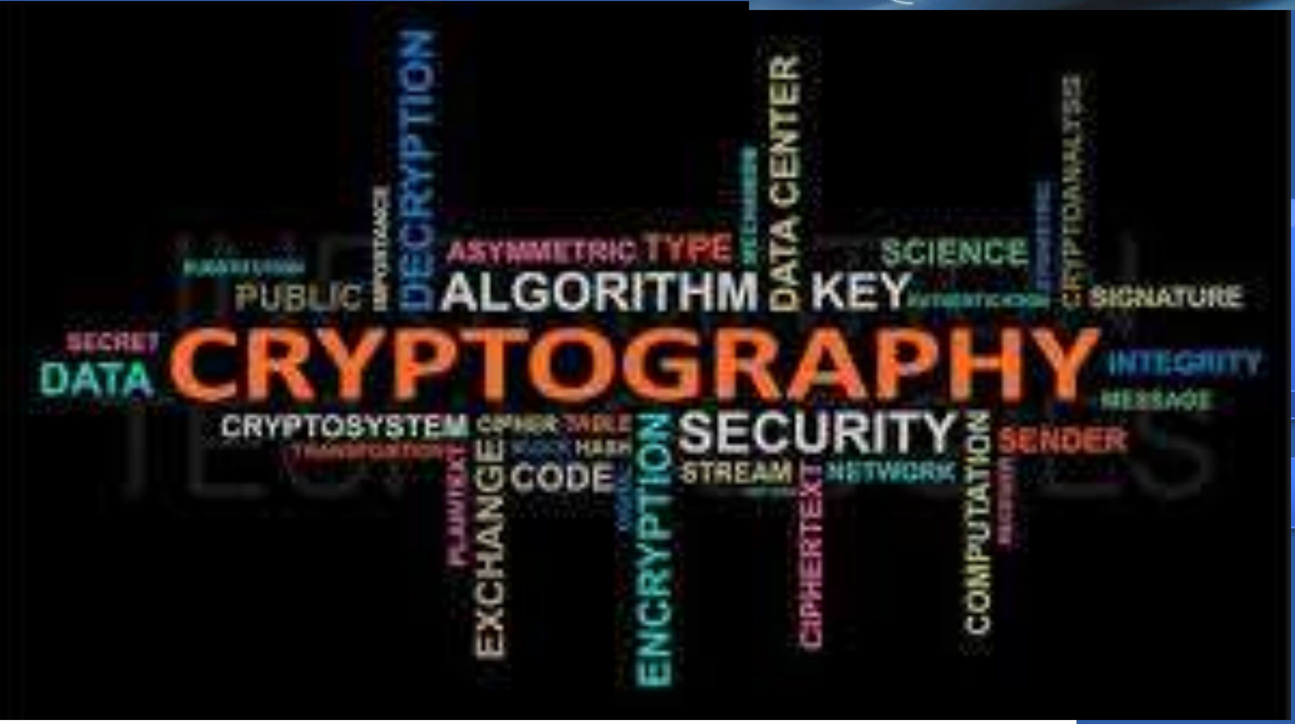
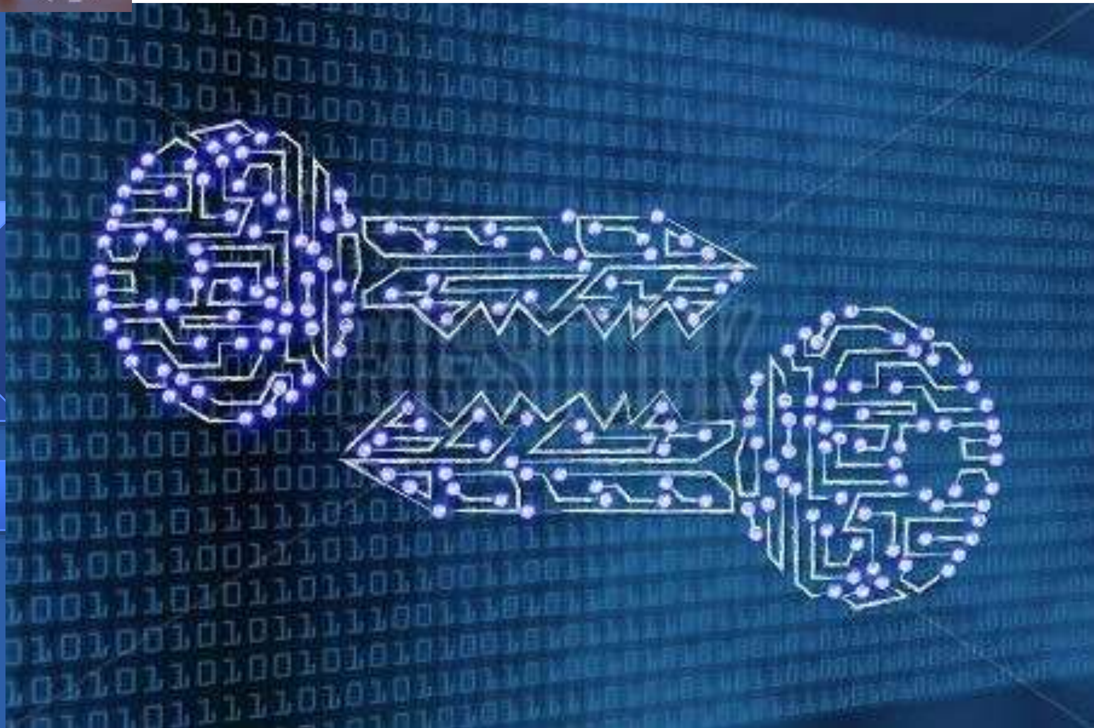
Universidad Francisco de Vitoria UFV Madrid

Criptografía

Tema 3 Cifrado Simétrico





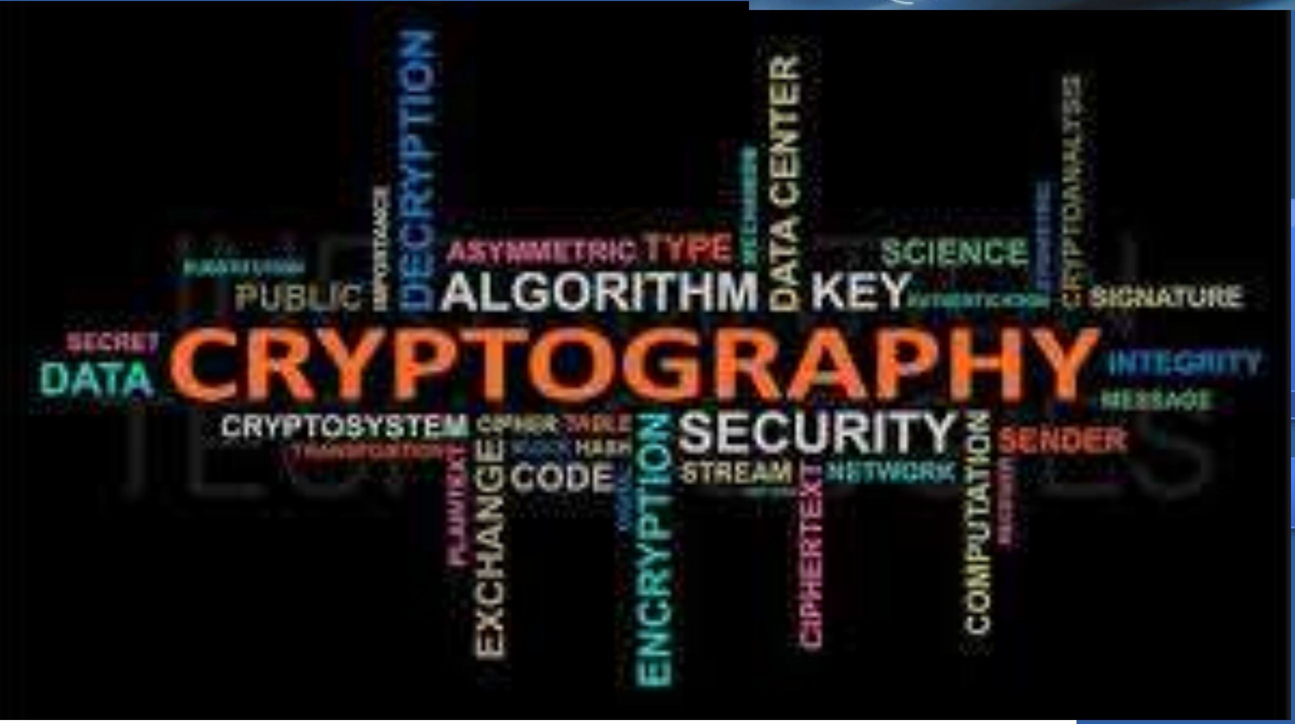
Criptografía Clásica

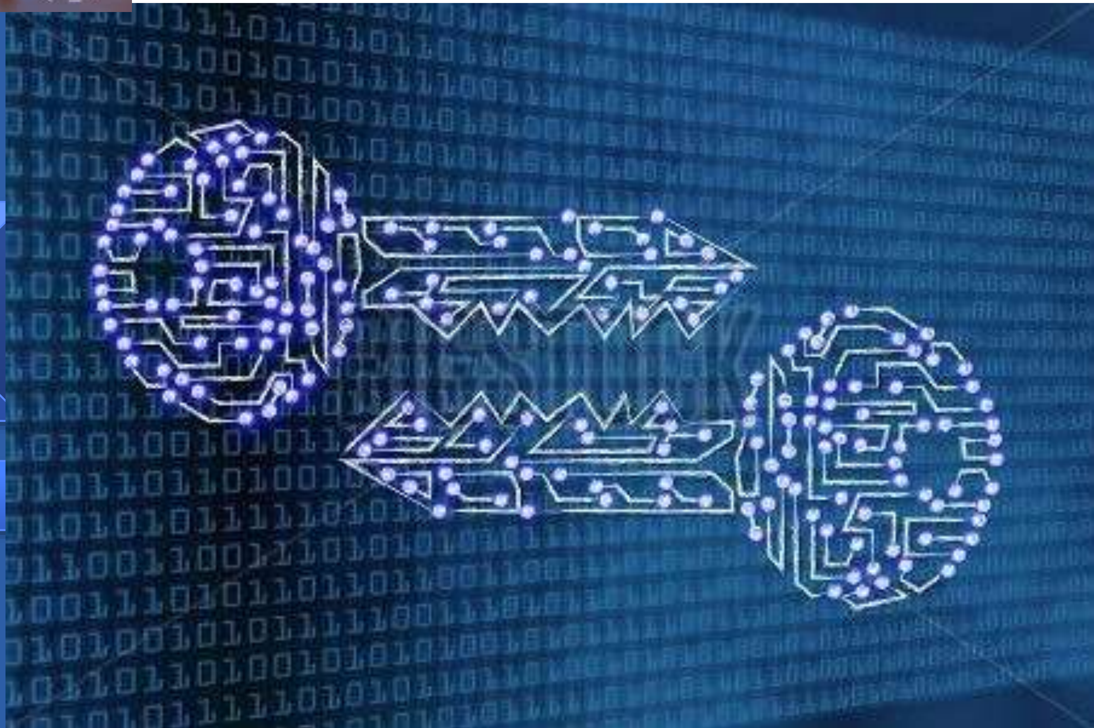
- Introducción
- Algoritmos de cifrado simétrico de flujo
- Algoritmos de cifrado simétrico de bloque
- Modos de operación

Criptografía

Universidad Francisco de Vitoria UFV Madrid

Tema 3.1 Introducción al Cifrado Simétrico





Criptografía Moderna

Podemos clasificar los algoritmos según varias dimensiones:

- **Por el tipo de clave usada**
  - **Simétricos**
  - **Asimétricos**
- **Por la cantidad de elementos tomados a la vez:**
  - **Cifradores de flujo**
  - **Cifradores de bloque**

.

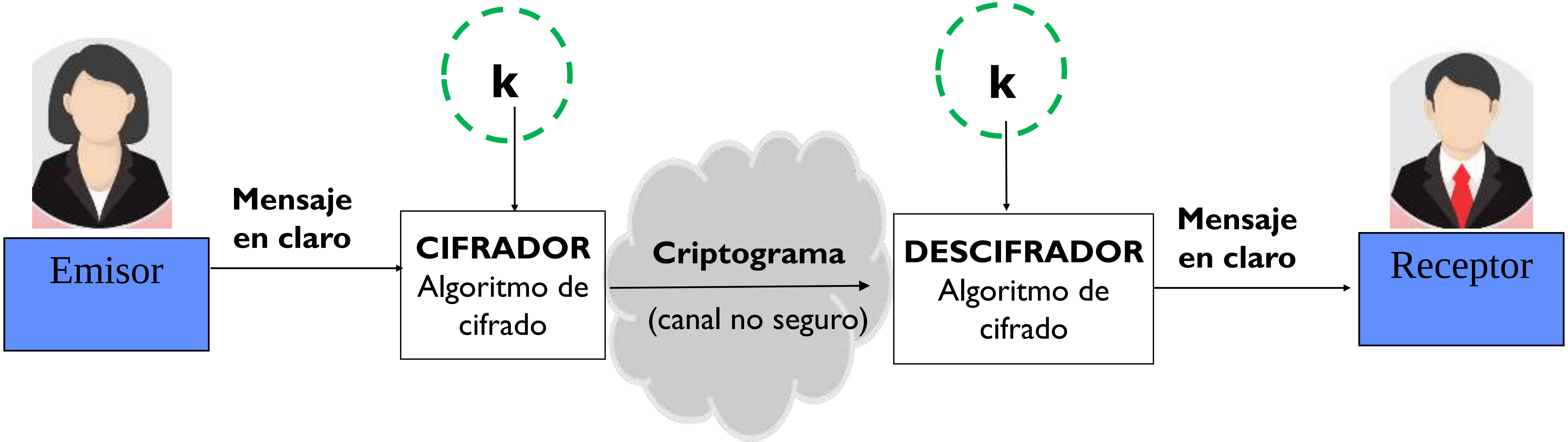
Criptografía Moderna

- **Por el tipo de clave usada**
  - Simétricos
  - Asimétricos

.

Modelo de Criptosistema

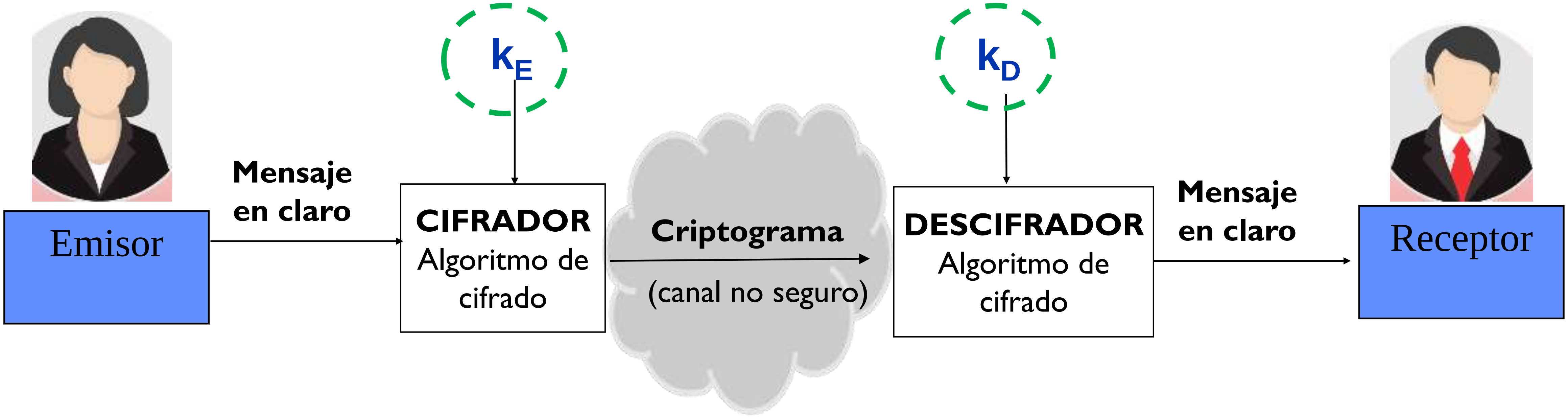
SIMÉTRICOS: Misma clave (k) para cifrar y descifrar



- La clave se acuerda y comparte entre las en modo secreto

Modelo de Criptosistema

ASIMÉTRICOS: Una clave (k_E) para cifrar y otra (k_D) para descifrar



- Base de los sistemas de clave pública

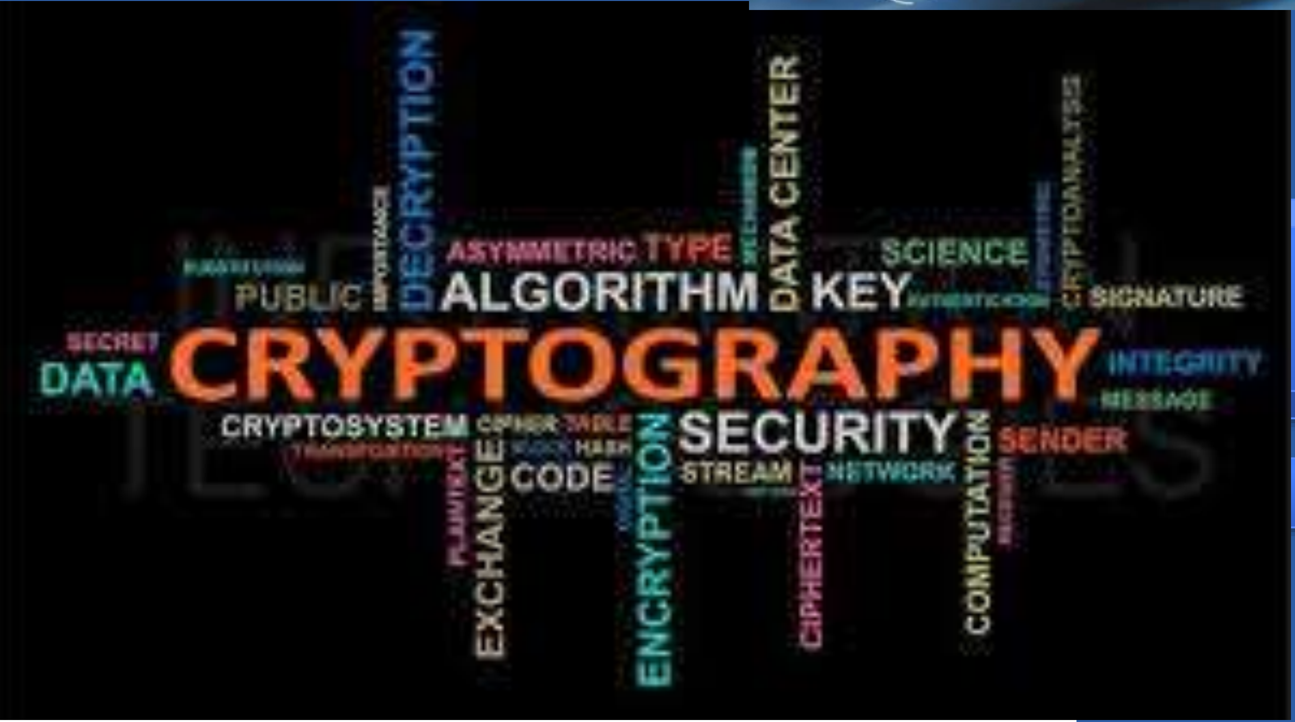
Criptografía Moderna

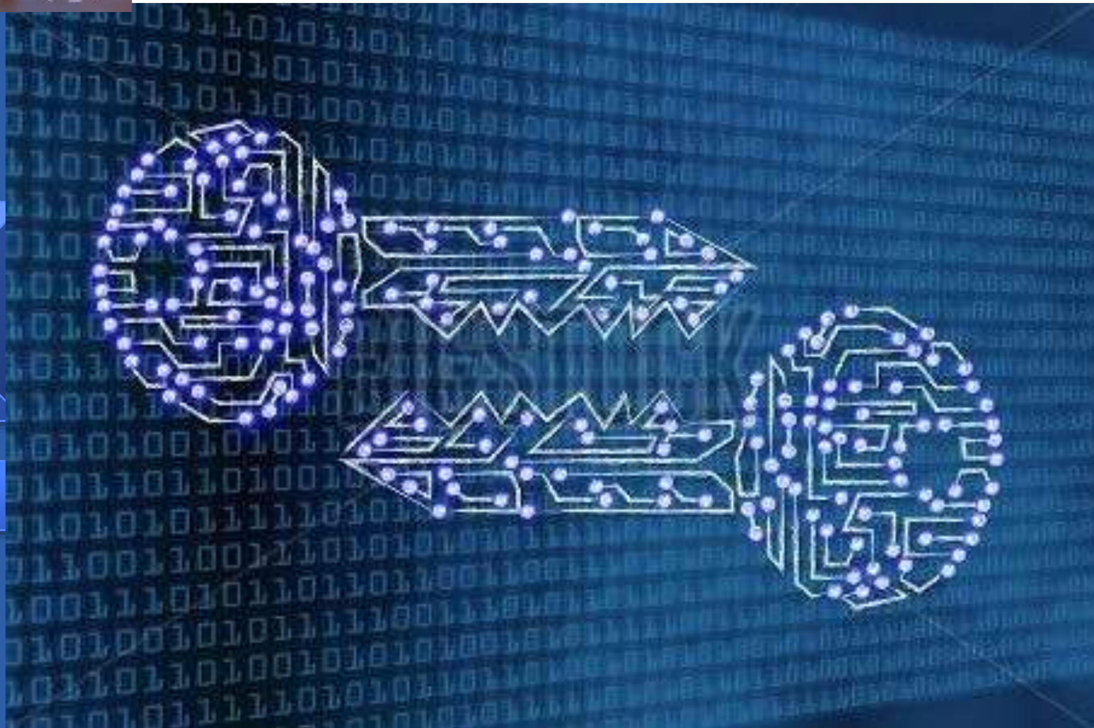
- **Por la cantidad de elementos tomados a la vez:**
  - **Cifradores de flujo**
  - **Cifradores de bloque**

.

