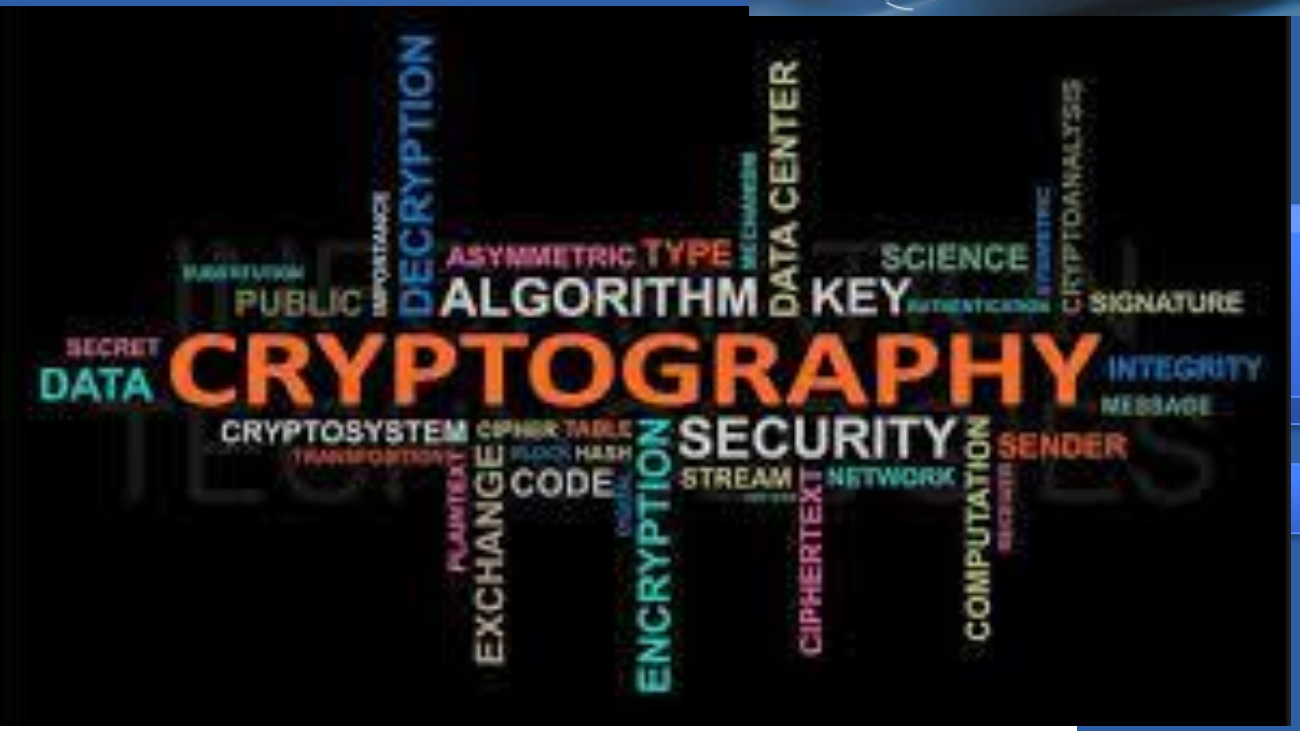
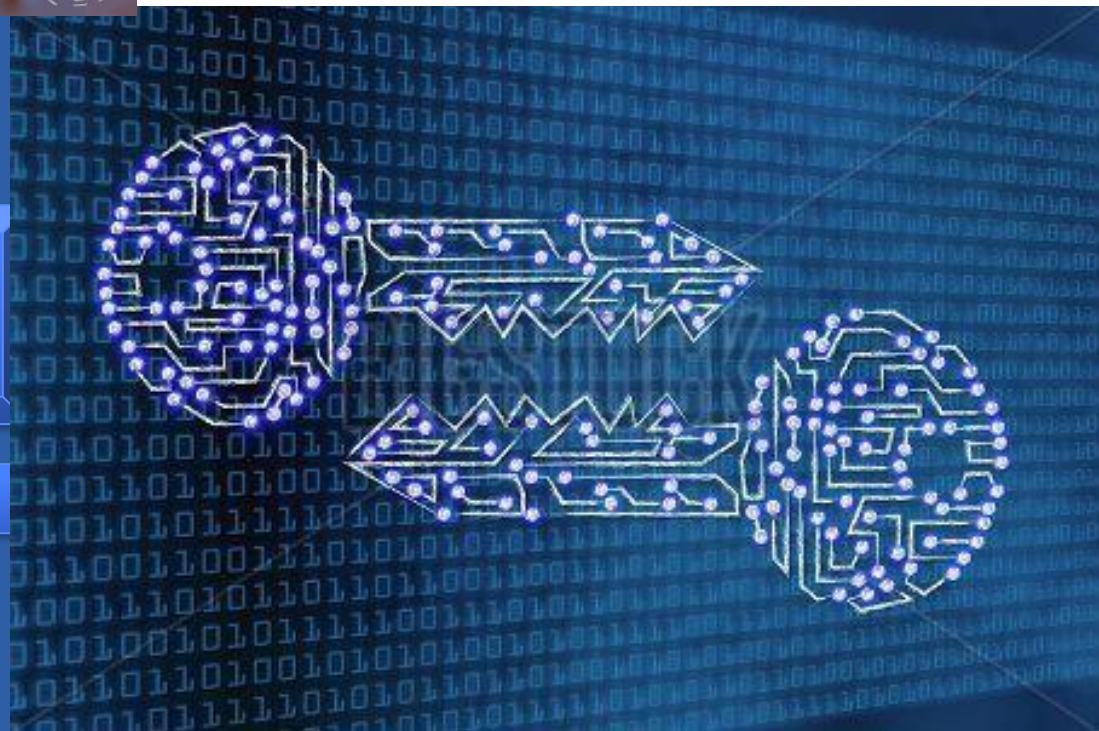
Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 2
Criptografía Clásica





Criptografía Clásica

- **Criptografía**
  - del griego κρύπτος (kryptós), «secreto», y
  - γραφή (graphé), «grafo» o «escritura»,
    - literalmente «***escritura secreta***»

➢ ámbito de la **criptología** que se ocupa de las técnicas de cifrado o codificado destinadas a alterar las representaciones lingüísticas de ciertos mensajes con el fin de hacerlos ininteligibles a receptores no autorizados.

Criptografía Clásica

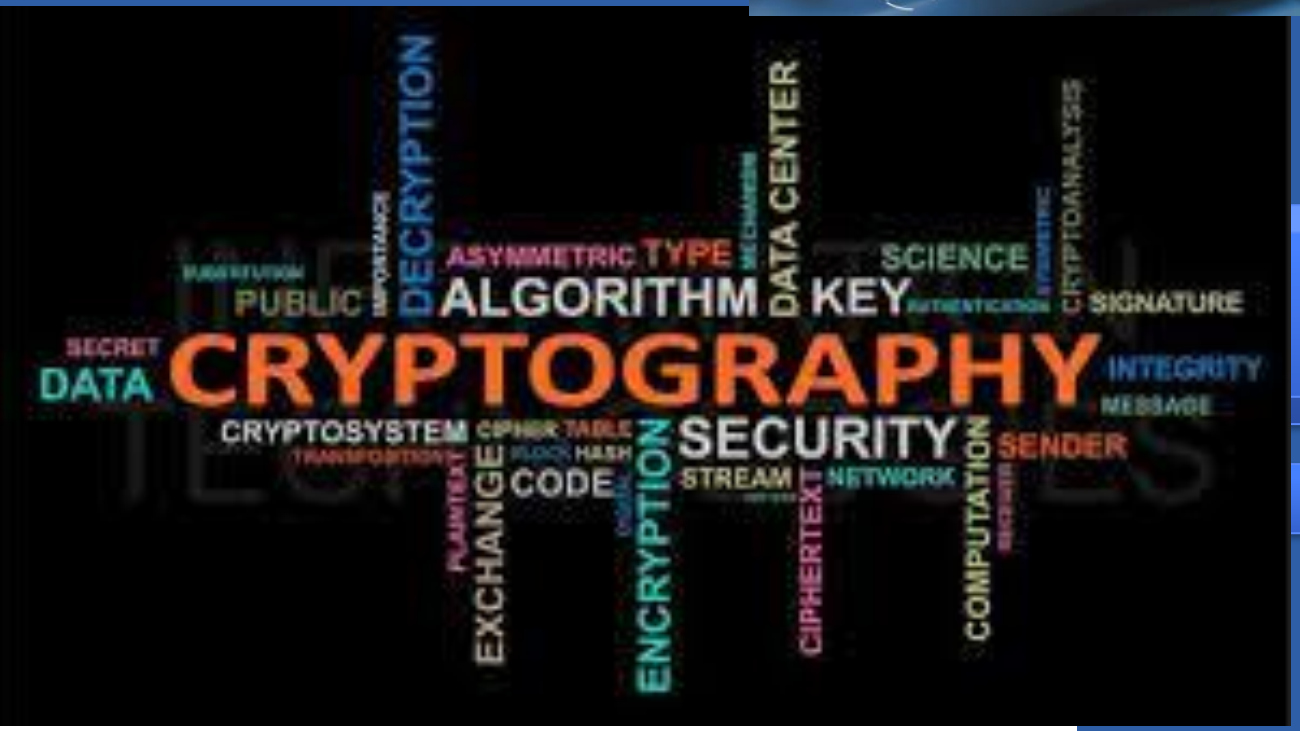
- ***¿Criptografía vs Criptología?***

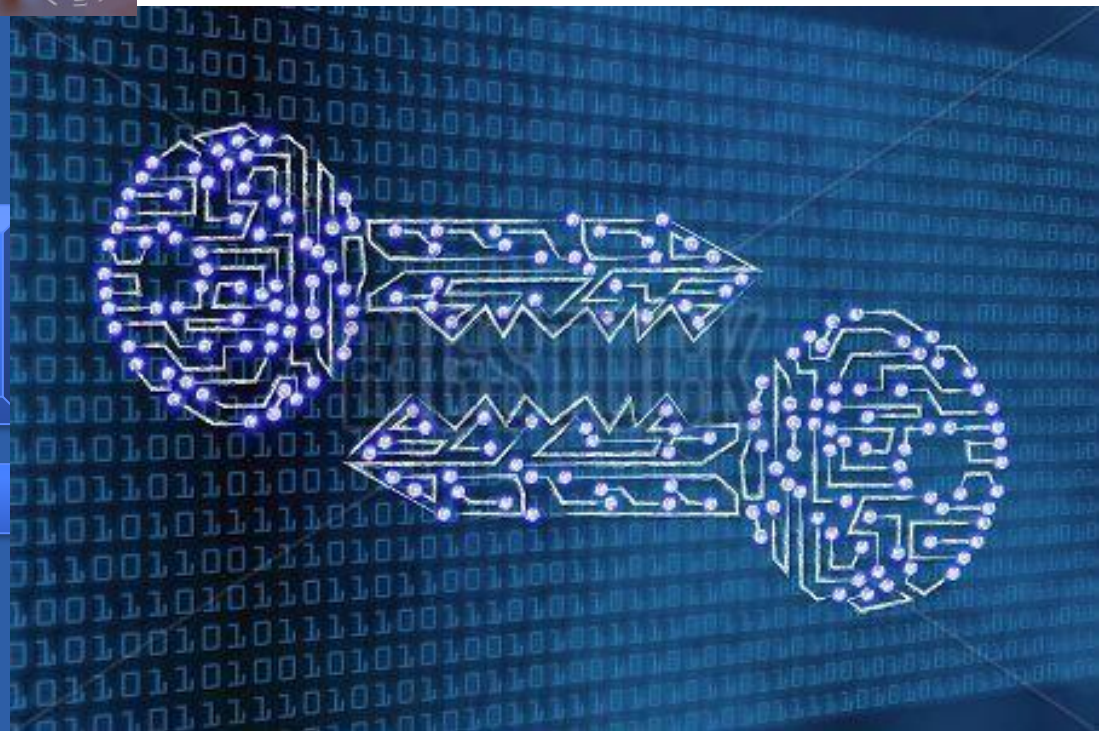
Criptología = Criptografía + Criptoanálisis

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 1.2
Métodos de cifrado clásico





Criptografía Clásica

- **Métodos** de cifrado clásico

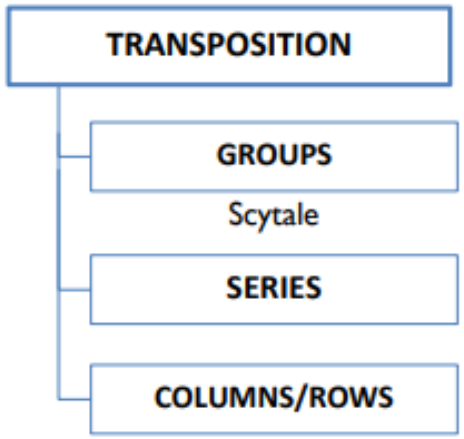
  - Los principales métodos se pueden clasificar en

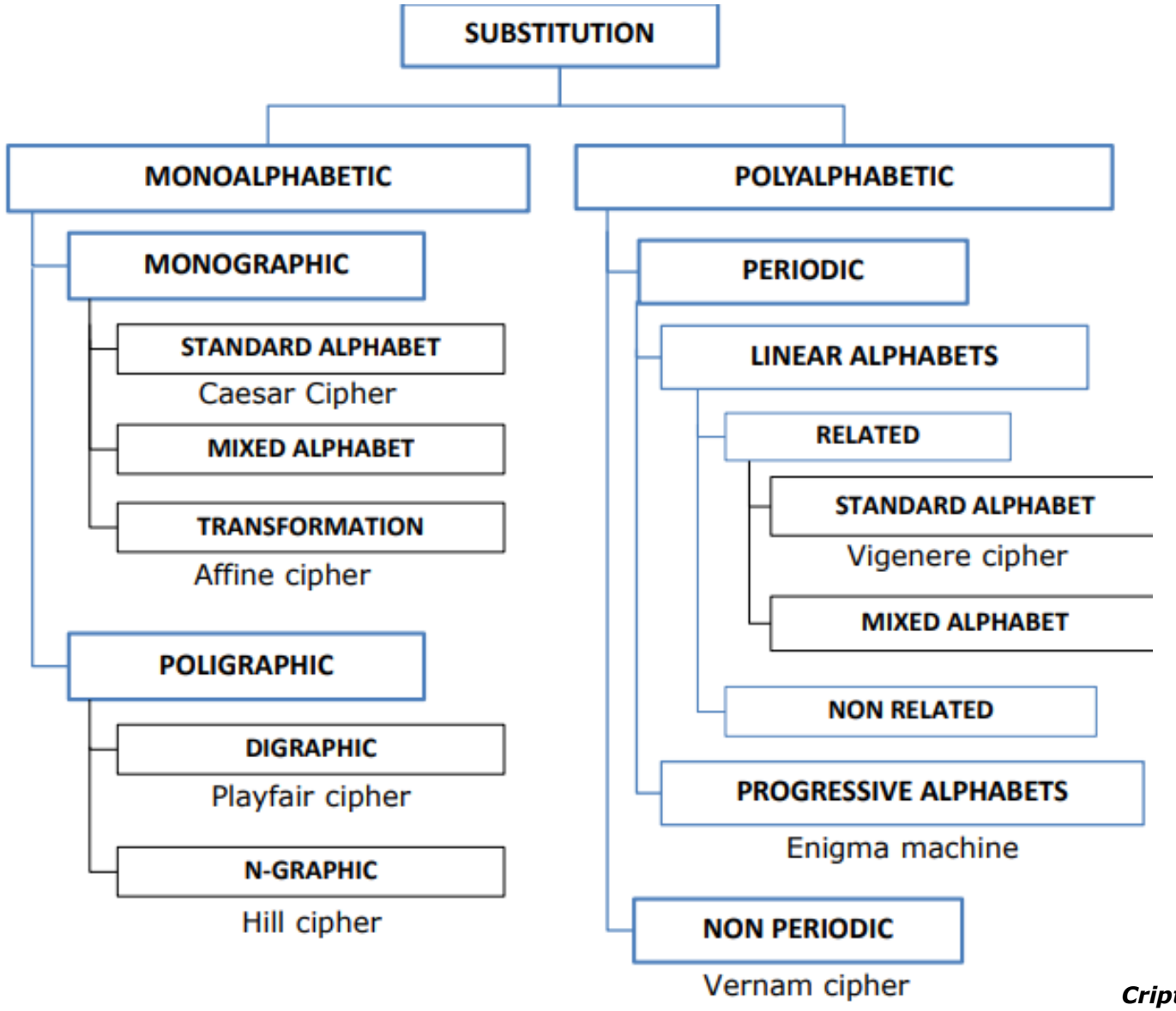
    - **Trasposición** (o permutación)
      - Todos los caracteres del texto plano aparecen en el texto cifrado, pero en diferente posición u orden, según un patrón determinado.

    - **Sustitución**
      - Cada carácter del texto plano se sustituye por otro carácter en el texto cifrado

Criptografía Clásica

- **Clasificación**





Criptografía Clásica

- **Métodos de <u>trasposición</u>**

  - **Por grupos**
    - Las posiciones de las letras dentro de un mismo grupo se determinan por una función de permutación Π_p

  - **Por series**
    - Los caracteres del mensaje se ordenan según las posiciones determinadas por una cadena de sub-mensajes, conforme a un patrón o criterio determinado.

➢ La **frecuencia de repetición** de caracteres en el texto cifrado es la **misma** que en el texto plano

Criptografía Clásica

- **Métodos de trasposición <u>por grupos</u>**

  - El orden (posición en el mensaje cifrado) de las letras dentro de un mismo grupo (de longitud p) se determinan por una función de permutación Π_p

    - **EJEMPLO**: Π_p = 24531

      - M = BUENO SDIAS
      - C = UNOEB DASIS

- *Incrementando el periodo "p" se incrementa la seguridad del cifrado (menos vulnerable)*

Métodos de trasposición por series

- Los caracteres del mensaje se ordenan según las posiciones determinadas por una cadena de sub-mensajes, conforme a un patrón o criterio determinado.

  - **EJEMPLO**: $C = M_{S1}M_{S2}M_{S3}$
    - Siendo:
      - **M_{S1}** las posiciones pares, excepto las posiciones en número primo (4,6,8,10,12,14,16)
      - **M_{S2}** las posiciones impares, excepto las posiciones en número primo (1,9,15)
      - **M_{S3}** las posiciones en número primo (2,3,5,7,11,13,17)

        - M = BUENOSDIASEQUIPO!

        - **C = ??**

Métodos de trasposición por series

- **SOLUCIÓN**: C = $M_{S1}M_{S2}M_{S3}$
  - Siendo:
    - **M_{S1}** las posiciones pares, excepto las posiciones en número primo (4,6,8,10,12,14,16)
    - **M_{S2}** las posiciones impares, excepto las posiciones en número primo (1,9,15)
    - **M_{S3}** las posiciones en número primo (2,3,5,7,11,13,17)

➢ M = BUENOSDIASEQUIPO!

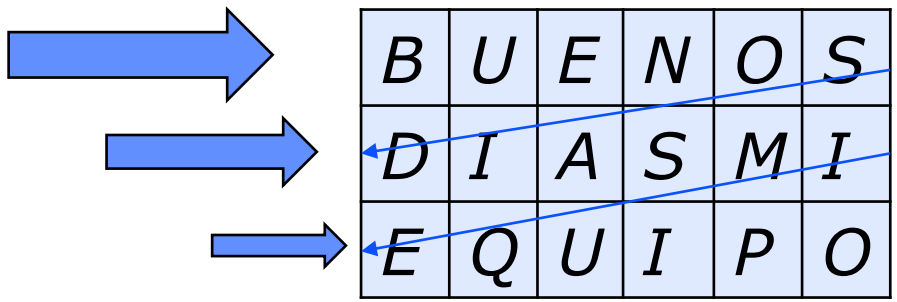
Pos:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
M	*B*	*U*	*E*	*N*	*O*	*S*	*D*	*I*	*A*	*S*	*E*	*Q*	*U*	*I*	*P*	*O*	*!*
M_{S1}				N		S		I		S		Q		I		O	
M_{S2}	B								A						P		
M_{S3}		U	E		O		D				E		U				!
C	*N*	*S*	*I*	*S*	*Q*	*I*	*O*	*B*	*A*	*P*	*U*	*E*	*O*	*D*	*E*	*U*	*!*

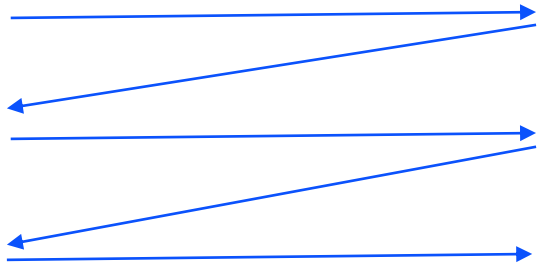
- **Métodos de trasposición <u>por filas / columnas</u>**

  - **ALGORITMO (CRITERIO):**

    - Los elementos **se introducen** según un patrón geométrico (por ejemplo, por filas)
    - Y se extraen según otro patrón (por ejemplo, por columnas)

Ejemplo: M = BUENOSDIASMIEQUIPO

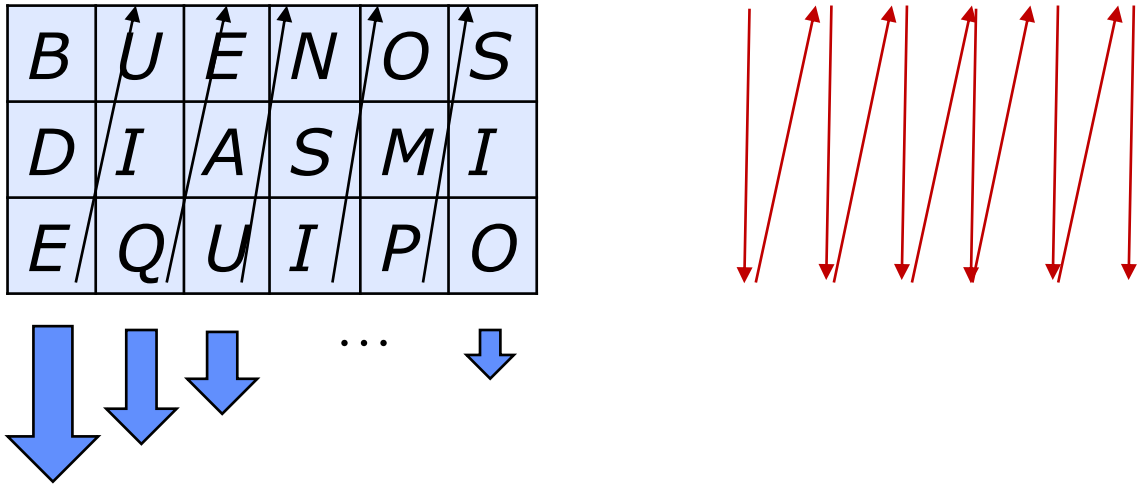




- **Métodos de trasposición <u>por filas / columnas</u>**

  - **ALGORITMO (CRITERIO):**
    - Los elementos se introducen según un patrón geométrico (por columnas)
    - Y **se extraen** según otro patrón (por ejemplo, por columnas)

Ejemplo: M = BUENOSDIASMIEQUIPO

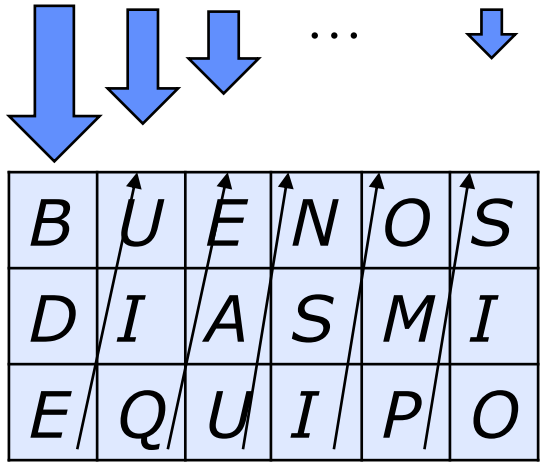


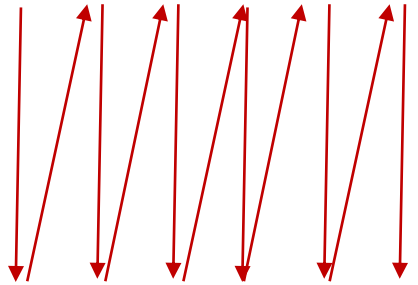
Cifrado: C = BDEUIQEAUNSIOMPSIO

- **Métodos de trasposición <u>por filas / columnas</u>**

  - **DESCIFRADO:**
    - **Se invierten los patrones**
  - Para el ejemplo anterior:
    - Los elementos **se introducen** por columnas (según el ejemplo)
    - Y **se extraen por filas (por columnas)**

Cifrado: C = BDEUIQEAUNSIOMPSIO

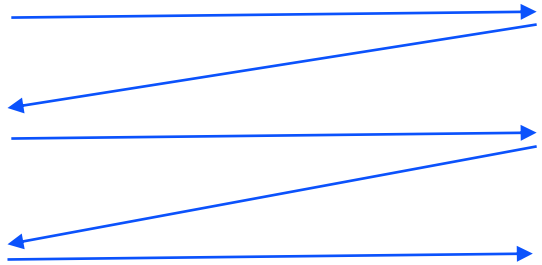




- **Métodos de trasposición por filas / columnas**

  - **DESCIFRADO:**
    - **Se invierten los patrones**
  - Para el ejemplo anterior:
    - Los elementos **se introducen por** columnas (según el ejemplo)
    - Y **se extraen por filas** (por columnas)

B	U	E	N	O	S
D	I	A	S	M	I
E	Q	U	I	P	O



M = BUENOSDIASMIEQUIPO

Criptografía Clásica

- **Métodos** de cifrado clásico

  Sustitución

  - Cada carácter del alfabeto del texto plano se sustituye por otro carácter del alfabeto del texto cifrado, según una función o algoritmo.

