6	8C	D8	95																																																																																		
47	40	A3	4C																																																																																		
37	D4	70	9F																																																																																		
94	E4	3A	42																																																																																		
ED	A5	A6	BC																																																																																		
AC	19	28	57																																																																																		
77	FA	D1	5C																																																																																		
66	DC	29	00																																																																																		
F3	21	41	6E																																																																																		
10	<table border="1"> <tr><td>EB</td><td>59</td><td>8B</td><td>1B</td></tr> <tr><td>40</td><td>2E</td><td>A1</td><td>C3</td></tr> <tr><td>F2</td><td>38</td><td>13</td><td>42</td></tr> <tr><td>1E</td><td>84</td><td>E7</td><td>D2</td></tr> </table>	EB	59	8B	1B	40	2E	A1	C3	F2	38	13	42	1E	84	E7	D2	<table border="1"> <tr><td>E9</td><td>CB</td><td>3D</td><td>AF</td></tr> <tr><td>09</td><td>31</td><td>32</td><td>2E</td></tr> <tr><td>89</td><td>07</td><td>7D</td><td>2C</td></tr> <tr><td>72</td><td>5F</td><td>94</td><td>B5</td></tr> </table>	E9	CB	3D	AF	09	31	32	2E	89	07	7D	2C	72	5F	94	B5	<table border="1"> <tr><td>E9</td><td>CB</td><td>3D</td><td>AF</td></tr> <tr><td>31</td><td>32</td><td>2E</td><td>09</td></tr> <tr><td>7D</td><td>2C</td><td>89</td><td>07</td></tr> <tr><td>B5</td><td>72</td><td>5F</td><td>94</td></tr> </table>	E9	CB	3D	AF	31	32	2E	09	7D	2C	89	07	B5	72	5F	94		<table border="1"> <tr><td>D0</td><td>C9</td><td>E1</td><td>B6</td></tr> <tr><td>14</td><td>EE</td><td>3F</td><td>63</td></tr> <tr><td>F9</td><td>25</td><td>0C</td><td>0C</td></tr> <tr><td>A8</td><td>89</td><td>C8</td><td>A6</td></tr> </table>	D0	C9	E1	B6	14	EE	3F	63	F9	25	0C	0C	A8	89	C8	A6																
EB	59	8B	1B																																																																																		
40	2E	A1	C3																																																																																		
F2	38	13	42																																																																																		
1E	84	E7	D2																																																																																		
E9	CB	3D	AF																																																																																		
09	31	32	2E																																																																																		
89	07	7D	2C																																																																																		
72	5F	94	B5																																																																																		
E9	CB	3D	AF																																																																																		
31	32	2E	09																																																																																		
7D	2C	89	07																																																																																		
B5	72	5F	94																																																																																		
D0	C9	E1	B6																																																																																		
14	EE	3F	63																																																																																		
F9	25	0C	0C																																																																																		
A8	89	C8	A6																																																																																		
salida	<table border="1"> <tr><td>39</td><td>02</td><td>DC</td><td>19</td></tr> <tr><td>25</td><td>DC</td><td>11</td><td>6A</td></tr> <tr><td>84</td><td>09</td><td>85</td><td>0B</td></tr> <tr><td>1D</td><td>FB</td><td>97</td><td>32</td></tr> </table>	39	02	DC	19	25	DC	11	6A	84	09	85	0B	1D	FB	97	32																																																																				
39	02	DC	19																																																																																		
25	DC	11	6A																																																																																		
84	09	85	0B																																																																																		
1D	FB	97	32																																																																																		

<http://www.youtube.com/watch?v=mlzxpkdXP58>



4.4 Modos de cifrado en bloque

- Se ha convenido en denominar al uso directo de un cifrador en bloque como modo de «Libro Electrónico de Códigos» (Electronic Codebook, **ECB**).
- Entre otros, el NIST (U.S. National Institute for Standards and Technology) recomienda cuatro modos de uso, tanto para AES como para cualquier cifrado en bloque:
 - Encadenamiento de bloques cifrados (*Cipher Block Chaining*, **CBC**)
 - Realimentación del texto cifrado (*Cipher Feedback*, **CFB**)
 - Realimentación de la salida (*Output Feedback*, **OFB**)
 - Modo contador (*Counter mode*, **CTR**)



4.4.1 Modo ECB (Electronic CodeBook)

MODO ECB

Electronic CodeBook: cifra cada bloque con la clave k de forma independiente como si fuese un gran *libro electrónico de códigos*.

Debilidades:

- Se podría reconstruir ese libro electrónico sin necesidad de conocer la clave.
- Aparece el problema denominado de comienzos y finales fijos que permiten un tipo de ataque.
- Ataque mediante repetición de bloques similares.



4.4.1 Modo ECB (Electronic CodeBook)

- ▶ El modo ECB es el método más sencillo y obvio de aplicar a un algoritmo de cifrado por bloques.
- ▶ Simplemente se subdivide la cadena que se quiere cifrar en bloques del tamaño adecuado y se cifran todos ellos empleando la misma clave.
- ▶ Entre las **ventajas** de este método destaca la posibilidad de dividir el mensaje en bloques y cifrarlos en paralelo o el acceso aleatorio a diferentes bloques.

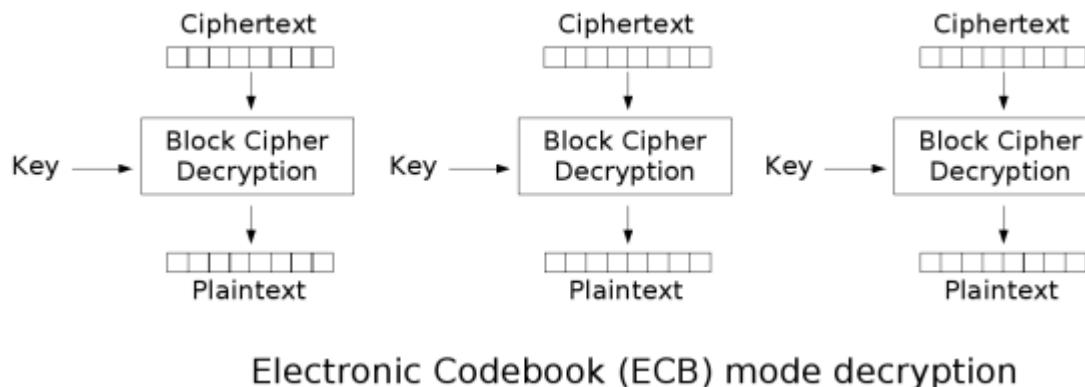
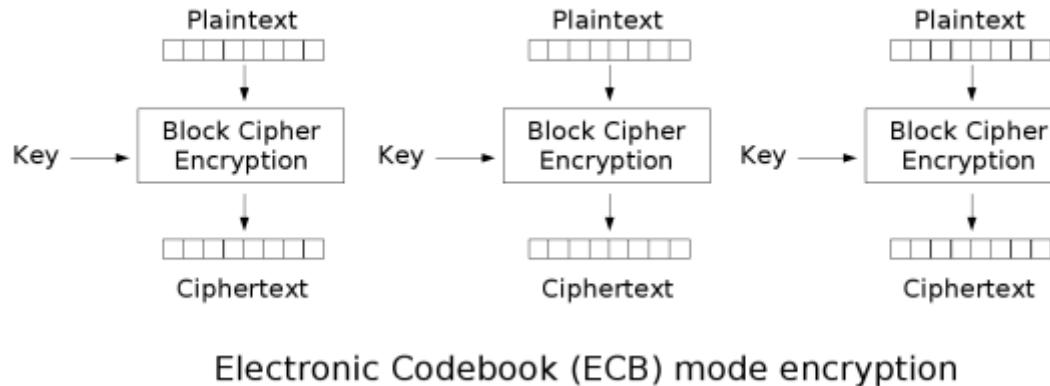


4.4.1 Modo ECB (Electronic CodeBook)

- ▶ Sin embargo, las **desventajas** de este modo de cifrado son enormes, por lo que se usa cada vez menos.
 - ▶ El hecho de cifrar los bloques por separado implica que **cuando se cifice un bloque con cierto valor, siempre** se obtendrá el **mismo resultado**.
 - ▶ Esto hace posible los ataques de diccionario.
- ▶ Además, cuando se cifran varios bloques y se envían por un canal inseguro, es posible que un adversario elimine ciertos bloques sin ser detectado, o que capture algunos bloques y los reenvíe más adelante.