Criptografía

Tema 3.2 Algoritmos de cifrado simétrico de flujo





Cifradores de flujo

- **Dividen el mensaje en caracteres o símbolos**
  - **M = m1 m2 m3 m4…**

- **El flujo de cifrado (clave) también se divide en caracteres o símbolos**
  - **K = k1 k2 k3 k4…**

- **Cada carácter se cifra con el carácter correspondiente de la clave:**
  - **c1 = E(m1, k1)**
  - **c2 = E(m2, k2)**
  - **c3 = E(m3, k3)**
  - **…**

- **<u>Ejemplos</u>**: Vernam, RC4, etc.

.

Cifradores de flujo

- **EJERCICIO: Cifrador Vernam**
  - **Mensaje = “HOLA”**
  - **Clave = “key”**
  - **Tamaño de símbolo = 1 byte (Codificación ASCII)**
  - **Texto cifrado = ??**

.

- **EJERCICIO: Cifrador Vernam**

  - **Mensaje = “HOLA”**
    - “H” = HEX(48) = 0100 1000
    - “O” = HEX(4F) = 0100 1111
    - “L” = HEX(4C) = 0100 1100
    - “A” = HEX(41) = 0100 0001

  - **Clave = “KEY”**
    - “k” = HEX(6B) = 0110 1011
    - “e” = HEX(65) = 0110 0101
    - “y” = HEX(79) = 0111 1001

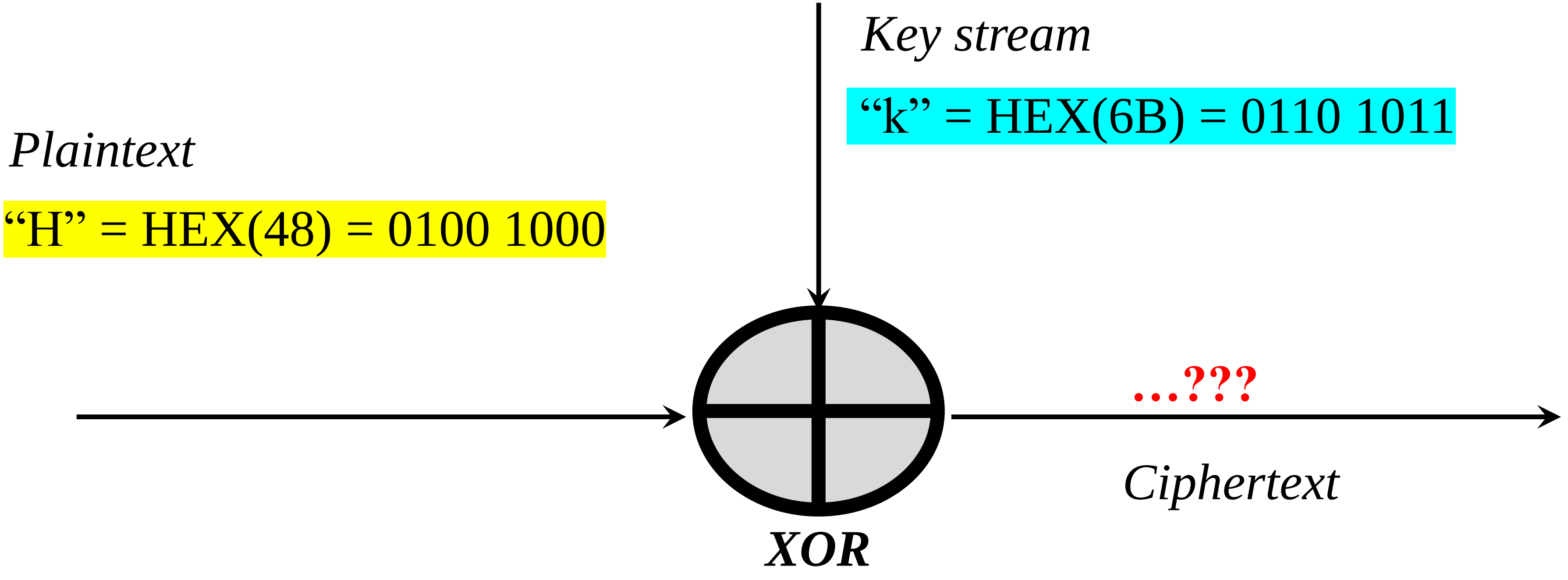
Cifradores de flujo

- **EJERCICIO: Cifrador Vernam**

  - **Mensaje = “HOLA”**
    - “H” = HEX(48) = 0100 1000
    - “O” = HEX(4F) = 0100 1111
    - “L” = HEX(4C) = 0100 1100
    - “A” = HEX(41) = 0100 0001

  - **Clave = “KEY”**
    - “k” = HEX(6B) = 0110 1011
    - “e” = HEX(65) = 0110 0101
    - “y” = HEX(79) = 0111 1001

Cifradores de flujo



"H" = HEX(48) = 0100 1000

"k" = HEX(4B) = 0110 1011 ⊕

= 0010 0011 = HEX(23) = "#"

Cifradores de flujo

- **Generación del keystream**

  – **En la práctica se emplean generadores pseudo-aleatorios**
    - **Semilla inicial (clave compartida, usada sólo una vez)**
    - **Algoritmo <u>determinista</u> (misma entrada genera misma salida)**
      - **Útil para que el transmisor y el receptor se entiendan**

  – **Ejemplo: LFSR**

$$f(x) = C_n\, x^{n} + C_{n.1}\, x^{n\text{-}1} + \ldots + C_2\, x^{2} + C_1\, x + 1$$

$$Seed = S_n\,,\, S_{n.1}\,,\ \ldots\ , S_{2,}\, S_1$$

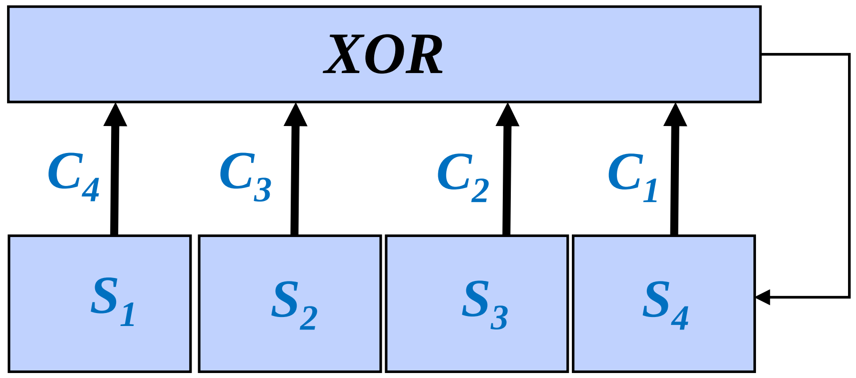
Cifradores de flujo

LFSR *(Linear Feedback Shift Register)*

- ***Expresión general***
  - ***(para orden de polinomio n=4)***

$$f(x) = C_4\, x^4 + C_3\, x^3 + C_2\, x^2 + C_1\, x + 1$$

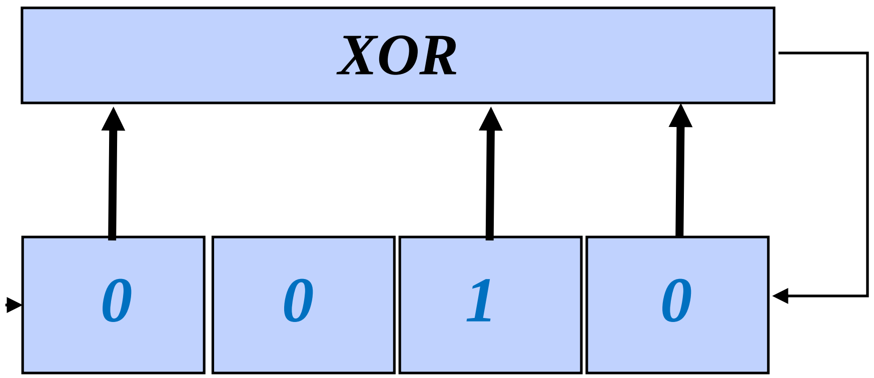
$$Seed = (S1, S2, S3, S4)$$



Ejemplo:

$$f(x) = 1 \cdot x^4 + 0 \cdot x^3 + 1 \cdot x^2 + 1 \cdot x + 1$$

$$Seed = 0\ 0\ 1\ 0$$



Cifradores de flujo

PRESTACIONES

- **VENTAJAS:**
  - Muy rápidos
  - Ideales para prestaciones que requieran tiempo real
    - Ejemplo: streaming de video / audio
  - Los errores de transmisión no se propagan a otros símbolos

- **INCONVENIENTES:**
  - Mala difusión de la información
    - La información de cada símbolo se traslada íntegramente a su símbolo cifrado
  - Seguridad y gestión de la clave
    - (puramente aleatoria, más larga que el texto a cifrar, sólo usada una vez…)

Cifradores de flujo

ECRYPT II

eSTREAM portfolio

Home

Profile 1 (SW)

HC-128
Rabbit
Salsa20/12
SOSEMANUK

Profile 2 (HW)

Grain v1
MICKEY 2.0
Trivium

eSTREAM: the ECRYPT Stream Cipher Project

Welcome to the home page of eSTREAM, the ECRYPT Stream Cipher Project. The eSTREAM project was a multi-year effort, running from 2004 to 2008, to promote the design of efficient and compact stream ciphers suitable for widespread adoption. As a result of the project, a portfolio of new stream ciphers was announced in April 2008. The eSTREAM portfolio was revised in September 2008, and currently contains seven stream ciphers. This website is dedicated to ciphers in this final portfolio. For information on the eSTREAM *project* and selection process, including a timetable of the project and further technical background, please visit the original eSTREAM Project website.

The eSTREAM Portfolio

The short report from April 2008 discussing the initial portfolio (with eight stream ciphers) and the end of the eSTREAM project can be found here. The eSTREAM portfolio was revised in September 2008, following the announcement of cryptanalytic results against one of the original algorithms (see here). The portfolio is periodically revisited, as the algorithms mature: the first review of the eSTREAM portfolio was published in October 2009, and is available here; the second review from January 2012 can be found here.

The eSTREAM portfolio ciphers fall into two profiles. Profile 1 contains stream ciphers more suitable for software applications with high throughput requirements. Profile 2 stream ciphers are particularly suitable for hardware applications with restricted resources such as limited storage, gate count, or power consumption.

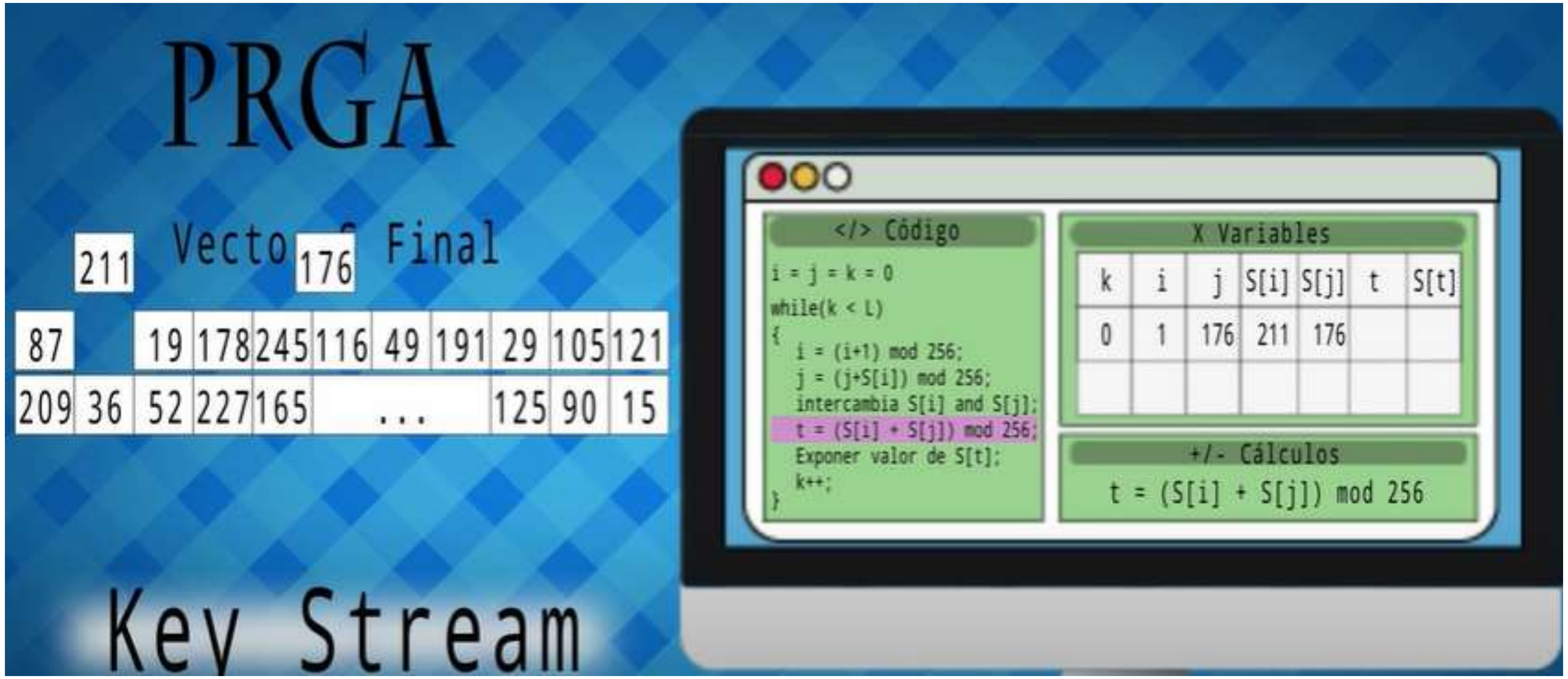
The eSTREAM portfolio contains the following ciphers:

Profile 1 (SW)	Profile 2 (HW)
HC-128	Grain v1
Rabbit	MICKEY 2.0
Salsa20/12	Trivium
SOSEMANUK	

https://www.ecrypt.eu.org/stream/index.html

Cifradores de flujo

- Algoritmo RC4 (no entra en examen!)

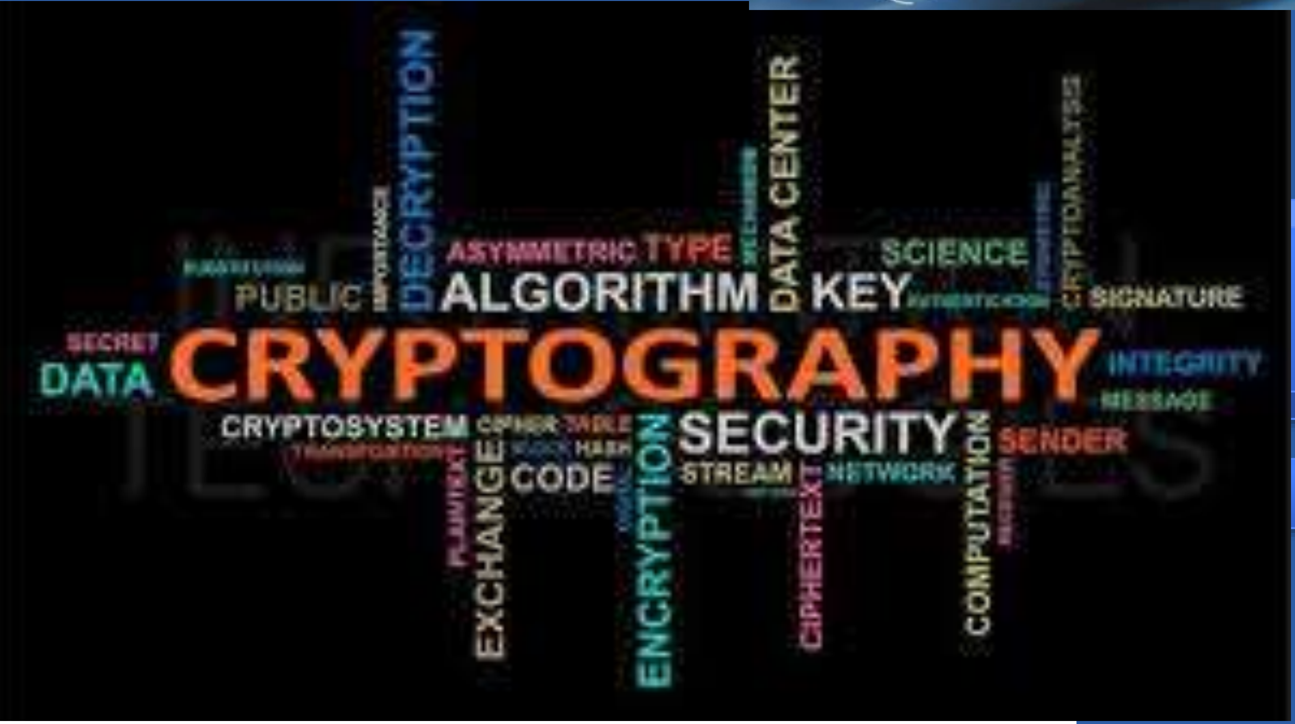


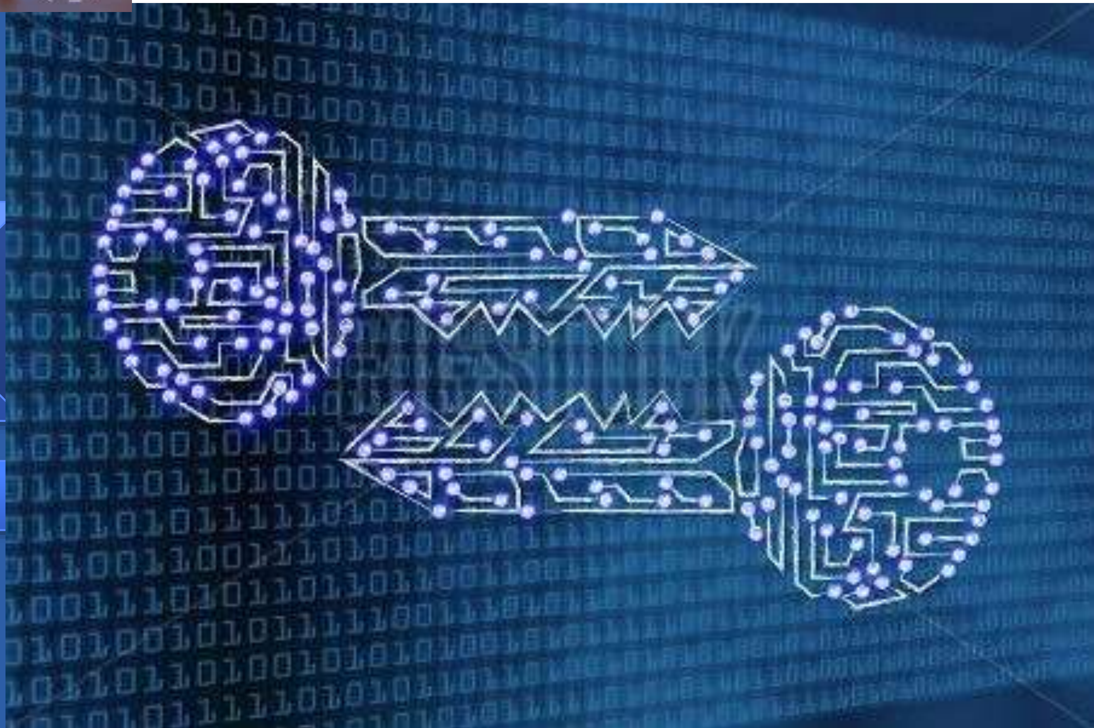
https://www.youtube.com/watch?v=G3HajuqYH2U

Criptografía

Universidad Francisco de Vitoria UFV Madrid

Tema 3.2 Algoritmos de cifrado simétrico de bloque





Cifradores de bloque

- **Dividen el mensaje en bloques de igual tamaño *(*)***
  - **M = M1 M2 M3 M4…**

(*)Habitualmente:
64, 128** o **256 bits

- **Se utiliza la misma clave para el cifrar (y descifrar) todos los bloques**
  - **C1 = E(M1, K) → M1 = D(C1, K)**
  - **C2 = E(M2, K) → M2 = D(C2, K)**
  - **C3 = E(M3, K) → M3 = D(C3, K)**
  - **…**

- **Ejemplos**: DES, AES, etc.