Criptografía Clásica

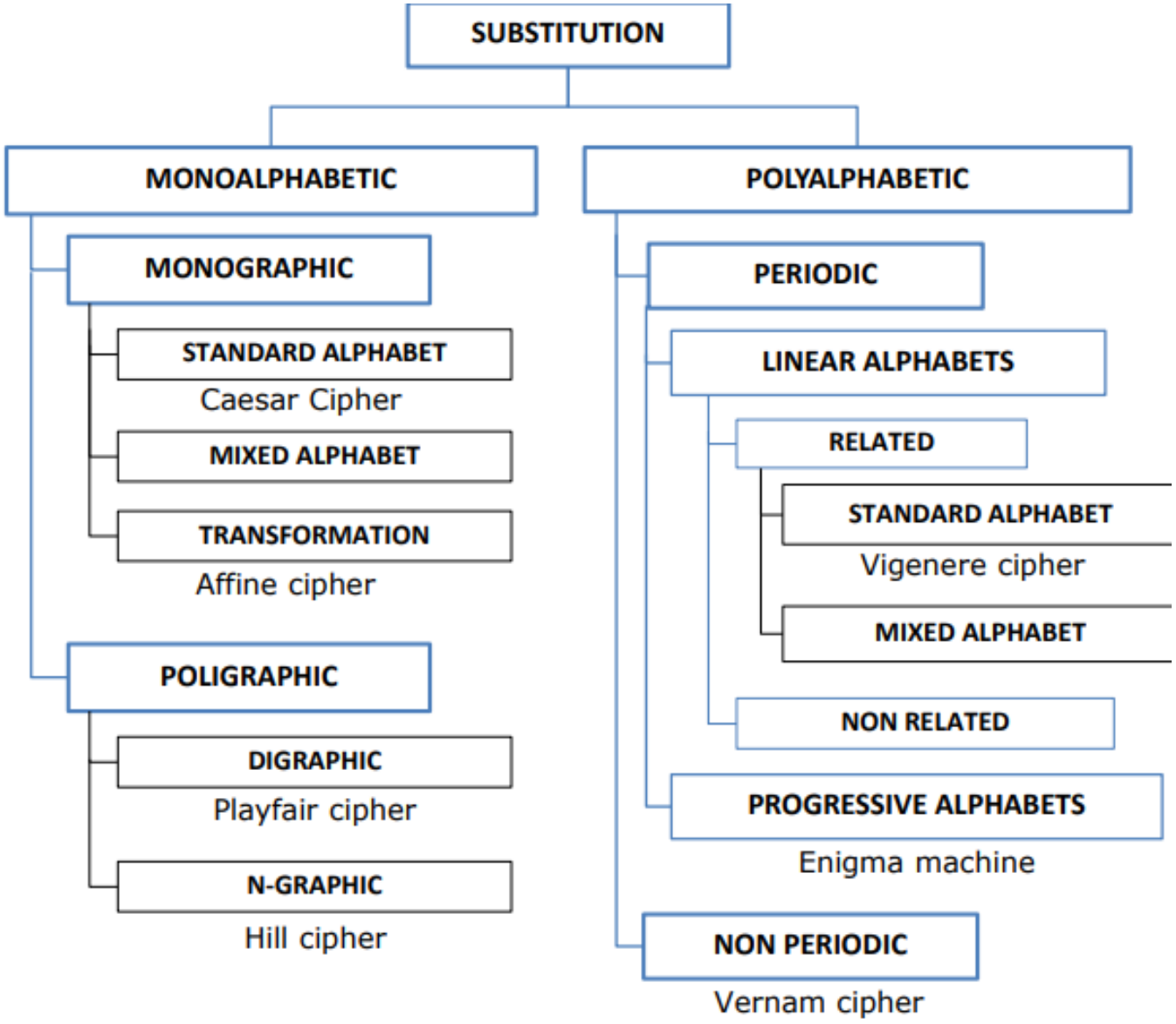
- **Métodos de <u>sustitución</u>**

  - Se define **alfabeto** como el conjunto de símbolos o caracteres utilizados.
    - Alfabeto para el texto plano
    - Alfabeto para el texto cifrado

  - Se suele emplear una conversión **numérica** para cada símbolo o carácter del alfabeto
    - para poder hacer operaciones matemáticas con ellos
    - *Ejemplo: A=0; B=1; C=2;… Z=25*

Criptografía Clásica

- **Clasificación**



Criptografía Clásica

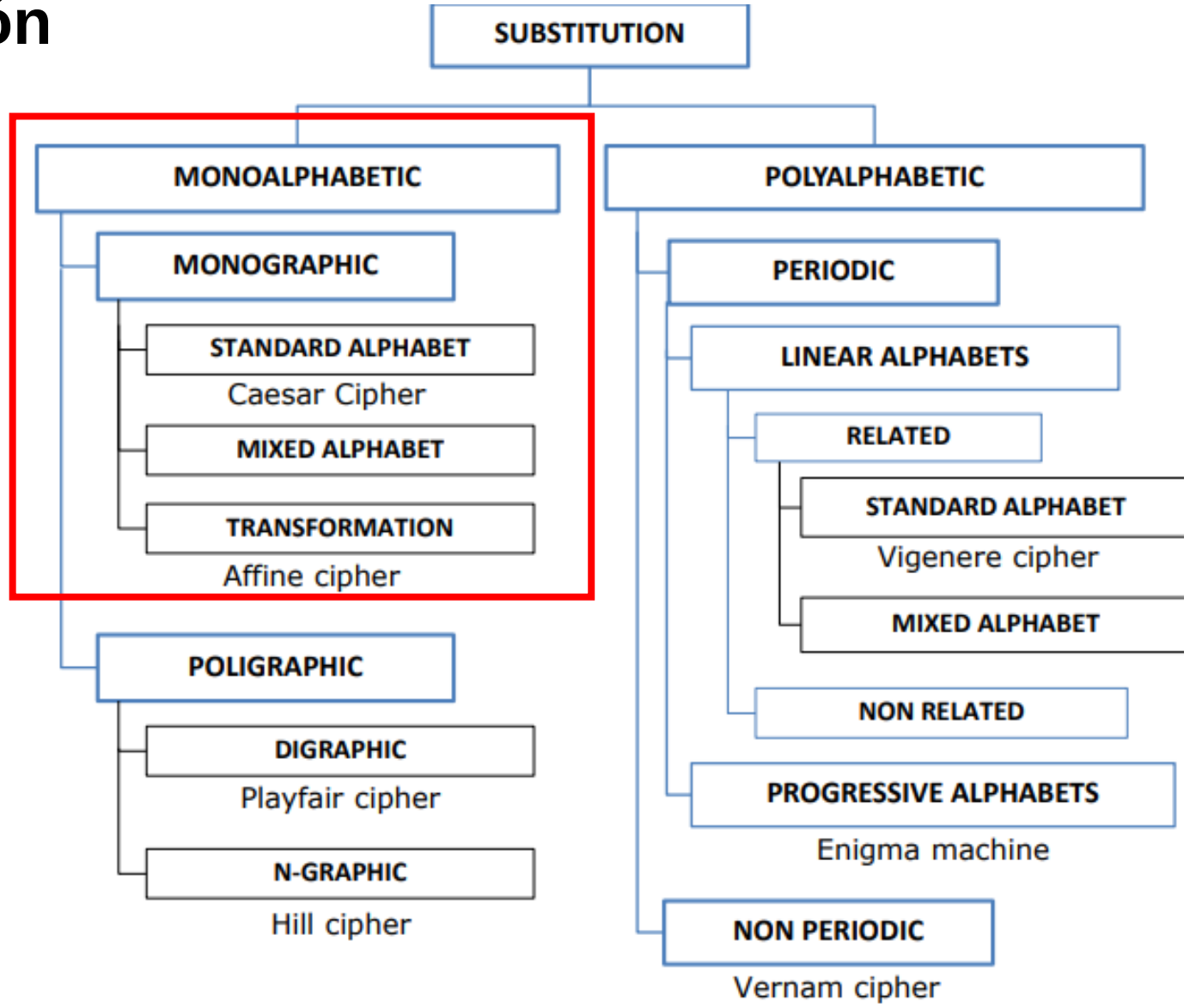
- **Métodos de sustitución**

  - **Monoalfabéticos**
    - Se sustituye un caracter del alfabeto del texto original por un carácter del alfabeto del texto cifrado

  - **Polialfabéticos**
    - La sustitución de un carácter del alfabeto del texto original se corresponde con un caracter de uno de los posibles alfabetos del texto cifrado.

Criptografía Clásica

- **Clasificación**



Criptografía Clásica

- **Métodos de <u>sustitución monoalfabéticos</u>**
  - Se sustituye un caracter del alfabeto del texto original por un carácter del alfabeto del texto cifrado
  - ***Ejemplo***, para alfabeto británico de 26 caracteres (n=26)
    - Existen **n! diferentes alfabetos** para el texto cifrado

Criptografía Clásica

- **Métodos de sustitución monoalfabéticos**
  - Se sustituye un caracter del alfabeto del texto original por un carácter del alfabeto del texto cifrado
  - ***Expresión general del cifrador afín para sustitución monoalfabética:***

$$C(m_i) = (a \cdot m_i + b) \bmod n$$

Sólo si: mcd (a,n) = 1 (coprimos!!)

- Siendo:
  - m_i = valor numérico del carácter *m* en la posición "*i*" del texo plano (mensaje)
  - *a* = constante de diezmado
  - *b* = constante de desplazamiento

  } $Clave = \{a,b\}$
  - *n* = número de elementos en el alfabeto (módulo)
  - $C(m_i)$ = valor numérico correspondiente al carácter *m* en la posición "*i*" del texto cifrado.

- **Métodos de <u>sustitución monoalfabéticos</u>**
  - Casos particulares:

➢ Cifrador **CAESAR**: *a = 1, b = 3*

$$C(m_i) = (m_i + 3) \bmod n$$

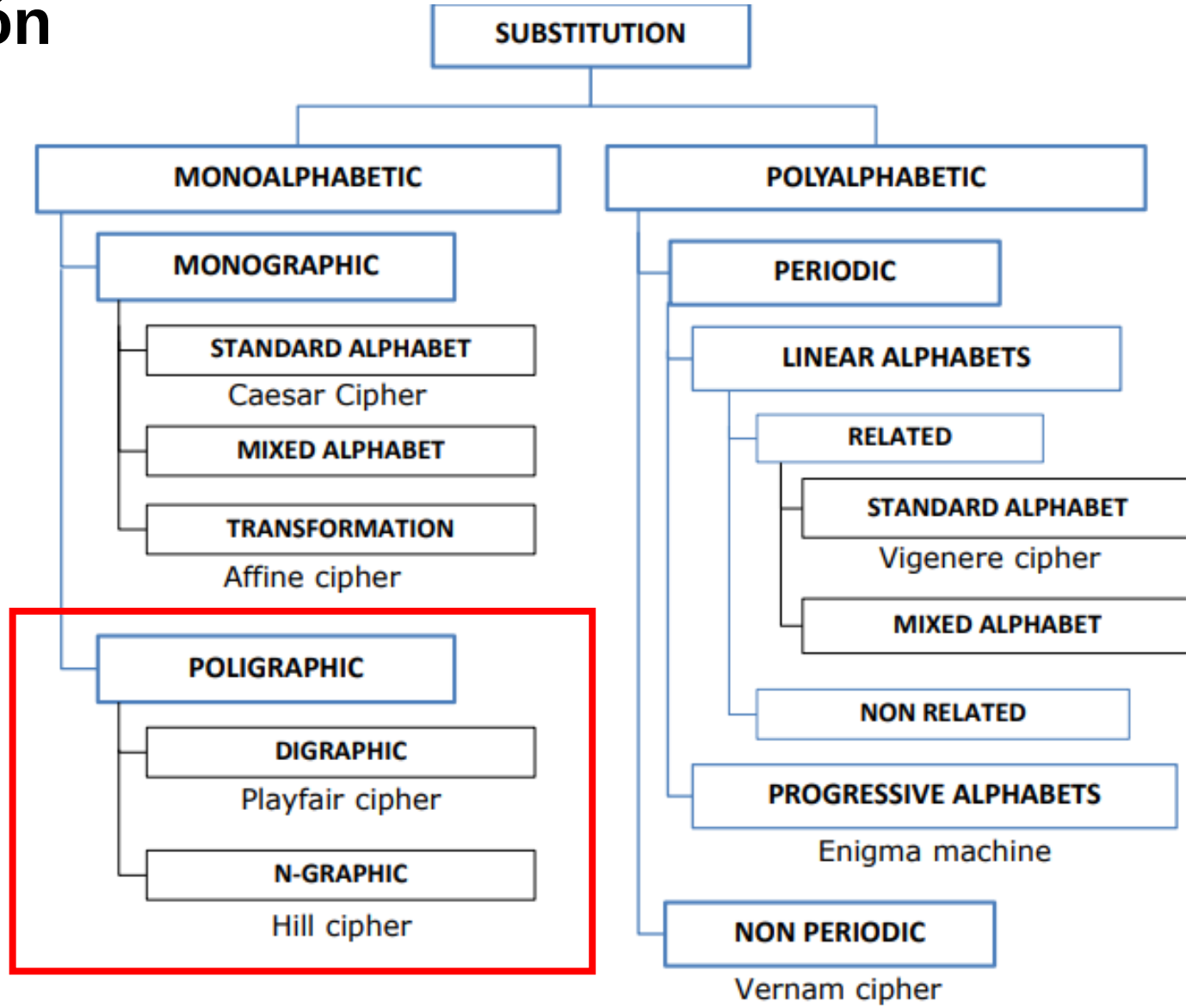
➢ Cifrador **ROT13**: *a = 1, b = 13*

$$C(m_i) = (m_i + 13) \bmod n$$

A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Criptografía Clásica

- **Clasificación**



Criptografía Clásica

- **Métodos de sustitución monoalfabéticos (poligráficos)**

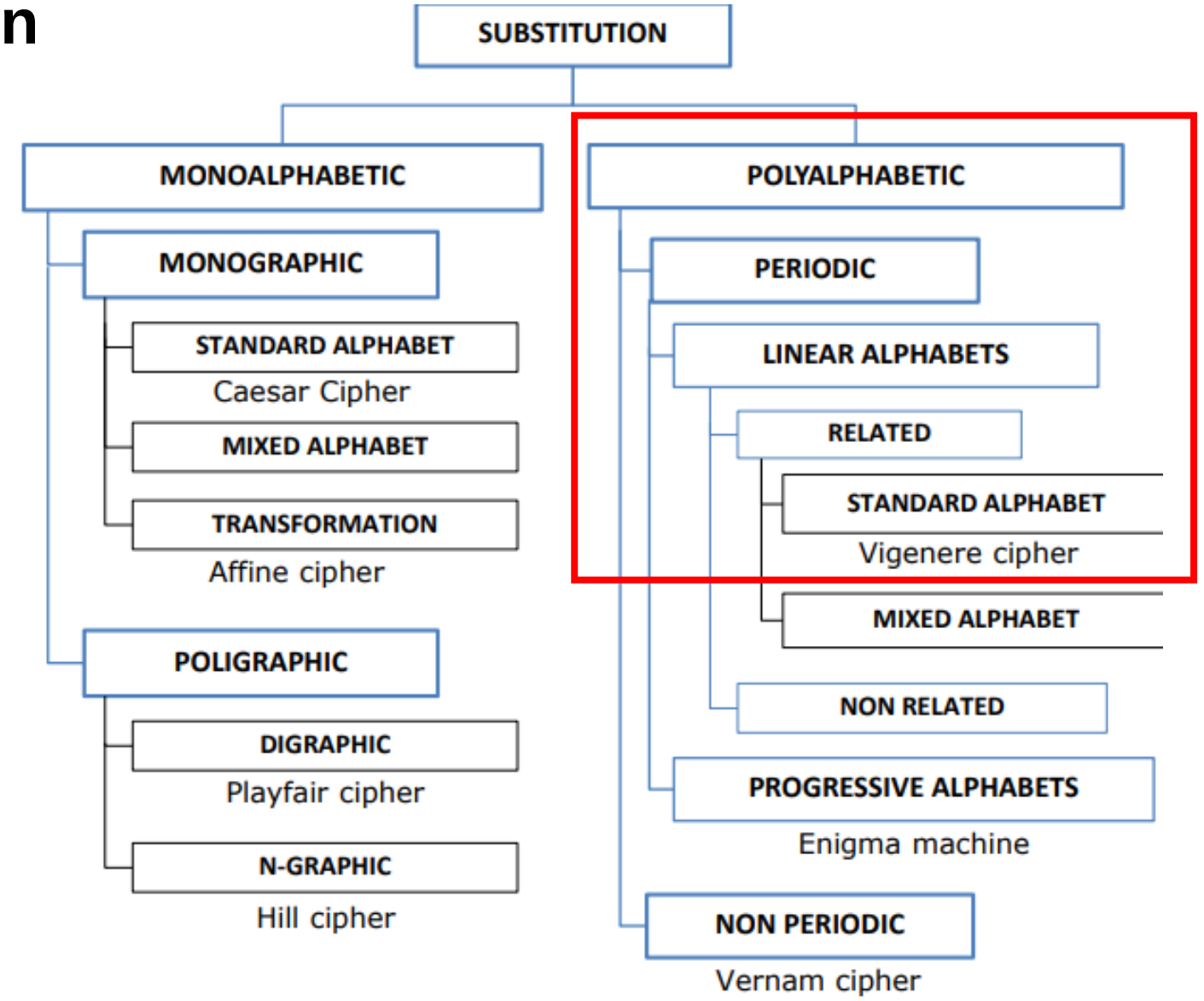
  - **PlayFair**

    - Se sustituyen **dígrafos**, no caracteres individuales
    - Reglas sencillas:
      - Misma fila → se toman los caracteres según desplazamiento a la derecha
      - Misma columna → se toman los caracteres según desplazamiento hacia abajo
      - Rectángulo → se toman los caracteres de la diagonal opuesta
      - Carácter de relleno (normalmente, la “X”), para:
        - Padding
        - Separar letras dobles

    - *https://www.youtube.com/watch?v=UURjVI5cw4g*

Criptografía Clásica

- **Clasificación**



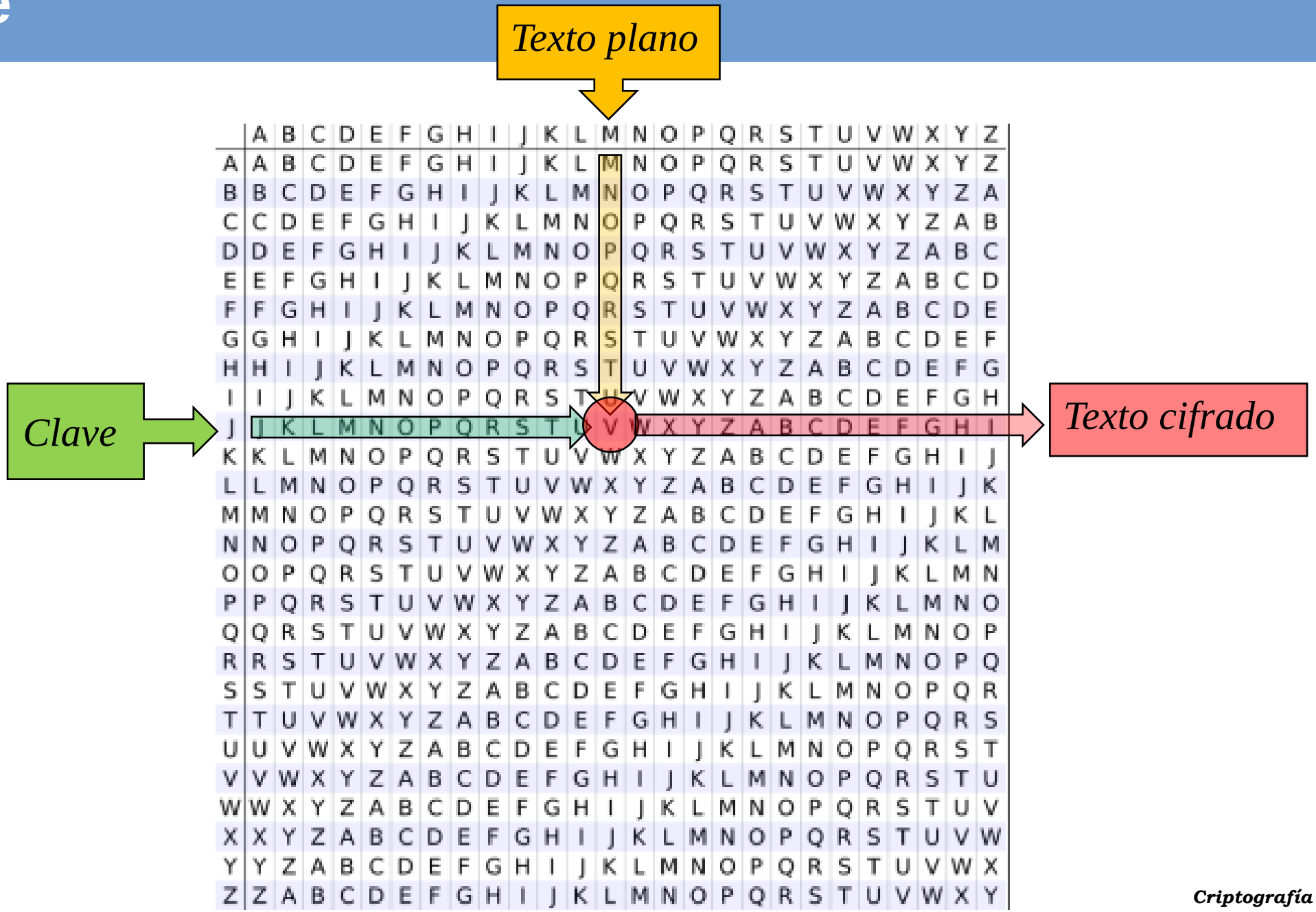
- **Métodos de sustitución polialfabéticos**
  - La sustitución de un carácter del alfabeto del texto original se corresponde con un caracter de **uno de los posibles alfabetos** del texto cifrado
  - ***Ejemplo***, ***Vigenere***

$$C(m_j) = (m_j + k_{(j \bmod n)}) \bmod 26$$

    - para alfabeto británico de 26 caracteres (n=26) → Existen **26 diferentes sustituciones de monoalfabetos**
  - Siendo:
    - m = longitud de la clave
    - k_j = desplazamiento para el alfabeto j
    - m_j = carácter en la posición j del texto plano
    - $C(m_j)$ = valor cifrado del carácter en la posición mj del texto plano

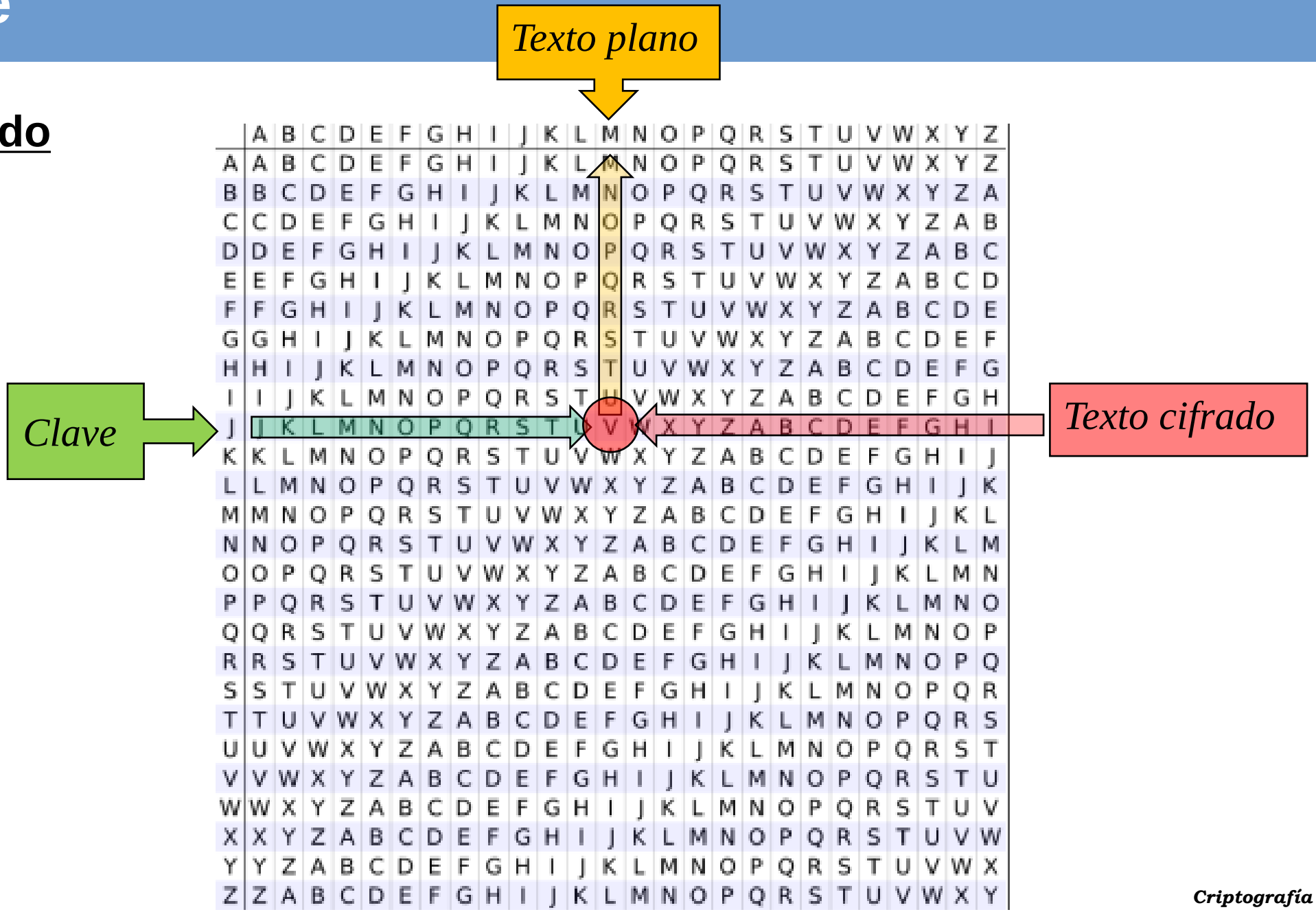
Vigenere

Cifrado



Vigenere

Descifrado



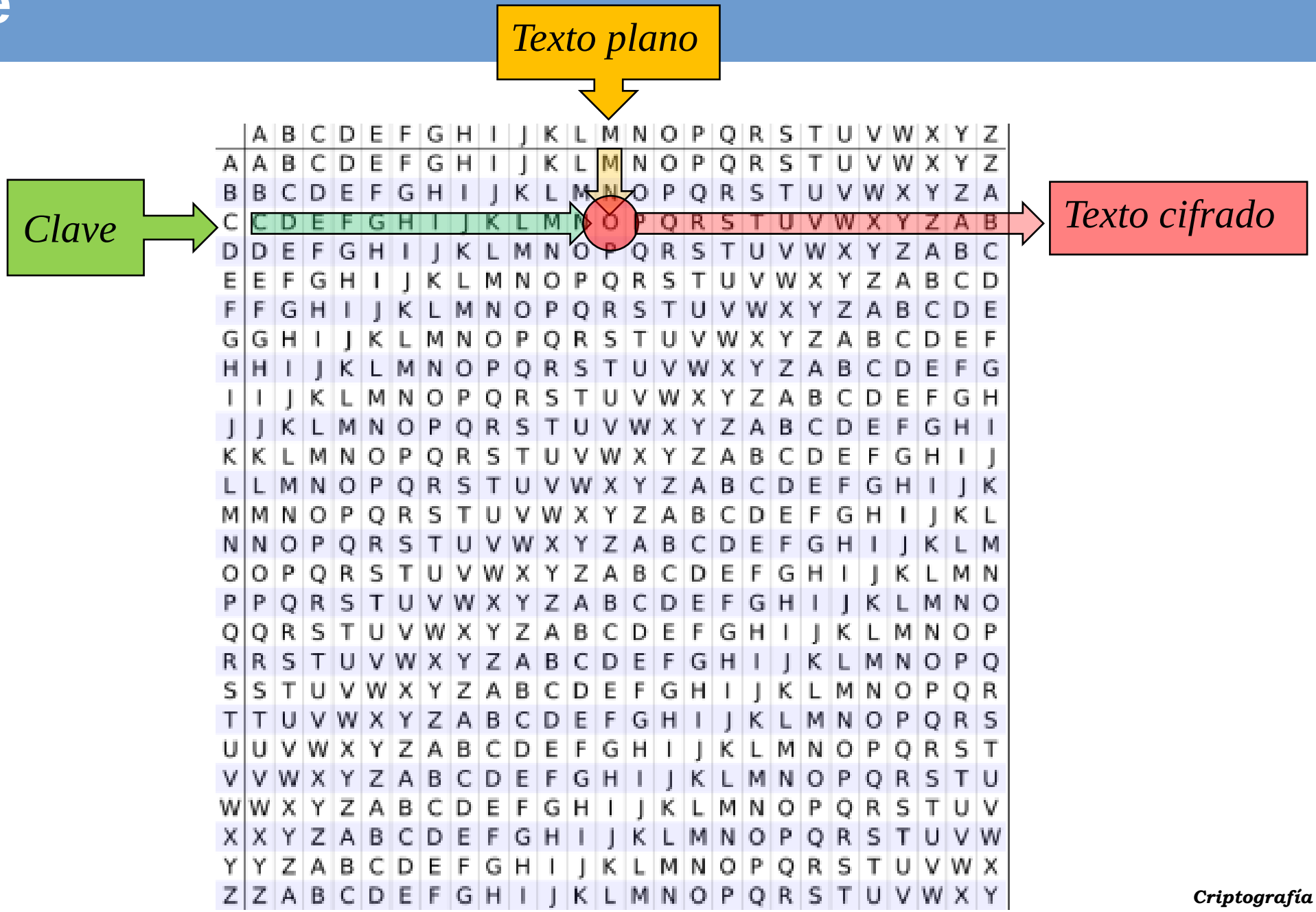
Vigenere

Ejemplo

Texto plano	***M***	*E*	*N*	*S*	*A*	*J*	*E*	*U*	*N*	*O*
Clave	***C***	*L*	*A*	*V*	*E*	*C*	*L*	*A*	*V*	*E*
Texto cifrado	***?***	*?*	*?*	*?*	*?*	*?*	*?*	*?*	*?*	*?*