4.4.1 Modo ECB (Electronic CodeBook)



4.4.2 Modo CBC (Cipher Book Chaining)

- Este modo de cifrado es una extensión de ECB que añade cierta seguridad.
- El modo de cifrado CBC divide el mensaje en bloques y usa XOR para combinar el cifrado del bloque anterior con el texto en claro del bloque actual.



4.4.2 Modo CBC (Cipher Book Chaining)

- ▶ Como no se dispone de un texto cifrado con el que combinar el primer bloque, se usa un vector de inicialización VI (número aleatorio que puede ser públicamente conocido).
- ▶ El uso del vector de inicialización es importante, pues de no usarlo, podría ser susceptible de ataques de diccionario.
- ▶ También es necesario que el VI sea aleatorio y no un número secuencial o predecible.



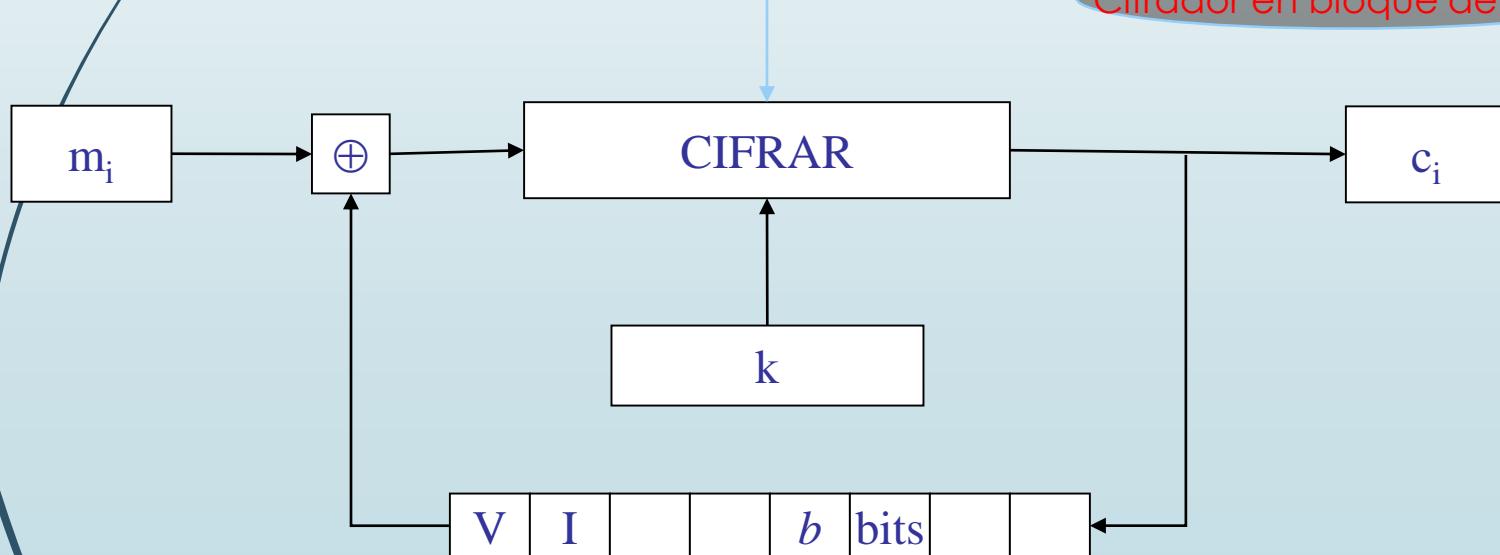
4.4.2 Modo CBC (Cipher Book Chaining)

- Para empezar, se carga inicialmente el registro de b bits con un vector inicial (VI) que no importa que sea secreto, pero sí conviene que sea aleatorio.
- Cada bloque m_i de b bits del texto en claro se cifra con la misma clave k y el bloque de salida c_i se realimenta hacia la entrada mediante el registro de b bits
- Se aplica la siguiente recurrencia para cifrar:

$$c_1 = E_k(m_1 \oplus VI); \quad c_i = E_k(m_i \oplus c_{i-1}), \text{ para } i = 2, 3, \dots, n$$

donde n es el número de bloques a cifrar.