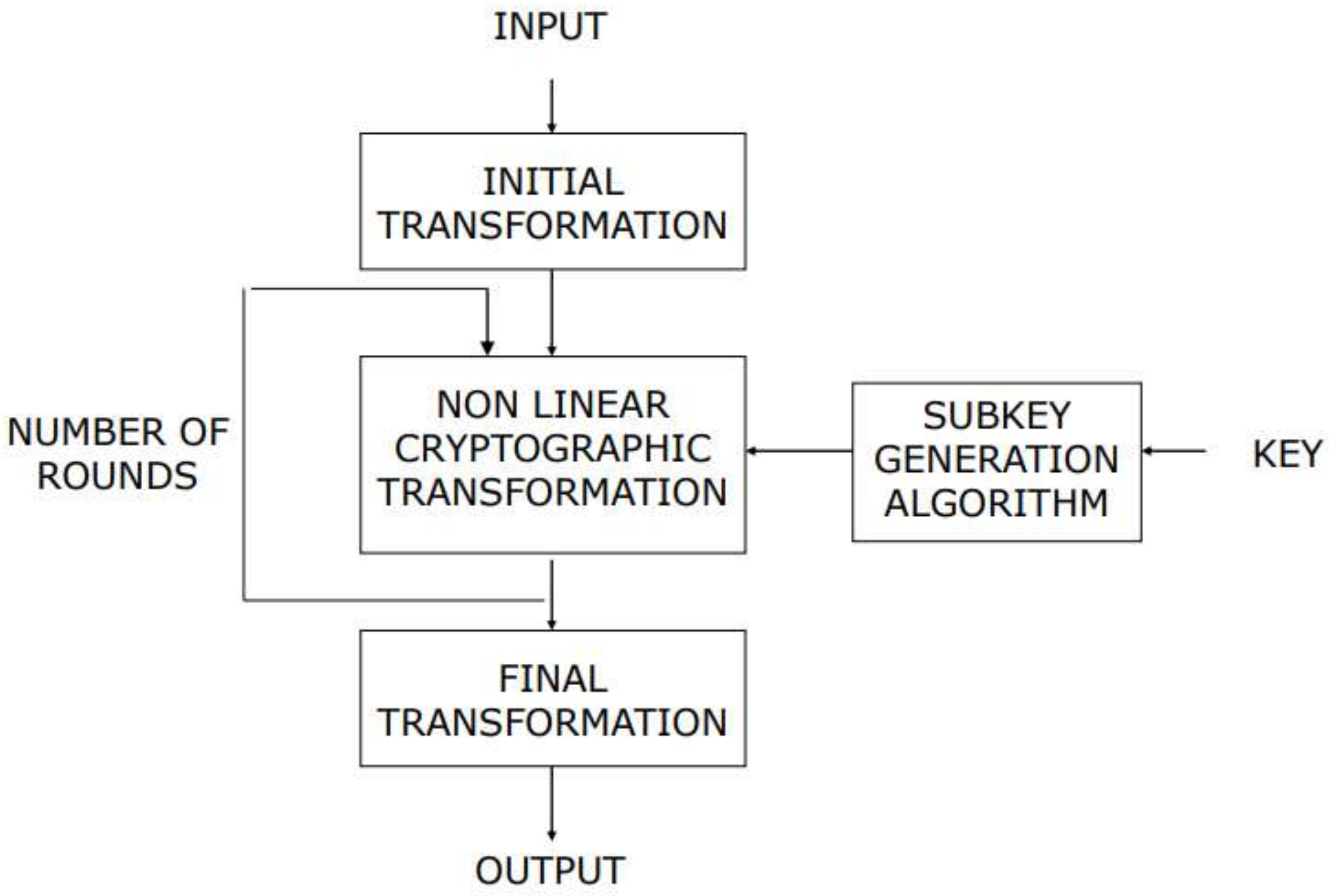
.

Cifradores de bloque

- **Objetivos de seguridad (para dificultar criptoanálisis):**
  - **<u>Difusión</u>**
    - Que la estructura del mensaje quede disipada en la estructura del texto cifrado
      - Dificultar análisis de frecuencias, etc.
      - Cada modificación de 1 bit del mensaje tiene que afectar a muchos bits del texto cifrado, y viceversa.
    - Se consigue mediante funciones de permutación
  - **Confusión**
    - Que la relación entre la clave y el texto cifrado sea tan compleja como sea posible
      - Y no se infiera información de la clave a partir del texto cifrado.
    - Se consigue mediante algoritmos de sustitución

- **Esquema general**

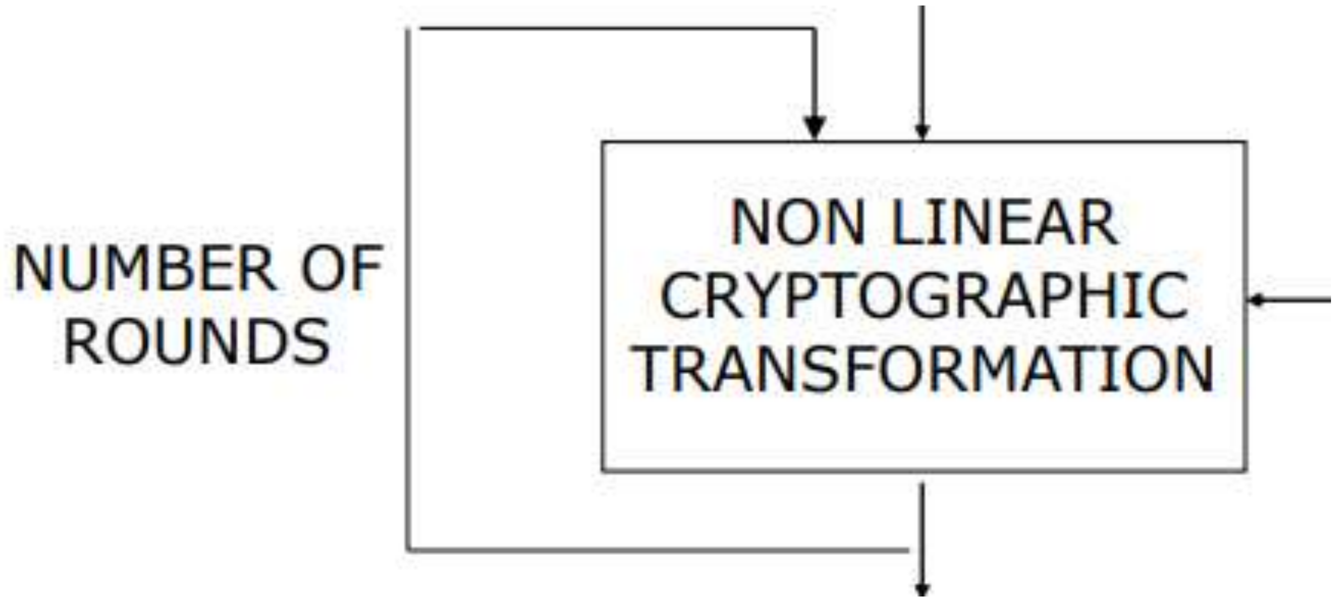
.



Cifradores de bloque

- **Principales esquemas de transformación criptográfica no lineal:**
  - **Feistel** (*ejemplo: DES*)
  - **Esquema de Sustitución – Permutación** (*ejemplo: AES*)

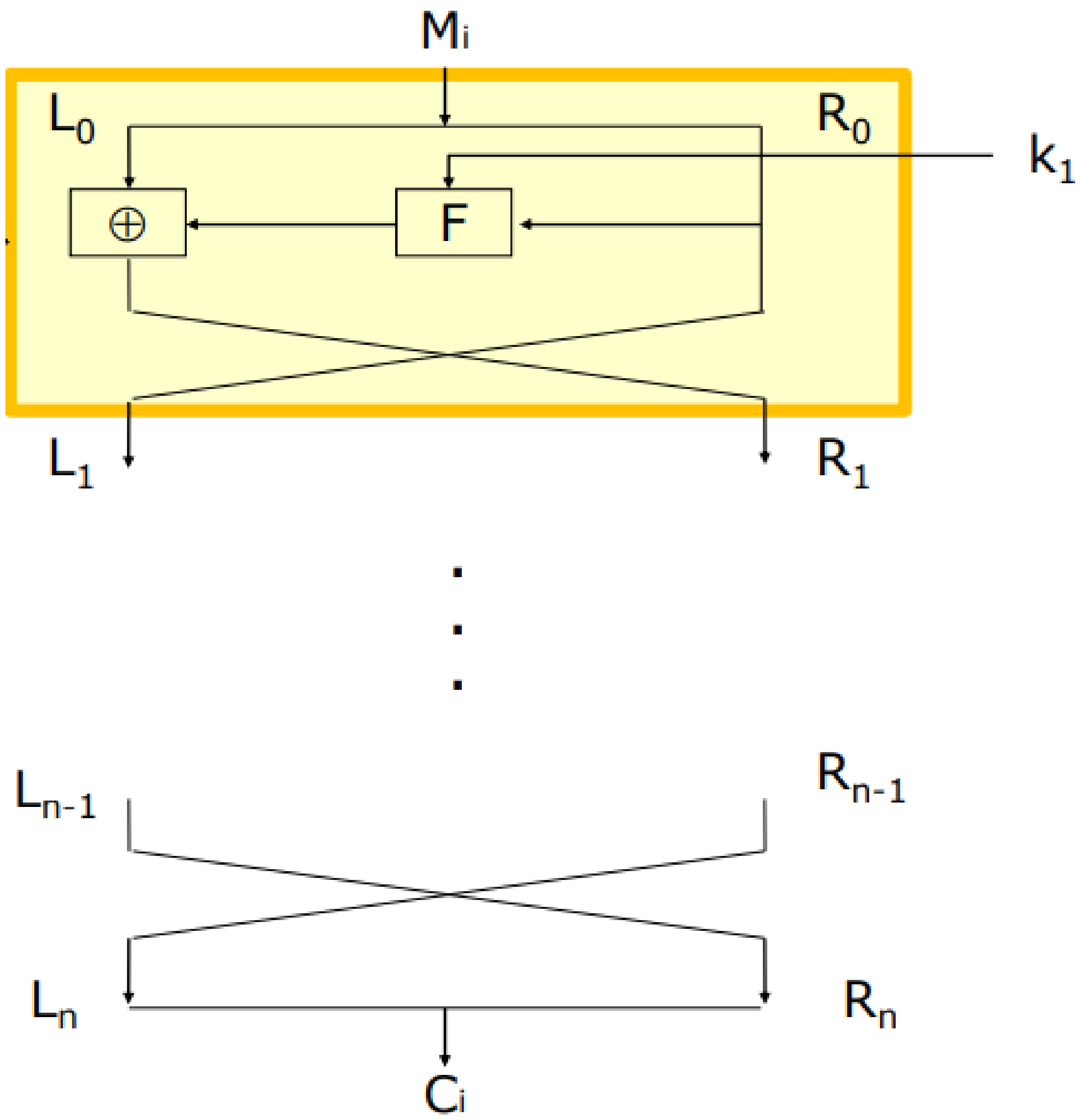
.



Esquema de Feistel

- **Procedimiento del esquema de Feistel:**

1. **Divide** el bloque M en <u>dos mitades</u>, $\mathbf{L_0}$ y $\mathbf{R_0}$
2. **Repite** los siguientes pasos durante **<u>n rondas</u>**. En cada ronda "i":
   1. Aplica una **función F** sobre la <u>mitad derecha</u> ($\mathbf{R_i}$) y la subclave de ronda $\mathbf{k_i}$)
   2. Realiza un **XOR** entre la **salida de F** y la mitad izquierda $\mathbf{L_i}$
   3. **Intercambia** la mitad derecha y la izquierda



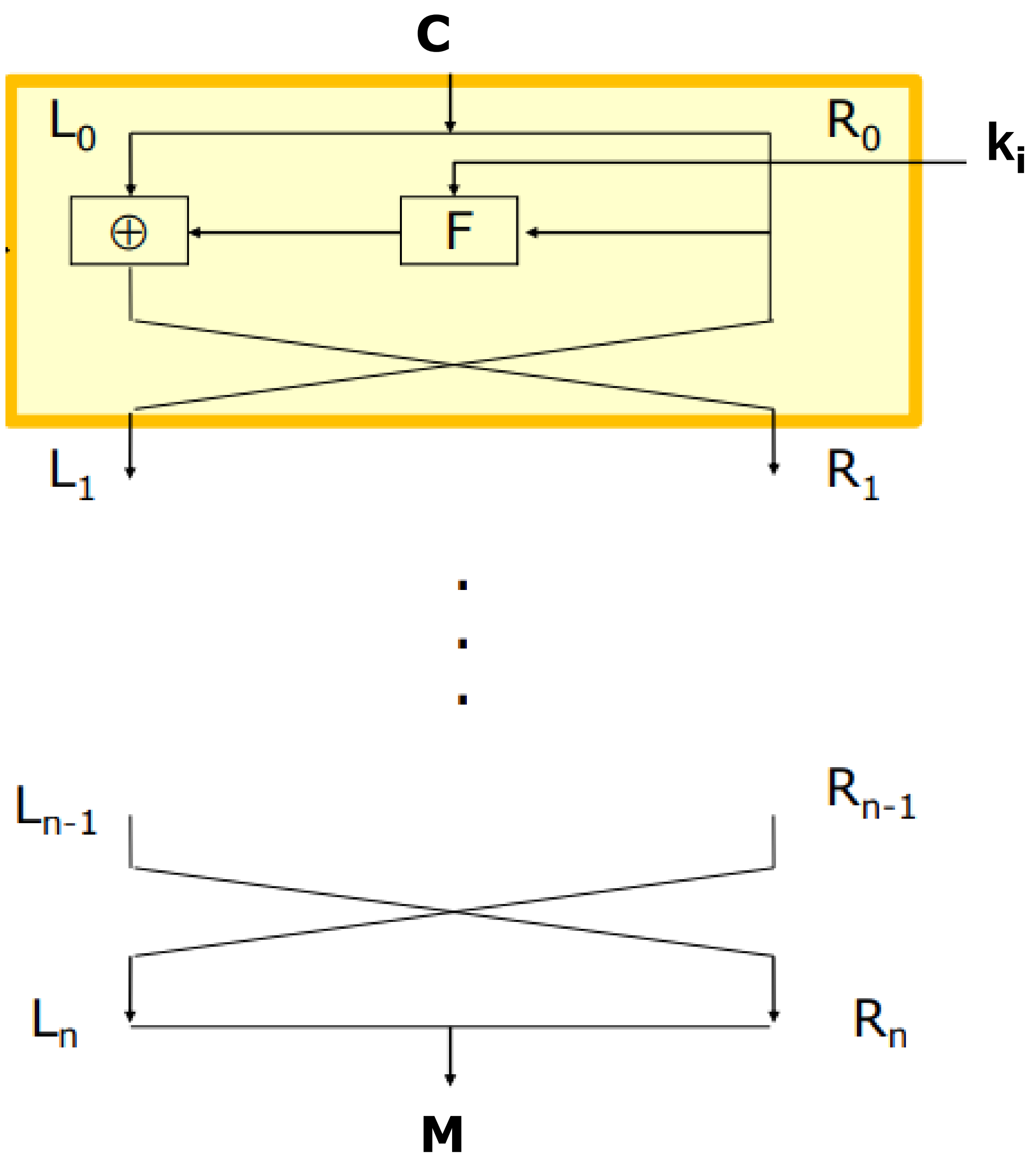
.

Esquema de Feistel

- **Para descifrar:**

1. Se utiliza el mismo circuito que para cifrar
2. Pero las subclaves de ronda se utilizan en el orden inverso
   - ✓ $k_n \rightarrow k_{n-1} \rightarrow \ldots k_1$
3. Ojo con el orden de la **permutación** inicial y final

.



Esquema de Feistel

- **Seguridad en esquema de Feistel basada en:**

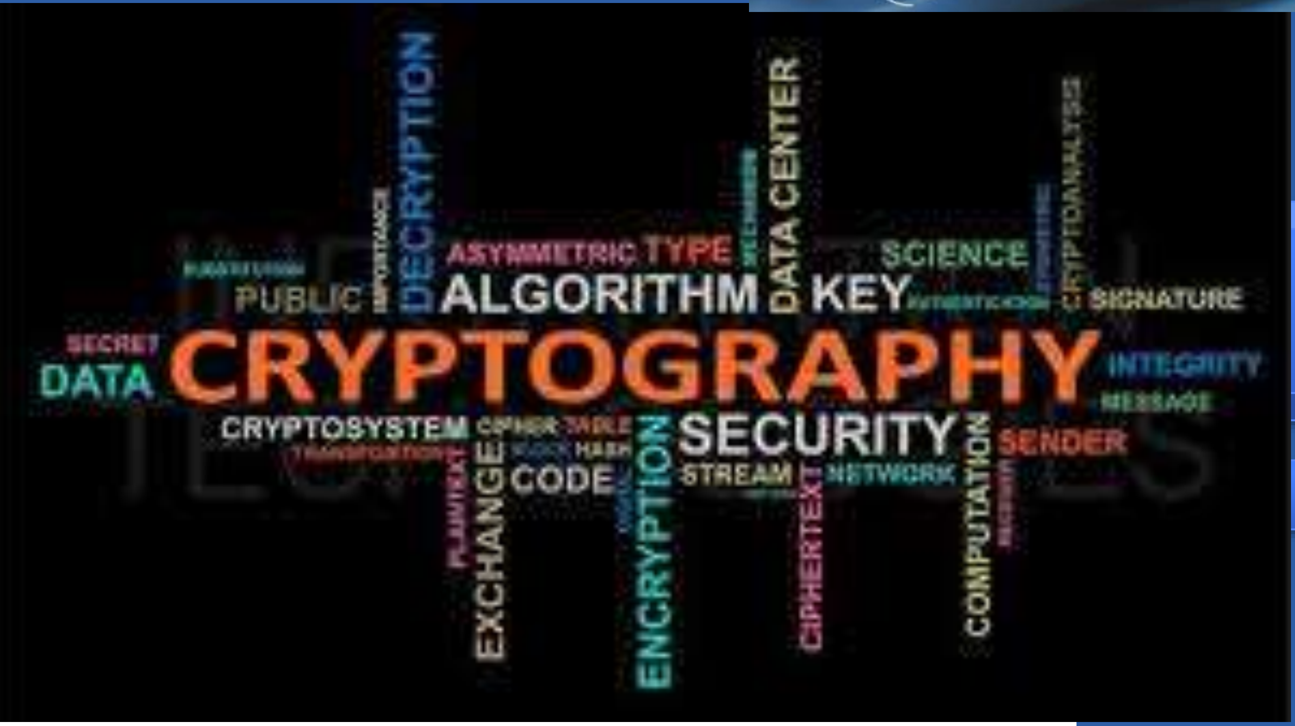
1. Diseño de una buena función F
2. Diseño de una buen algoritmo de generación de subclaves

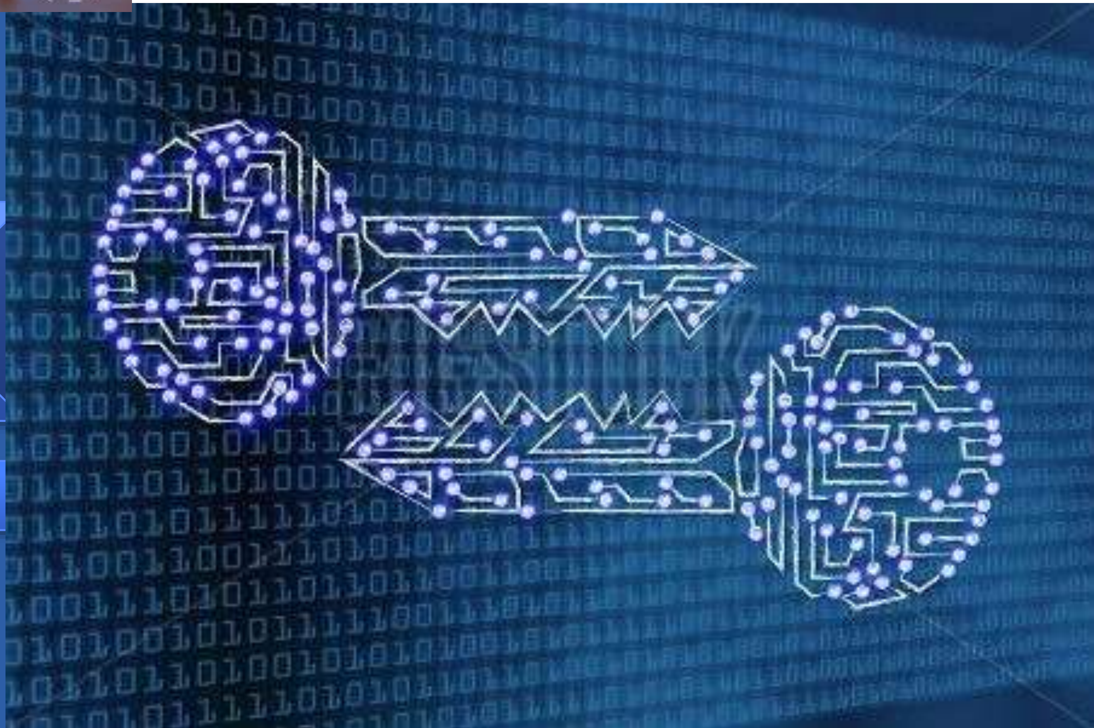
- **Parámetros típicos del esquema de Feistel:**

1. Tamaño de bloque: cuanto más grande, mayor seguridad pero más lento
   - ✓ 64 bits o más.
2. Tamaño de clave: cuanto más grande, mayor seguridad pero más lento
   - ✓ 64 bits o más.
3. Número de rondas: cuanto más grande, mayor seguridad pero más lento
   - ✓ Valor típico: 16 rondas.