Vigenere

Cifrado



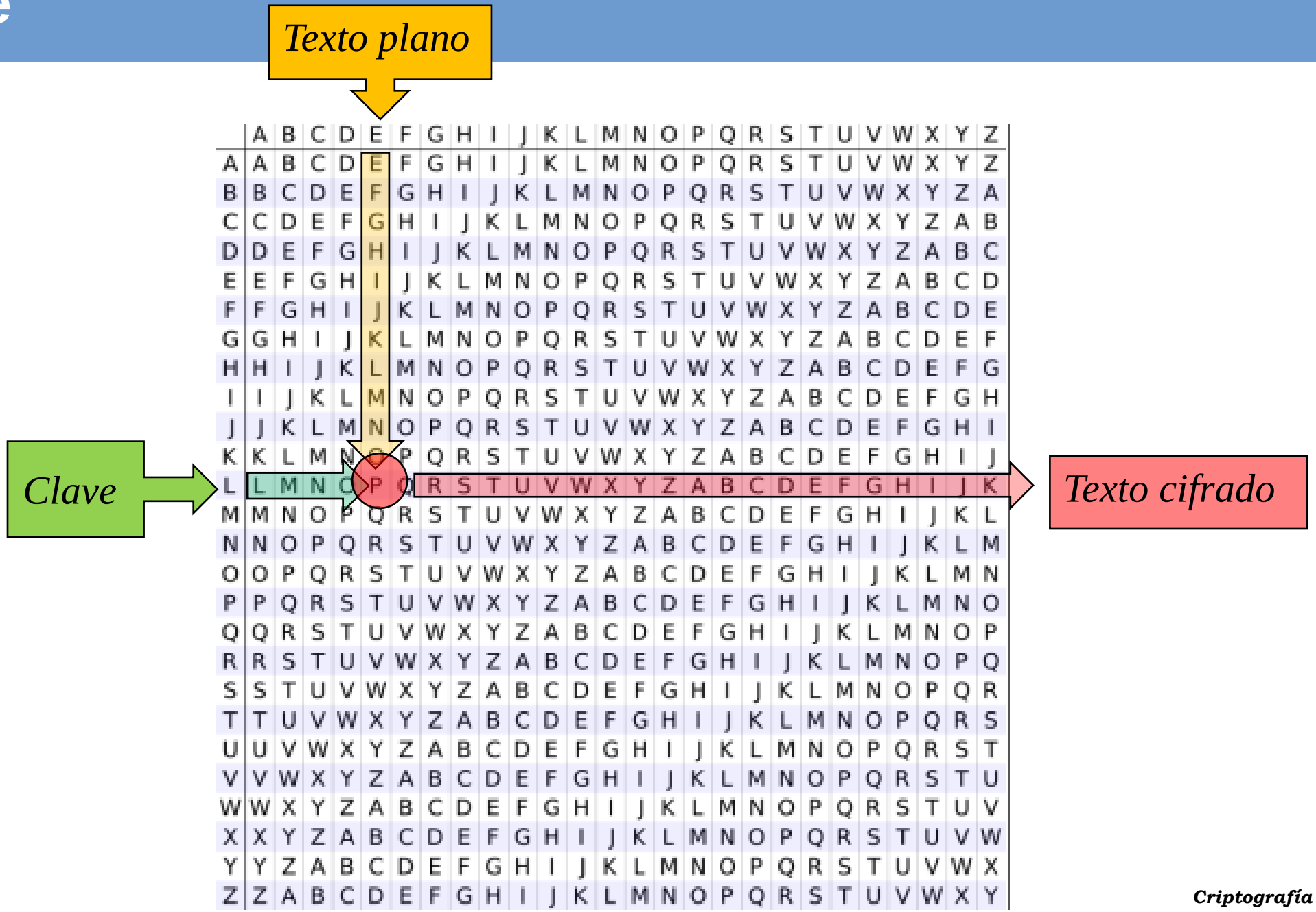
Vigenere

Ejemplo

Texto plano	***M***	*E*	*N*	*S*	*A*	*J*	*E*	*U*	*N*	*O*
Clave	***C***	*L*	*A*	*V*	*E*	*C*	*L*	*A*	*V*	*E*
Texto cifrado	***O***	*?*	*?*	*?*	*?*	*?*	*?*	*?*	*?*	*?*

Vigenere

Cifrado



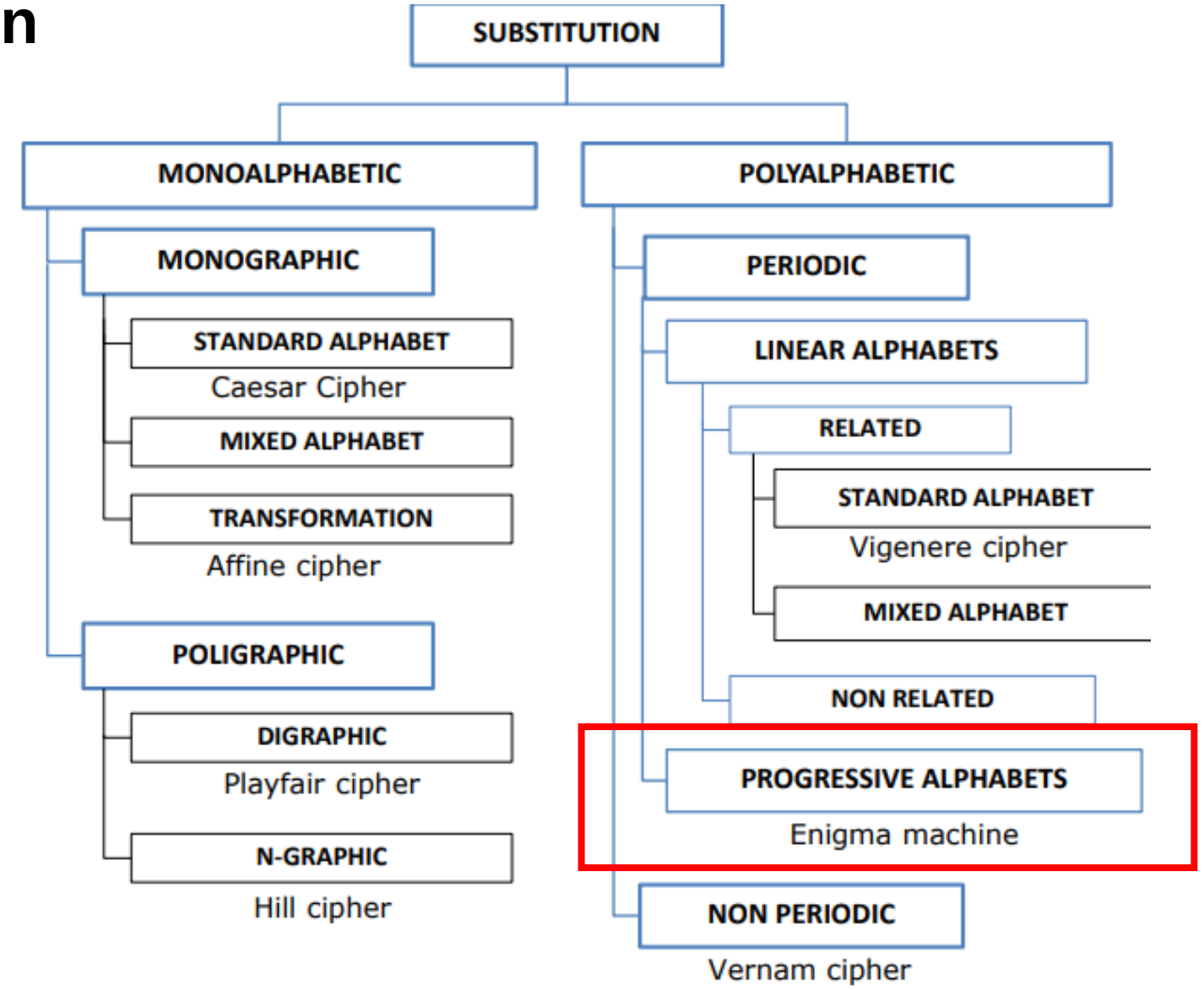
Vigenere

Ejemplo

Texto plano	***M***	***E***	*N*	*S*	*A*	*J*	*E*	*U*	*N*	*O*
Clave	***C***	***L***	*A*	*V*	*E*	*C*	*L*	*A*	*V*	*E*
Texto cifrado	***O***	***P***	*?*	*?*	*?*	*?*	*?*	*?*	*?*	*?*

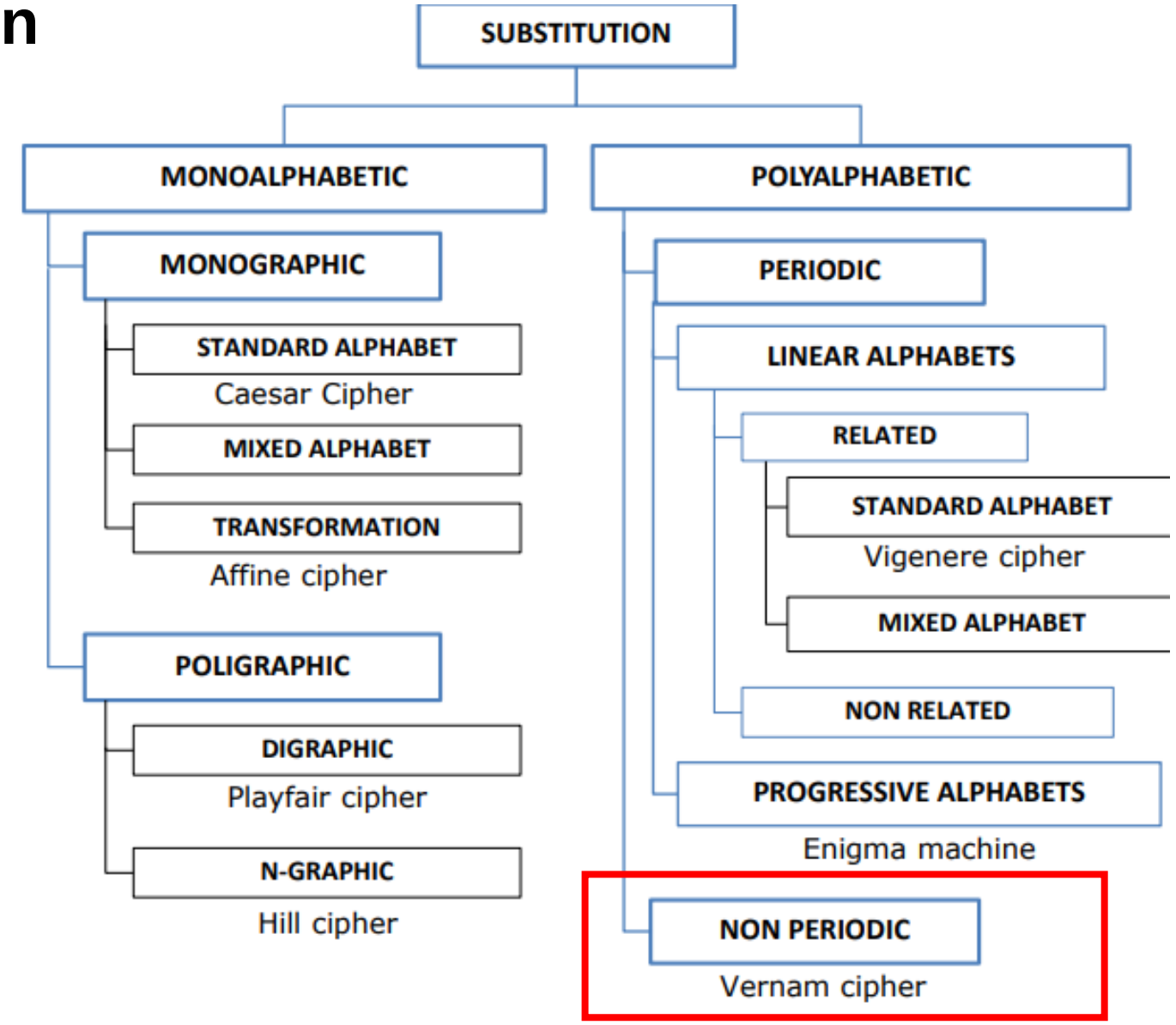
Criptografía Clásica

- **Clasificación**



Criptografía Clásica

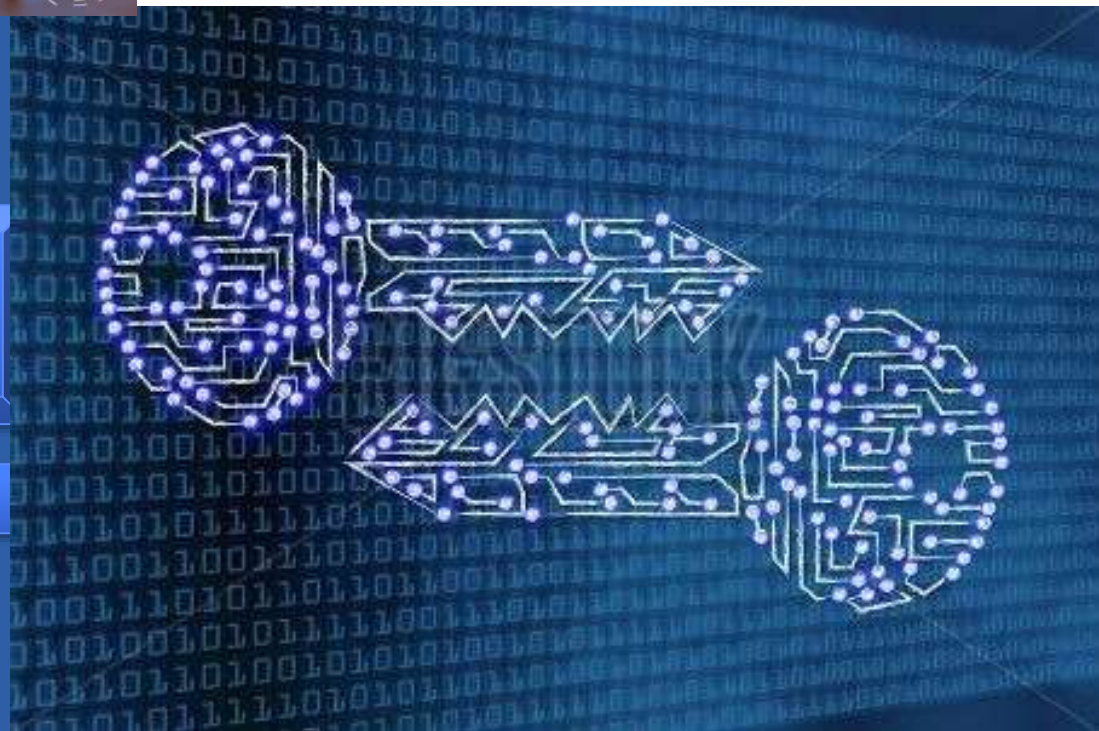
- **Clasificación**



Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 3
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**

.

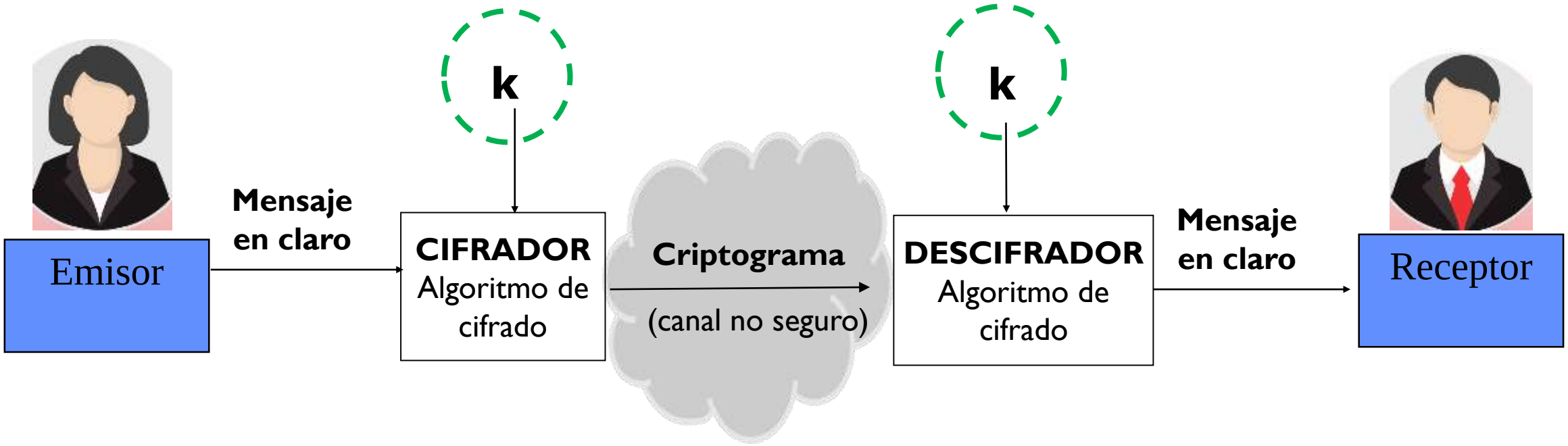
Criptografía Moderna

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

.

Modelo de Criptosistema

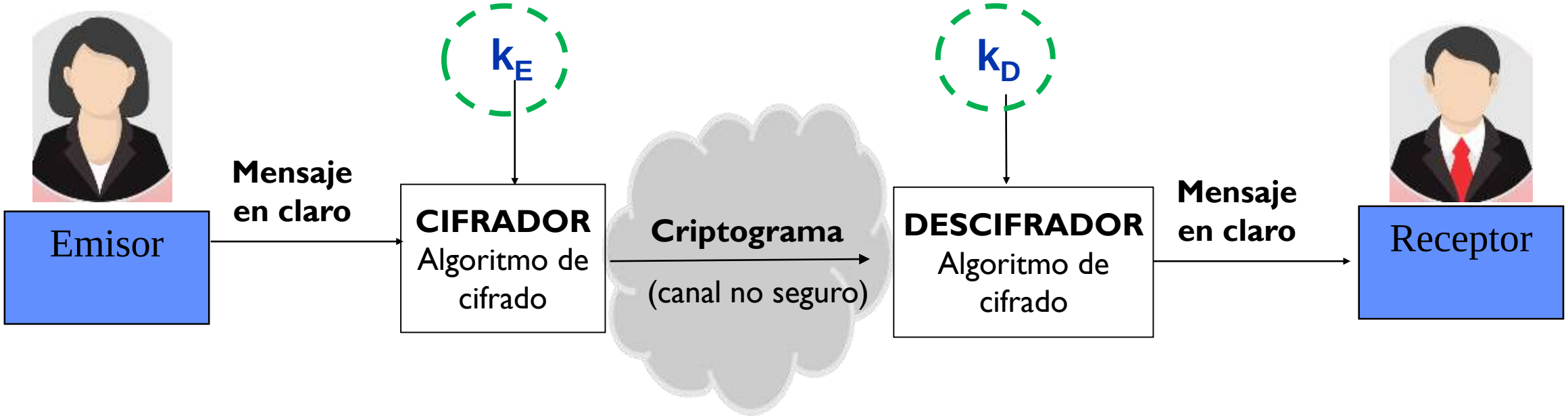
SIMÉTRICOS: Misma clave (k) para cifrar y descifrar



- 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

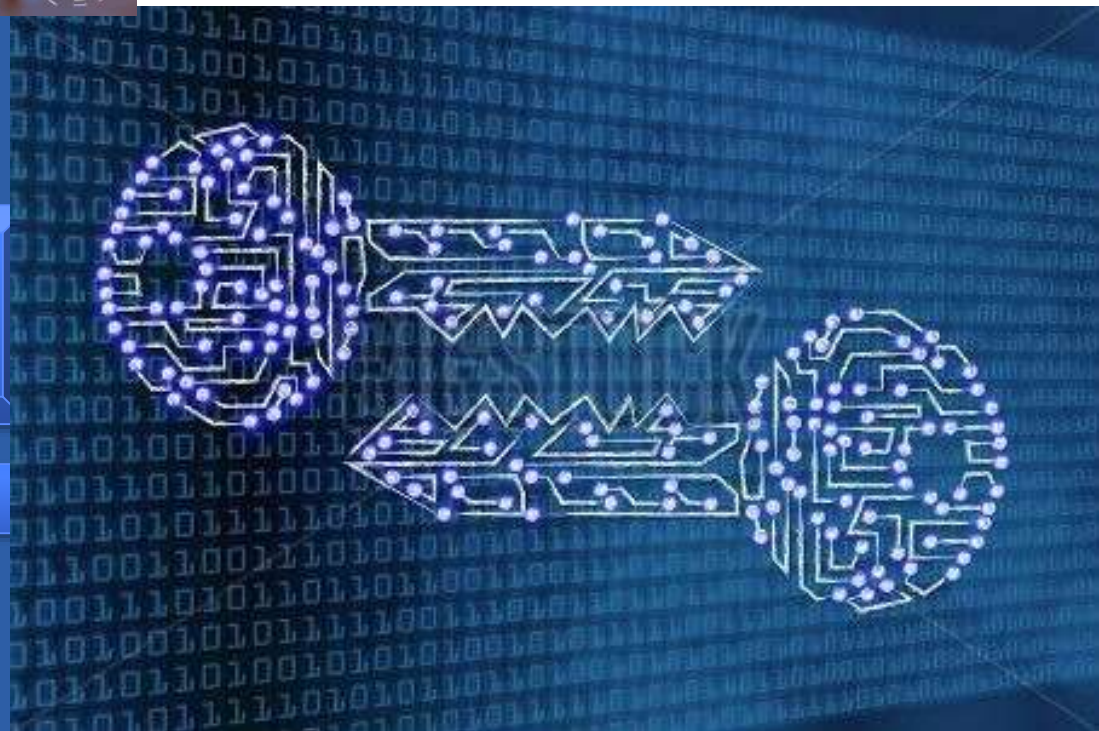
- **Por la cantidad de elementos tomados a la vez:**
  - **Cifradores de flujo**
  - **Cifradores de bloque**

.

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

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)**
  - **…**
- **Ejemplos**: Vernam, RC4, etc.

.

Cifradores de flujo

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

.

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

- **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

- **Generación del keystream**
  - **En la práctica se emplean generadores pseudo-aleatorios**
    - **Semilla inicial (clave compartida, usada sólo una vez)**
    - **Algoritmo determinista (misma entrada genera misma salida)**
      - **Útil para que el transmisor y el receptor se entiendan**
  - **Ejemplo: LFSR**

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

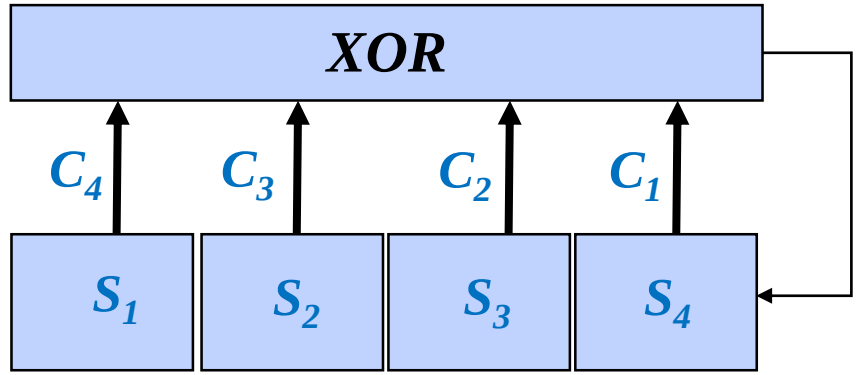
$$Seed = S_n , S_{n.1} , \ldots , S_2 , S_1$$

LFSR *(Linear Feedback Shift Register)*

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

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

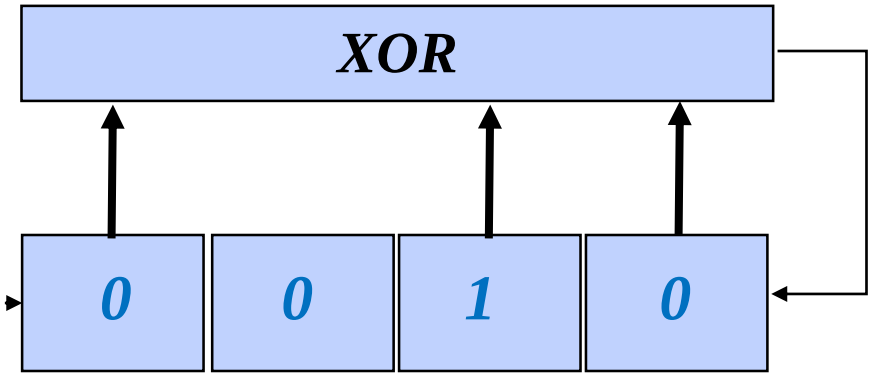
$$Seed = (S1, S2, S3, S4)$$



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

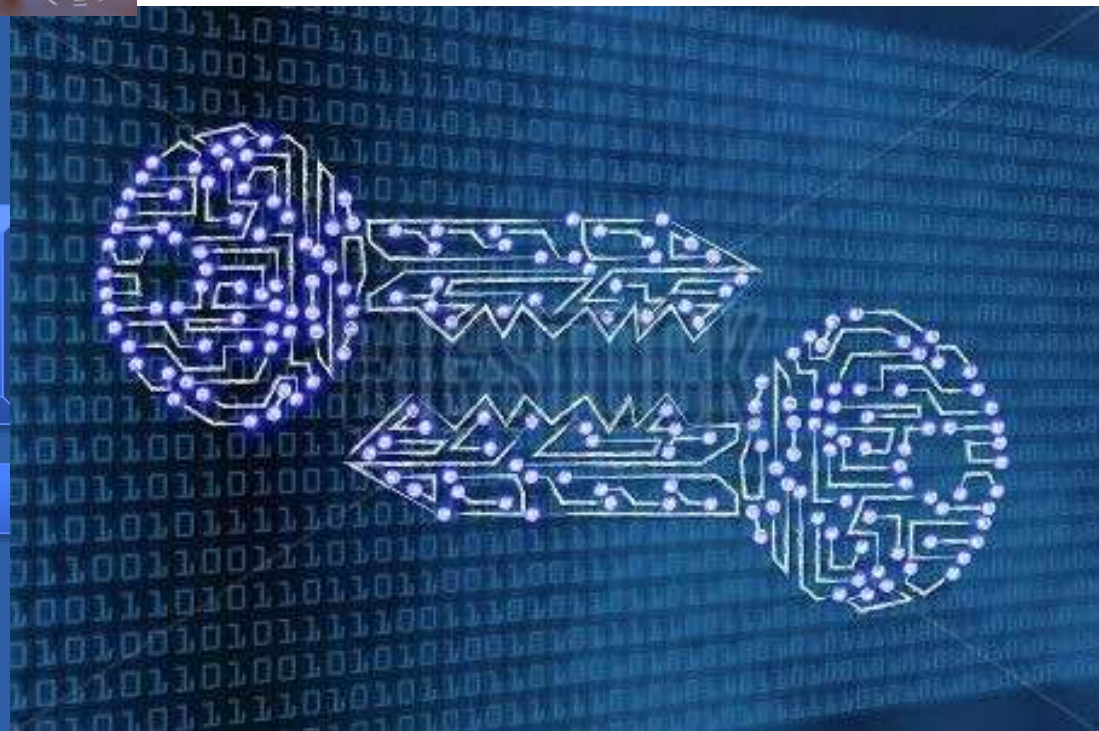
- **VENTAJAS**:
  - Muy rápidos
  - Ideales para prestaciones que requieran tiempo real
    - Ejemplo: streaming de video / audio
  - Los errores de transmisión no se propagan a otros símbolos

- **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…)

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.