Cifrador en bloque de b bits

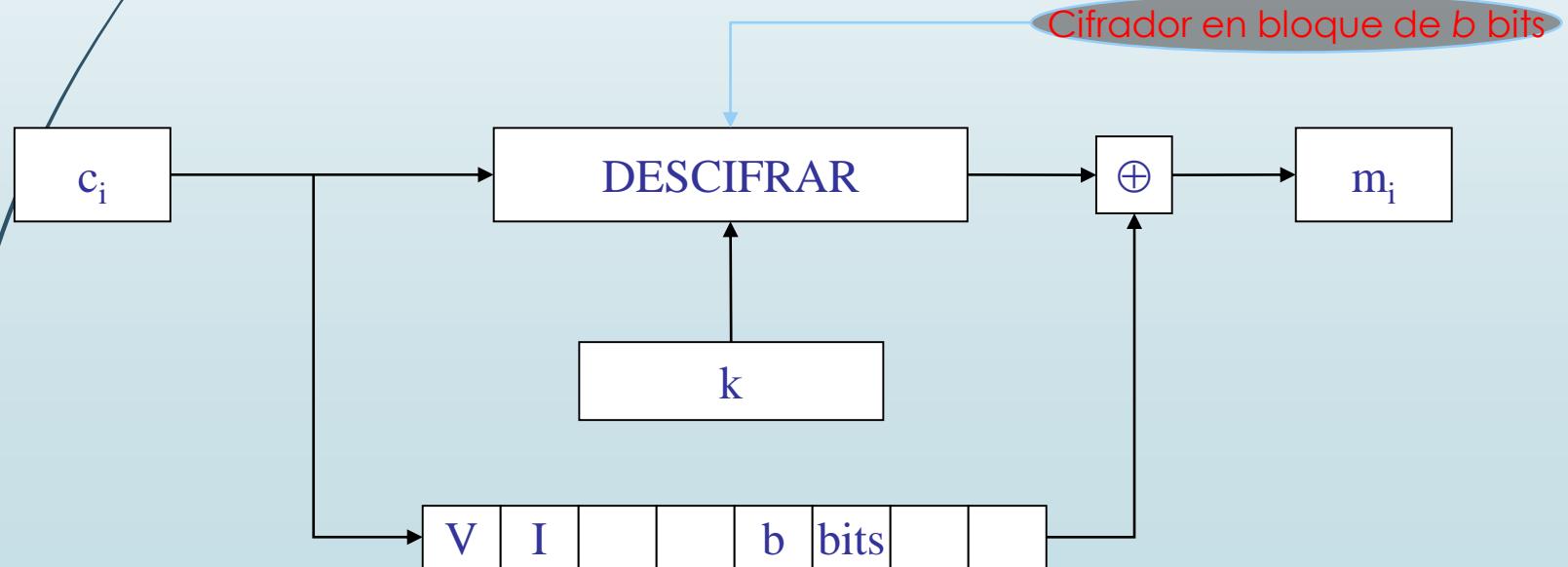


4.4.2 Modo CBC (Cipher Book Chaining)

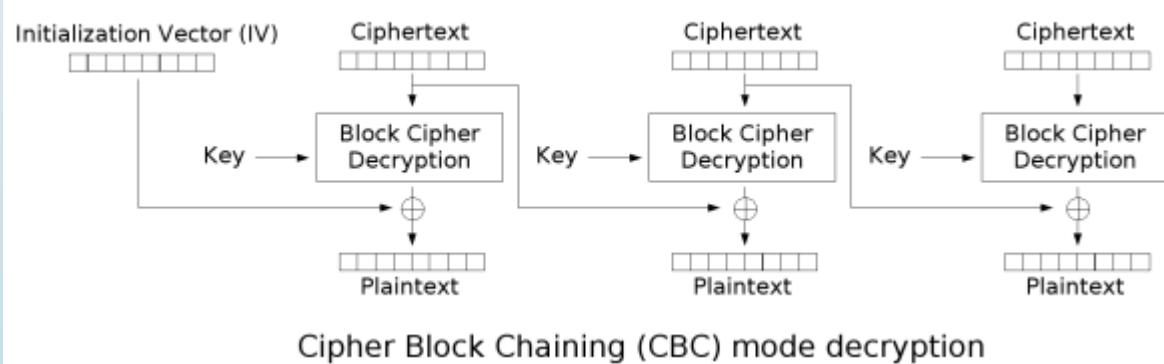
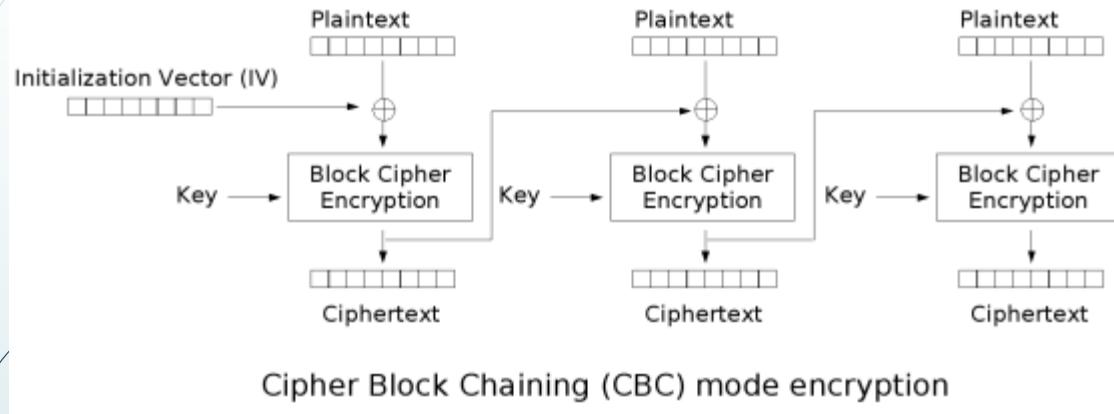
- Para **descifrar**, cada bloque c_i de b bits del criptograma se descifra con la misma clave k y alimenta el registro de b bits que se suma módulo 2 con la salida $D(c_i)$.
- Se tiene

$$D_k(c_1) = D_k[E_k(m_1 \oplus VI)] = m_1 \oplus VI, \text{ luego } m_1 = D_k(c_1) \oplus VI.$$

$$D_k(c_i) = D_k[E_k(m_i \oplus c_{i-1})] = m_i \oplus c_{i-1}, \text{ luego } m_i = D_k(c_i) \oplus c_{i-1} \text{ para } i = 2, 3, \dots, n.$$



4.4.2 Modo CBC (Cipher Book Chaining)



4.4.2 Modo CBC (Cipher Book Chaining)

- ▶ Sus propiedades son:
 - ▶ Cada bloque depende de todos los bloques que le anteceden.
 - ▶ Convierte el cifrador en bloque en un cifrador en flujo y, por tanto, oculta los perfiles del mensaje claro.
 - ▶ Se puede hacer que cifre mensajes iguales de forma diferente con sólo cambiar cada vez el VI.
 - ▶ Limita la propagación de cada error de transmisión a dos bloques.
 - ▶ No cambia el tamaño del espacio de claves.



4.4.3 Modo CFB (Cipher-Feedback)

- ▶ El modo CBC no empieza a cifrar (o descifrar) hasta que no se tiene que transmitir (o se ha recibido) un bloque completo de información (128 bits, por ejemplo, para AES).
- ▶ Esta circunstancia puede convertirse en un serio inconveniente, por ejemplo en el caso de terminales, que deberían poder transmitir cada carácter que pulsa el usuario de manera individual.
- ▶ Una posible solución sería emplear un bloque completo para transmitir cada byte y llenar el resto con ceros, pero esto hará que tengamos únicamente 256 mensajes diferentes en nuestra transmisión y que un atacante pueda efectuar un sencillo análisis estadístico para comprometerla.



4.4.3 Modo CFB (Cipher-Feedback)

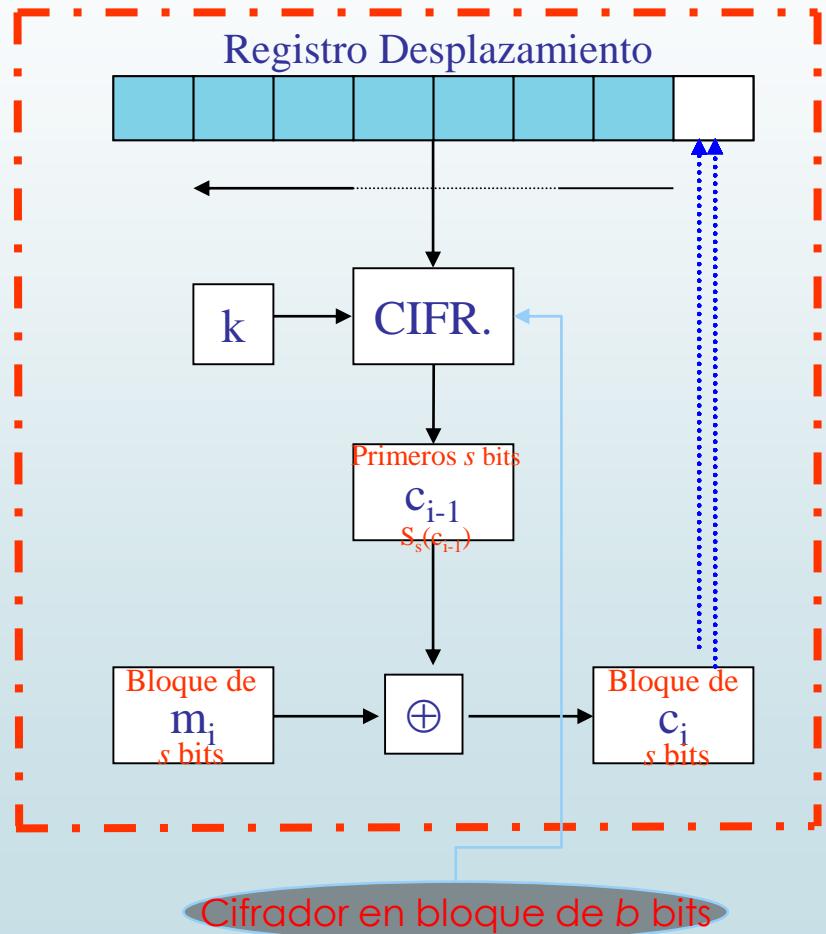
- ▶ Otra opción sería llenar el bloque con información aleatoria, aunque seguiríamos desperdiciando gran parte del ancho de banda de la transmisión.
- ▶ El modo de operación CFB (Cipher-Feedback Mode) permitirá cifrar la información en unidades inferiores al tamaño del bloque (128 bits, por ejemplo, para AES), lo cual permite aprovechar totalmente la capacidad de transmisión del canal de comunicaciones, manteniendo además un nivel de seguridad adecuado.