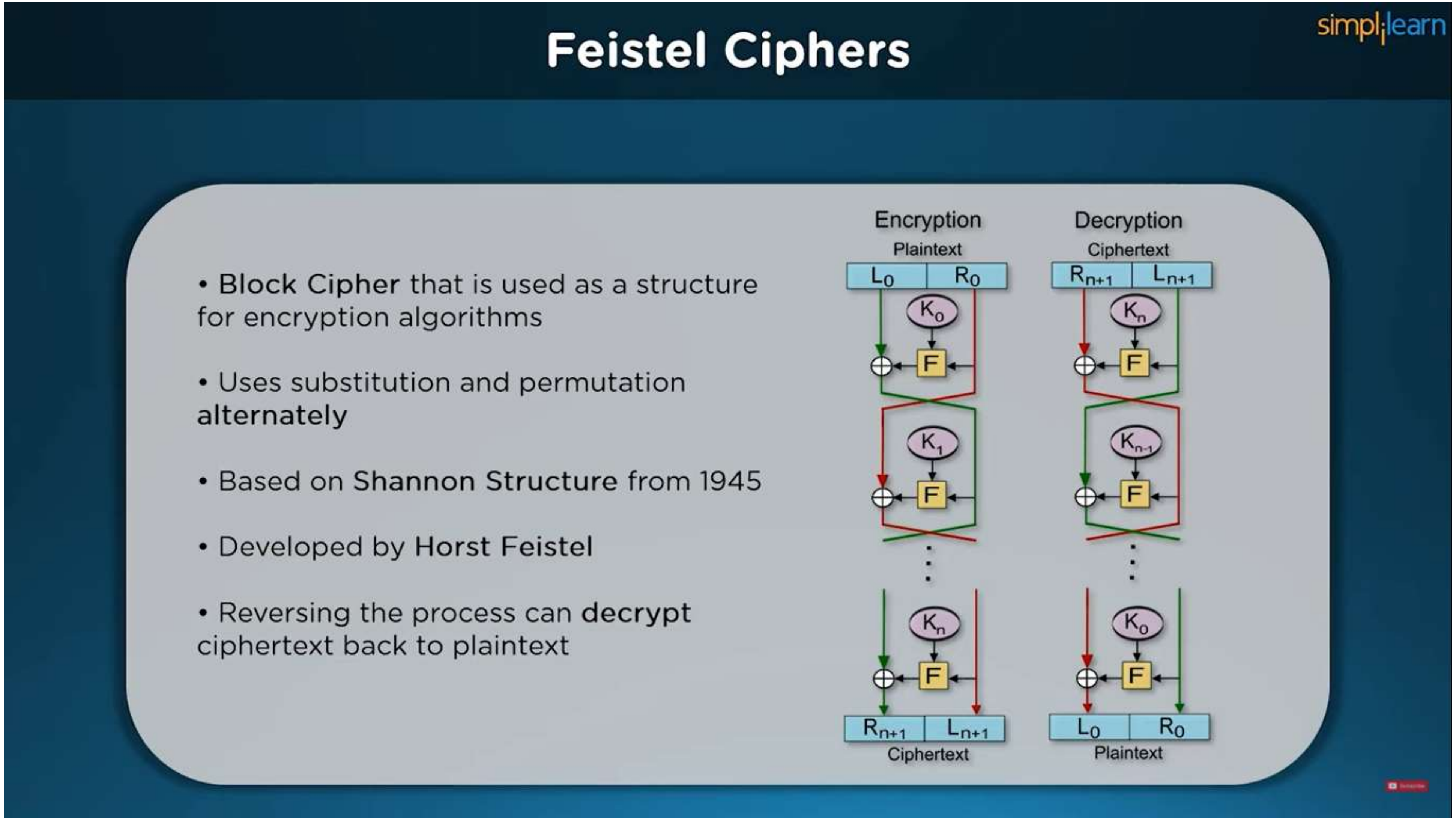
Criptografía

Tema 3.2.1 Algoritmos de cifrado simétrico de bloque: DES





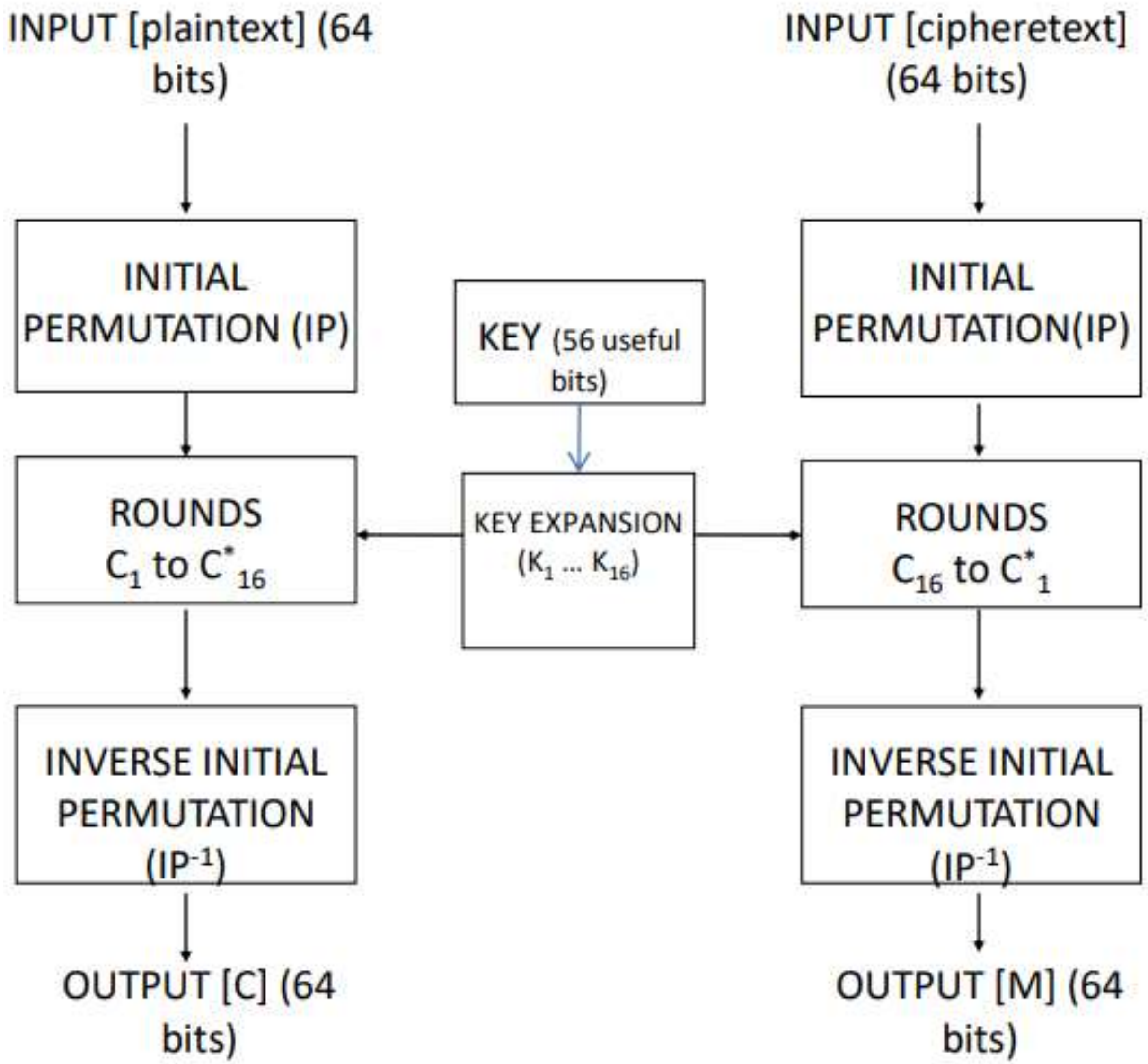
Data Encryption Standard (DES)



https://www.youtube.com/watch?v=S918rR4VdqQ

Data Encryption Standard (DES)

- **Basado en esquema de Feistel:**
  1. Tamaño de **bloque: 64 bits**
  2. Tamaño de **clave**: 64 bits
     - 8 son de paridad
     - Clave inicial de **56** (64-8=56)
  3. Número de **rondas**: **16**
     - **Algoritmo de expansión de claves**
       - 16 **Subclaves** de **48 bits**
  4. Complejidad basada en:
     1. Sustituciones lineales
     2. Sustituciones no lineales
     3. Permutaciones



Data Encryption Standard (DES)

- **Especificación técnica**

Federal Information
Processing Standards Publication 46-3

1999 October 25

Announcing the

DATA ENCRYPTION STANDARD

Federal Information Processing Standards Publications (FIPS PUBS) are issued by the National Institute of Standards and Technology after approval by the Secretary of Commerce pursuant to Section 5131 of the Information Technology Management Reform Act of 1996 (Public Law 104-106), and the Computer Security Act of 1987 (Public Law 100-235).

1. **Name of Standard.** Data Encryption Standard (DES).

2. **Category of Standard.** Computer Security, Cryptography.

3. **Explanation.** The Data Encryption Standard (DES) specifies two FIPS approved cryptographic algorithms as required by FIPS 140-1. When used in conjunction with American National Standards Institute (ANSI) X9.52 standard, this publication provides a complete description of the mathematical algorithms for encrypting (enciphering) and decrypting (deciphering) binary coded information. Encrypting data converts it to an unintelligible form called cipher. Decrypting cipher converts the data back to its original form called plaintext. The algorithms described in this standard specifies both enciphering and deciphering operations which are based on a binary number called a key.

FIPS PUB 46-3

FEDERAL INFORMATION
PROCESSING STANDARDS PUBLICATION

Reaffirmed
1999 October 25

U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and Technology

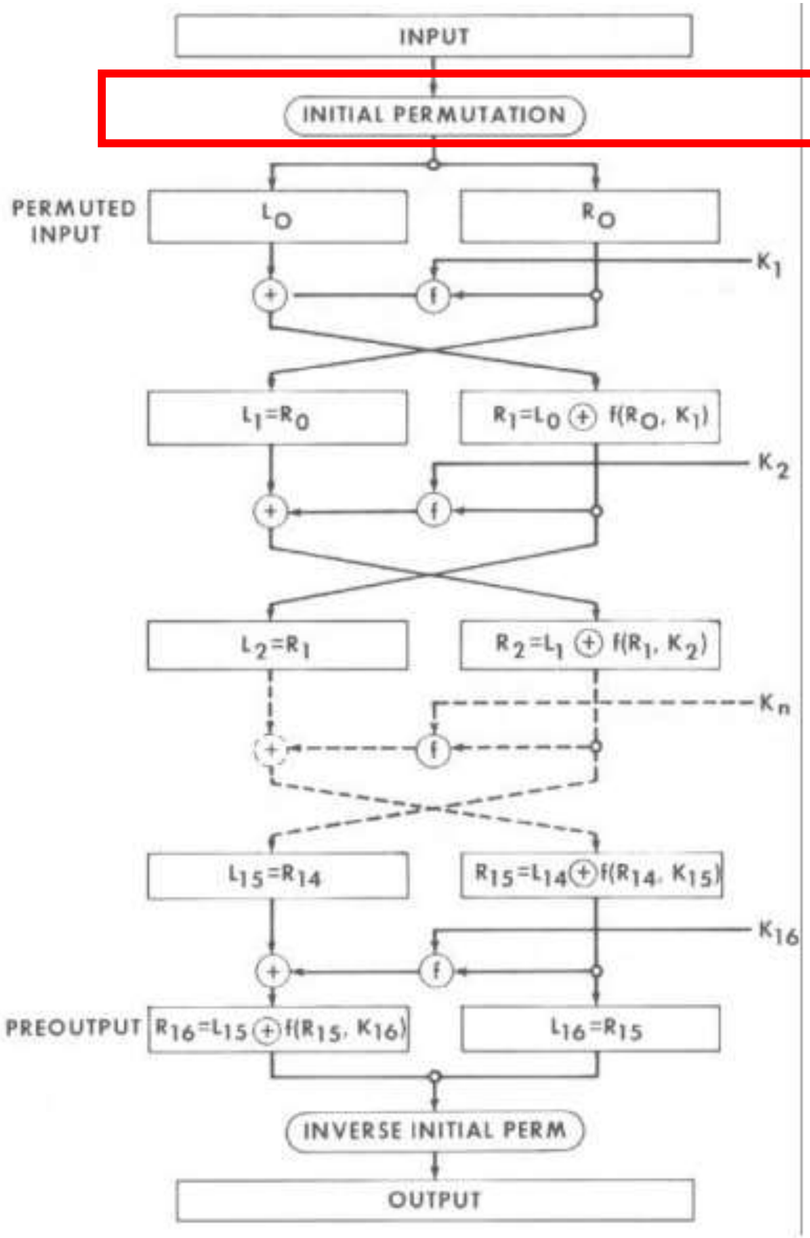
DATA ENCRYPTION STANDARD (DES)

CATEGORY: COMPUTER SECURITY
SUBCATEGORY: CRYPTOGRAPHY

Data Encryption Standard (DES)

- **Permutación Inicial**

.

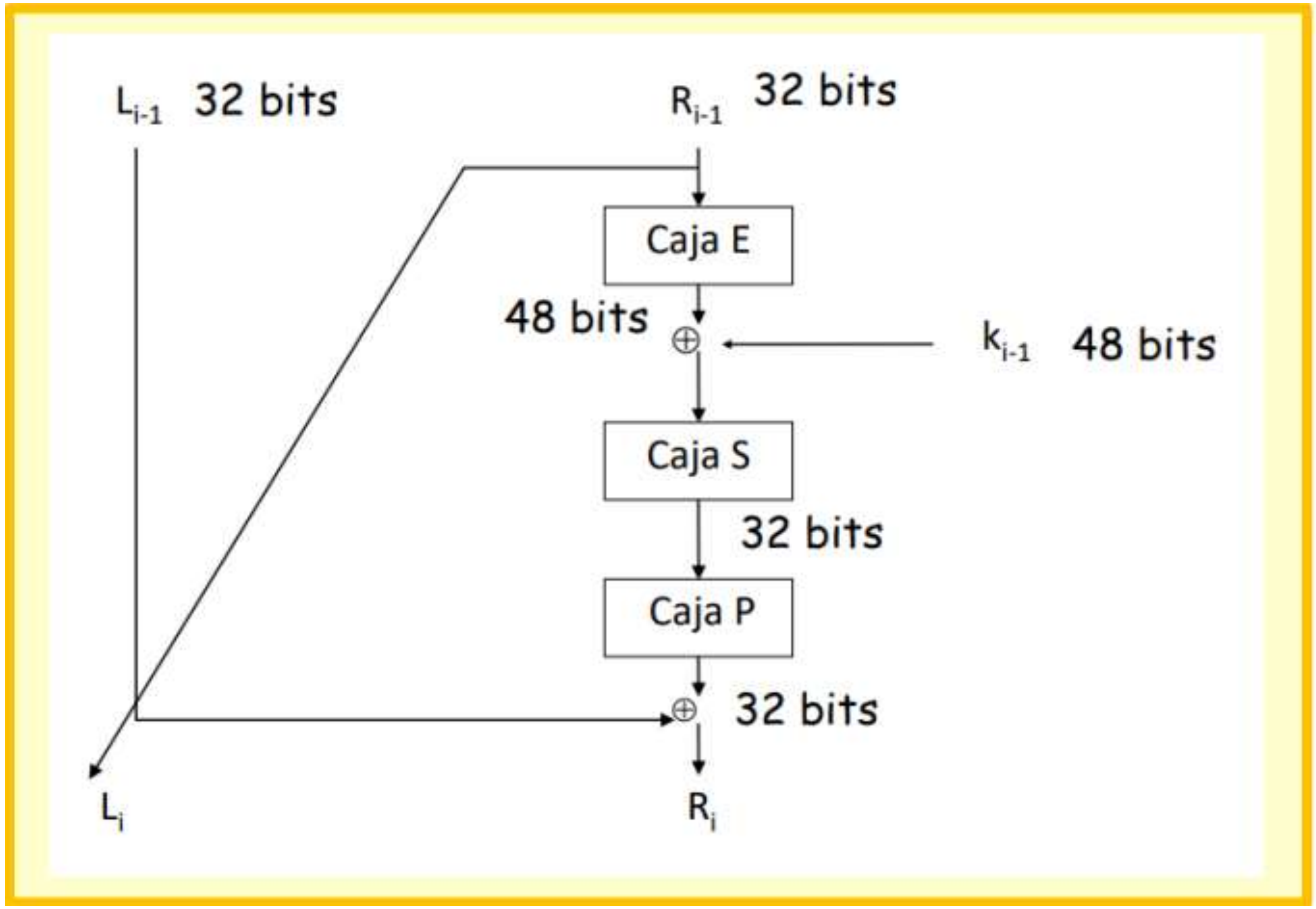


IP

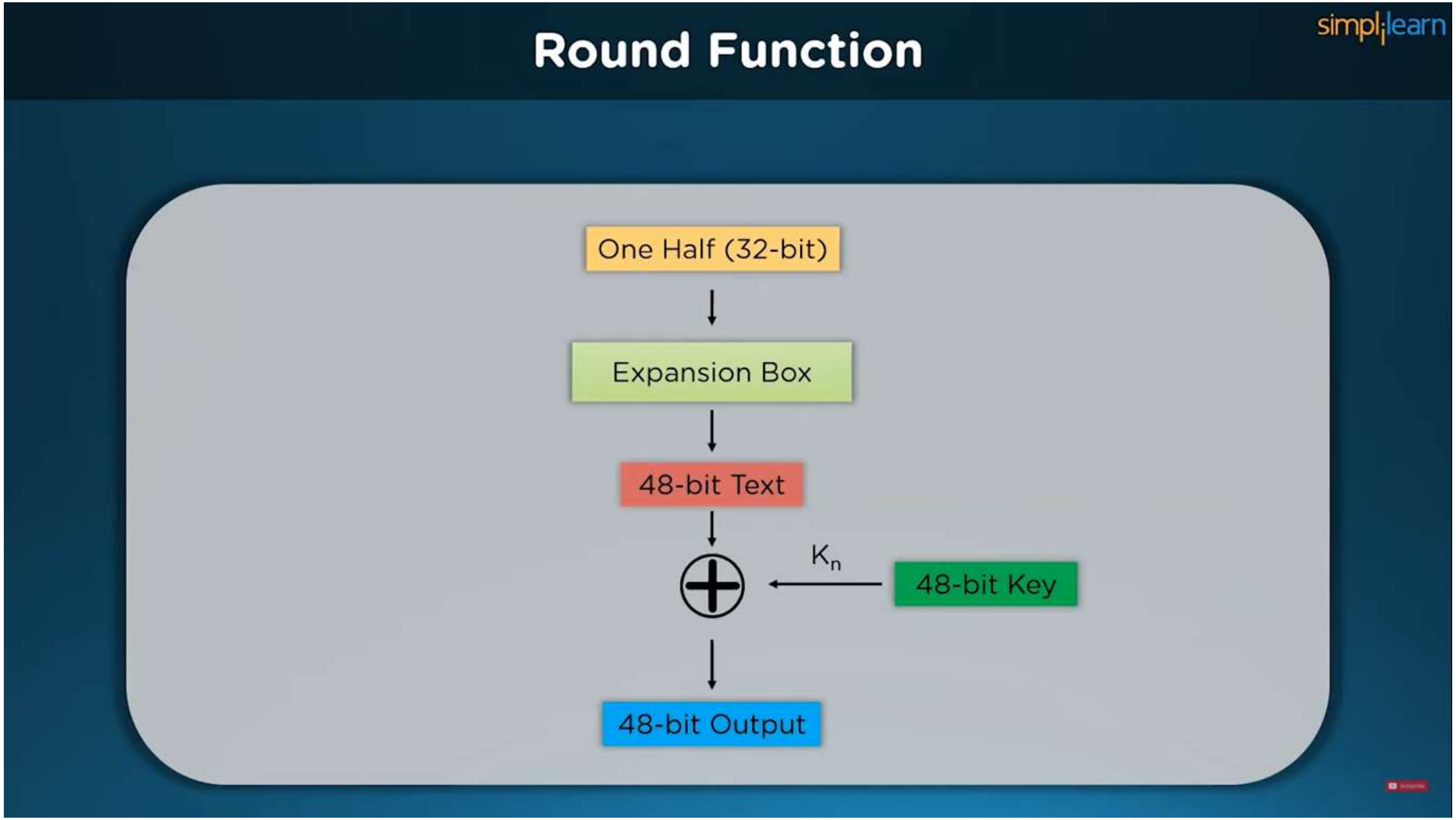
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Data Encryption Standard (DES)