.

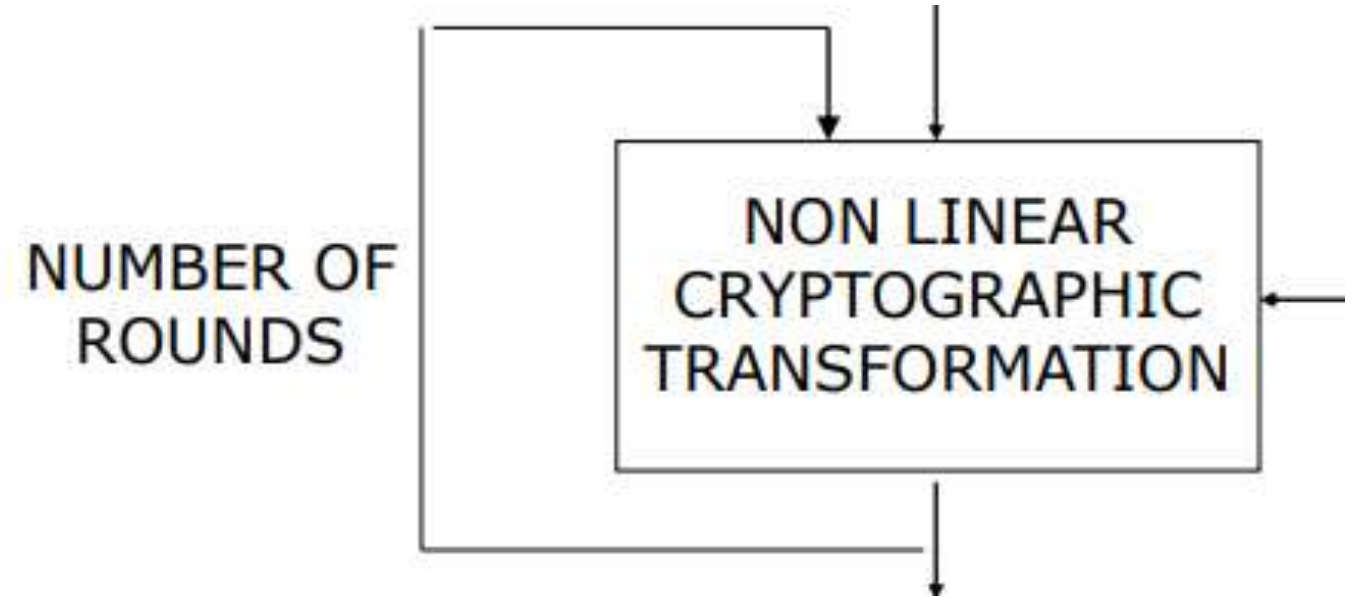
Cifradores de bloque

- **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

Cifradores de bloque

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

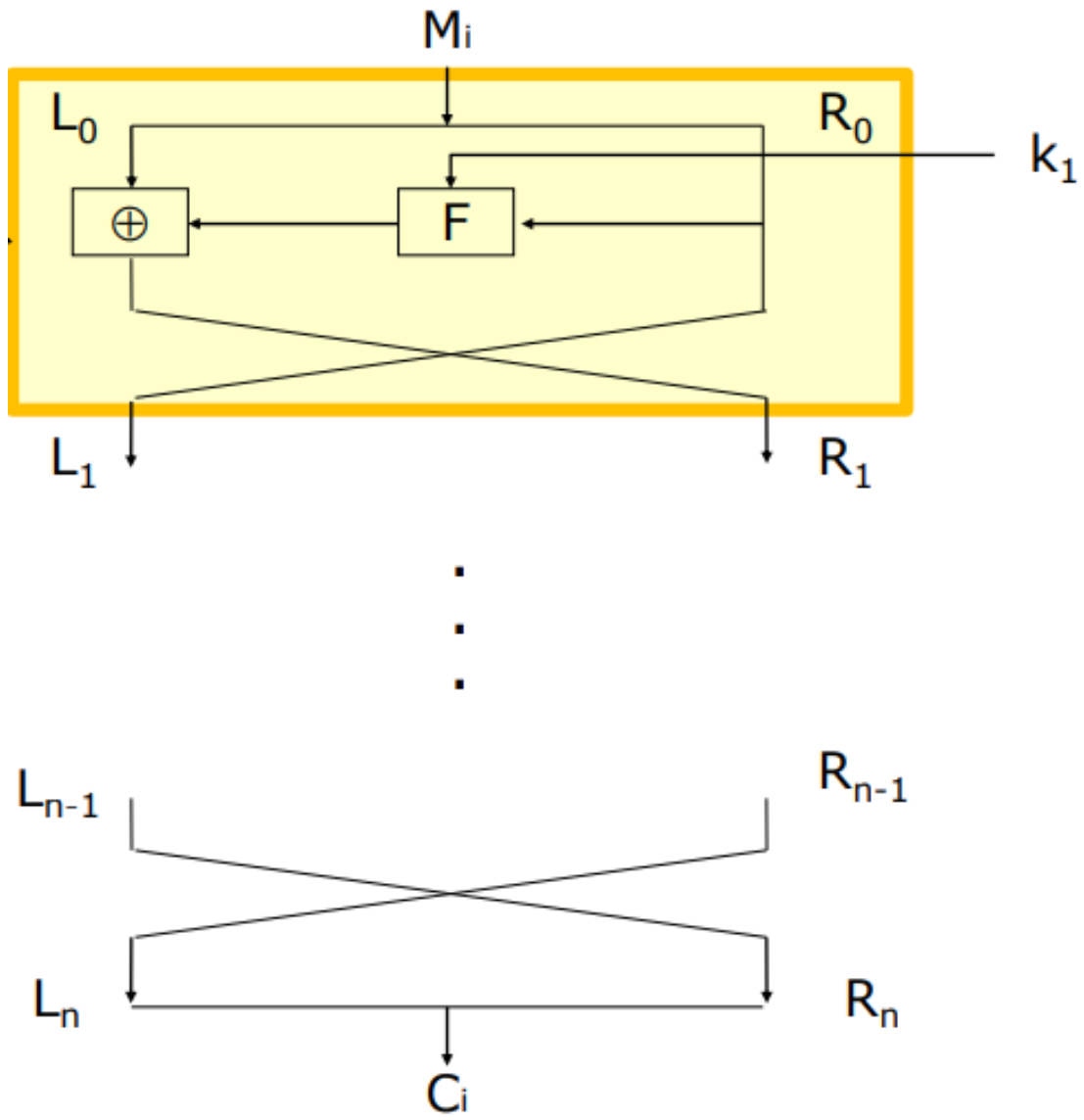
.



Esquema de Feistel

- **Procedimiento del esquema de Feistel:**

1. **Divide** el bloque M en dos mitades, $\mathbf{L_0}$ y $\mathbf{R_0}$
2. **Repite** los siguientes pasos durante **n rondas**. En cada ronda "i":
   1. Aplica una **función F** sobre la mitad derecha ($\mathbf{R_i}$) 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

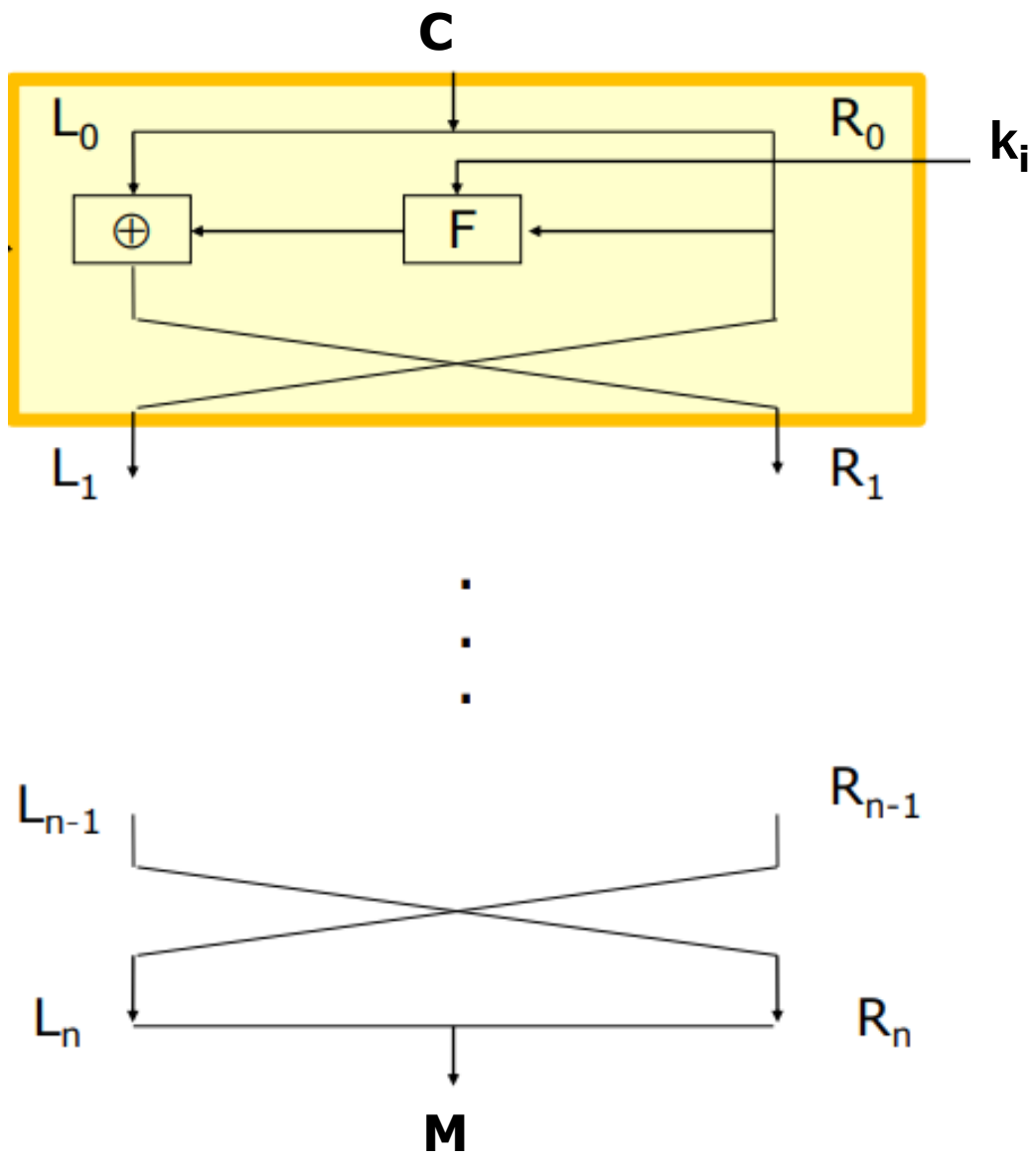


Esquema de Feistel

- **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:**

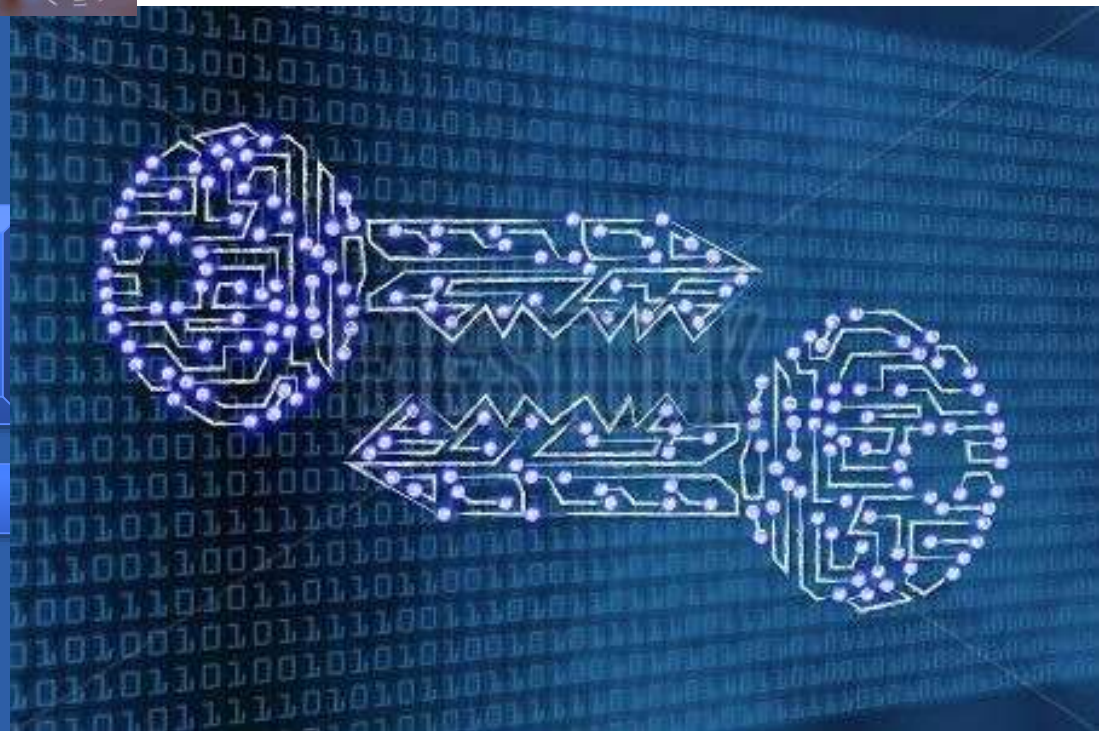
  1. Diseño de una buena función F
  2. Diseño de una buen algoritmo de generación de subclaves

- **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

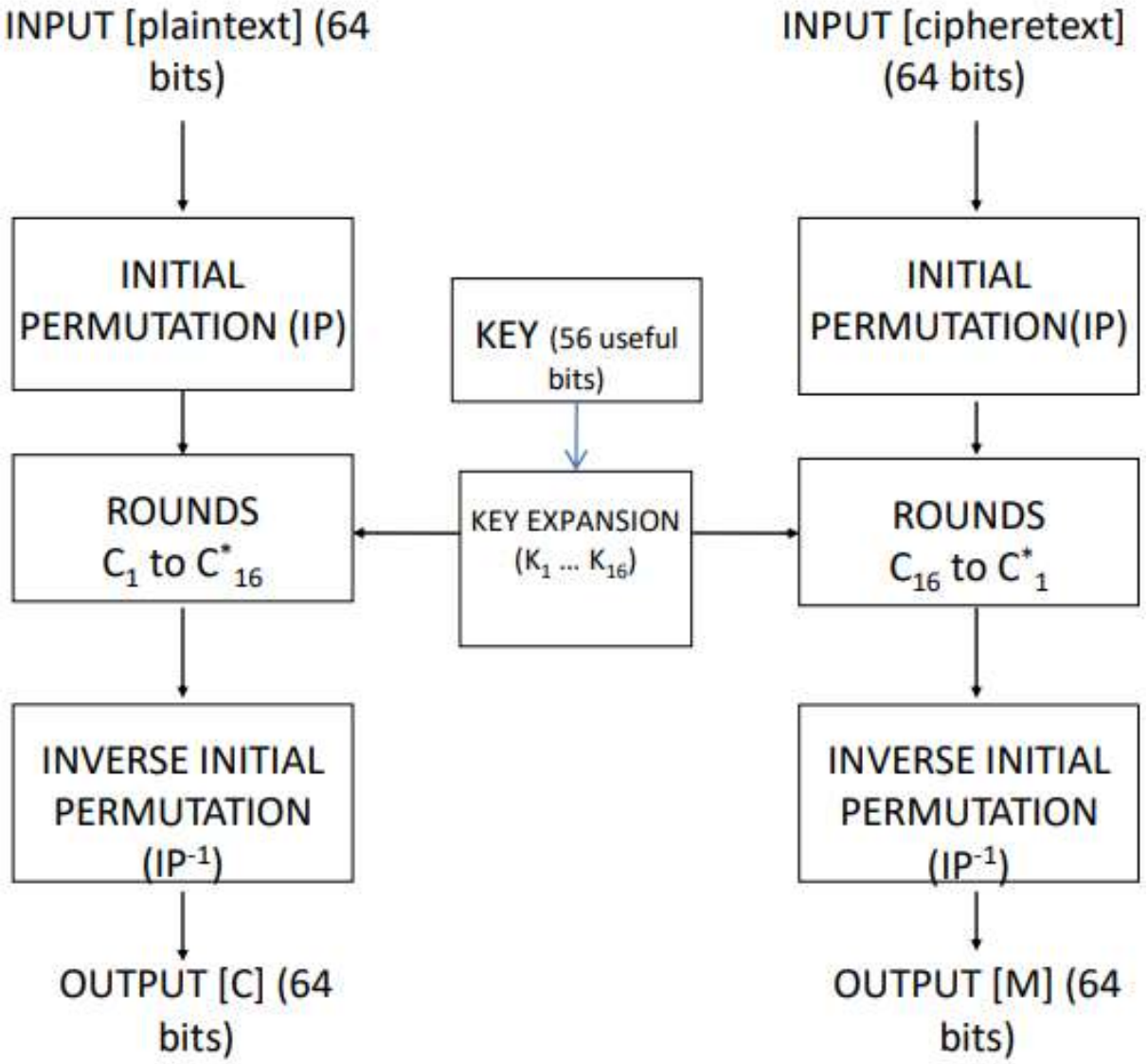
Universidad Francisco de Vitoria
UFV Madrid

Tema 3.2.1
Algoritmos de cifrado simétrico de bloque: DES



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

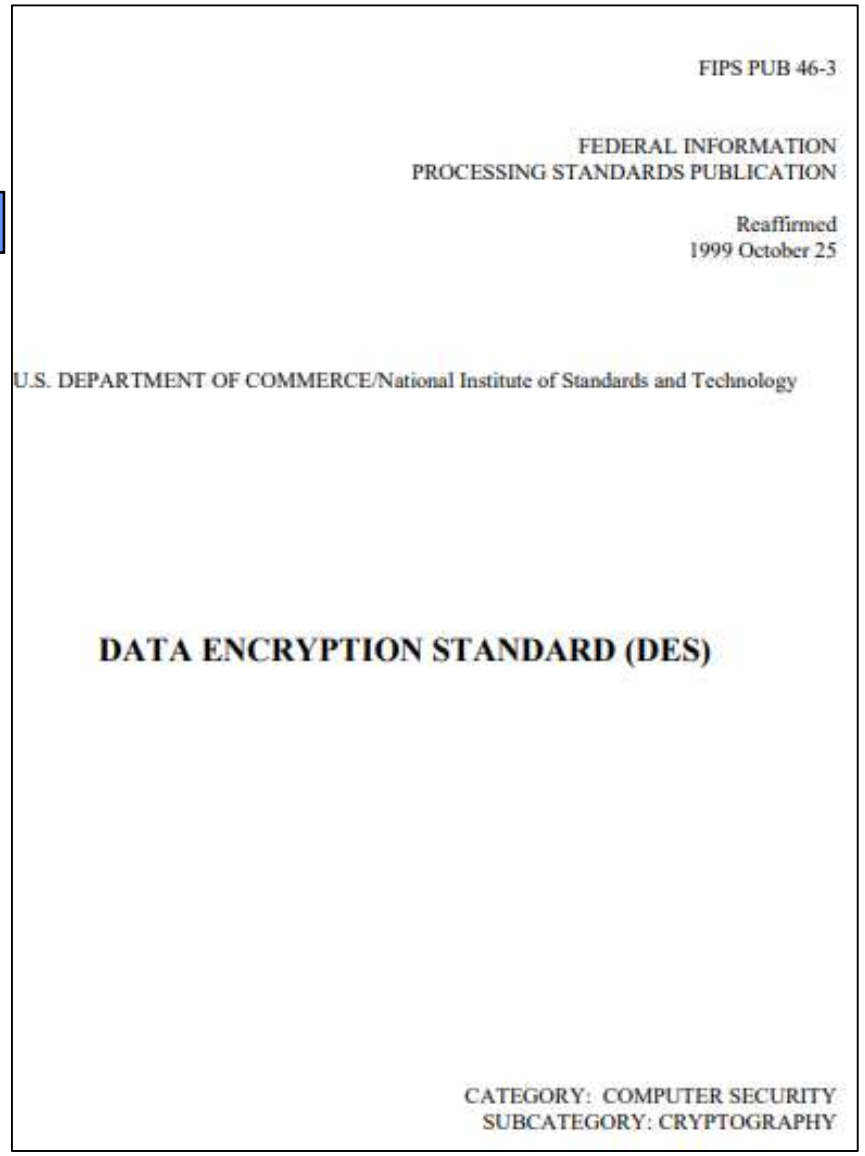
DATA ENCRYPTION STANDARD

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

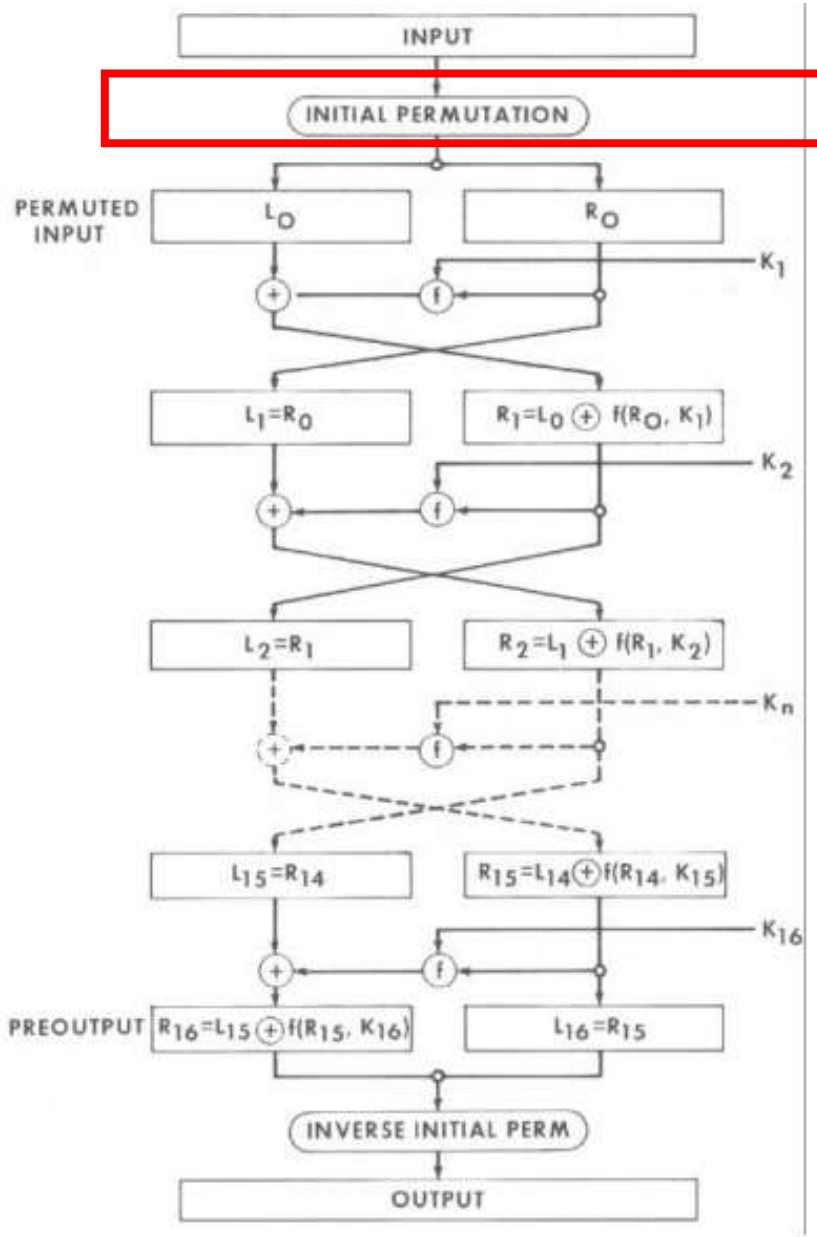
U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and Technology

DATA ENCRYPTION STANDARD (DES)

CATEGORY: COMPUTER SECURITY
SUBCATEGORY: CRYPTOGRAPHY

Data Encryption Standard (DES)