4.4.3 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 b 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
- $$c_i = m_i \oplus S_s[E_k(RD)], i = 1, 2, \dots$$

El bloque se va desplazando por el registro



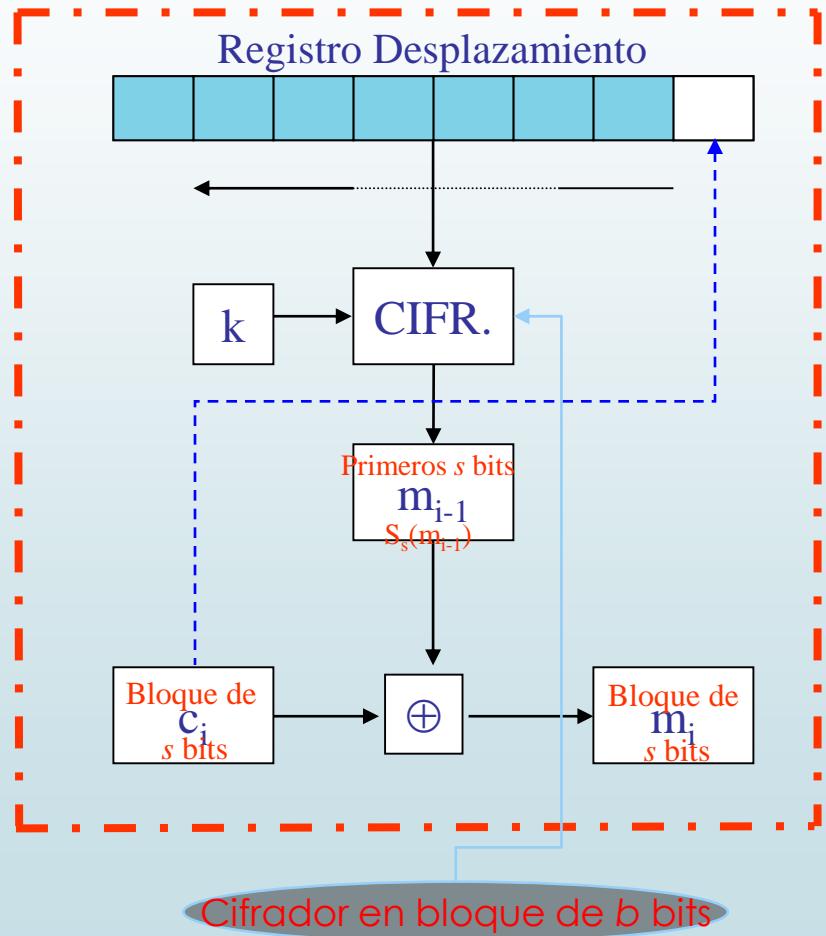
Cifrado en bloque con clave secreta



4.4.3 Modo CFB (Cipher-Feedback)

- Para la operación de descifrado, el esquema es básicamente el mismo: se intercambian los valores de c con m y la realimentación hacia el registro de desplazamiento sigue siendo desde el bloque de s bits del criptograma.
 - Si $S_s(x)$ representa los s bits más significativos de x , se tiene
- $$m_i = c_i \oplus S_s[E_k(RD)], i = 1, 2, \dots$$

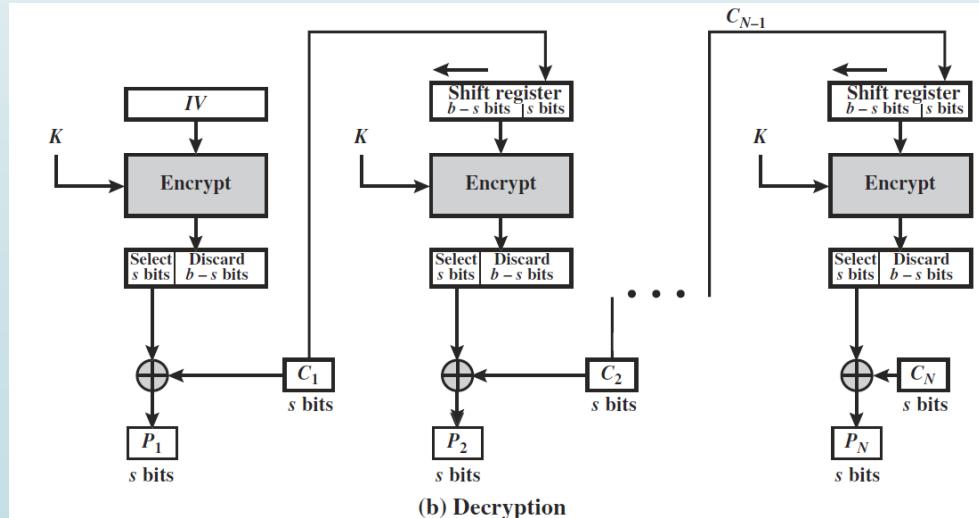
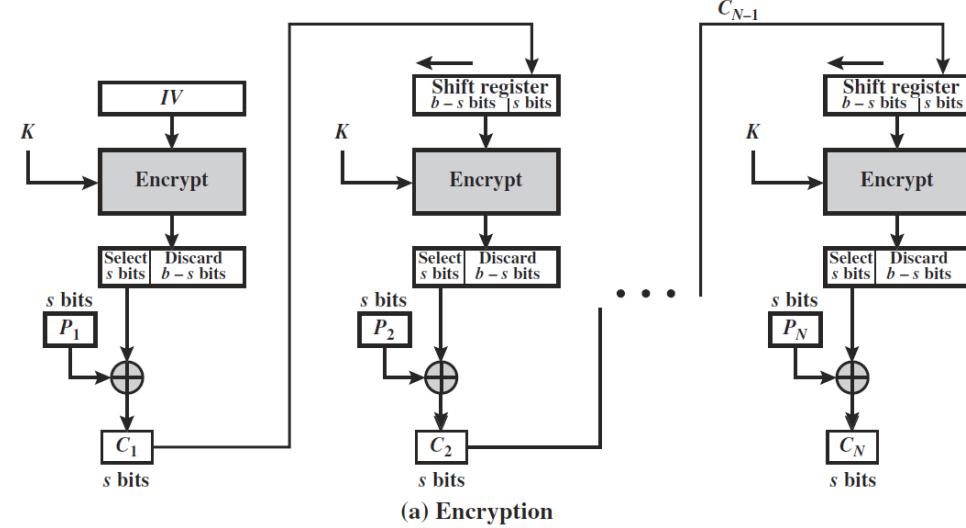
El bloque se va desplazando por el registro



4.4.3 Modo CFB (Cipher-Feedback)

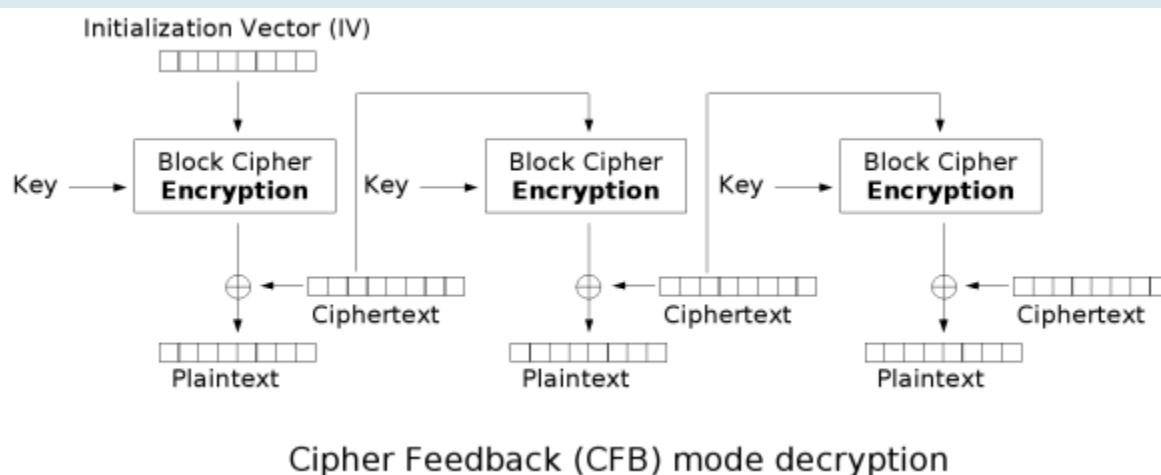
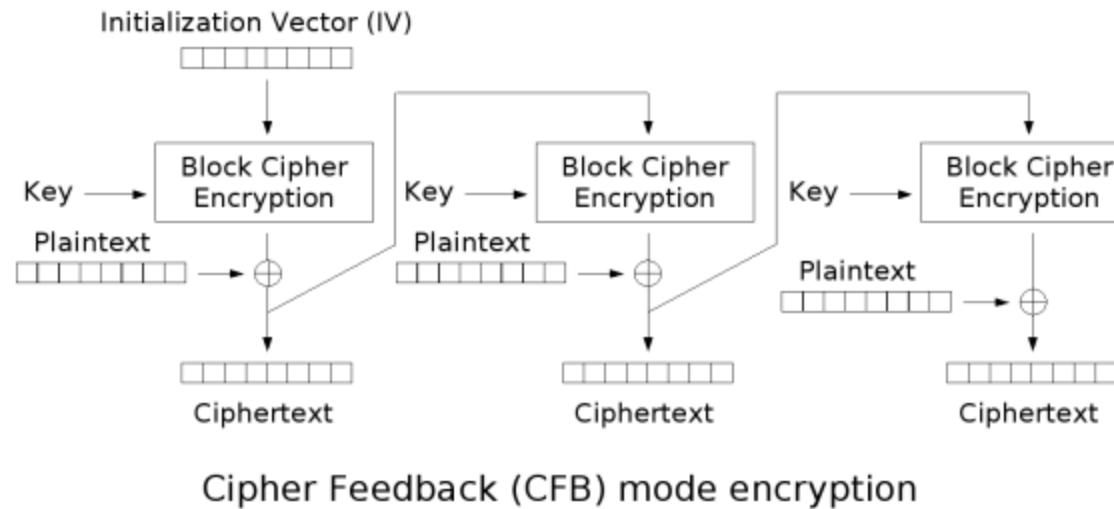
→ Modo CFB tomando el tamaño de bloque s (habitualmente 1 byte)