- **Basado en esquema de Feistel:**
  1. Tamaño de **bloque: 64 bits**
  2. Tamaño de **clave**: 64 bits
     - 8 son de paridad
     - Clave inicial de **56** (64-8=56)
  3. Número de **rondas**: **16**
     - **Algoritmo de expansión de claves**
       - 16 **Subclaves** de **48 bits**



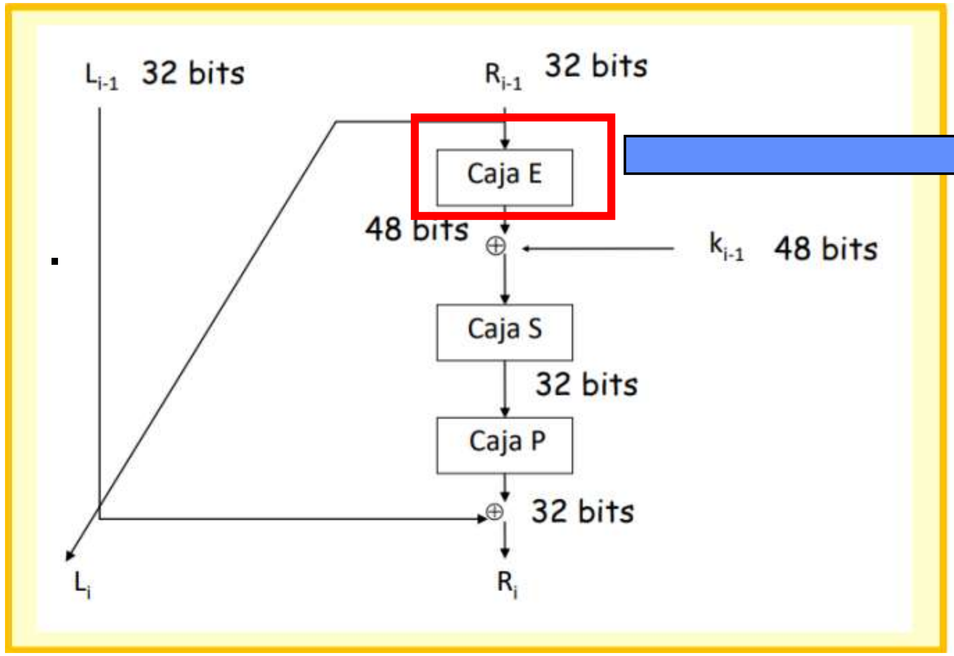
Data Encryption Standard (DES)



https://www.youtube.com/watch?v=S918rR4VdqQ

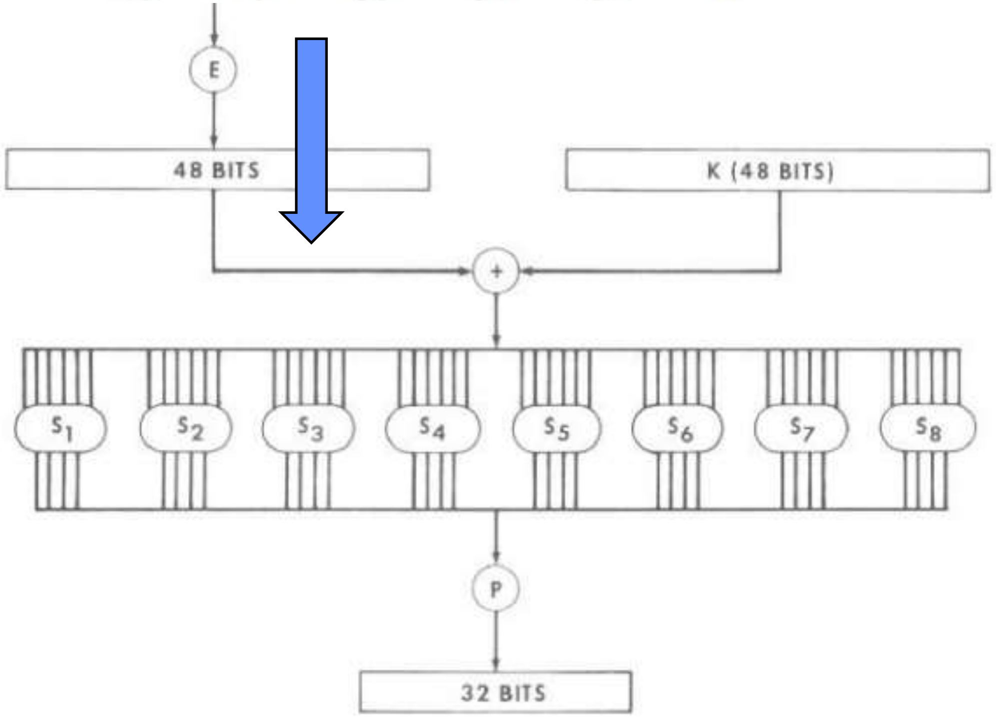
Data Encryption Standard (DES)

- **Función de ronda. Expansión.**

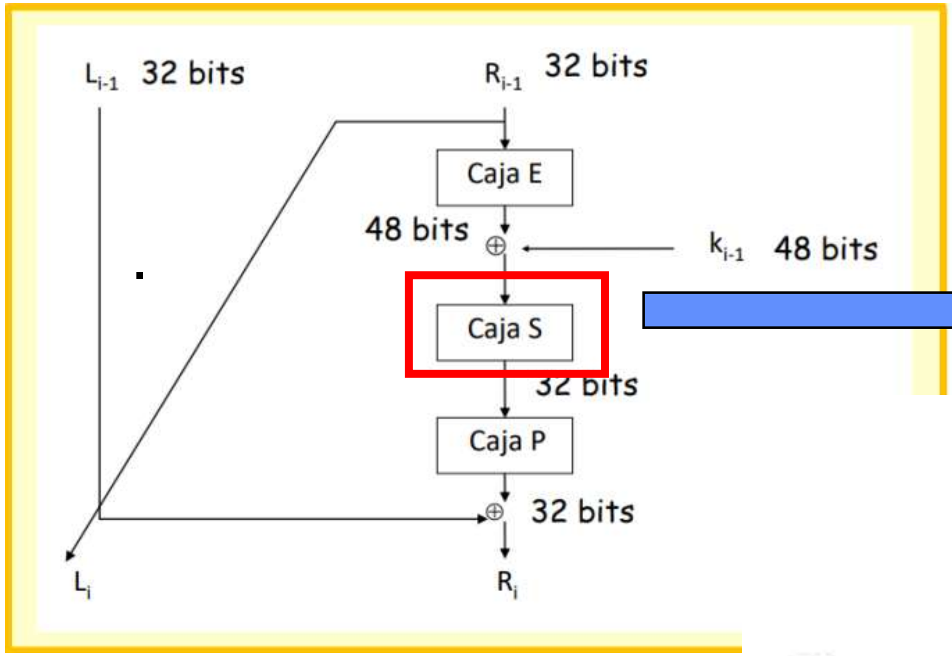


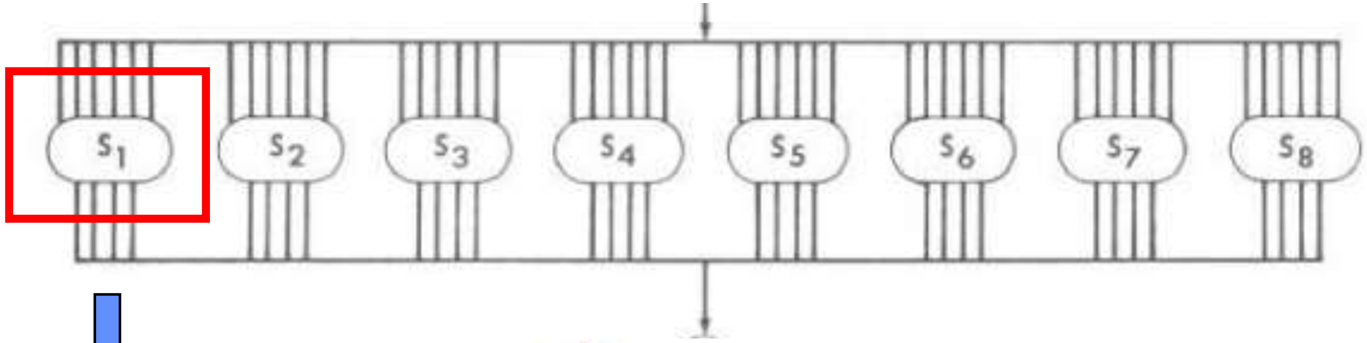
E BIT-SELECTION TABLE

32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	1



- **Función de ronda. Sustitución.**



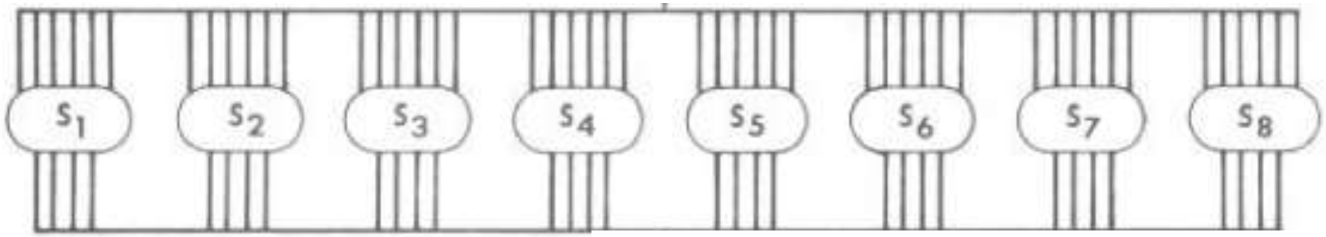


S_1

Column Number

Row No.	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
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Data Encryption Standard (DES)



S_1

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

S_2

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

S_3

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

S_4

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

S_5

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

S_6

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

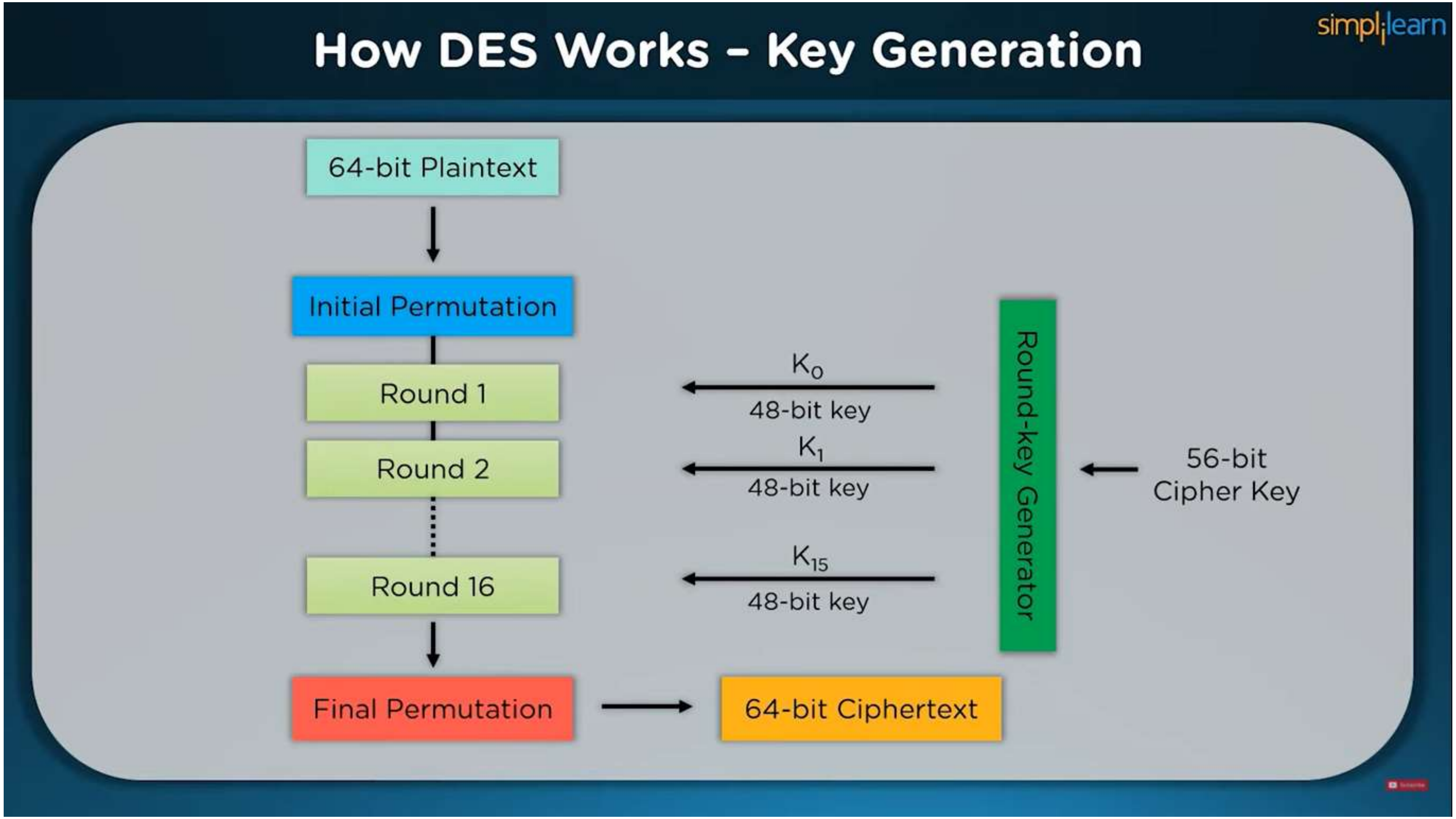
S_7

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

S_8

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

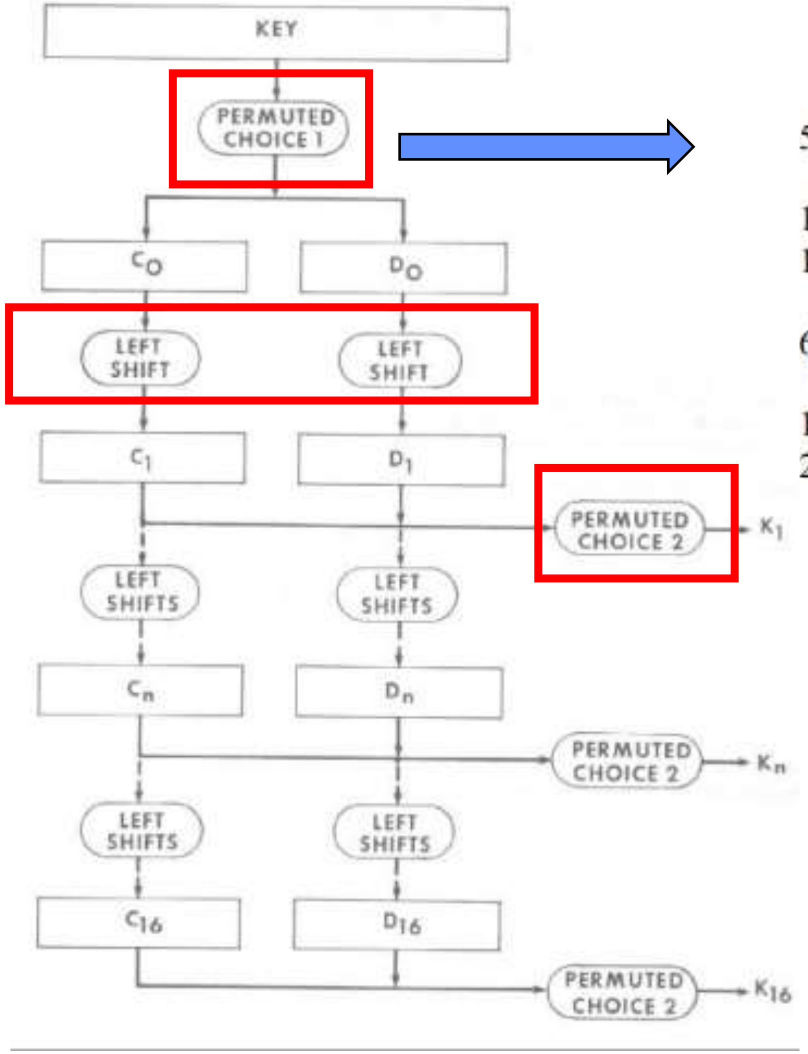
Data Encryption Standard (DES)



https://www.youtube.com/watch?v=S918rR4VdqQ

Data Encryption Standard (DES)

- **Función de ronda. Expansión de Claves.**



PC-1

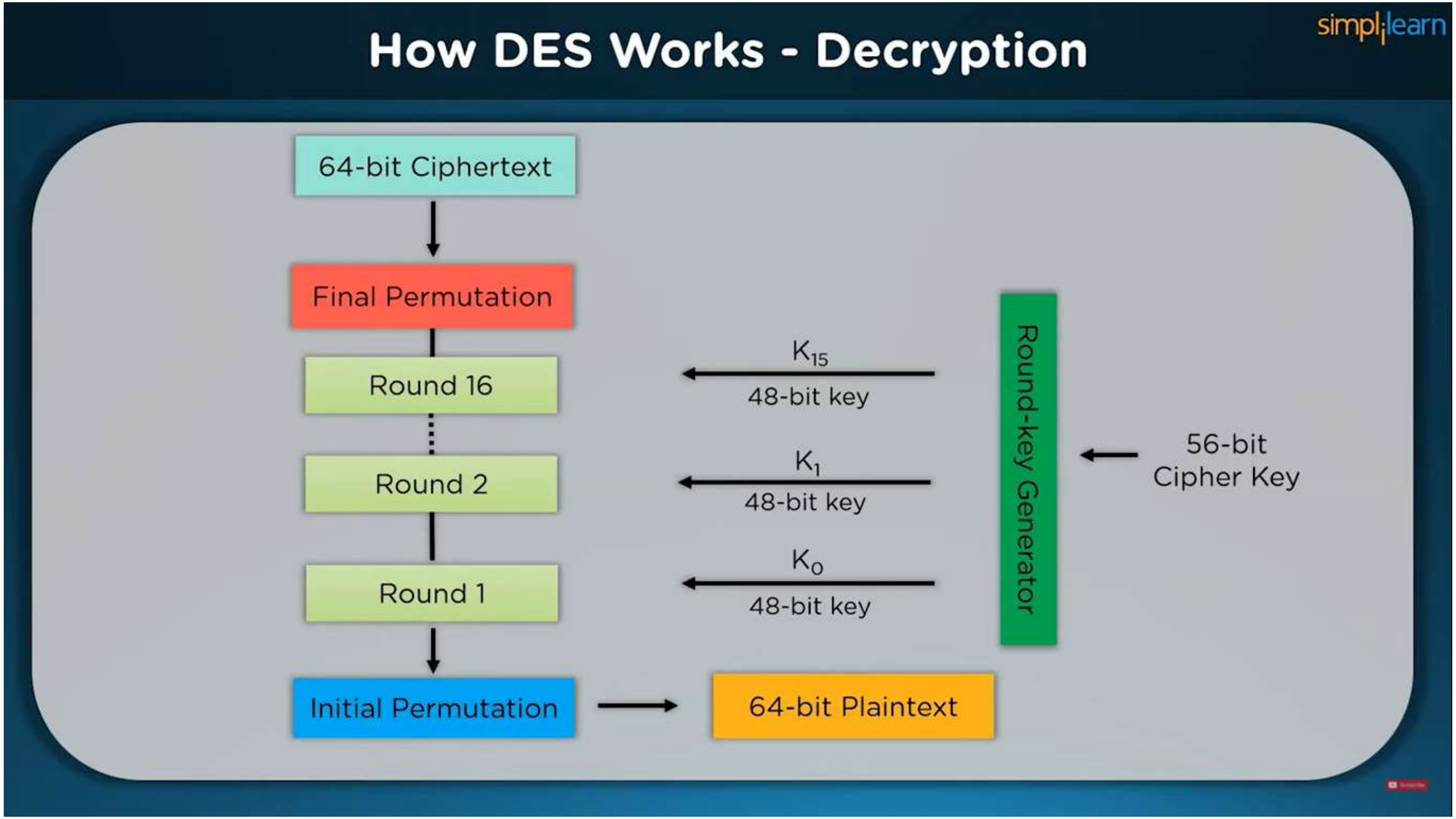
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