- **Permutación Inicial**

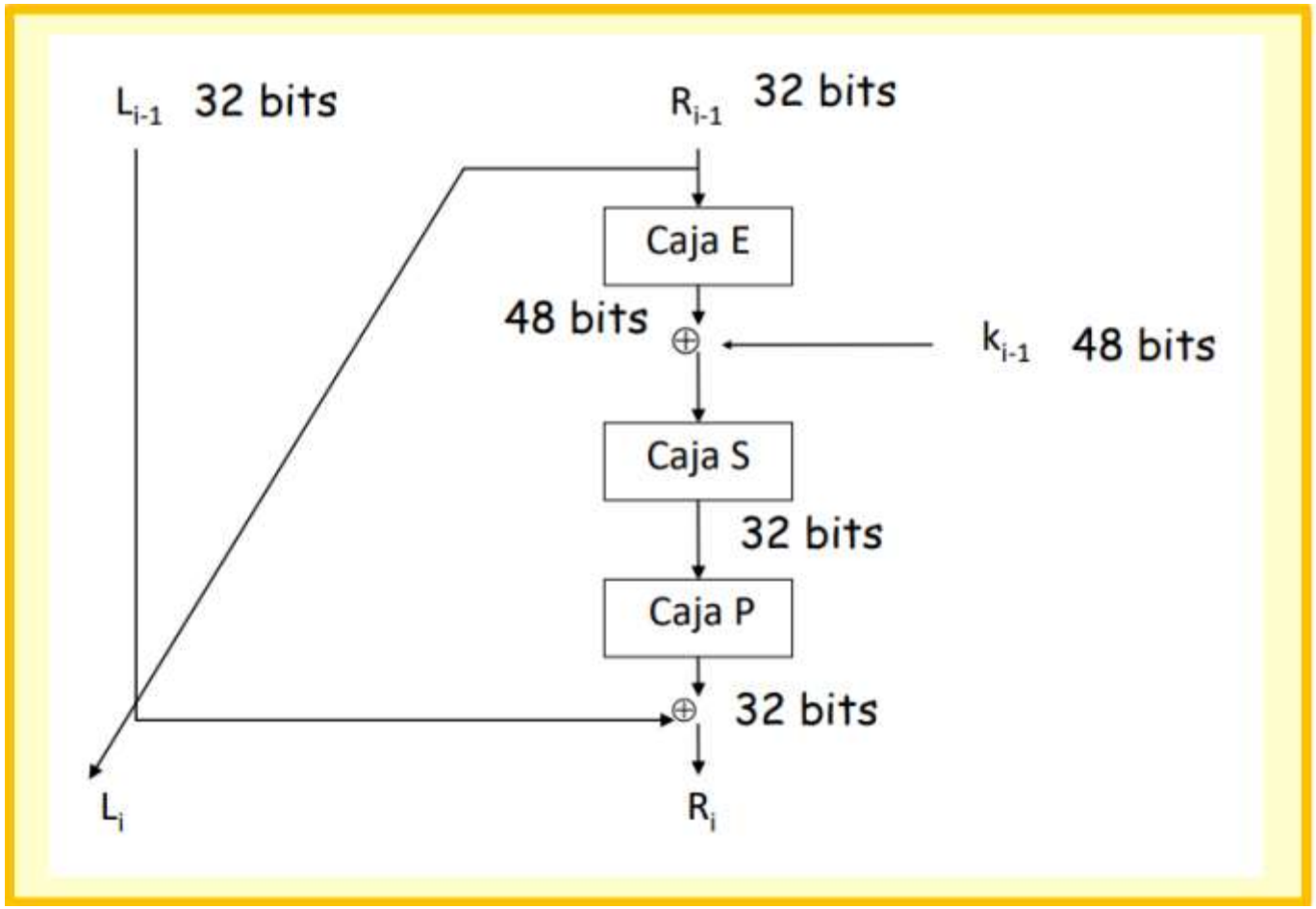


IP

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

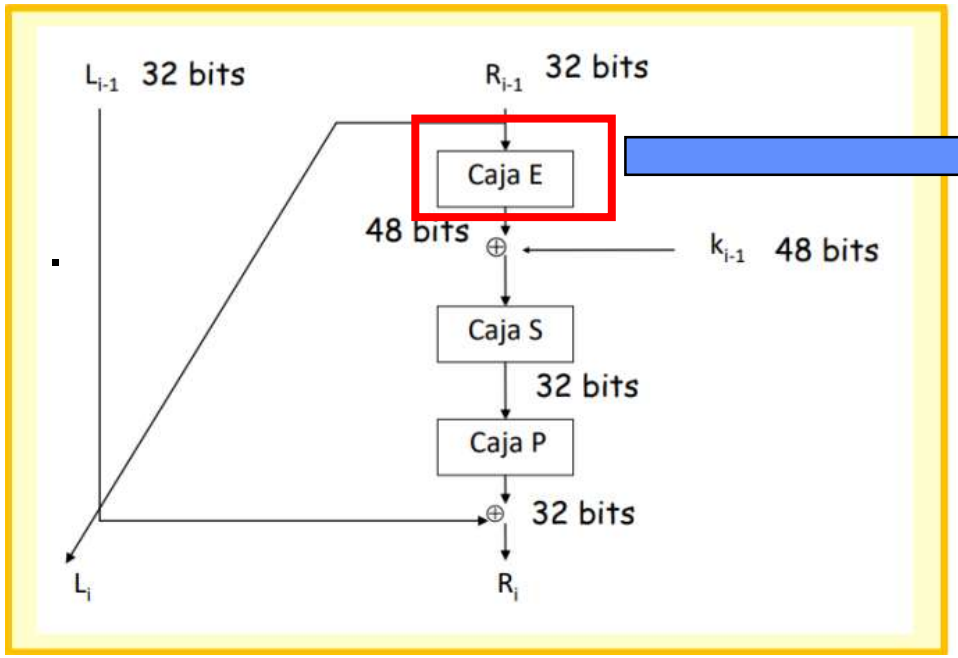
Data Encryption Standard (DES)

- 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**



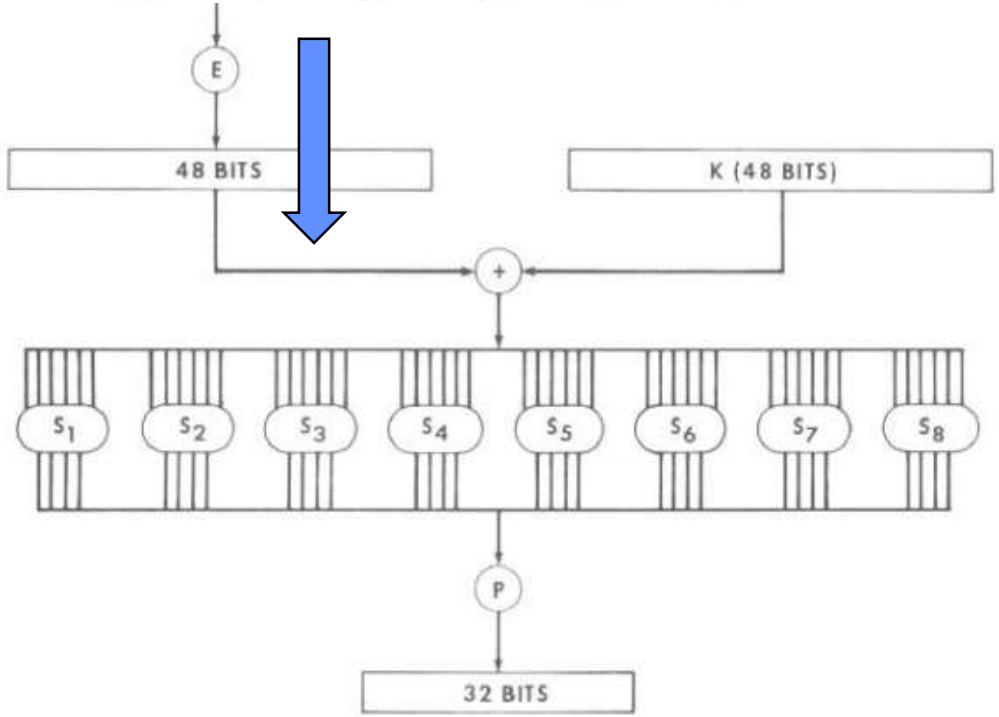
Data Encryption Standard (DES)

- **Función de ronda. Expansión.**



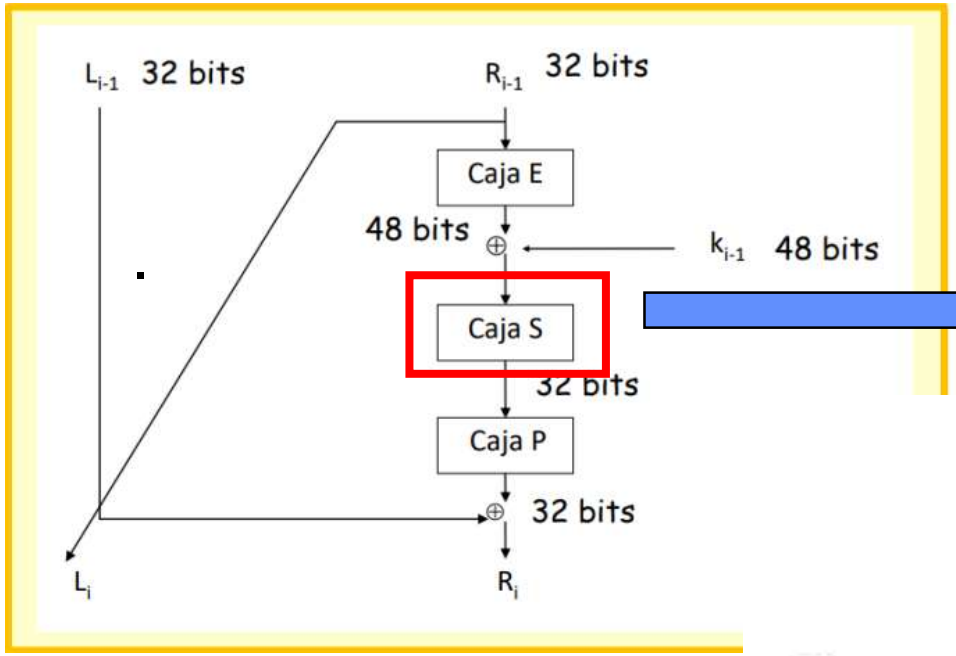
E BIT-SELECTION TABLE

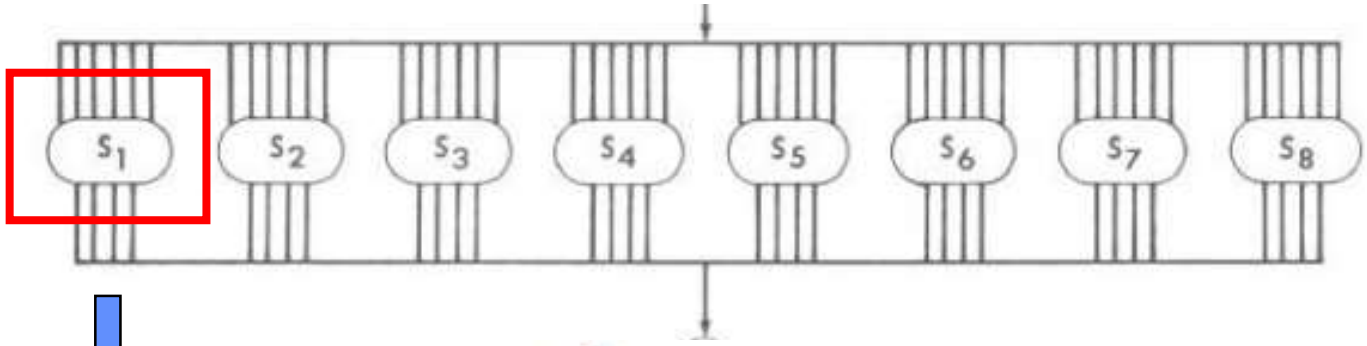
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1



Data Encryption Standard (DES)

- **Función de ronda. Sustitución.**



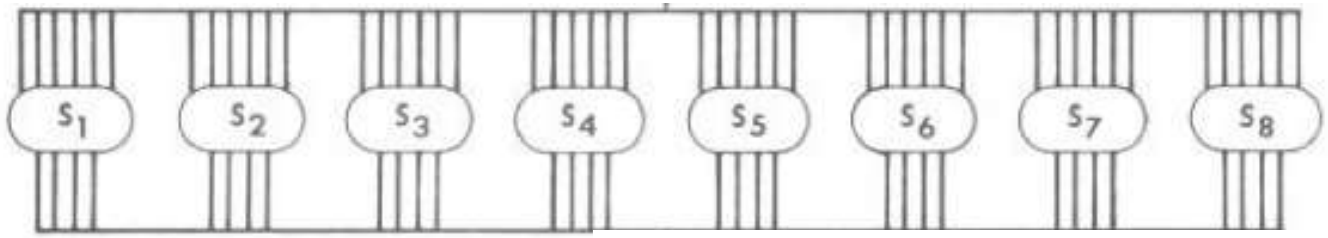


S_1

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

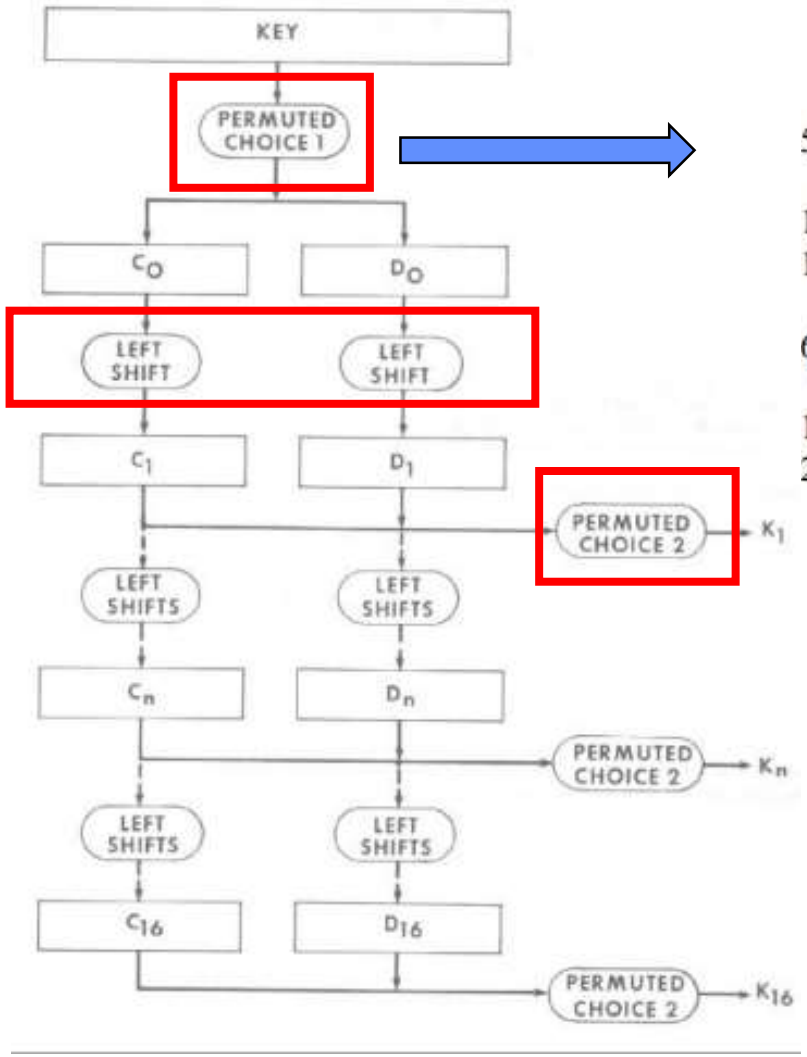
4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

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)

- **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

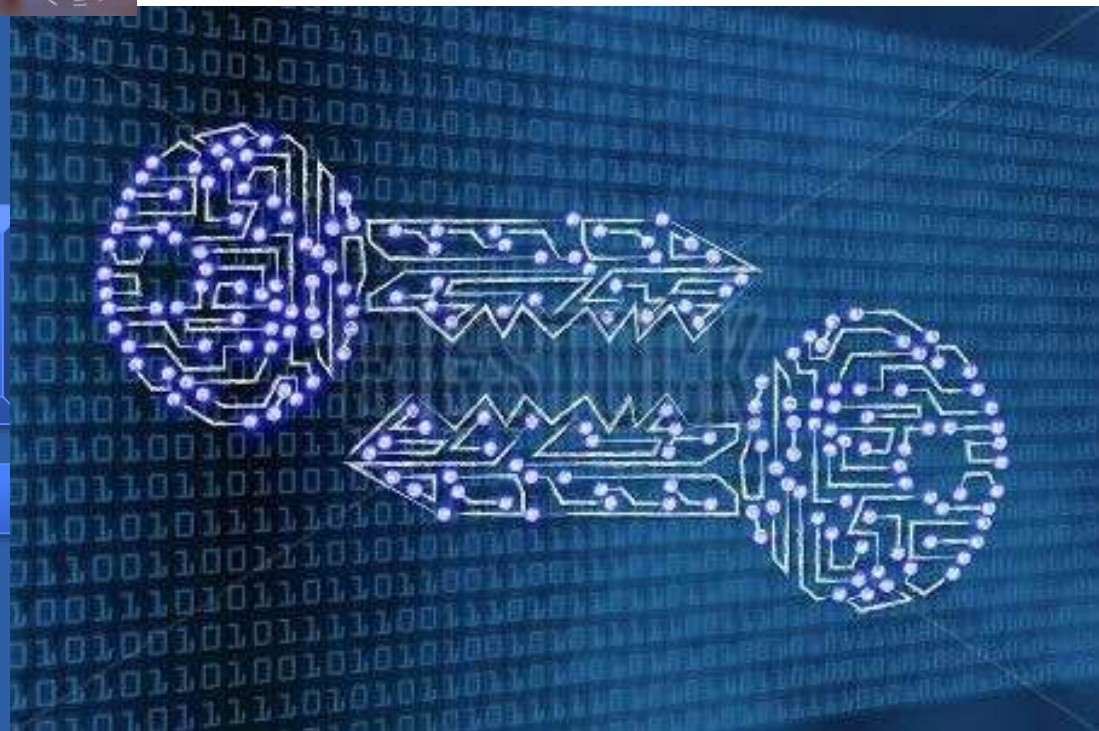
14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

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

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

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.

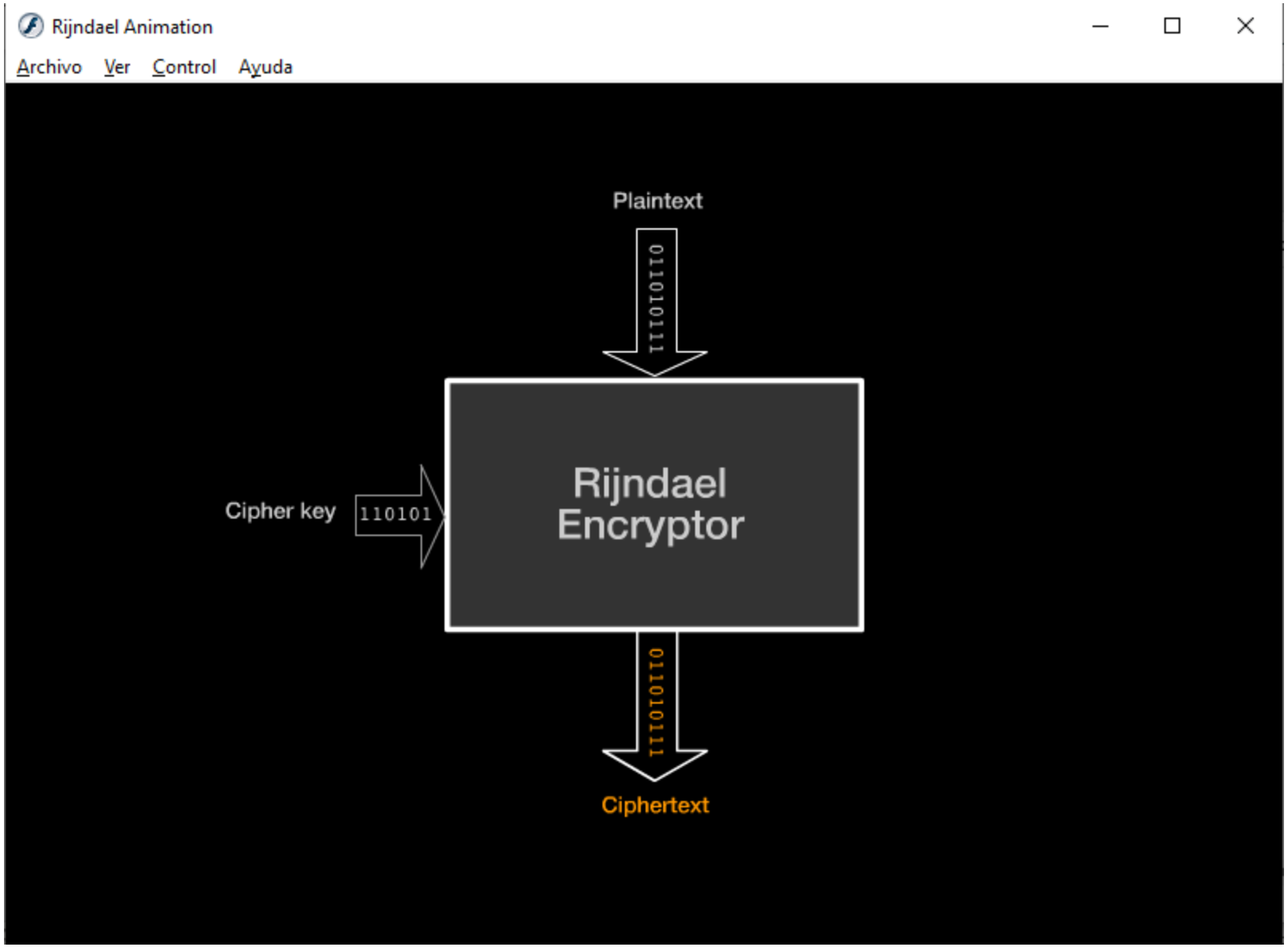
   The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits.

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

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

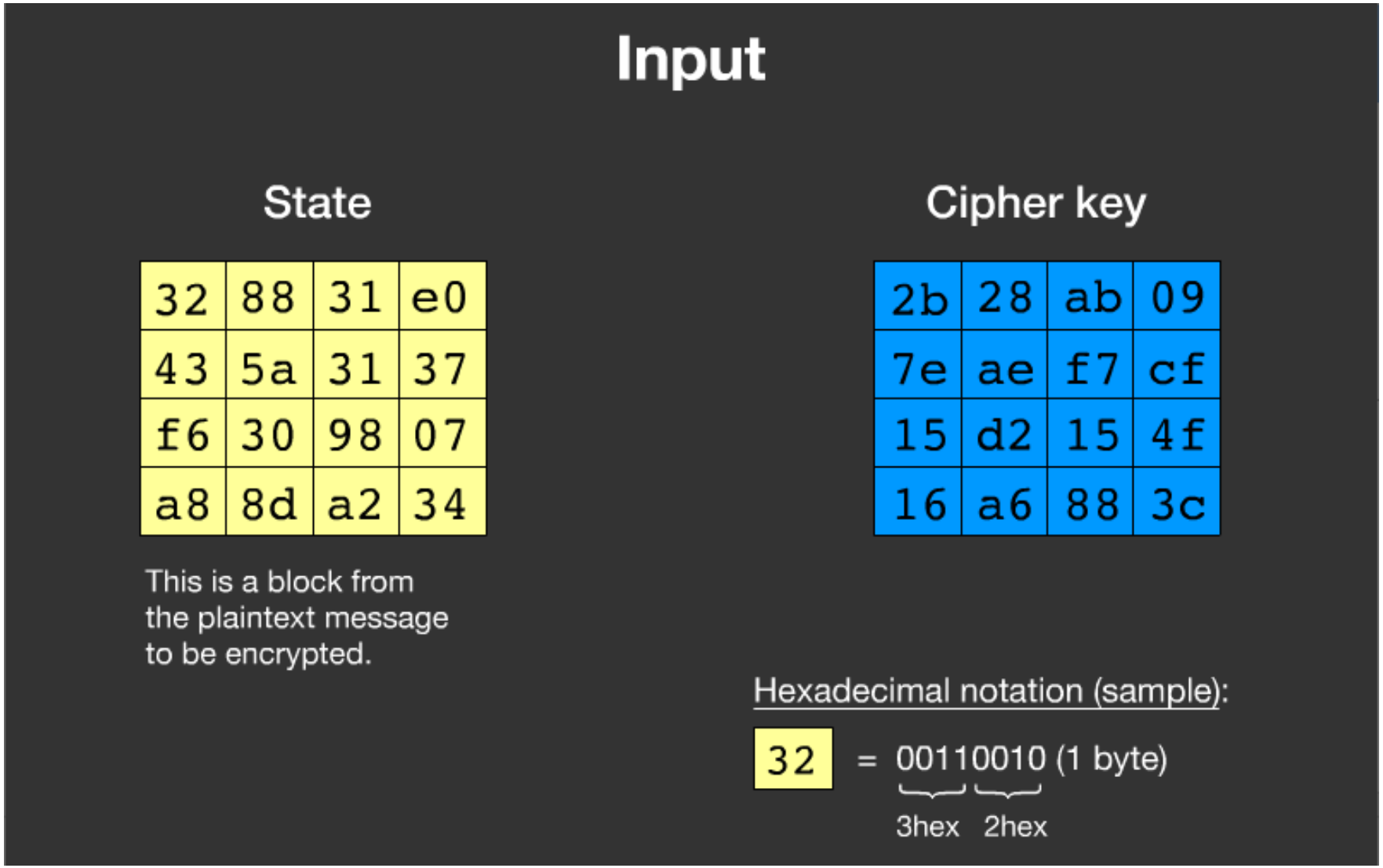
Advanced Encryption Standard (AES)

- **Animation**



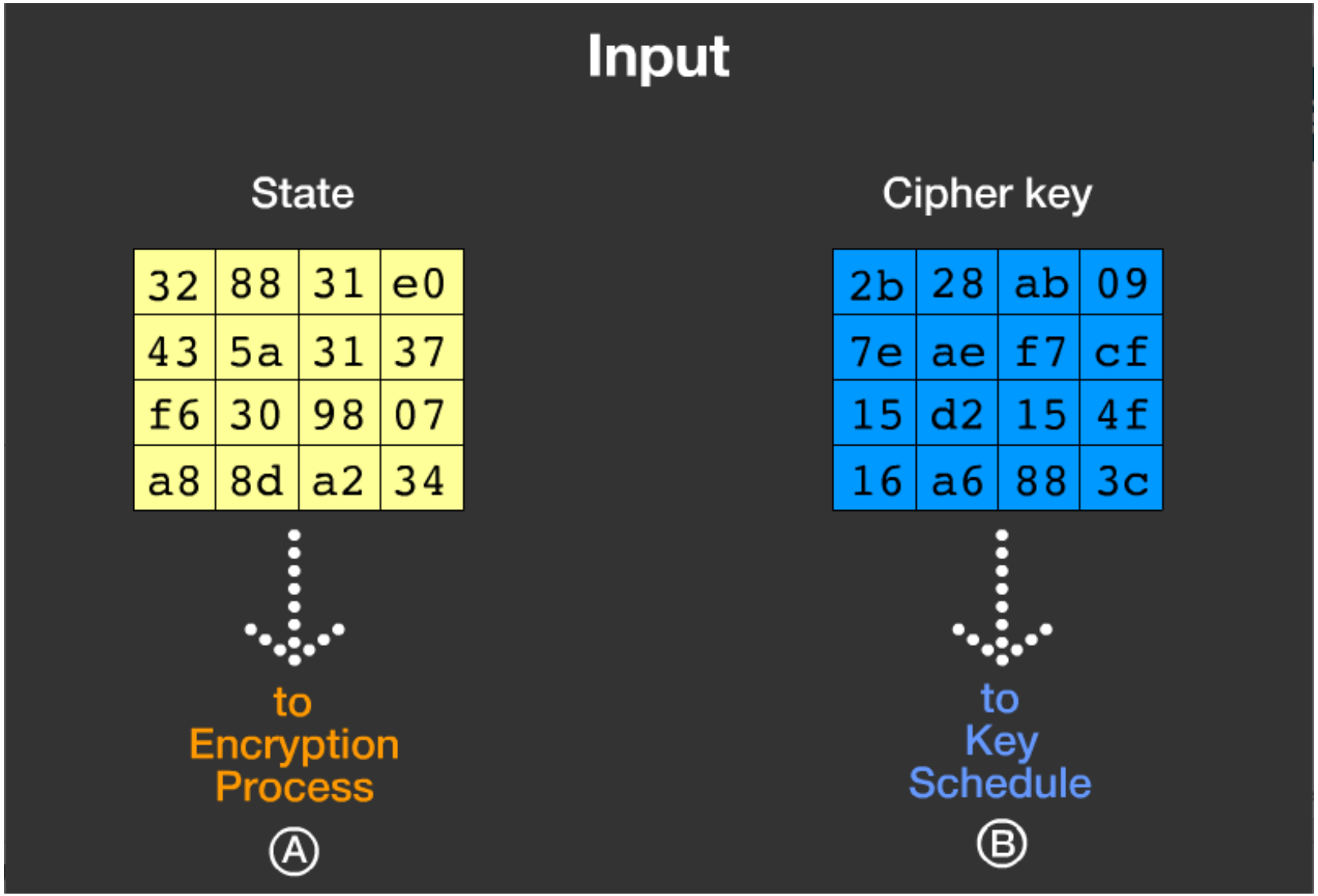
Advanced Encryption Standard (AES)