b tamaño de bloque
 P_i texto en claro
 C_i criptograma



4.4.3 Modo CFB (Cipher-Feedback)

- Modo CFB tomando s =tamaño de bloque del algoritmo (para AES $s=128$ bits)



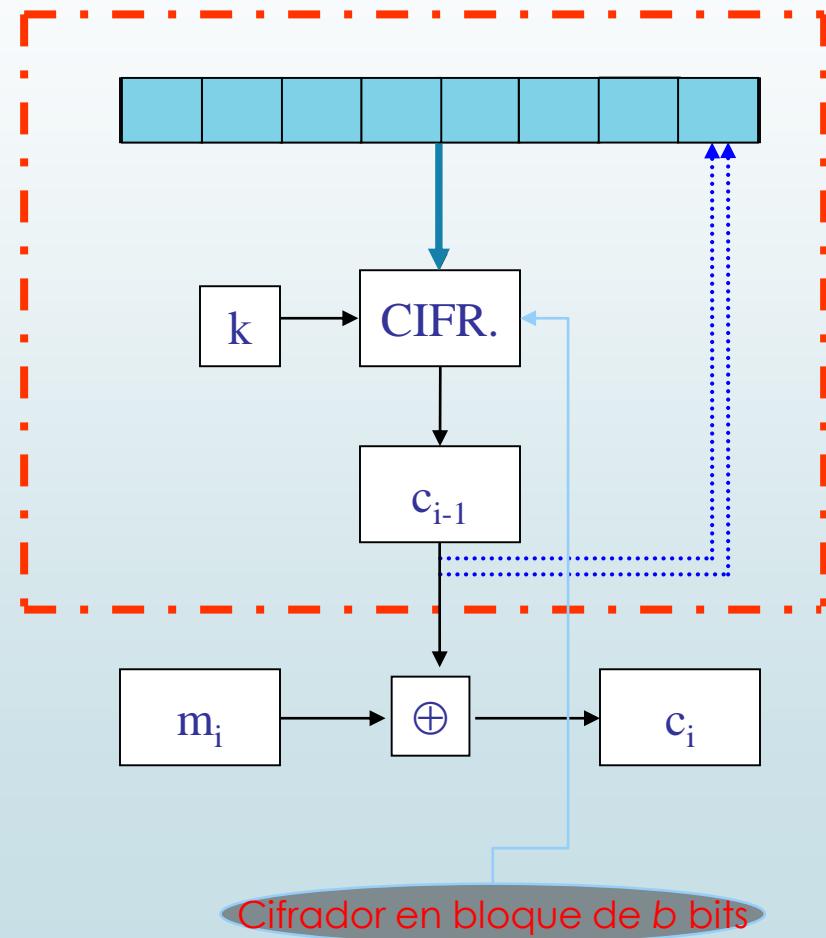
4.4.3 Modo CFB (Cipher-Feedback)

- ▶ Sus propiedades son:
 - ▶ Convierte el cifrador en bloque en un cifrador en flujo y, por tanto, oculta los perfiles del mensaje claro.
 - ▶ Se puede hacer que cifre mensajes iguales de forma diferente con sólo cambiar cada vez el VI.
 - ▶ Limita la propagación de cada error de transmisión a los s bits posteriores al bit afectado.
 - ▶ No cambia el tamaño del espacio de claves.



4.4.4 Modo OFB (Output Feedback)

- El modo OFB (Output Feedback) es muy parecido al anterior (CFB) con la única diferencia de que la **retroalimentación con la señal de entrada al cifrador en bloque se realiza antes** de la operación or-exclusiva.
- En este caso la función de cifrado E_k y el registro actúan como un generador de secuencia cifrante de bloques de b bits con la particularidad que al incluirse el vector inicial VI en ella, el efecto es doblar el tamaño de la clave del cifrador.
- El generador de secuencia cifrante es el que se encuentra encerrado entre líneas de puntos rojos.
- Las operaciones de cifrado y descifrado son exactamente iguales.



Cifrado en bloque con clave secreta

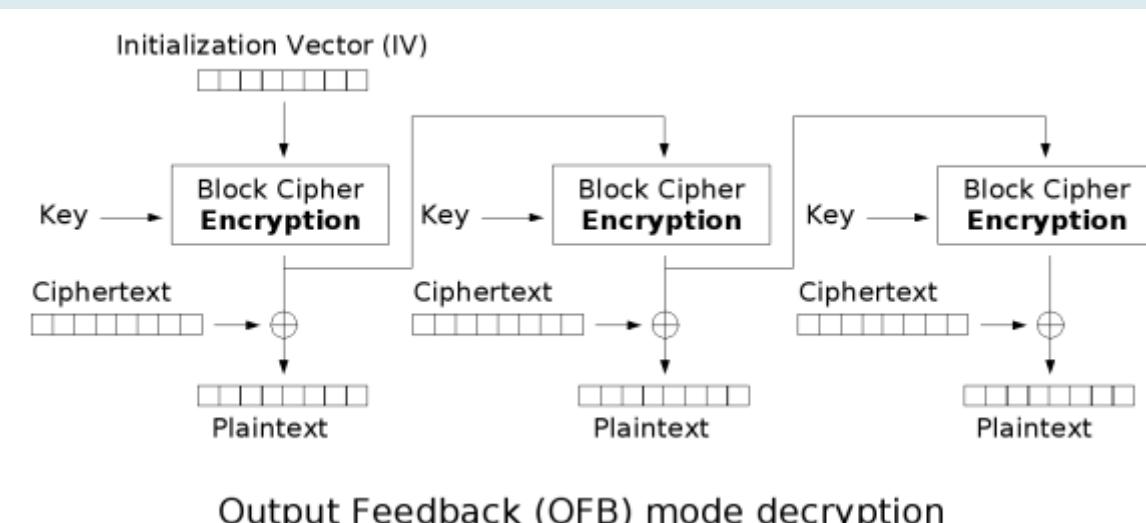
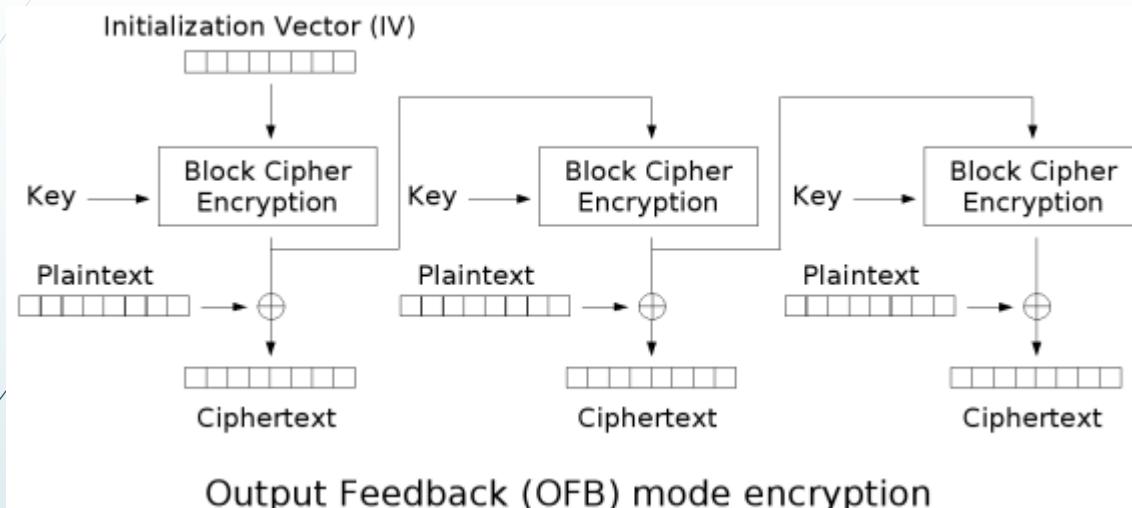