PC-2

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Iteration Number	Number of Left Shifts
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	1
10	2
11	2
12	2
13	2
14	2
15	2
16	1

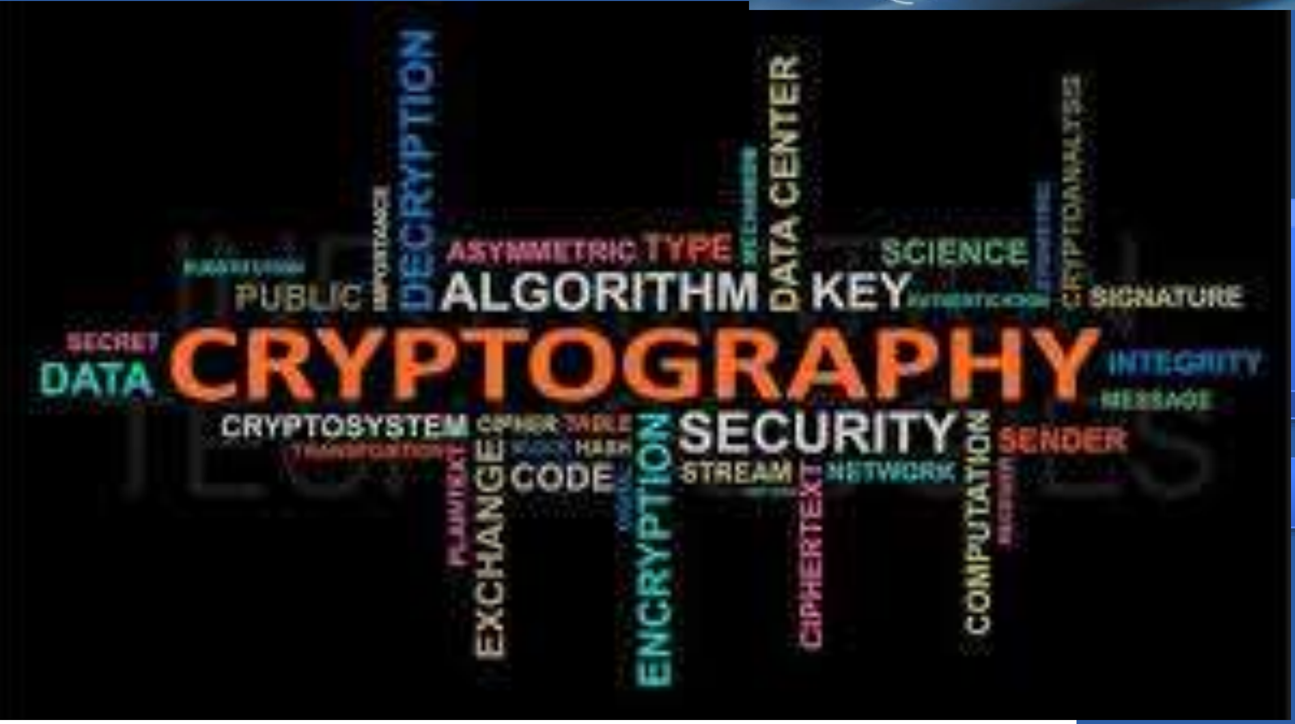
Data Encryption Standard (DES)

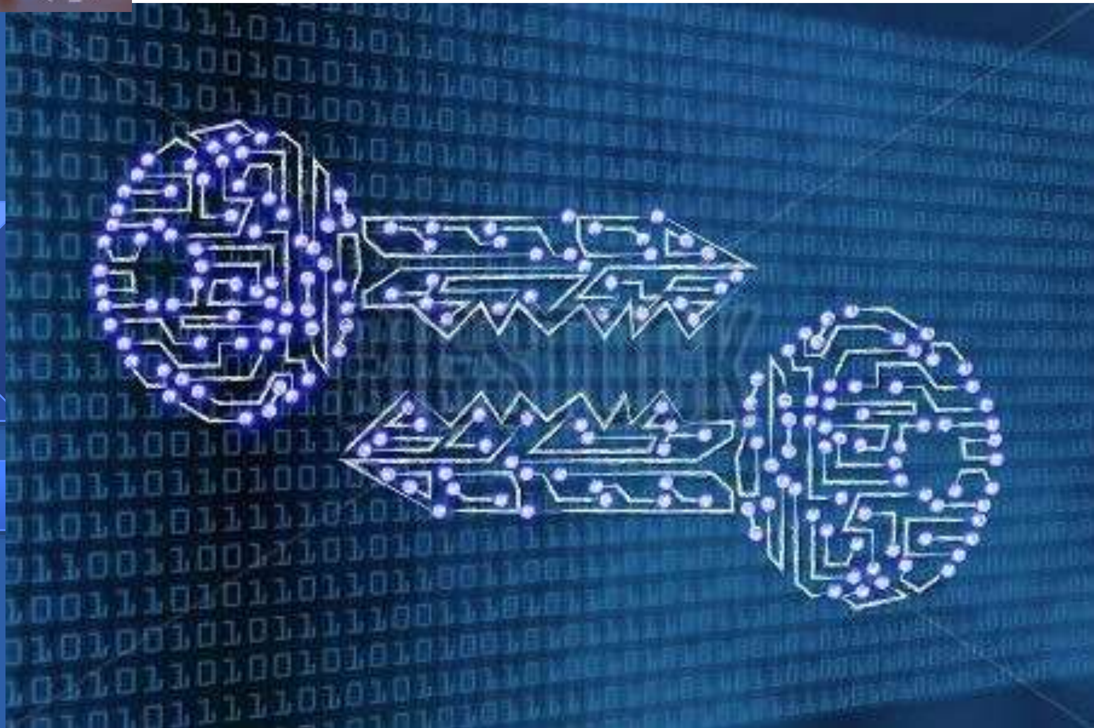


https://www.youtube.com/watch?v=S918rR4VdqQ

Criptografía

Tema 3.2.2 Algoritmos de cifrado simétrico de bloque: AES





Advanced Encryption Standard (AES)

- **Especificación técnica**

.

Federal Information

Processing Standards Publication 197

November 26, 2001

Announcing the

ADVANCED ENCRYPTION STANDARD (AES)

Federal Information Processing Standards Publications (FIPS PUBS) are issued by the National Institute of Standards and Technology (NIST) after approval by the Secretary of Commerce pursuant to Section 5131 of the Information Technology Management Reform Act of 1996 (Public Law 104-106) and the Computer Security Act of 1987 (Public Law 100-235).

1. **Name of Standard.** Advanced Encryption Standard (AES) (FIPS PUB 197).

2. **Category of Standard.** Computer Security Standard, Cryptography.

3. **Explanation.** The Advanced Encryption Standard (AES) specifies a FIPS-approved cryptographic algorithm that can be used to protect electronic data. The AES algorithm is a symmetric block cipher that can encrypt (encipher) and decrypt (decipher) information. Encryption converts data to an unintelligible form called ciphertext; decrypting the ciphertext converts the data back into its original form, called plaintext.

The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits.

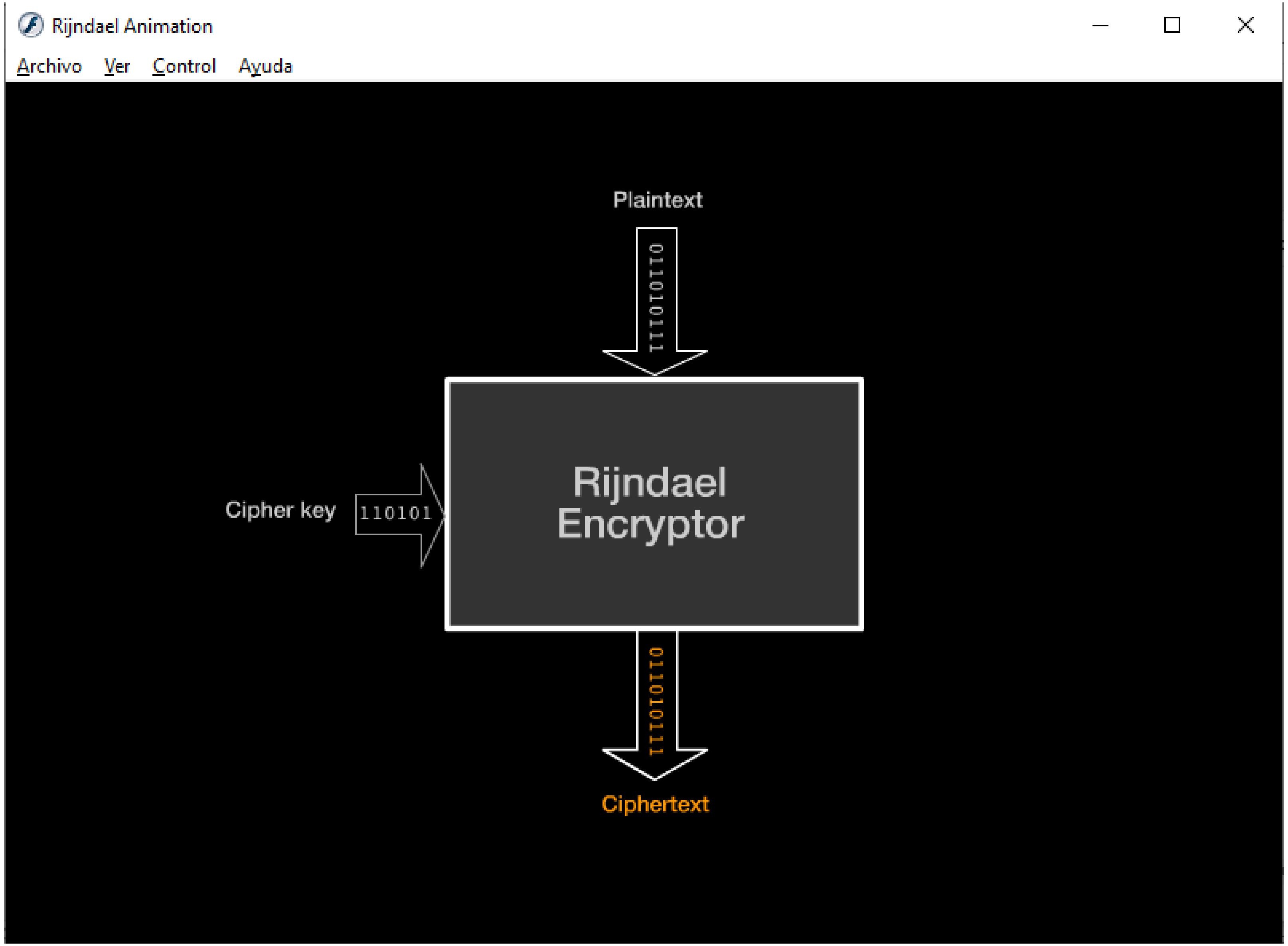
4. **Approving Authority.** Secretary of Commerce.

5. **Maintenance Agency.** Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory (ITL).

Advanced Encryption Standard (AES)

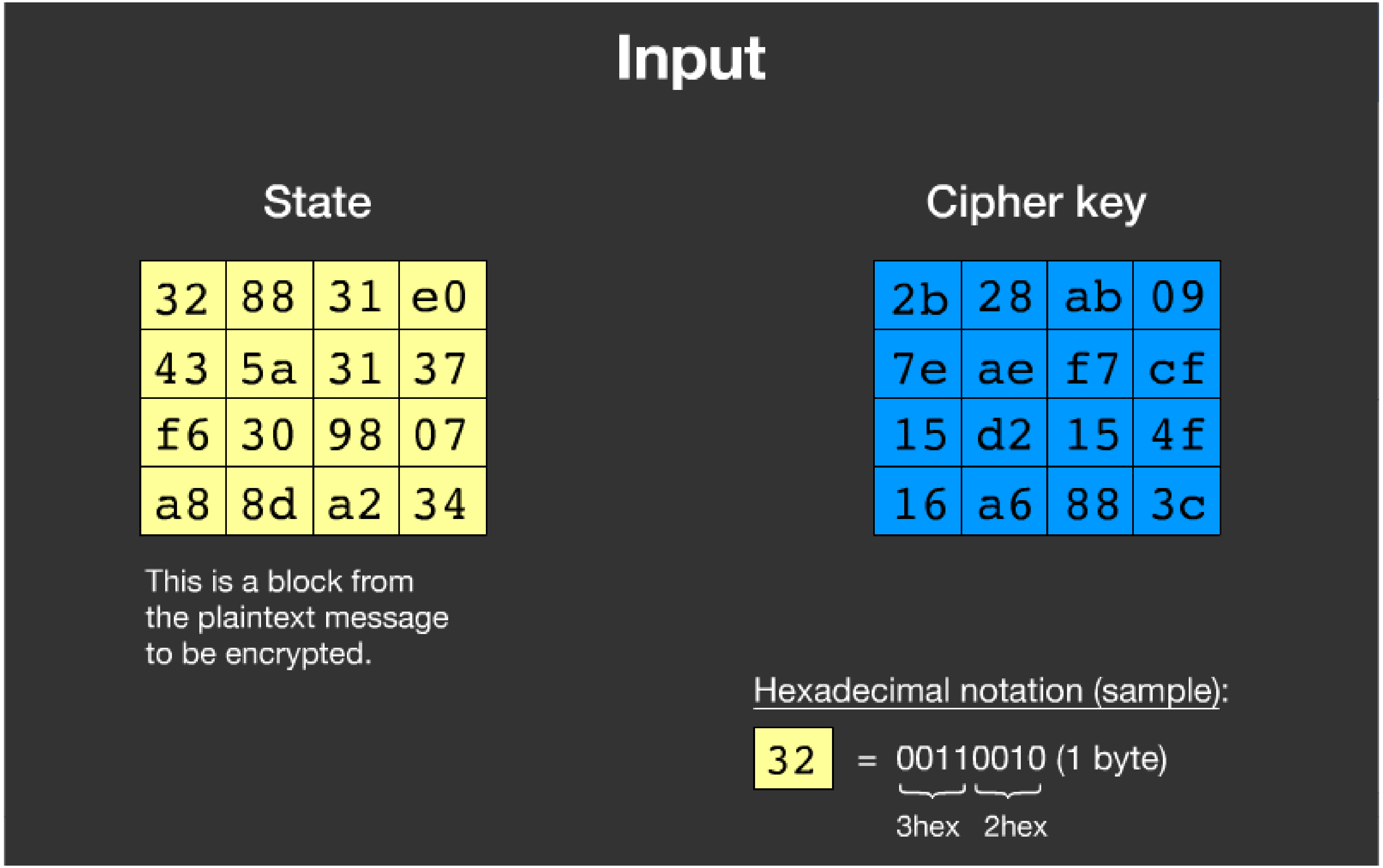
- **Animation**

.



Advanced Encryption Standard (AES)

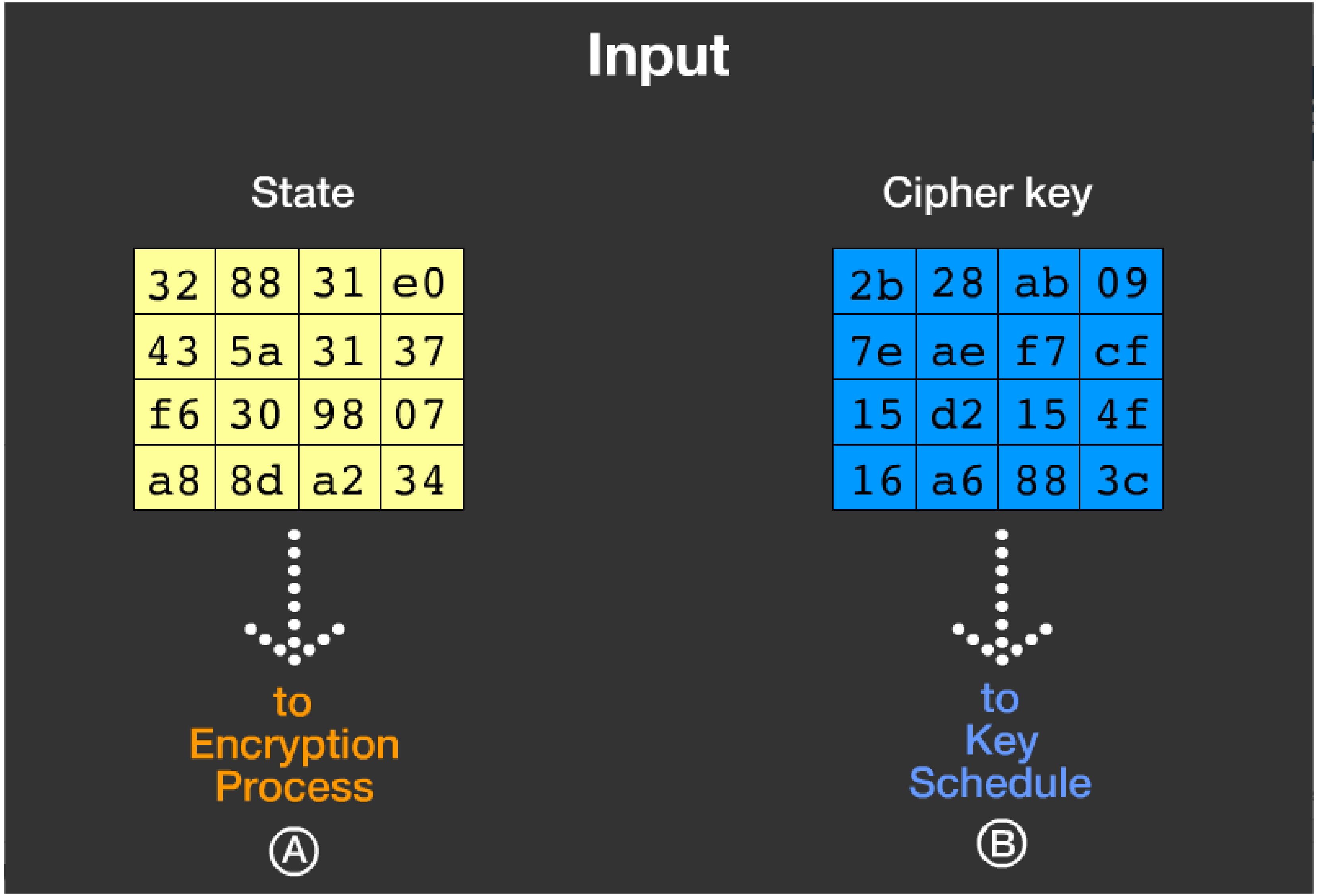
- **Animation**



Advanced Encryption Standard (AES)

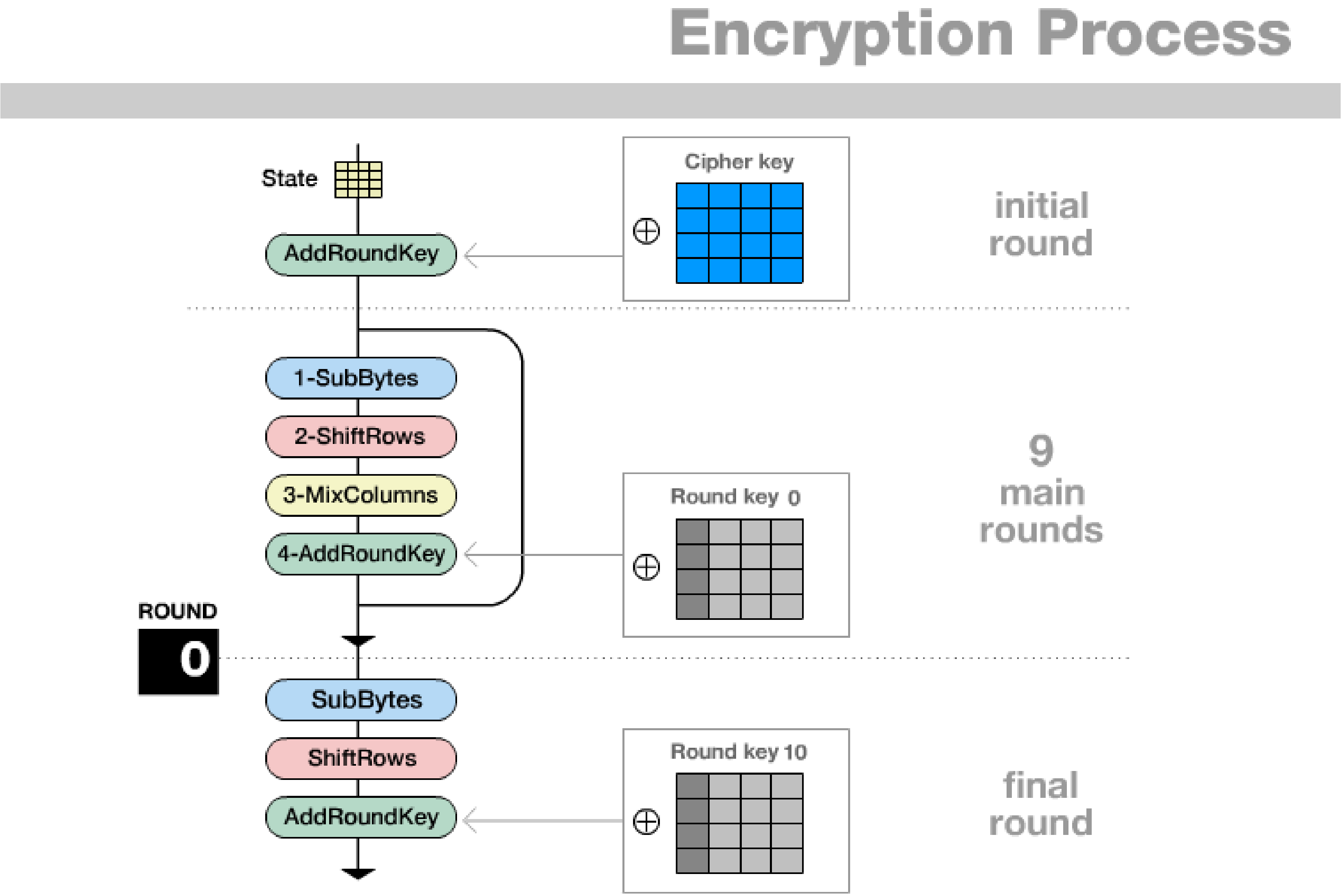
- **Animation**

.