- **Animation**



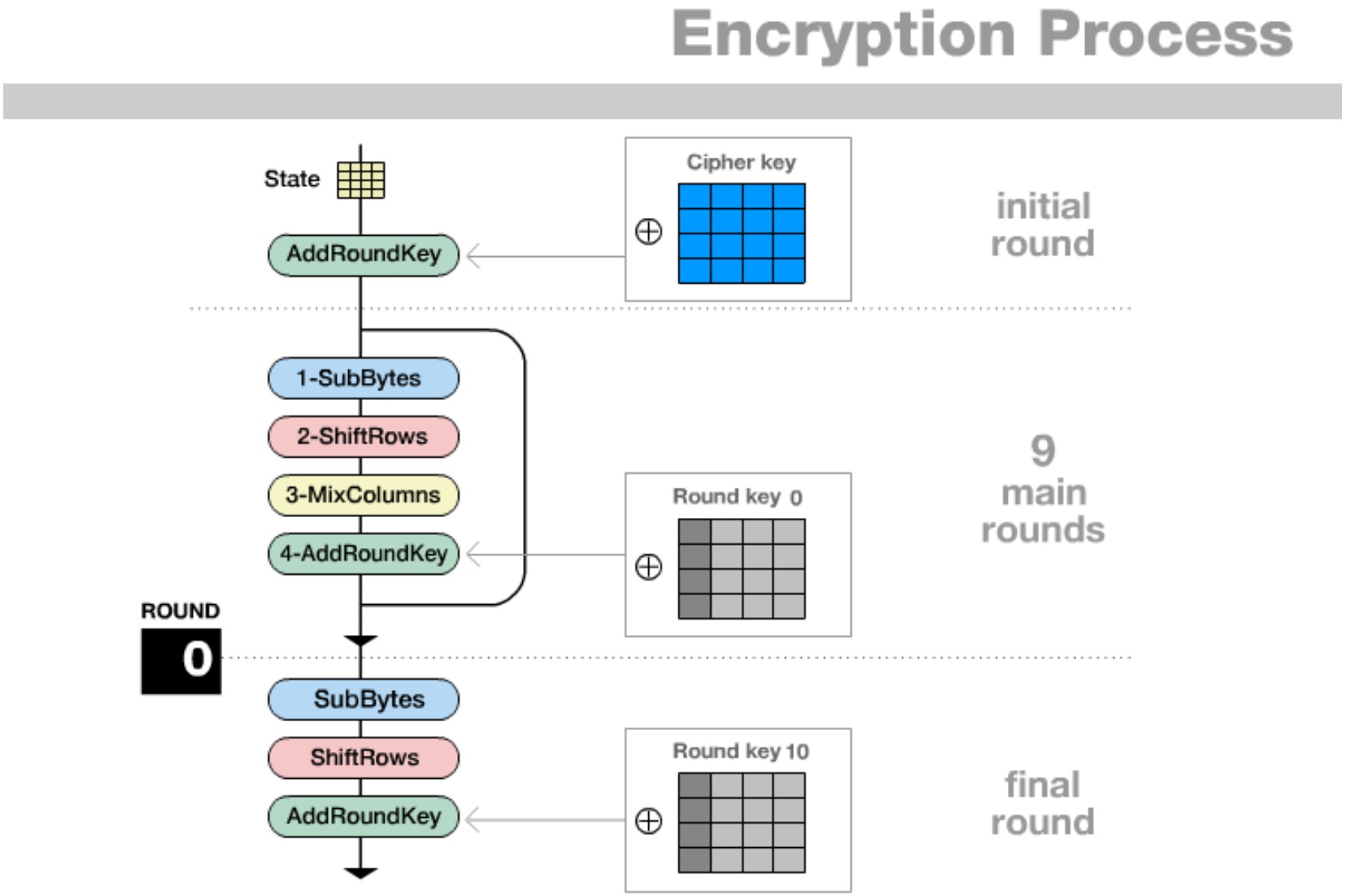
Advanced Encryption Standard (AES)

- **Animation**



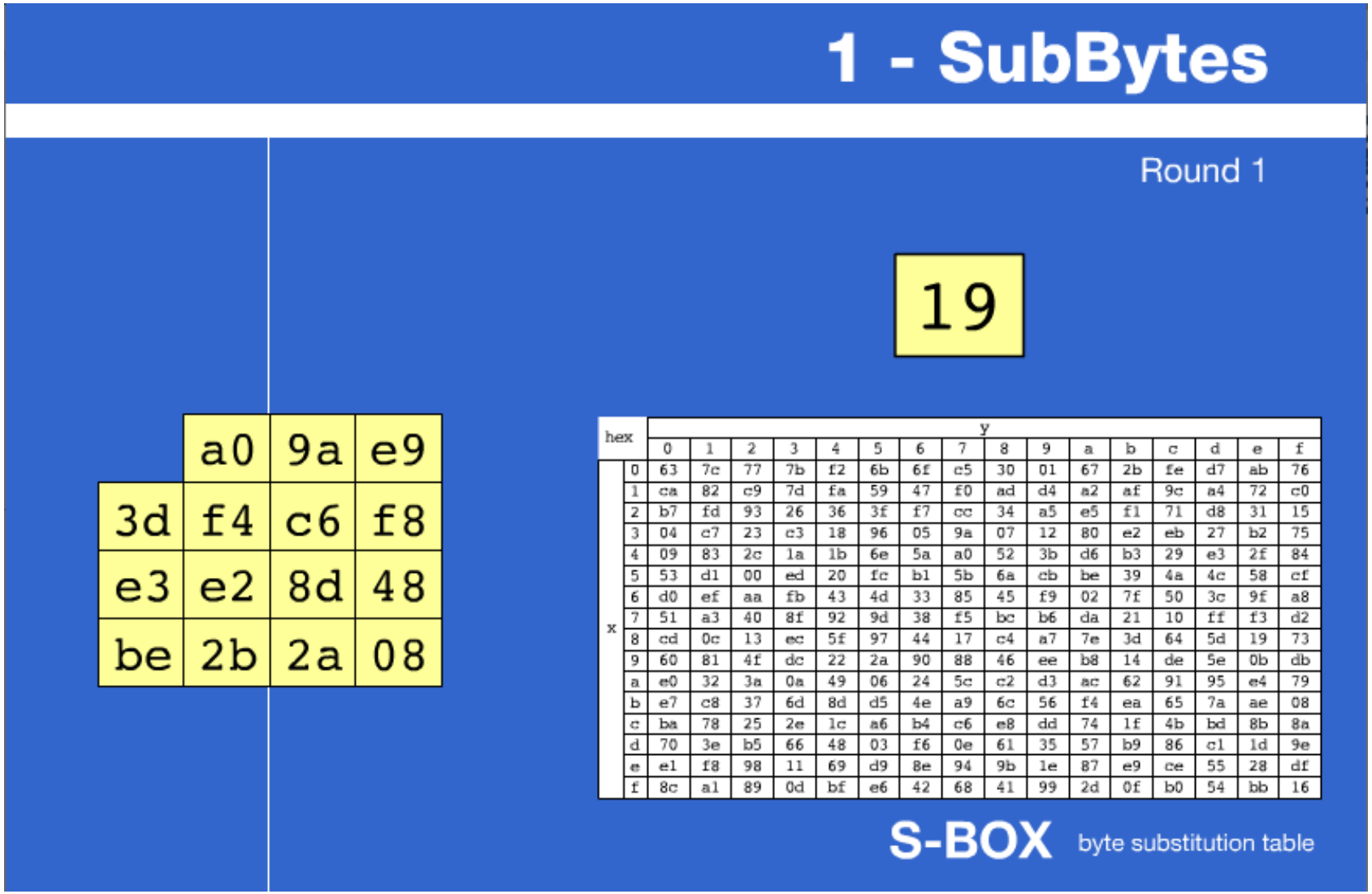
Advanced Encryption Standard (AES)

- **Animation**



Advanced Encryption Standard (AES)

- **Animation**



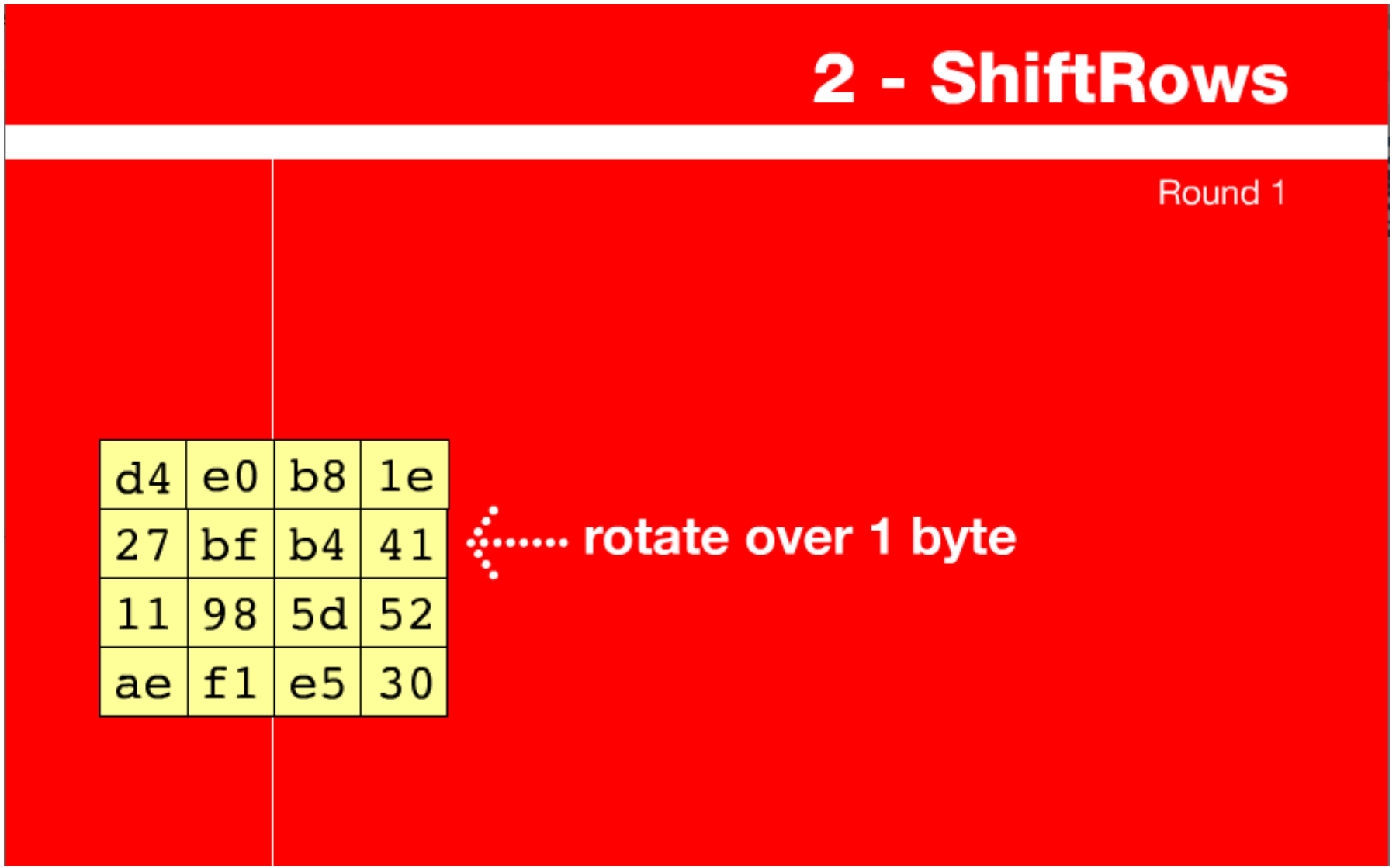
hex	y															
	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	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

x

Advanced Encryption Standard (AES)

- **Animation**

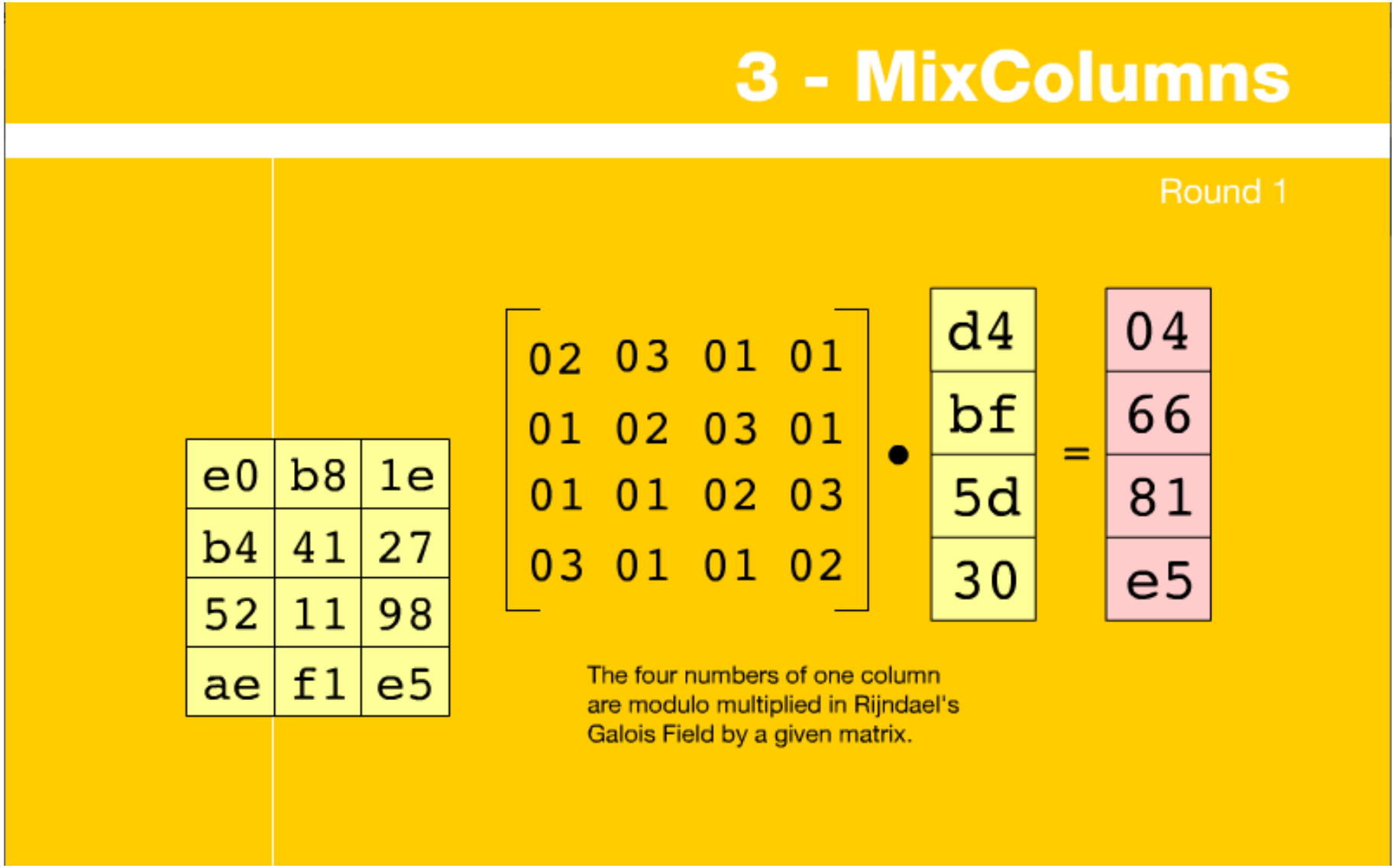
.



Advanced Encryption Standard (AES)

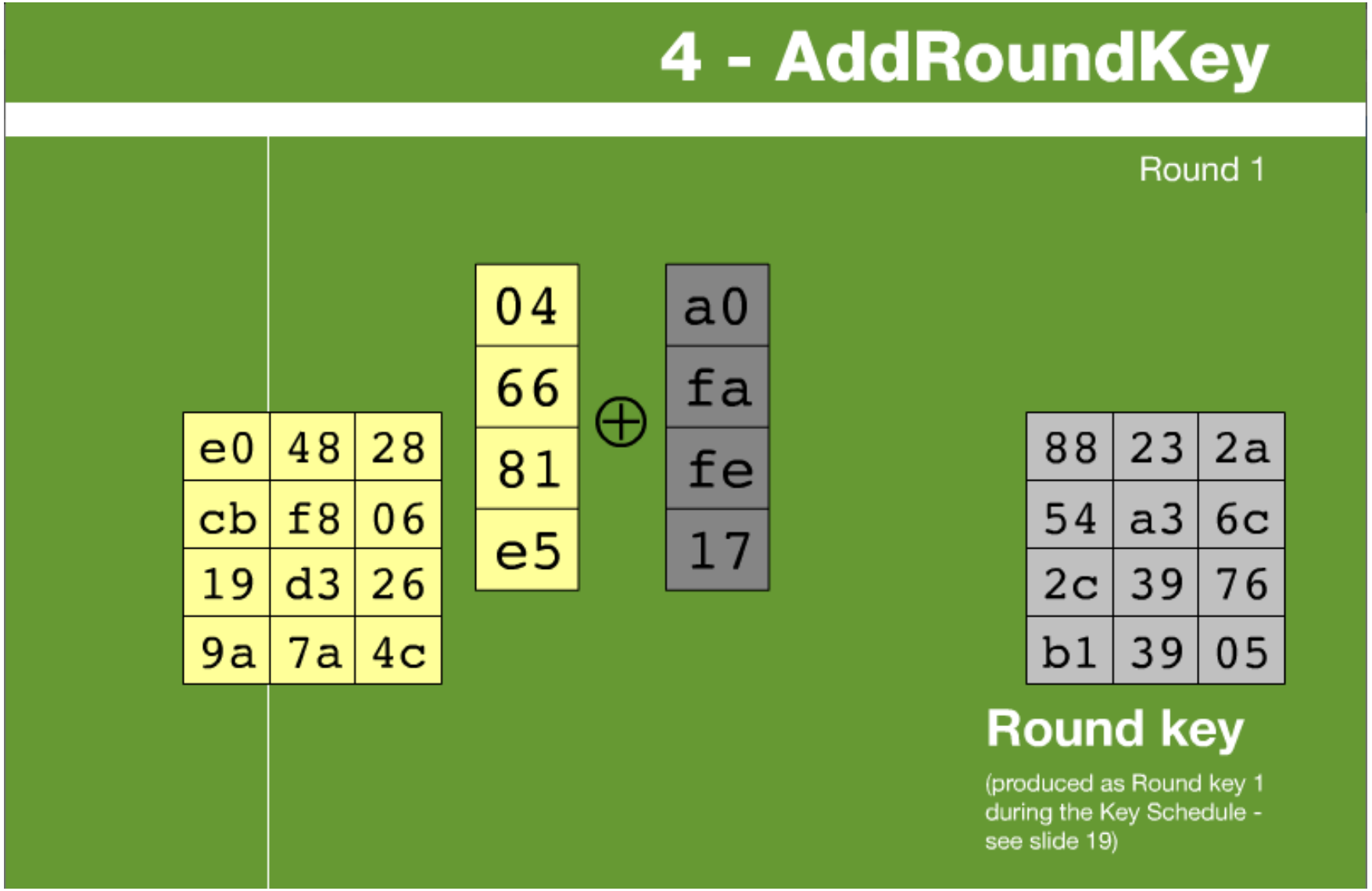
- **Animation**

.



Advanced Encryption Standard (AES)

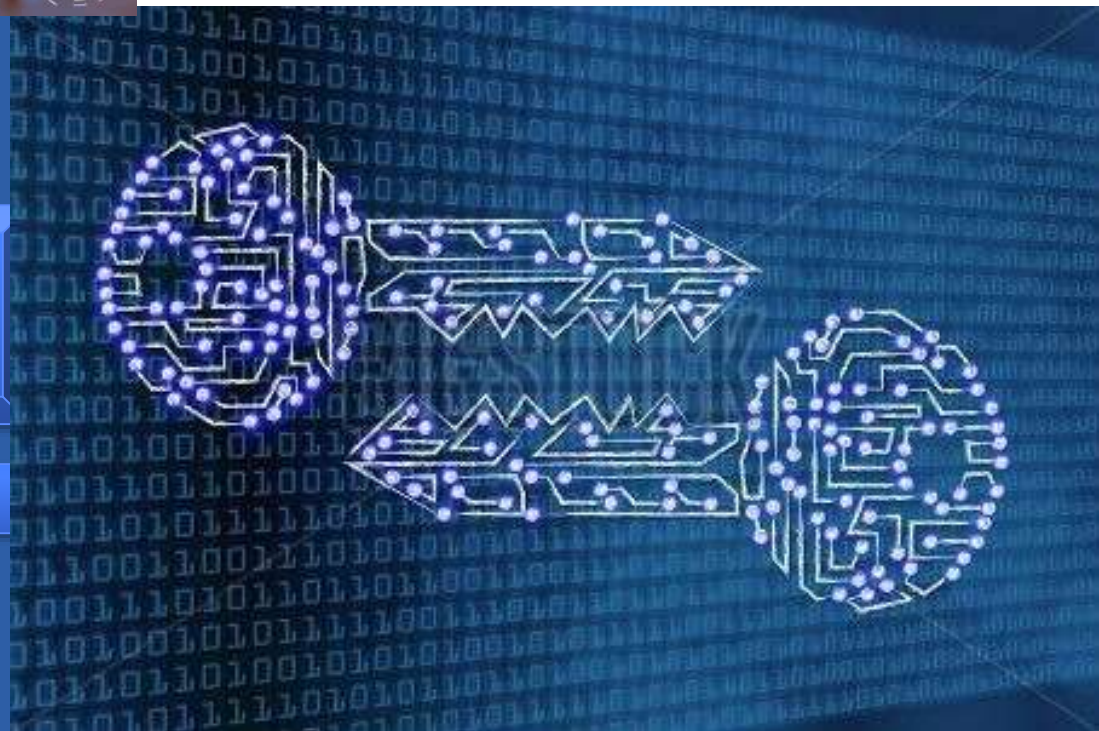
- **Animation**



Criptografía

Universidad Francisco de Vitoria
UFV Madrid

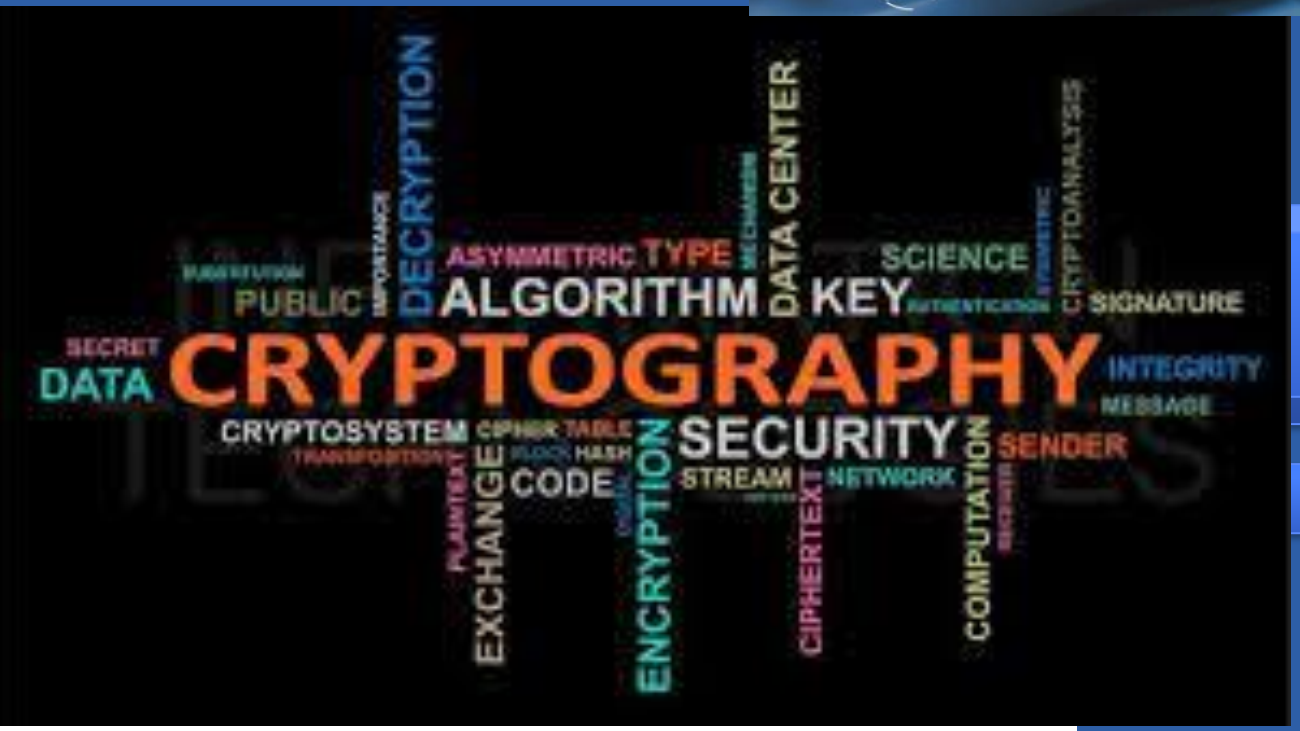
Tema 3
Cifrado Simétrico

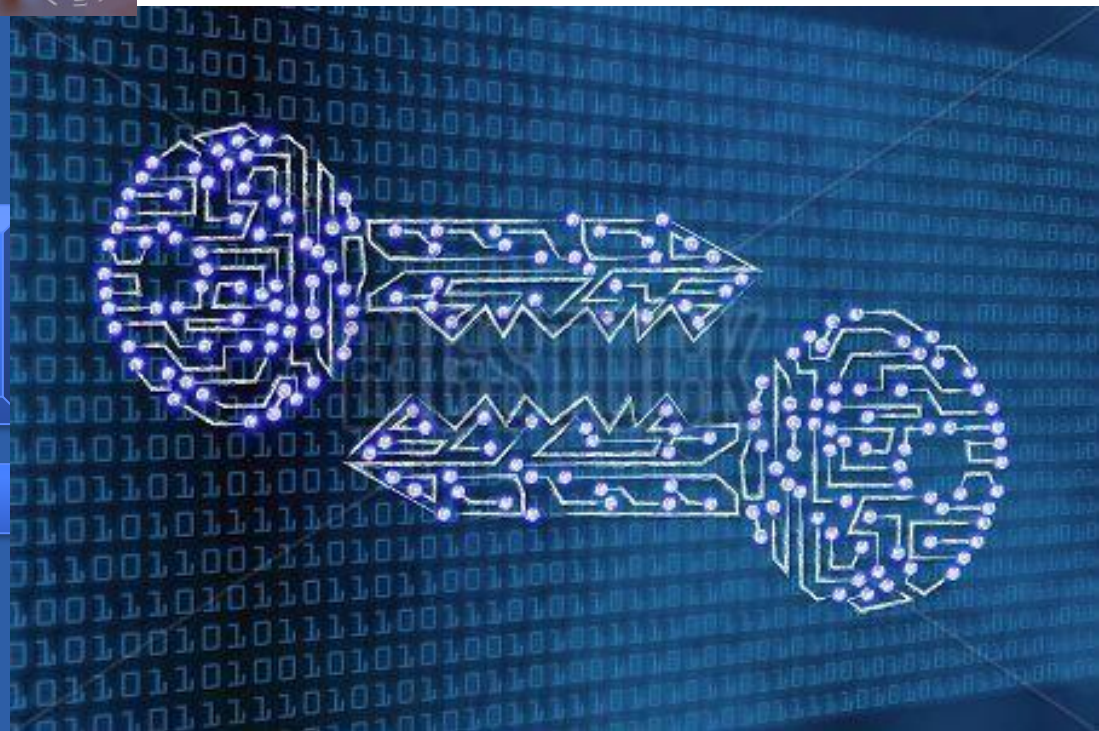


Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 4
Criptografía de Clave Pública