4.4.4 Modo OFB (Output Feedback)

- ▶ Sus propiedades son:
 - ▶ Convierte el cifrador en bloque en un cifrador en flujo y, por tanto, oculta los perfiles del mensaje claro.
 - ▶ Usa el cifrador en bloque como un generador de secuencia cifrante.
 - ▶ Se puede hacer que cifre mensajes iguales de forma diferente con sólo cambiar cada vez el VI.
 - ▶ Limita la propagación de cada error de transmisión al bit afectado; es decir, no hay propagación de errores.
 - ▶ Dobla el tamaño del espacio de claves, que ahora incluye el VI.
 - ▶ El cifrado y descifrado son idénticos.



4.4.4 Modo OFB (Output Feedback)



4.4.5 Modo CTR (Counter mode)

- Un inconveniente del cifrado por encadenamiento es que para cifrar cada bloque hay que tener previamente el cifrado del anterior.
- Esto añade **una dependencia de datos en el algoritmo** que impide usar procesamiento paralelo, lo que es una importante penalización de rendimiento en la computación actual.
- Una variante, que no requiere alimentación de los bloques anteriores, consiste en usar un contador (counter, abreviado CTR) como entrada del bloque de cifrado (AES o similar).



4.4.5 Modo CTR (Counter mode)

- ▶ Los datos propiamente dichos no llegan a pasar por el cifrado, sino que simplemente se combinan mediante or-exclusiva con el cifrado.
- ▶ Al contador se le adjunta un valor aleatorio (similar al vector de inicialización del CBC) para evitar que el cifrado sea siempre idéntico.
- ▶ Este valor se suele denominar por el término inglés nonce, que tiene difícil traducción y viene a significar "de un solo uso".



4.4.5 Modo CTR (Counter mode)

- Mientras que ECB y CBC son modos basados en bloques, CTR simula un cifrado en flujo.
- Es decir, se usa un cifrado de bloque para producir una secuencia pseudoaleatoria binaria (conocida como keystream).
- Esta secuencia se combina con el texto en claro mediante or-exclusiva dando lugar al cifrado (Vernam).



4.4.5 Modo CTR (Counter mode)

- ▶ Para generar la secuencia pseudoaleatoria se cifra un contador combinado con un nonce mediante ECB y se va incrementando.
- ▶ El valor del contador puede ser públicamente conocido, aunque es preferible guardarlo en secreto.
- ▶ Es necesario que el valor de [nonce + contador] lo conozcan ambos lados de la comunicación (donde el símbolo + significa concatenación)



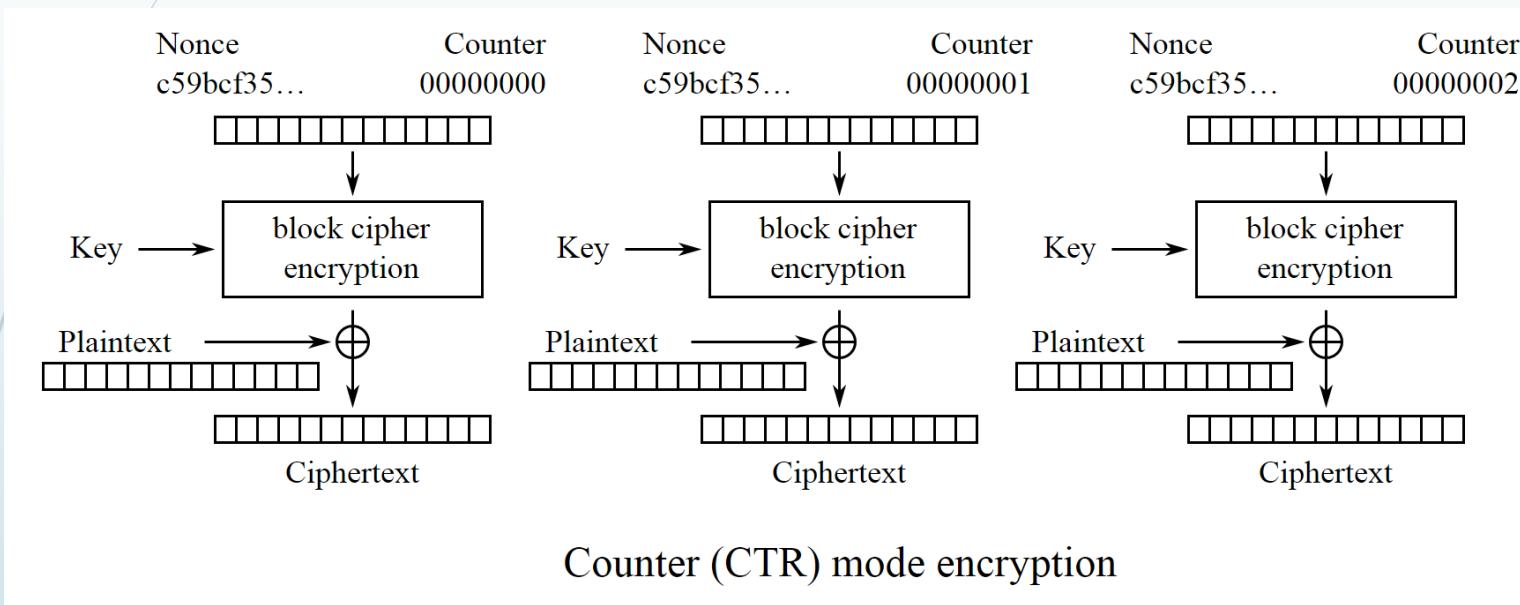
4.4.5 Modo CTR (Counter mode)

- ▶ En términos generales el NIST especifica **dos tipos de contadores**.
 - ▶ El **primero** se compone de un nonce y un contador. El nonce es aleatorio, y los bytes restantes son bytes de contador (que se incrementan).
 - ▶ Por ejemplo, un cifrado de bloque de 16 bytes podría utilizar los 8 bytes más significativos como un nonce y los 8 bytes menos significativos como un contador.
 - ▶ El **segundo** es un bloque de contador, donde todos los bytes son bytes de contador y se pueden incrementar a medida que se genera la secuencia cifrante.
 - ▶ Por ejemplo, en un cifrado de bloque de 16 bytes, los 16 bytes son bytes de contador.