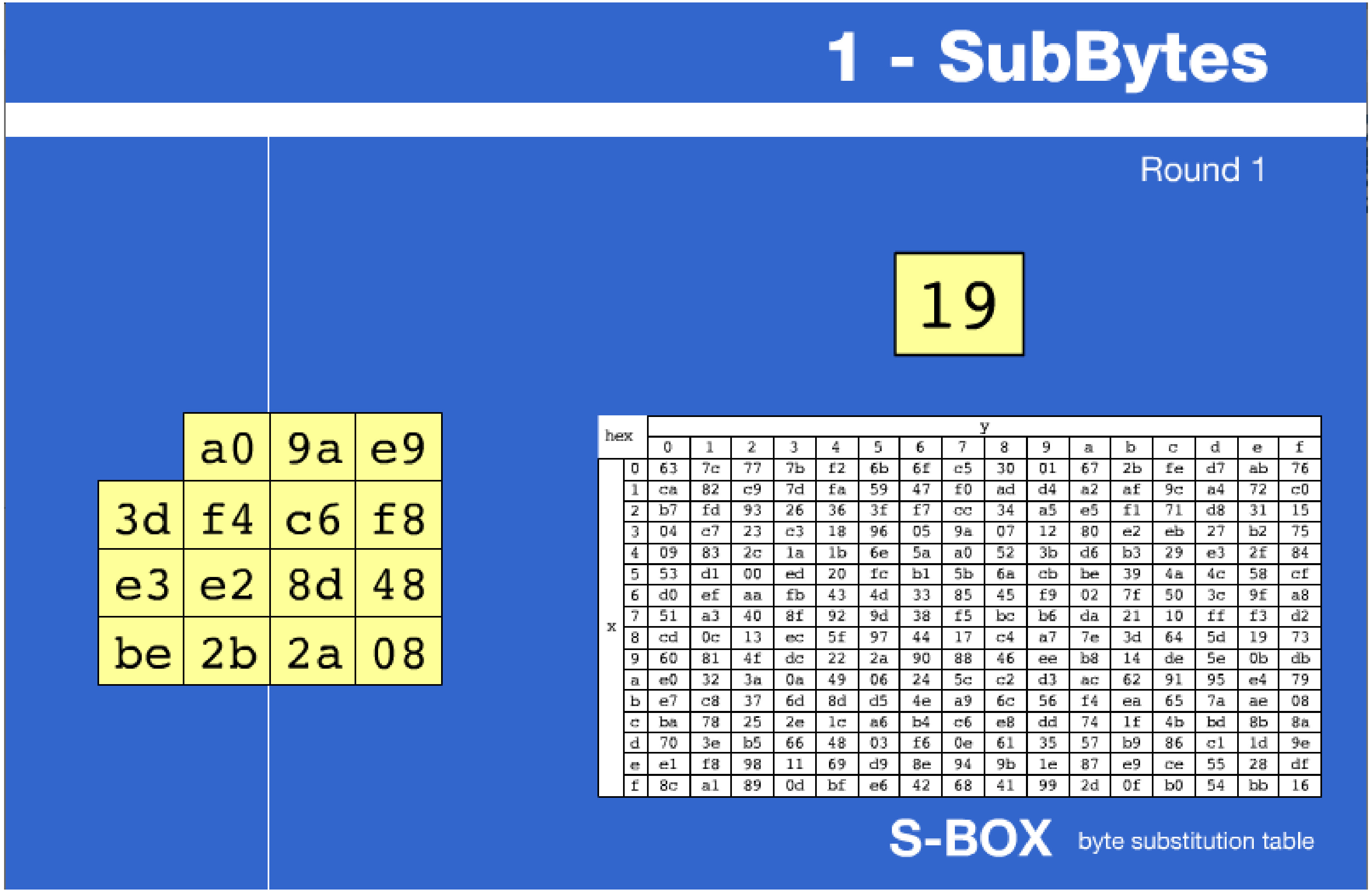
Advanced Encryption Standard (AES)

- **Animation**



- **Animation**

.

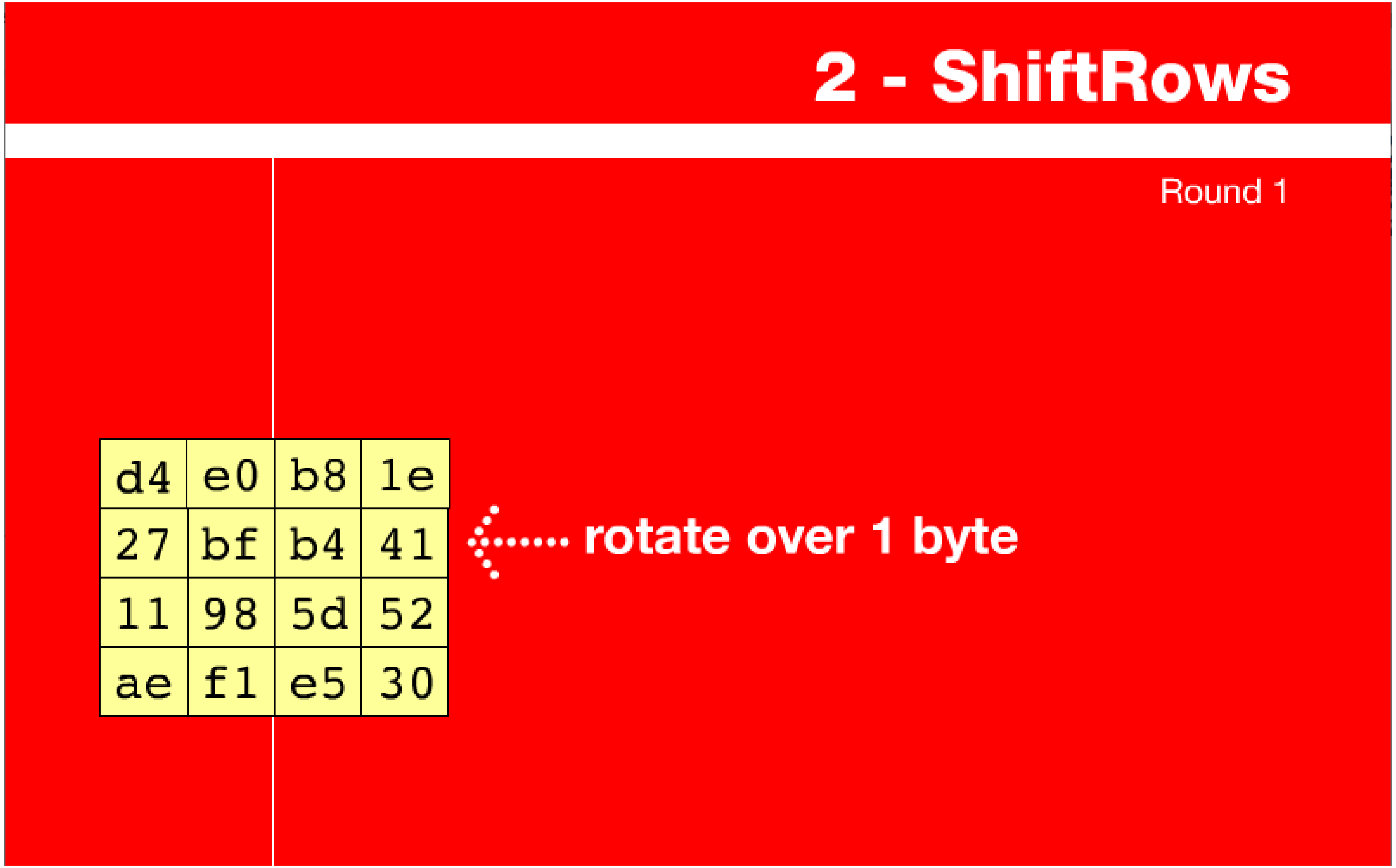


	a0	9a	e9
3d	f4	c6	f8
e3	e2	8d	48
be	2b	2a	08

hex	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

- **Animation**

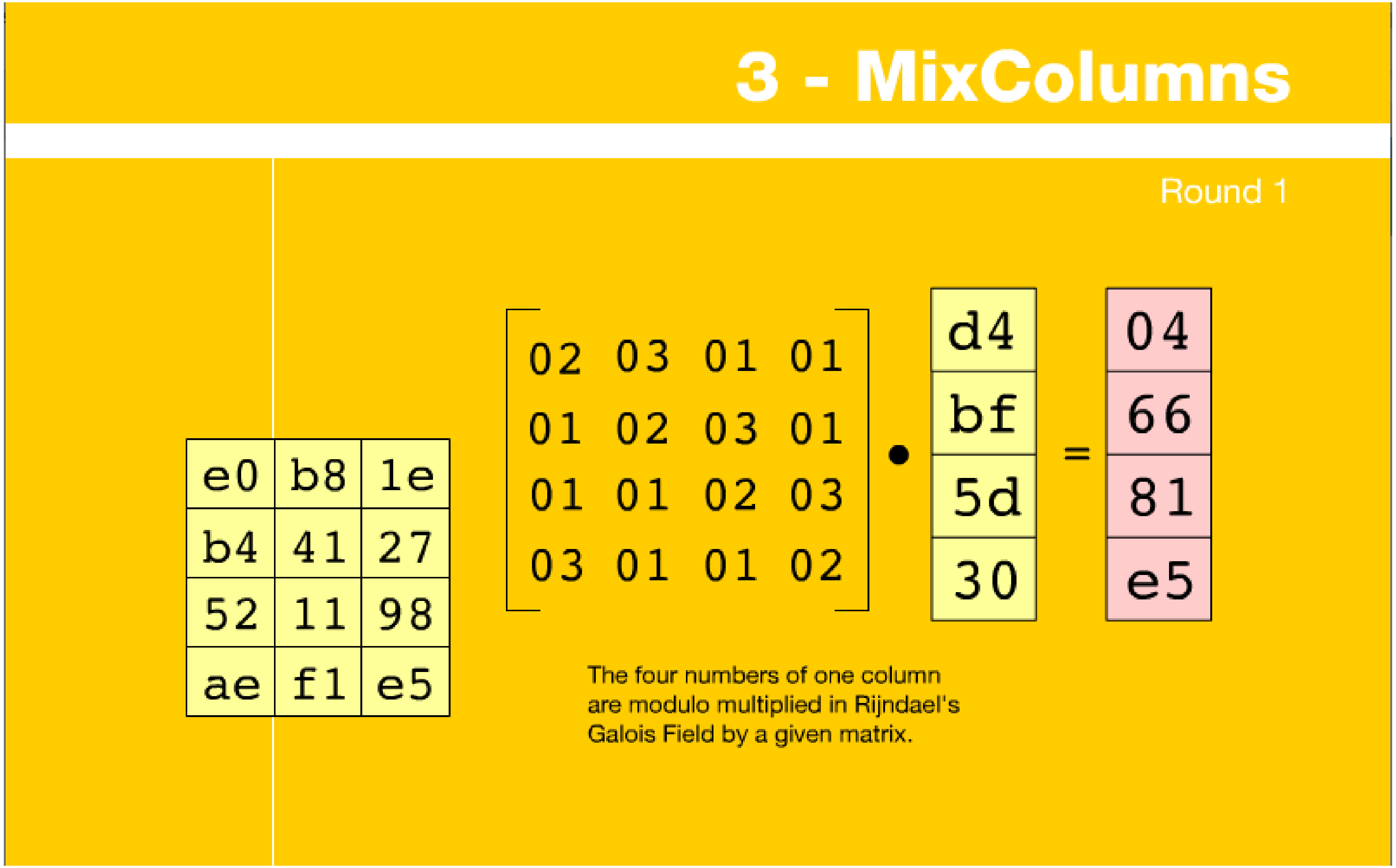
.


2 - ShiftRows
Round 1
d4 e0 b8 1e
27 bf b4 41
11 98 5d 52
ae f1 e5 30
rotate over 1 byte

Advanced Encryption Standard (AES)

- **Animation**

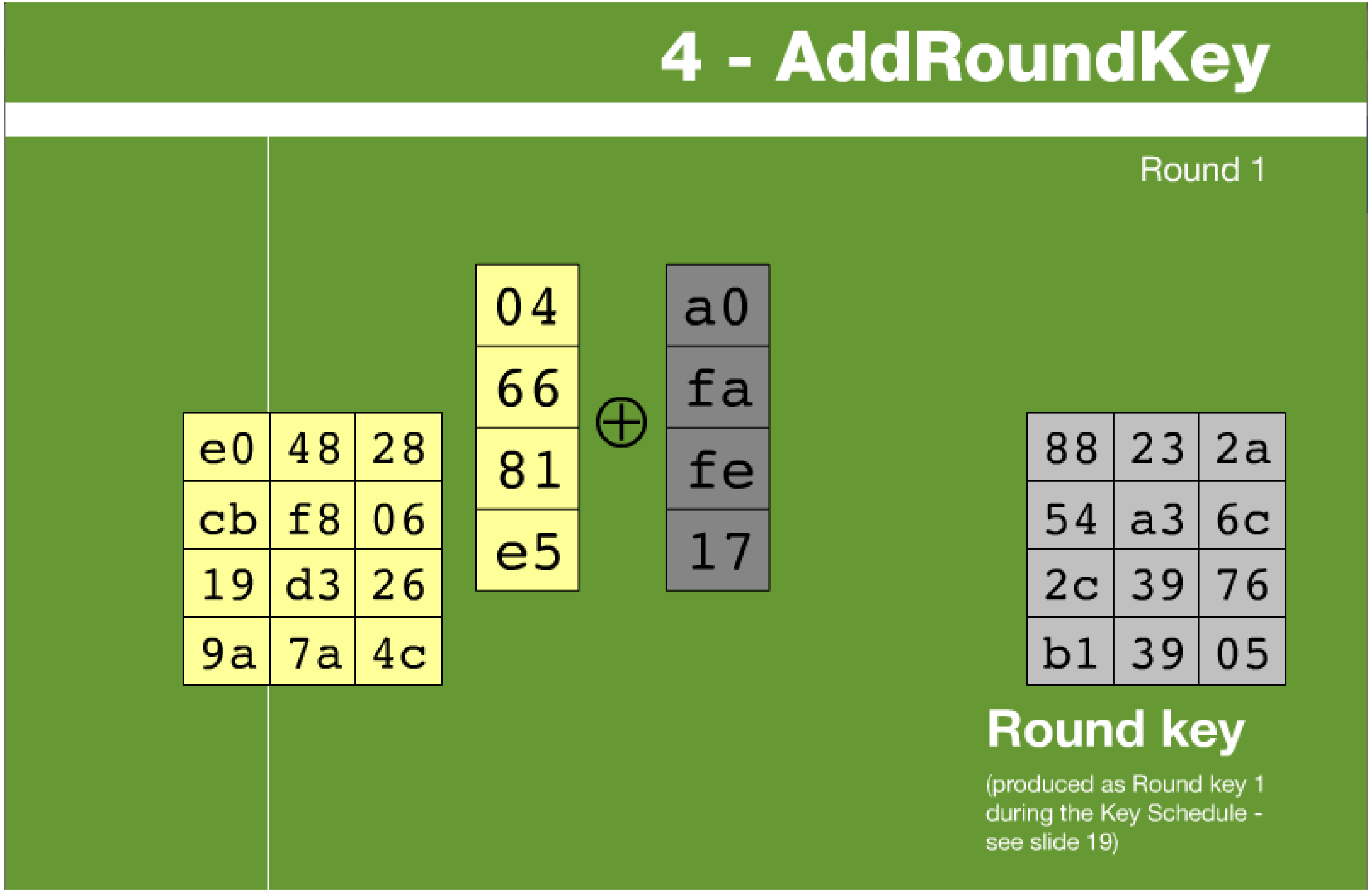
.



Advanced Encryption Standard (AES)

- **Animation**

.