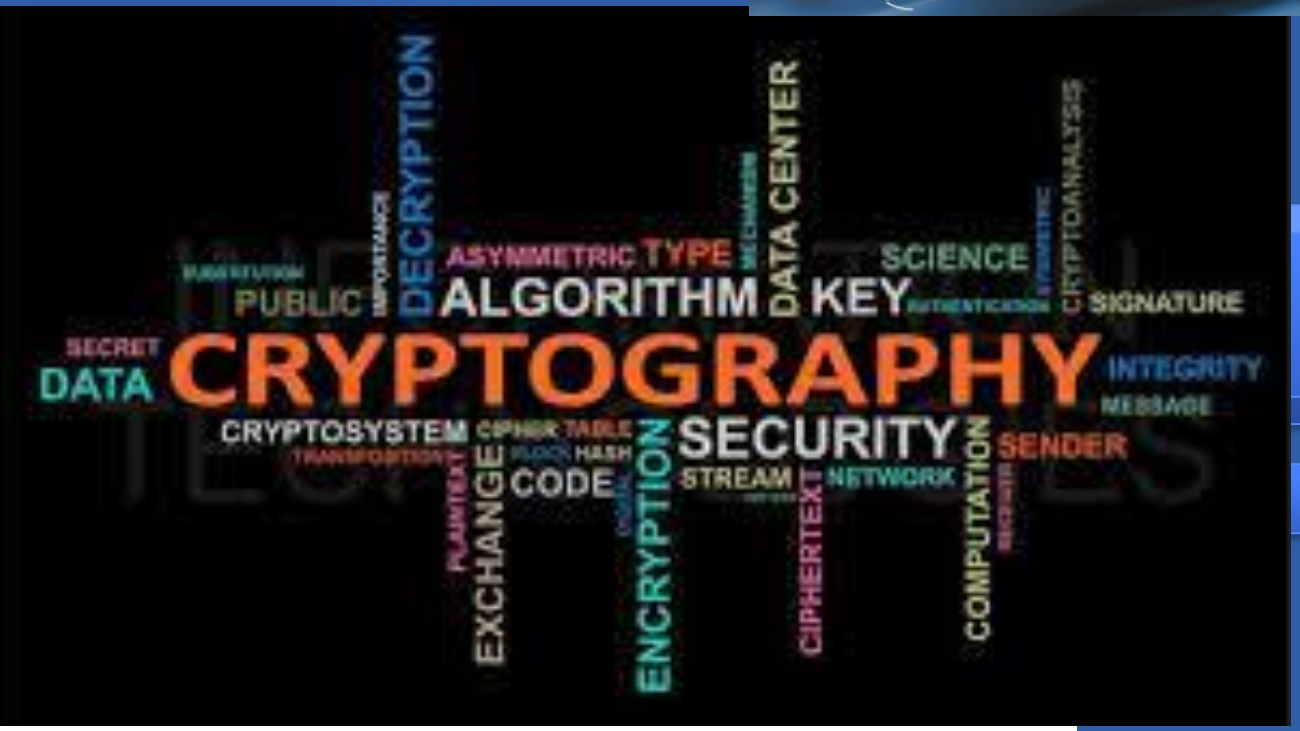


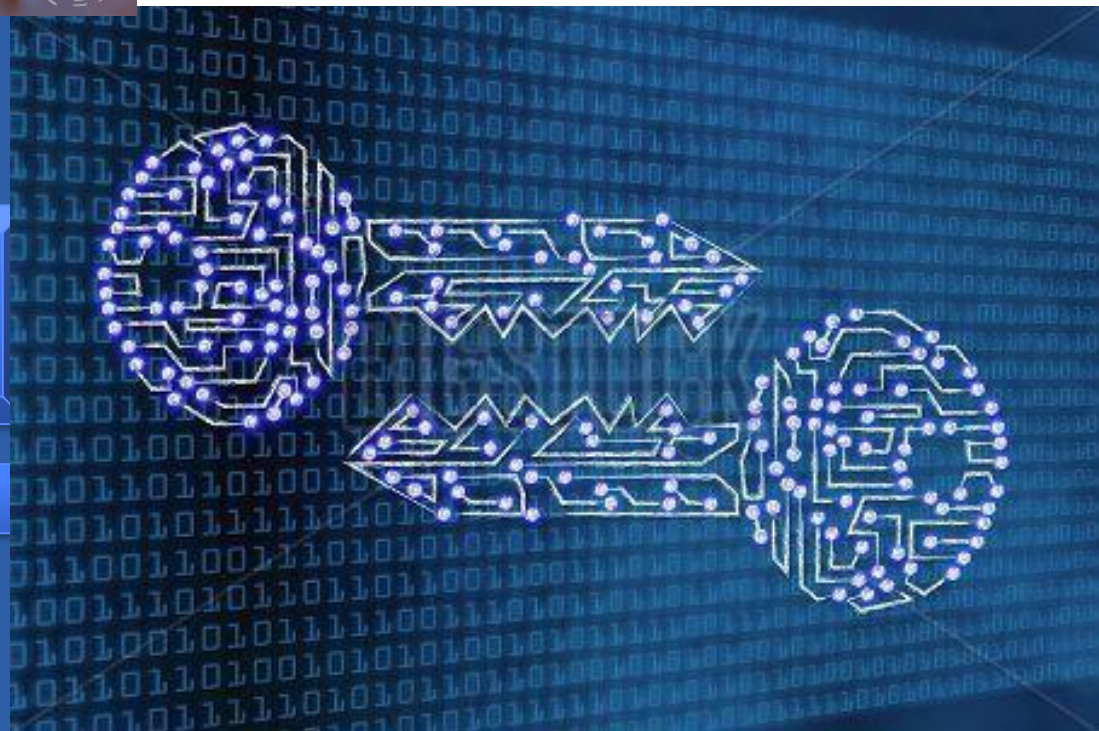


Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 4.1
Introducción





Problemas del cifrado simétrico

Distribución de claves

- Dos entidades, que no han hablado previamente, tienen que intercambiarse mensajes cifrados a través de un canal inseguro
- Para los algoritmos de cifrado simétrico, es necesario intercambiar la clave (simétrica) mediante un **canal seguro**.

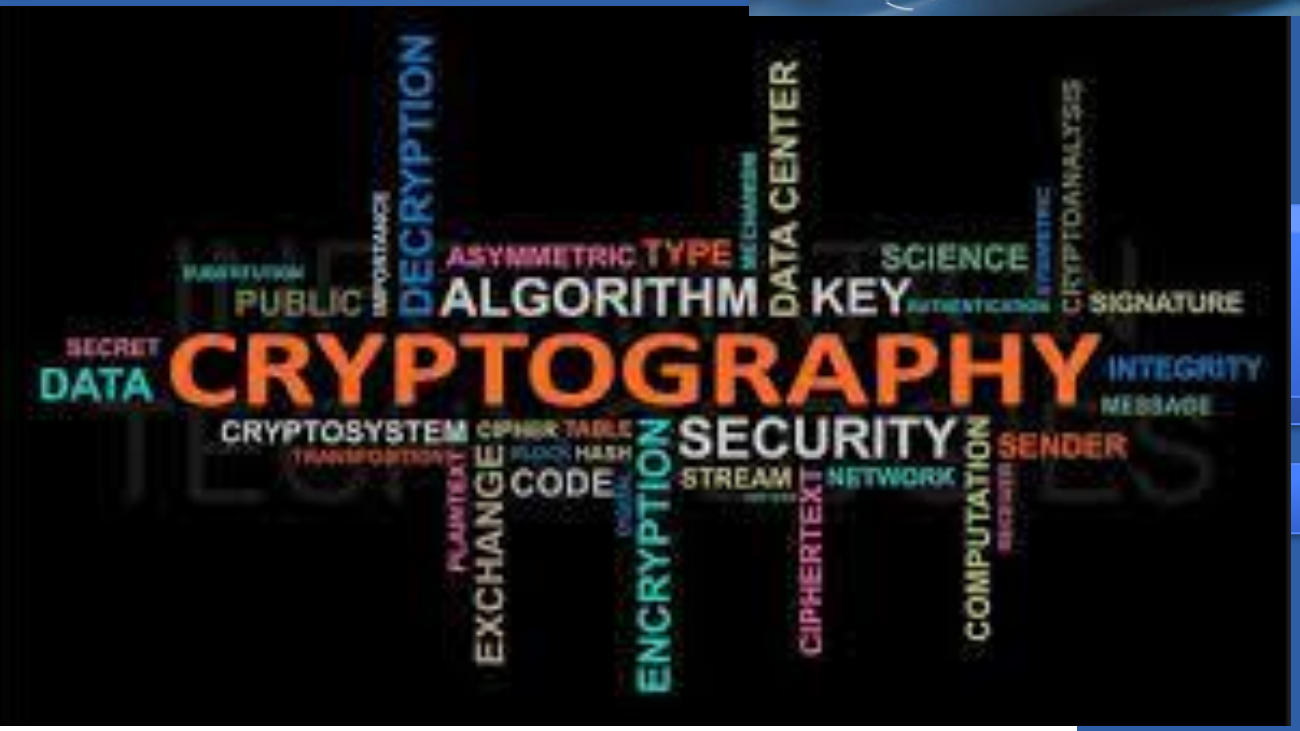
  - Pero…¿por qué no utilizar ese canal seguro para enviar los mensajes?
    - Distancia
    - Reutilización del canal seguro da pistas a los espías
    - Etc
  - Durante miles de años ha sido un problema sin solución

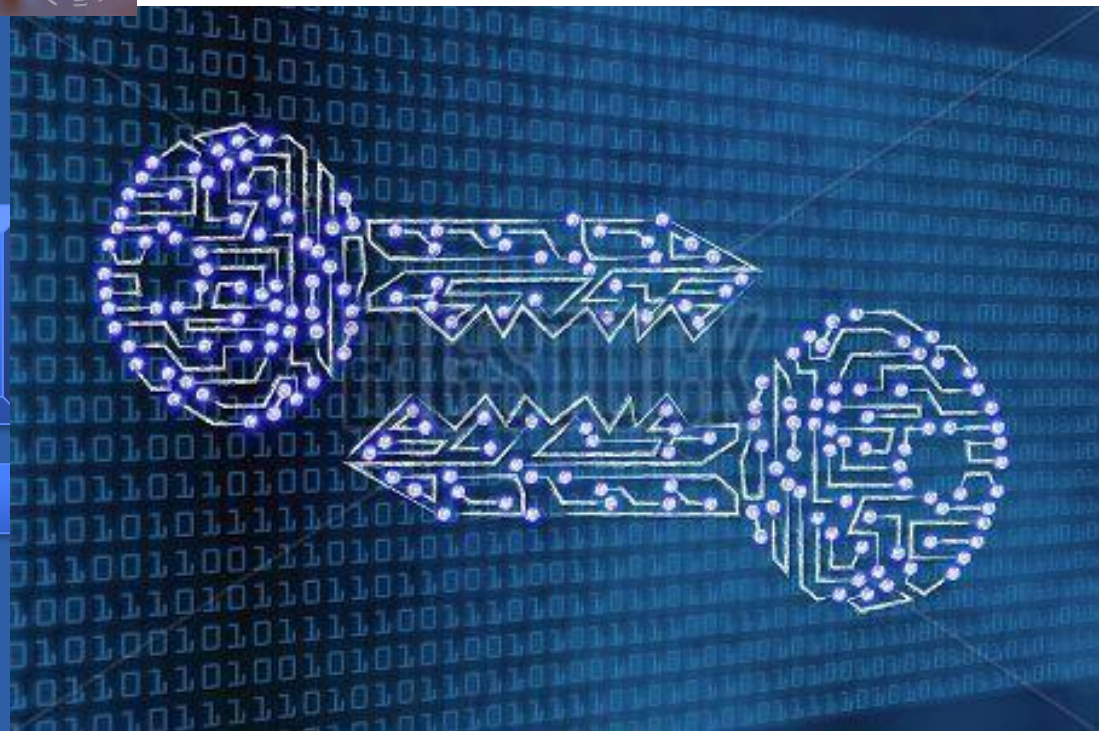
.

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 4.2
Diffie-Hellman





Distribución de claves

- Dos entidades, que no han hablado previamente, tienen que intercambiarse mensajes cifrados a través de un canal inseguro
- Para los algoritmos de cifrado simétrico, es necesario intercambiar la clave (simétrica) mediante un **canal seguro**.

  - Pero…¿por qué no utilizar ese canal seguro para enviar los mensajes?
    - Distancia
    - Reutilización del canal seguro da pistas a los espías
    - …
  - Durante miles de años ha sido un problema sin solución
    - Hasta….

Diffie - Hellman

- Introduce el concepto de **sistema de clave pública**
  - Dos valores relacionados:
    - Una **clave pública**, conocida por todos.
    - Una **clave privada**, mantenida secreta.
  - Fundamentos de seguridad:
    - Es sencillo derivar la pública a partir de la privada
    - Pero **casi imposible obtener la privada a partir de la pública**
  - Además…
    - Base para la **firma digital**

Diffie - Hellman

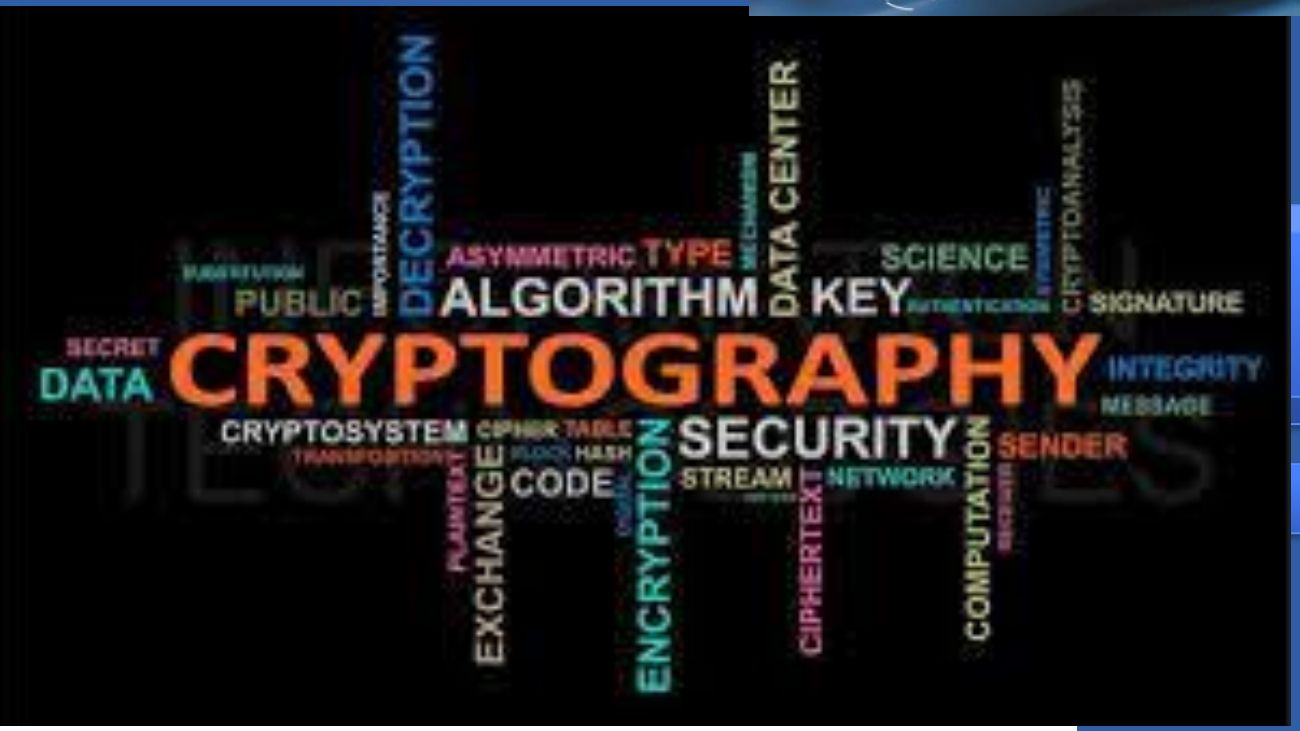
- **Propósito**: permite a dos usuarios (*A y B*) **intercambiar una clave**, de manera segura, **sobre un canal inseguro**
  - *Esa clave puede ser utilizada para cifrado simétrico posterior.*

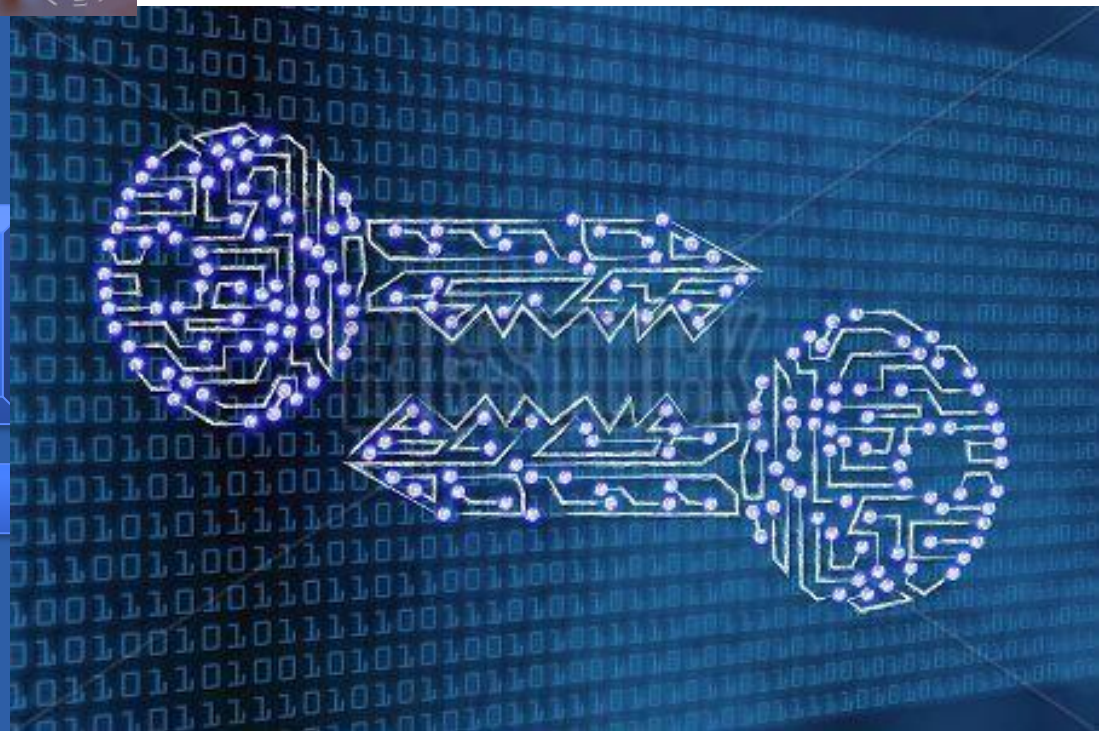
- **Efectividad** basada en:
  - Es poco costoso **calcular exponentes** de manera modular, **PERO**
  - Es muy costoso **calcular logaritmos en un campo discreto.**

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

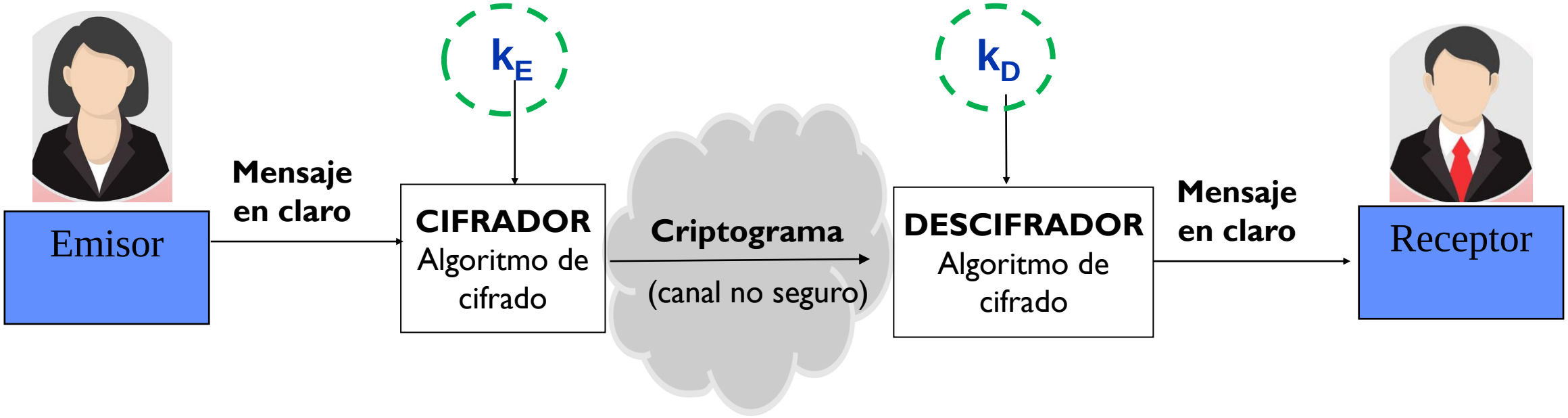
Tema 4.3
Algoritmos de cifrado asimétrico





Modelo de Criptosistema

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



- Base de los sistemas de clave pública

RSA

"A Method for Obtaining Digital Signature and Public-Key Cryptosystems."

R. L. **R**ivest, A. **S**hamir, L. **A**dleman.

Communications of the ACM, v. 21, nº 2, pp 120-126.

February 1978

- Primer **sistema de cifrado basado en clave pública**

- **Seguridad** basada en la **dificultad de factorizar** un número obtenido como producto de dos números primos muy grandes

Abstract (II)

- **Además** de cifrado asimétrico, especifica un protocolo para **Firma Digital**:
  - A message can be "**signed**" using a privately held decryption key.
  - **Anyone can verify this signature** using the corresponding publicly revealed encryption key.
  - Signatures cannot be forged, and a signer cannot later deny the validity of his signature. This has obvious applications in "electronic mail" and "electronic funds transfer" systems..

RSA

Proceso de generación de parejas de claves (pública / privada).