4.4.5 Modo CTR (Counter mode)

- Este es el diagrama de bloques donde se observa claramente el procesamiento paralelo:

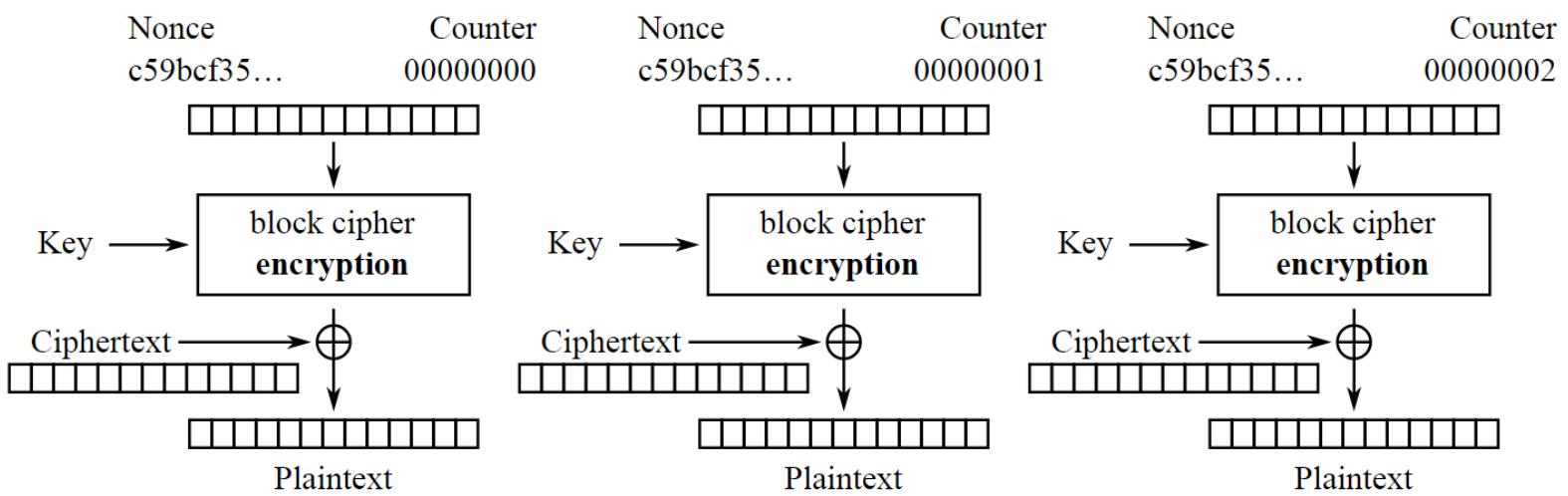


- En un primer vistazo puede parecer arriesgado depender de un valor previsible y sistemático para hacer el cifrado, pero lo cierto es que la caja negra de cifrado (AES, normalmente) inserta por sí mismo suficiente aleatoriedad que se propaga al cifrado final mediante la operación or-exclusiva.



4.4.5 Modo CTR (Counter mode)

- Otra ventaja de este modo es que el mecanismo de descifrado se hace simplemente invirtiendo el orden en la operación or-exclusiva.
- La parte dura del procesamiento (el bloque AES) es idéntica y no necesitamos un bloque de descifrado. Esto hace que la implementación (ya sea software o hardware) se simplifique enormemente.



4.4.5 Modo CTR (Counter mode)

► Otra representación esquemática del modo CTR

P_i texto en claro
 C_i criptograma

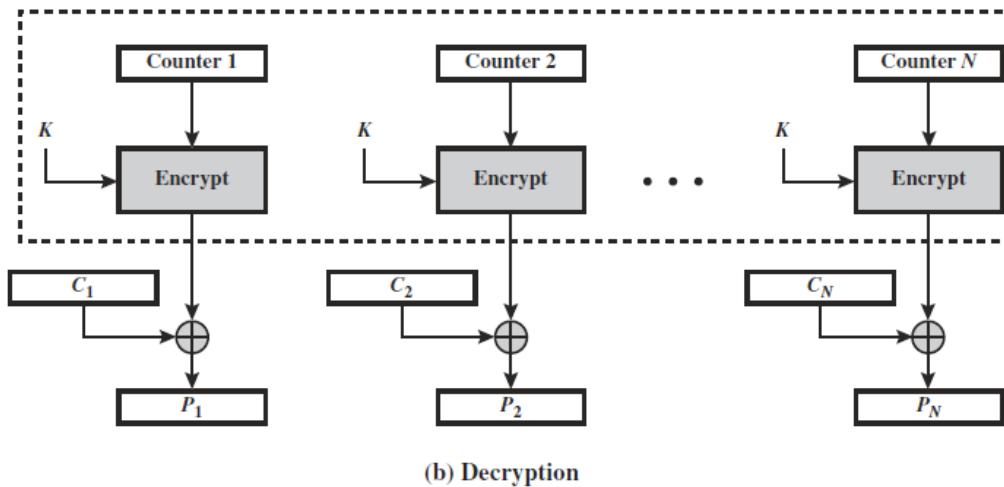
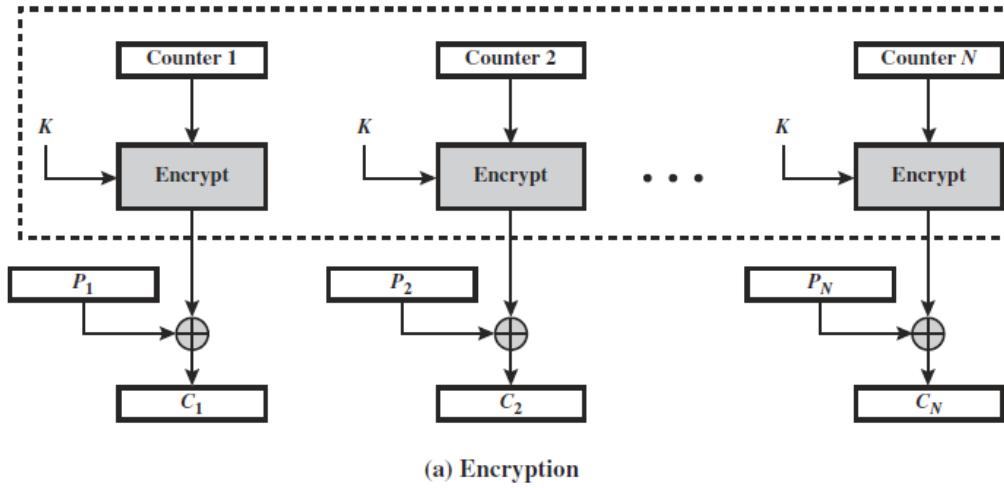


Figure 6.7 Counter (CTR) Mode



4.4.5 Modo CTR (Counter mode)

- Si consideramos que T_1, T_2, \dots, T_N es una secuencia de contadores (obtenida en cualquiera de los dos modos aceptados por el NIST), el cifrado se obtiene mediante las expresiones

$$c_i = m_i \oplus E_k(T_i), \quad i = 1, 2, \dots, N-1$$

$$c_N = m_N \oplus S_s[E_k(T_N)]$$

- donde $S_s(x)$ representa los s bits más significativos de x .
- Obsérvese que para el último bloque a cifrar (que tiene s bits), al contrario que en los modos ECB, CBC y CFB, no se necesita relleno (*padding*).



4.4.5 Modo CTR (Counter mode)

- El descifrado se obtiene mediante las expresiones.