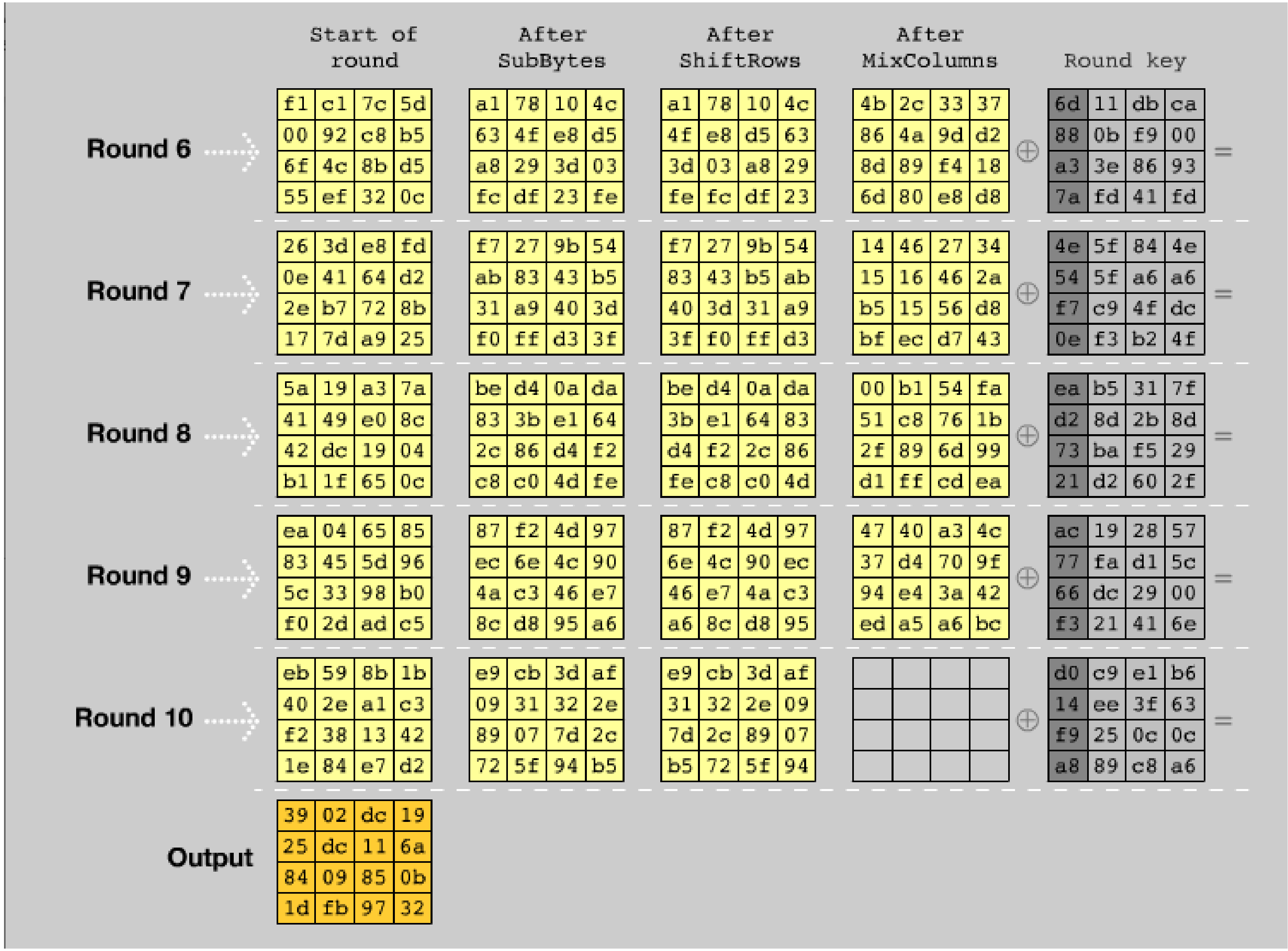
- **Animation**

	Start of round	After SubBytes	After ShiftRows	After MixColumns	Round key
Input	32 88 31 e0				2b 28 ab 09
	43 5a 31 37				7e ae f7 cf
	f6 30 98 07				15 d2 15 4f
	a8 8d a2 34				16 a6 88 3c
Round 1	19 a0 9a e9	d4 e0 b8 1e	d4 e0 b8 1e	04 e0 48 28	a0 88 23 2a
	3d f4 c6 f8	27 bf b4 41	bf b4 41 27	66 cb f8 06	fa 54 a3 6c
	e3 e2 8d 48	11 98 5d 52	5d 52 11 98	81 19 d3 26	fe 2c 39 76
	be 2b 2a 08	ae f1 e5 30	30 ae f1 e5	e5 9a 7a 4c	17 b1 39 05
Round 2	a4 68 6b 02	49 45 7f 77	49 45 7f 77	58 1b db 1b	f2 7a 59 73
	9c 9f 5b 6a	de db 39 02	db 39 02 de	4d 4b e7 6b	c2 96 35 59
	7f 35 ea 50	d2 96 87 53	87 53 d2 96	ca 5a ca b0	95 b9 80 f6
	f2 2b 43 49	89 f1 1a 3b	3b 89 f1 1a	f1 ac a8 e5	f2 43 7a 7f
Round 3	aa 61 82 68	ac ef 13 45	ac ef 13 45	75 20 53 bb	3d 47 1e 6d
	8f dd d2 32	73 c1 b5 23	c1 b5 23 73	ec 0b c0 25	80 16 23 7a
	5f e3 4a 46	cf 11 d6 5a	d6 5a cf 11	09 63 cf d0	47 fe 7e 88
	03 ef d2 9a	7b df b5 b8	b8 7b df b5	93 33 7c dc	7d 3e 44 3b
Round 4	48 67 4d d6	52 85 e3 f6	52 85 e3 f6	0f 60 6f 5e	ef a8 b6 db
	6c 1d e3 5f	50 a4 11 cf	a4 11 cf 50	d6 31 c0 b3	44 52 71 0b
	4e 9d b1 58	2f 5e c8 6a	c8 6a 2f 5e	da 38 10 13	a5 5b 25 ad
	ee 0d 38 e7	28 d7 07 94	94 28 d7 07	a9 bf 6b 01	41 7f 3b 00
Round 5	e0 c8 d9 85	e1 e8 35 97	e1 e8 35 97	25 bd b6 4c	d4 7c ca 11
	92 63 b1 b8	4f fb c8 6c	fb c8 6c 4f	d1 11 3a 4c	d1 83 f2 f9
	7f 63 35 be	d2 fb 96 ae	96 ae d2 fb	a9 d1 33 c0	c6 9d b8 15
	e8 c0 50 01	9b ba 53 7c	7c 9b ba 53	ad 68 8e b0	f8 87 bc bc

Advanced Encryption Standard (AES)

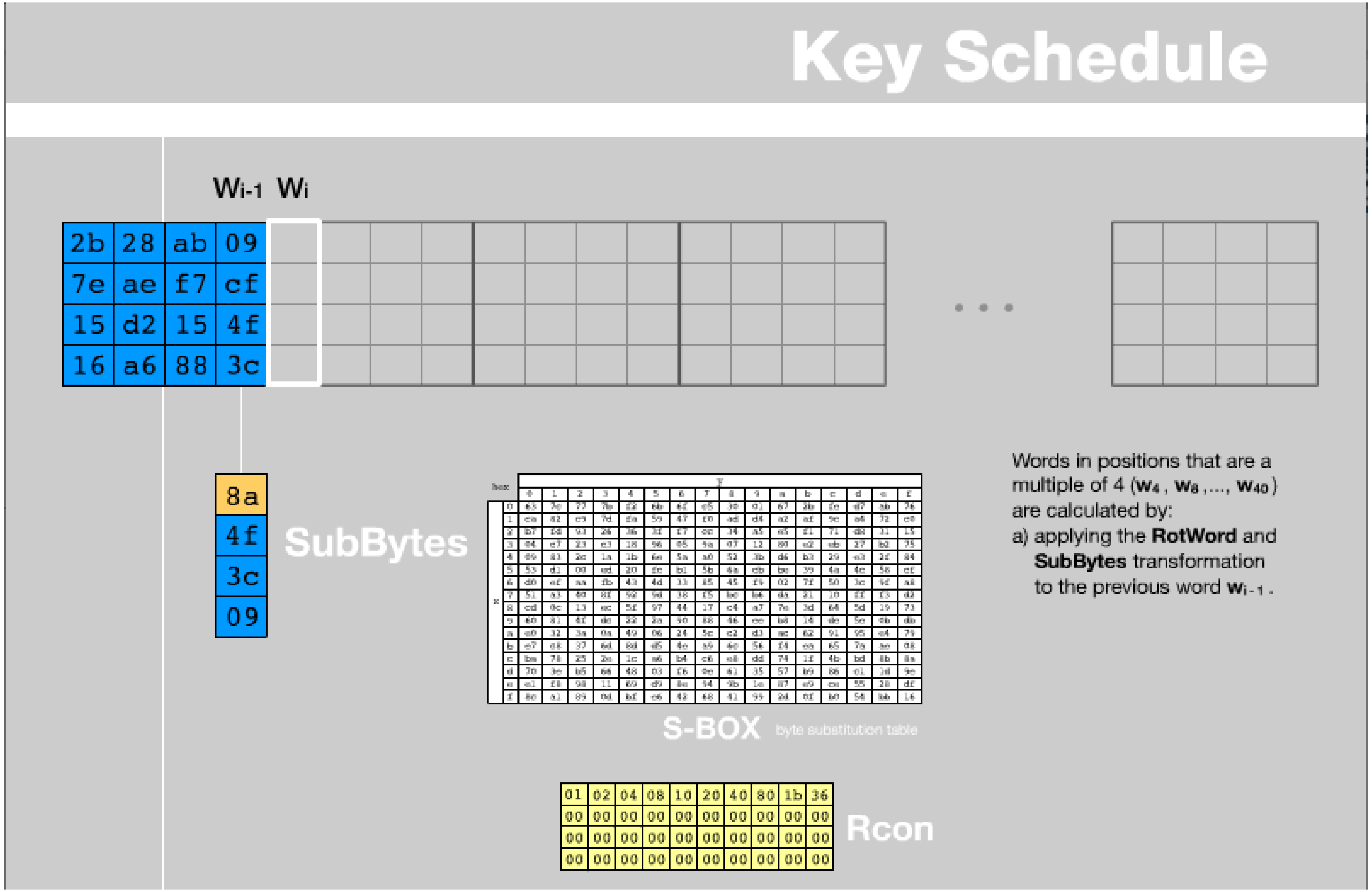
- **Animation**

.



Advanced Encryption Standard (AES)

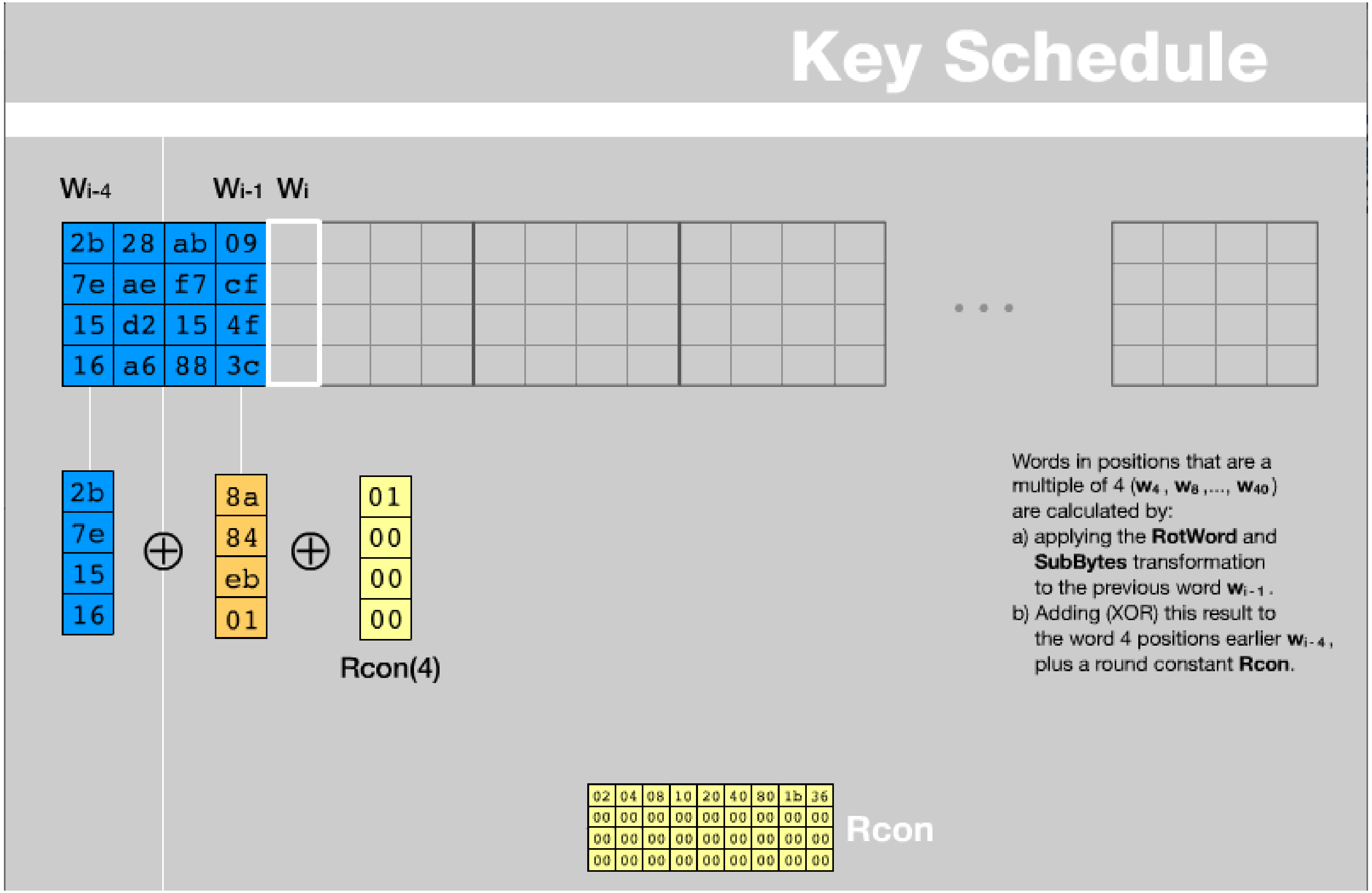
- **Animation**



Advanced Encryption Standard (AES)

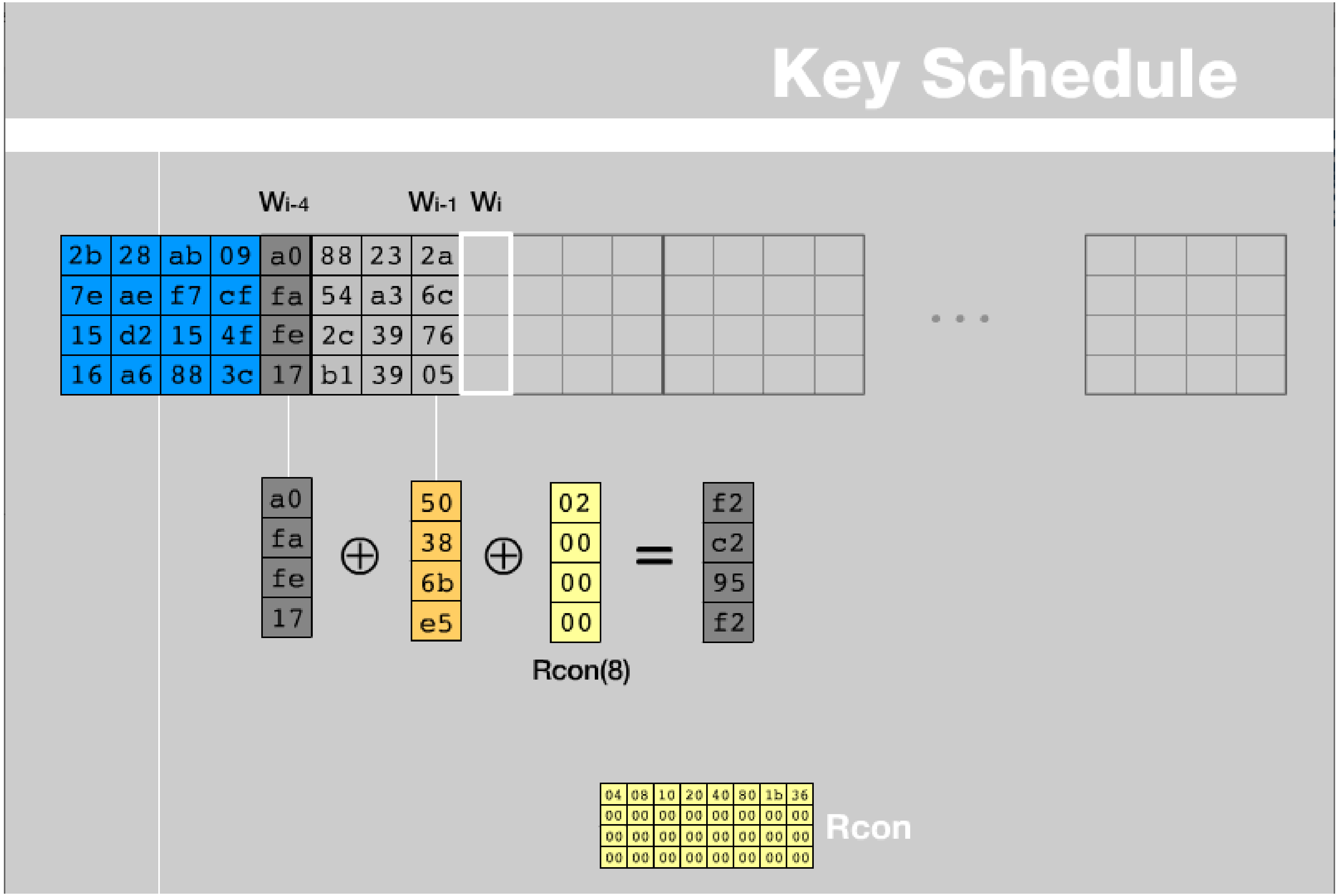
- **Animation**

.



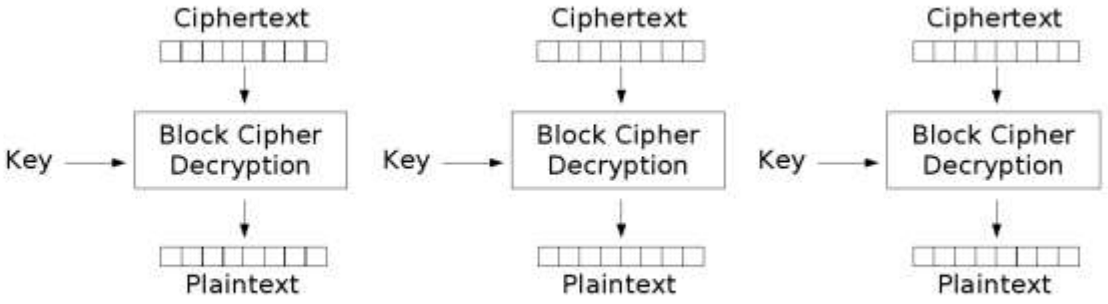
Advanced Encryption Standard (AES)

- **Animation**

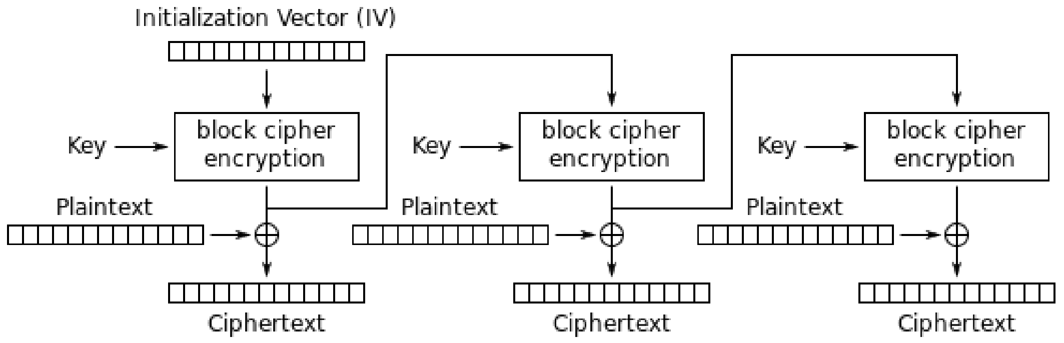


.

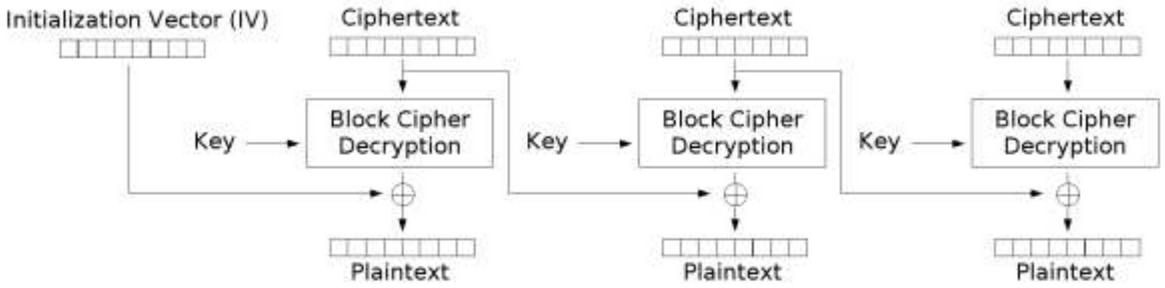
Modos de Operación



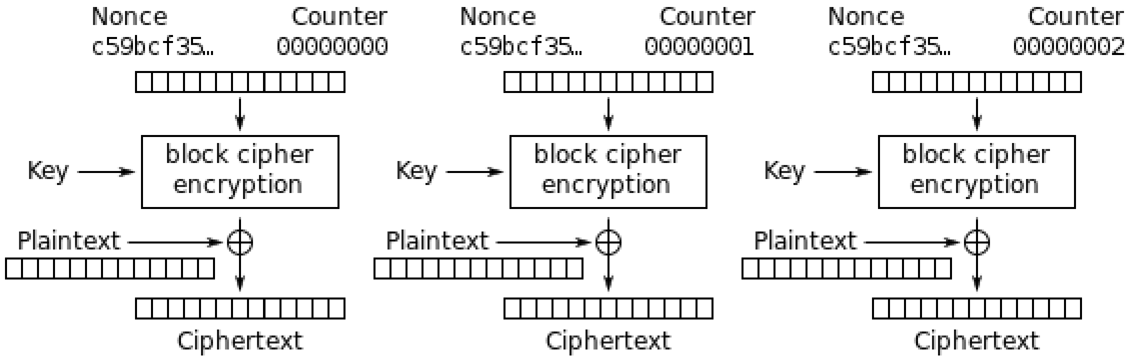
Electronic Codebook (ECB) mode decryption



Output Feedback (OFB) mode encryption



Cipher Block Chaining (CBC) mode decryption



Counter (CTR) mode encryption

Criptografía

Universidad
Francisco de Vitoria
UFV Madrid

Tema 3
Cifrado Simétrico