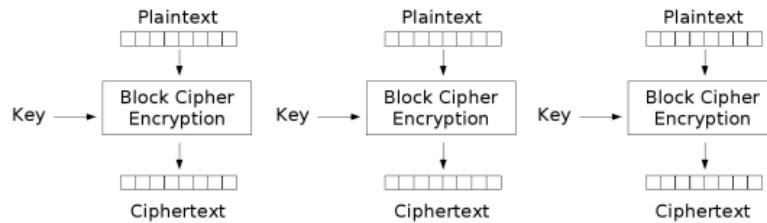
$$m_i = c_i \oplus E_k(T_i), \quad i = 1, 2, \dots, N-1$$

$$m_N = c_N \oplus S_s[E_k(T_N)]$$

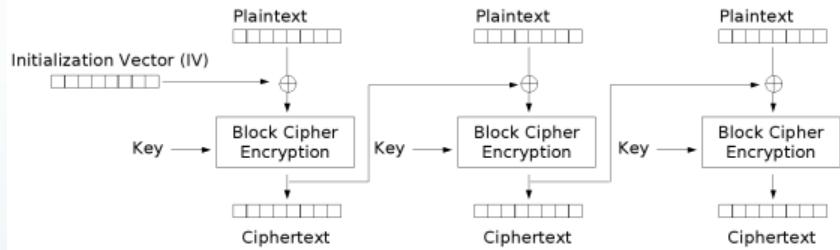
- donde $S_s(x)$ representa los s bits más significativos de x.



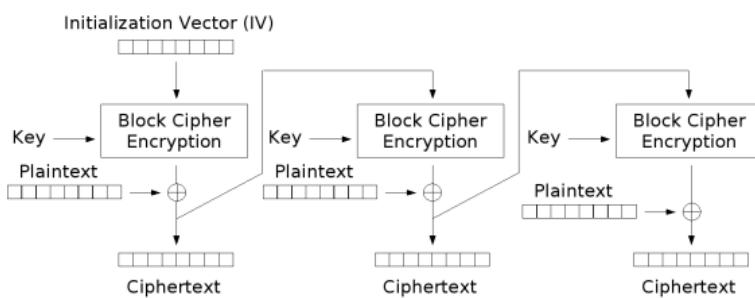
Resumen de modos de cifrados



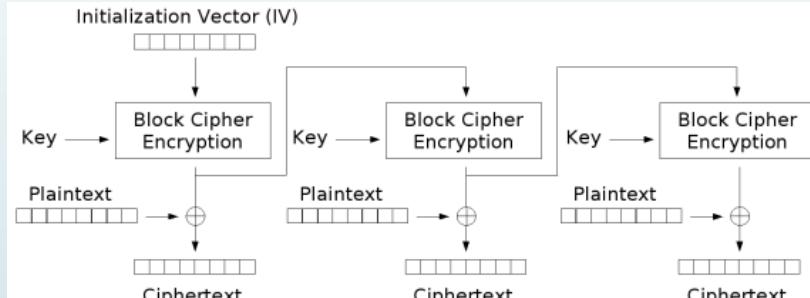
Electronic Codebook (ECB) mode encryption



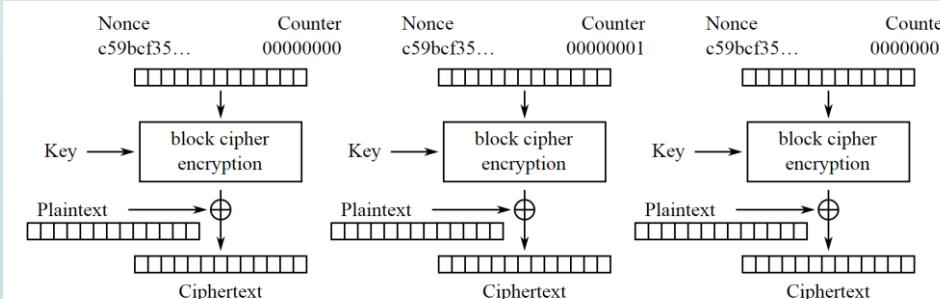
Cipher Block Chaining (CBC) mode encryption



Cipher Feedback (CFB) mode encryption



Output Feedback (OFB) mode encryption



Counter (CTR) mode encryption



4.5 Cifrado múltiple. Triple DES

- Si un sistema forma un grupo algebraico (esto es, si está cerrado bajo una operación de composición consistente en el cifrado repetido) cifrar un mensaje m con una clave k_1 y luego el resultado con una clave k_2 , es lo mismo que cifrar el mensaje con una única clave k_3 .
- Un ejemplo lo constituye el cifrado de Vigenère.
 - Sea $k_1 = \text{LUCIA}$ y $k_2 = \text{JUANA}$ y el mensaje a cifrar $m = \text{ESTO ES UN GRUPO}$.

$m_1 = \text{ESTOESUNGRUPO}$
 $k_1 = \text{LUCIALUCIALUC}$
 $c_1 = \text{ONVWEDOOÑRFKQ}$

$m_2 = \text{ONVWEDOOÑRFKQ}$
 $k_2 = \text{JUANAJUANAJUA}$
 $c_2 = \text{XHVJEMJOARÑEQ}$

Es fácil comprobar que se obtiene lo mismo al cifrar el texto en claro m_1 con la clave $k_3 = k_1 + k_2 = \text{LUCIA} + \text{JUANA} = \text{TOCUA}$.

DES no es un grupo y, por tanto, el cifrado múltiple permitirá aumentar el tamaño efectivo de la clave.



4.5 Cifrado múltiple. Triple DES

- El procedimiento para aumentar el espacio de claves de un cifrado en bloque consiste en hacer un cifrado múltiple, también denominado supercifrado en medios militares.
- Para una repetición del cifrado en DES n veces, usando n claves independientes, se puede demostrar que la longitud efectiva de la clave en bits es, aproximadamente:

$$l = 56 \left\lceil \frac{n}{2} \right\rceil$$

en vez de $56n$.

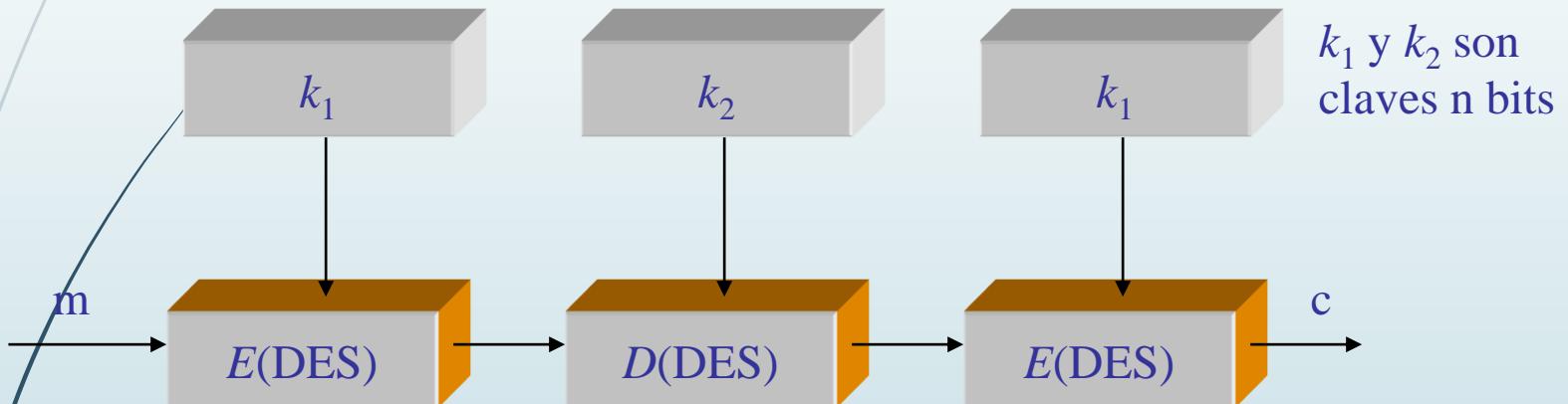
- En el caso particular de $n = 2$, la longitud exacta de la clave frente a un ataque por prueba exhaustiva de claves es solamente de 57 bits.
 - Para n igual a 3, la longitud de la clave se duplica (112 bits)



4.5 Cifrado múltiple. Triple DES

EDE: Encrypt-Decrypt-Encrypt

$$c = E_{k_1} \left(D_{k_2} \left[E_{k_1} (m) \right] \right)$$



- En este caso se logra un valor efectivo de longitud de clave igual a $2n$ bits, es decir $2 \cdot 56 = 112$ bits.
- El método fue propuesto por Matyas y Meyer de IBM y se denomina EDE: *Encrypt-Decrypt-Encrypt*.