Cada usuario debe:

1. **Seleccionar** dos números primos muy grandes **p** y **q**
2. **Calcular n = p·q**
3. **Calcular phi(n) = phi(p)· phi(q)**
4. **Elegir** el exponente público **"e", tal que:**
   1. e > 0
   2. e, phi(n) son coprimos
5. **Calcular d,** tal que ***e·d = 1 mod (phi(n))***

⇨ **PU = (e, n)**

PR = (d, n)

Proceso de cifrado

$PU = (e_B, n_B)$

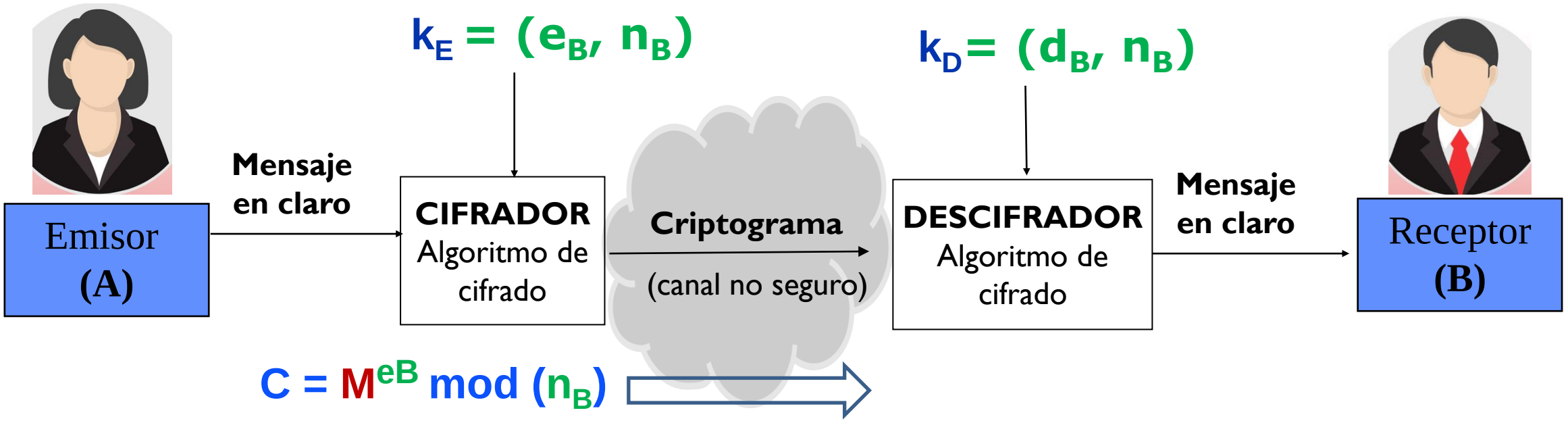
$PR = (d_A, n_A)$

El usuario (A) que envía el mensaje (cifrado) (a B) debe:

1. **Codificar** cada parte del mensaje en un número entero **M**
2. **Obtener la clave pública del receptor:** $PU_B = (e_B, n_B)$
3. **Calcular** $C = M^{eB} \bmod (n_B)$

Modelo de Criptosistema

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



- Se cifra con la **clave pública del receptor**

Proceso de descifrado

El usuario (B) que recibe el mensaje (cifrado) (de A) debe:

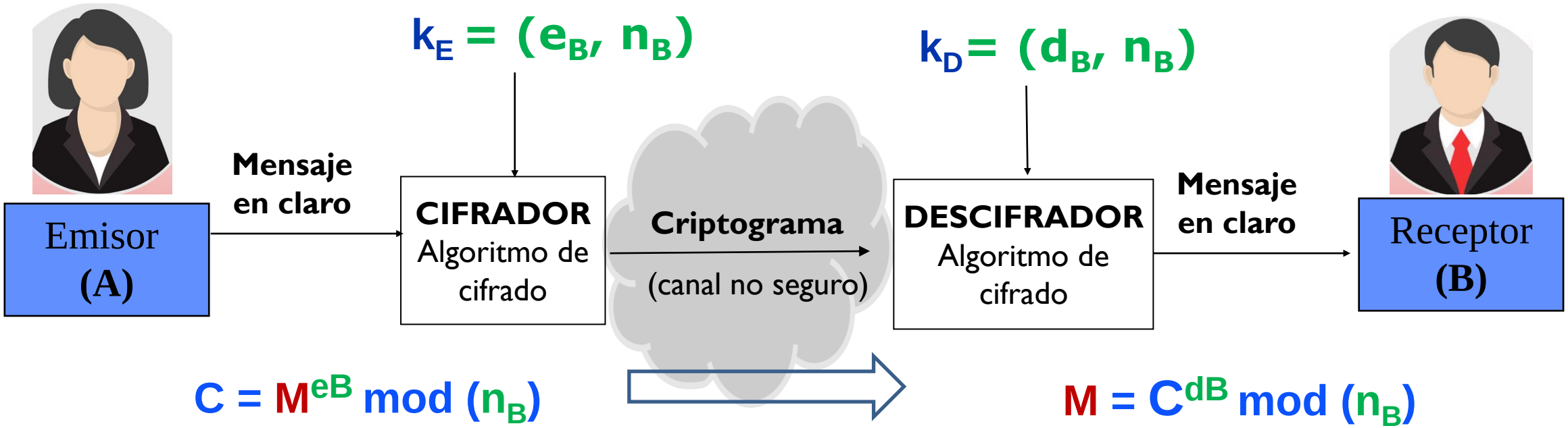
$PU = (e_B, n_B)$

$PR = (d_A, n_A)$

1. **Calcular** $M = C^{dB} \bmod (n_B)$
2. **Decodificar** el número entero **M** al alfabeto del mensaje

Modelo de Criptosistema

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

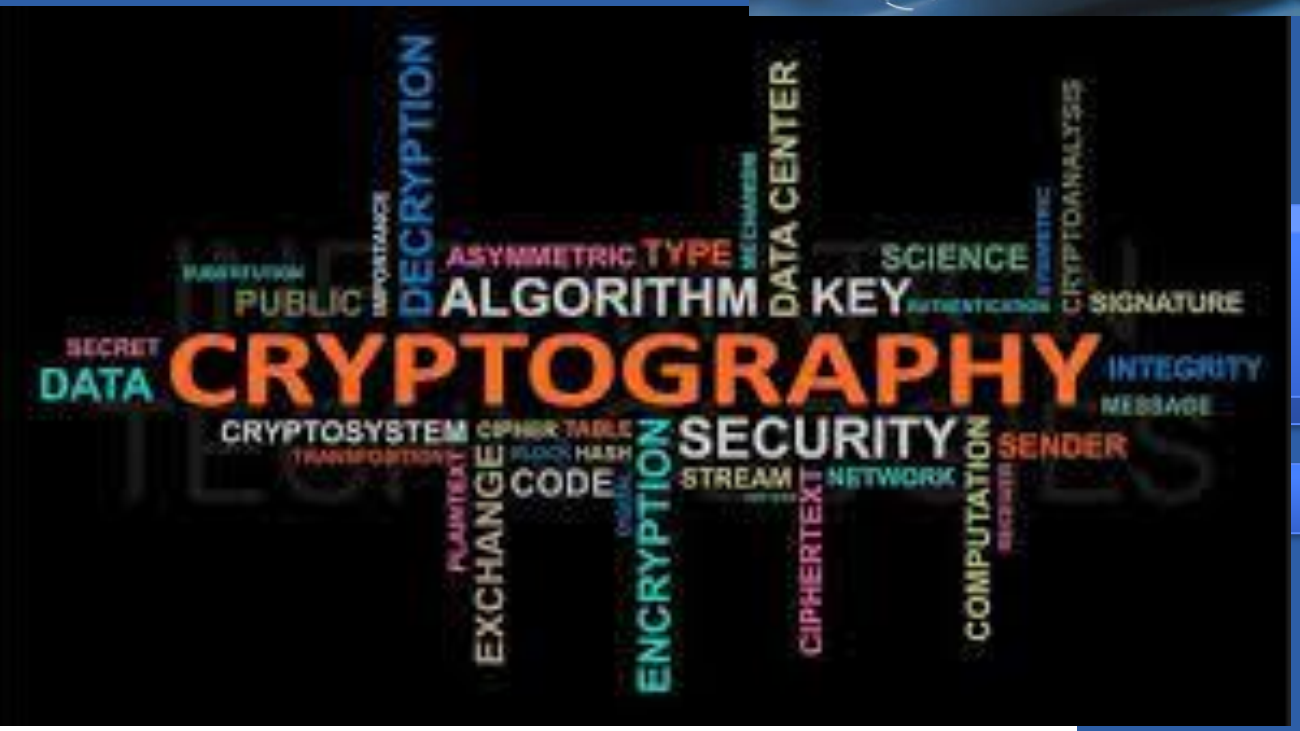


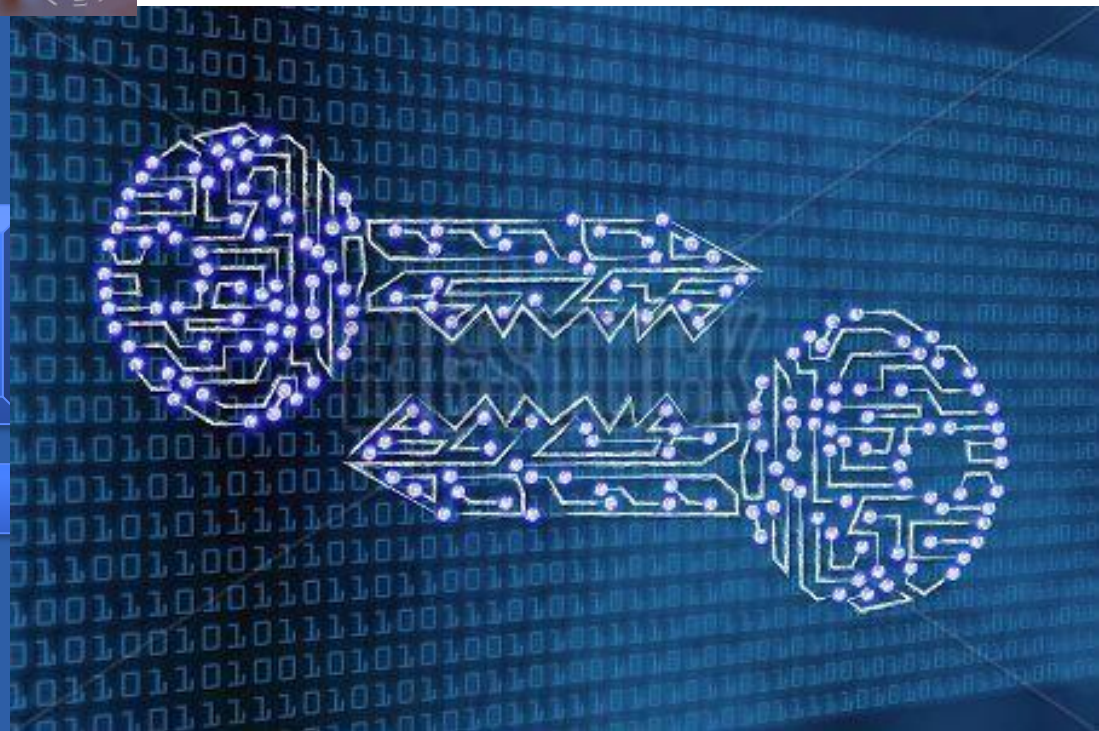
- Se descifra con la **clave <u>privada</u> del receptor**

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

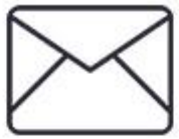
Tema 4.4
Firma Digital





Funciones Hash

¿Qué son y qué hacen las funciones Hash?

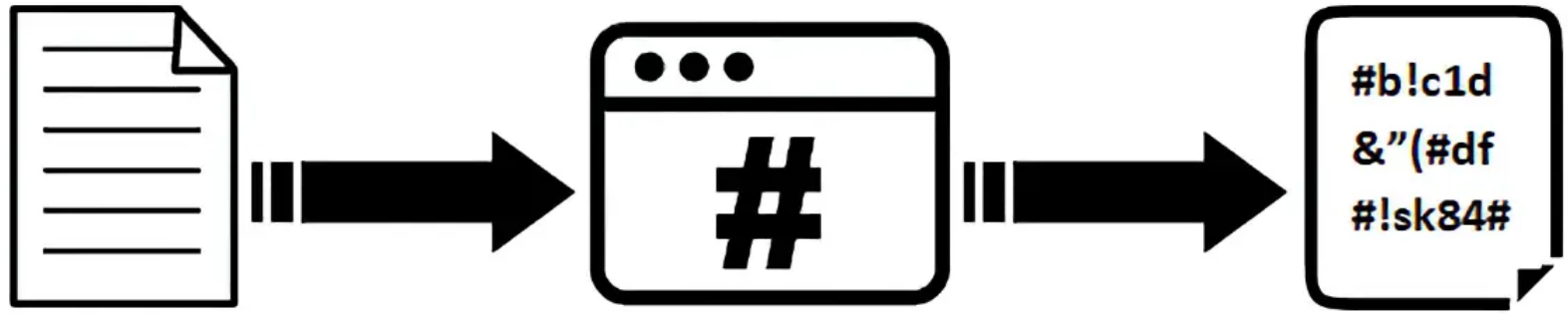


Longitud variable

Longitud fija

Funciones Hash

¿Qué son y qué hacen las funciones Hash?



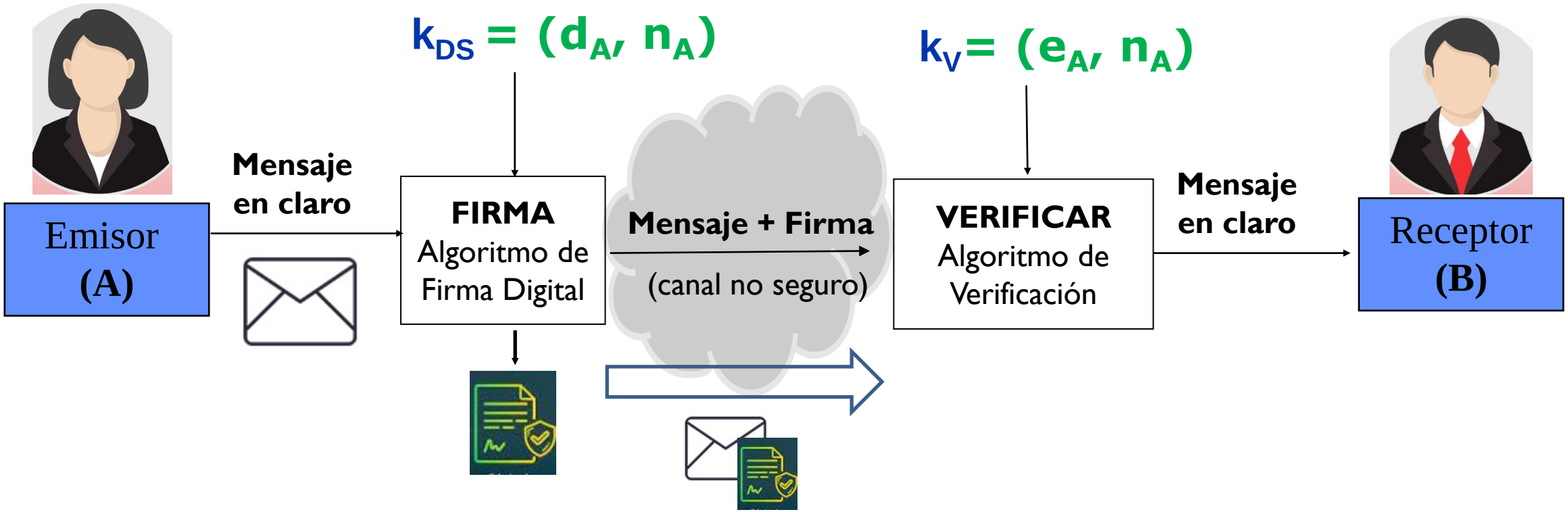
Longitud variable

Longitud fija

https://www.youtube.com/watch?v=2BldESGZKB8

Firma Digital

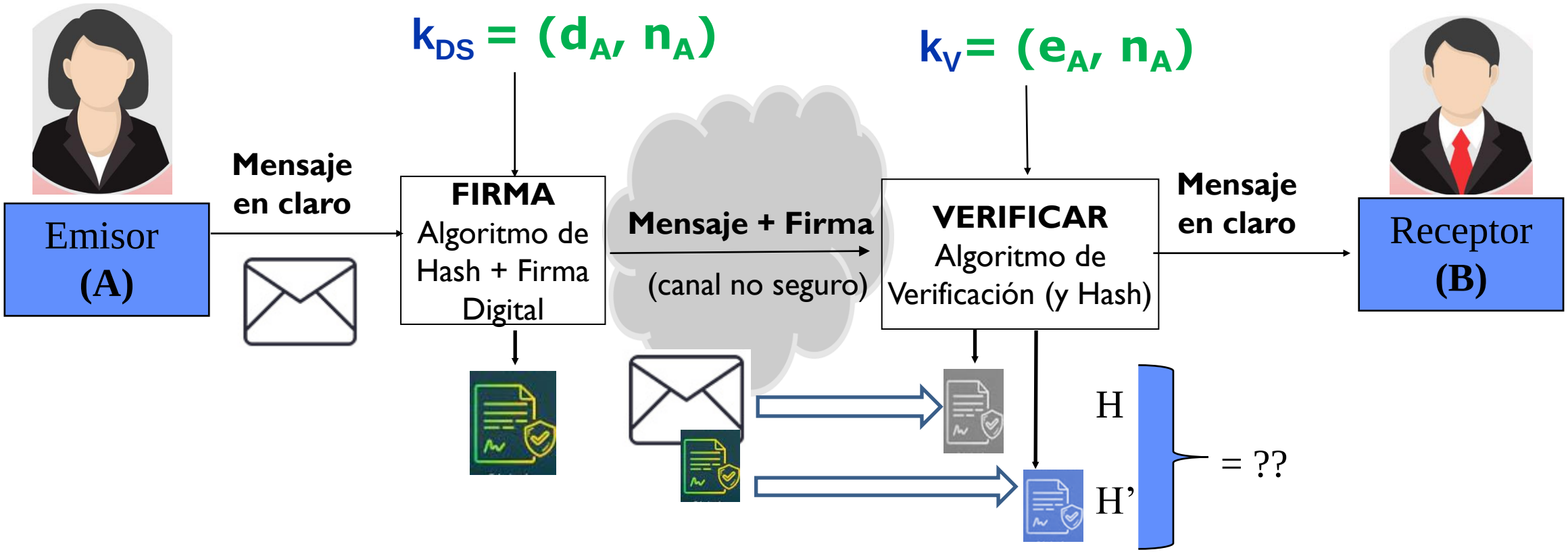
ASIMÉTRICOS: Una clave (k_{DS}) para firmar y otra (k_V) para verificar



- Se **firma** con la **clave privada del emisor**

Firma Digital

ASIMÉTRICOS: Una clave (k_{DS}) para firmar y otra (k_V) para verificar



- Se **verifica** con la **clave pública del emisor**

Firma Digital con RSA

Proceso de generación de parejas de claves (pública / privada).

(igual que para cifrado / descifrado)

Cada usuario debe:

1. **Seleccionar** dos números primos muy grandes **p** y **q**
2. **Calcular n = p·q**
3. **Calcular phi(n) = phi(p)· phi(q)**
4. **Elegir** el exponente público "**e**", **tal que:**
   1. e > 0
   2. e, phi(n) son coprimos
5. **Calcular d,** tal que ***e·d = 1 mod (phi(n))***

⇨ **PU = (e, n)**

PR = (d, n)

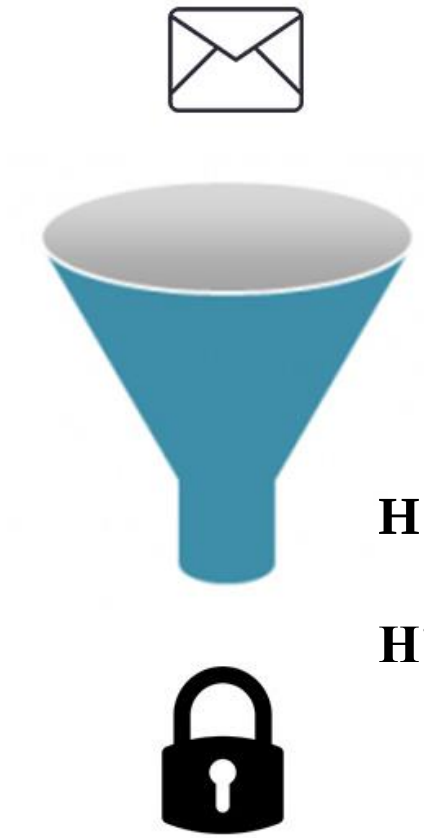
Y además…

Firma Digital con RSA

Utilización de la misma función Hash en emisión (firma) y en recepción (verificación de la firma)

H = Hash (M)

S = DS (H)



H = Hash (M)

H’ = V (S)

H = H’ ??

Firma Digital con RSA

Proceso de firma digital

El usuario (A) que envía el mensaje (cifrado) (a B) debe:

$PU = (e_B, n_B)$

$PR = (d_A, n_A)$

1. **Codificar** cada parte del mensaje en un número entero **M**
2. **Acceder a su clave privada (del emisor: $PR = (d_A, n_A)$)**
3. **Calcular H = Hash (M)**
4. **Calcular $S = H^{dA} \bmod (n_A)$**

Proceso de verificación

El usuario (B) que recibe el mensaje (cifrado) (de A) debe:

$PU = (e_B, n_B)$

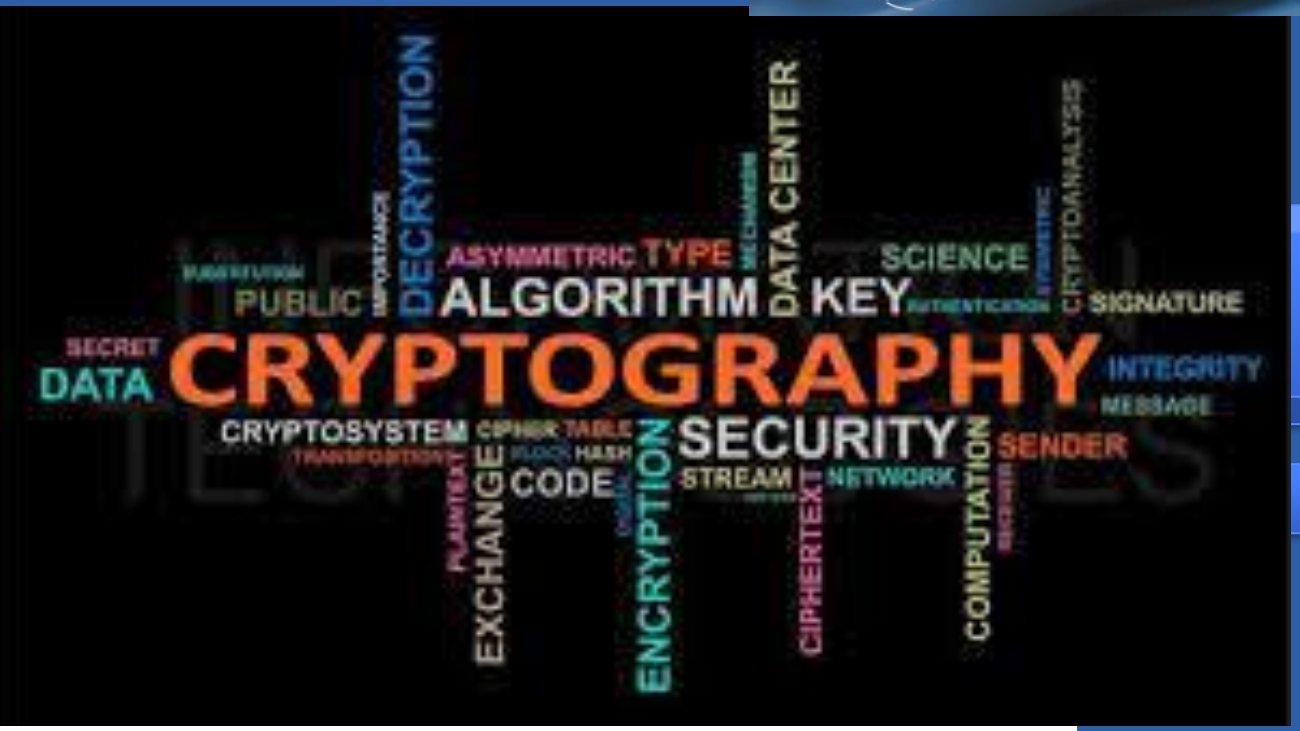
$PR = (d_A, n_A)$

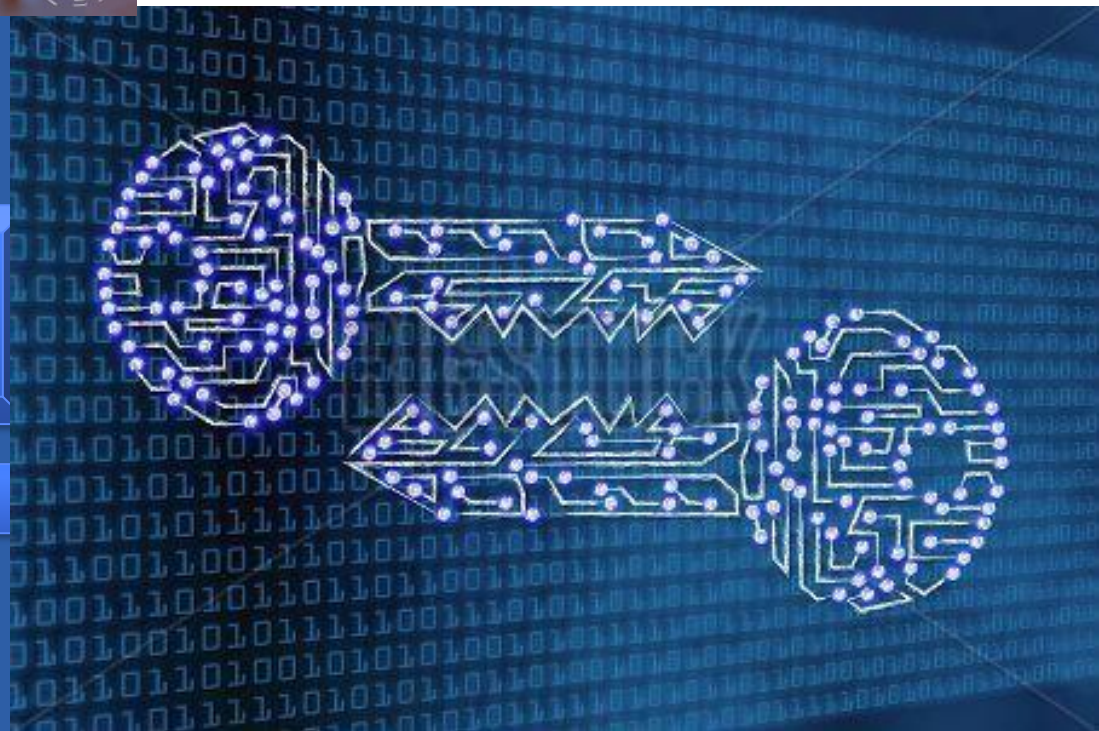
1. **Obtener la clave pública del emisor $PU_A = (e_A, n_A)$**
2. **Recibir el mensaje original M, y la firma asociada S**
3. **Calcular el Hash H = Hash(M)**
4. **Calcular $H' = S^{eA} \bmod (n_A)$**
5. **Comparar H y H'**
   1. **Si son iguales → mensaje válido y corresponde efectivamente al emisor.**

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 4.5
Modelos Híbridos





Modelo de Criptosistema

- Hasta ahora hemos visto…
  - **CONFIDENCIALIDAD**
    - **Cifrado simétrico**
      - ✓ **Eficiente para grandes volúmenes de datos**
    - **Cifrado asimétrico**
      - ✓ **Eficiente para intercambio de claves de manera segura y cifrado en canales no seguros**
  - **INTEGRIDAD Y AUTENTICACIÓN**
    - **Firma digital**

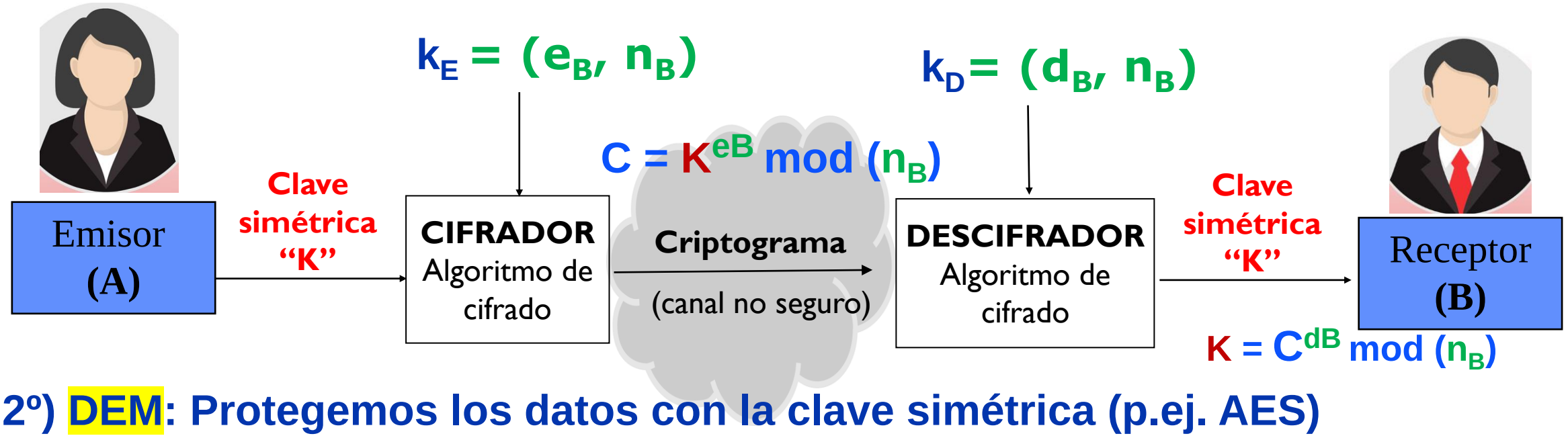
Modelo de Criptosistema

- Hasta ahora hemos visto…
  - **CONFIDENCIALIDAD**
    - **Cifrado simétrico**
      - ✓ **Eficiente para grandes volúmenes de datos**
    - **Cifrado asimétrico**
      - ✓ **Eficiente para intercambio de claves de manera segura y cifrado en canales no seguros**
  - **INTEGRIDAD Y AUTENTICACIÓN**
    - **Firma digital**

¿Y si los combinamos?

Modelo de Criptosistema: KEM/DEM

1º) KEM: Protegemos la clave simétrica con cifrado asimétrico

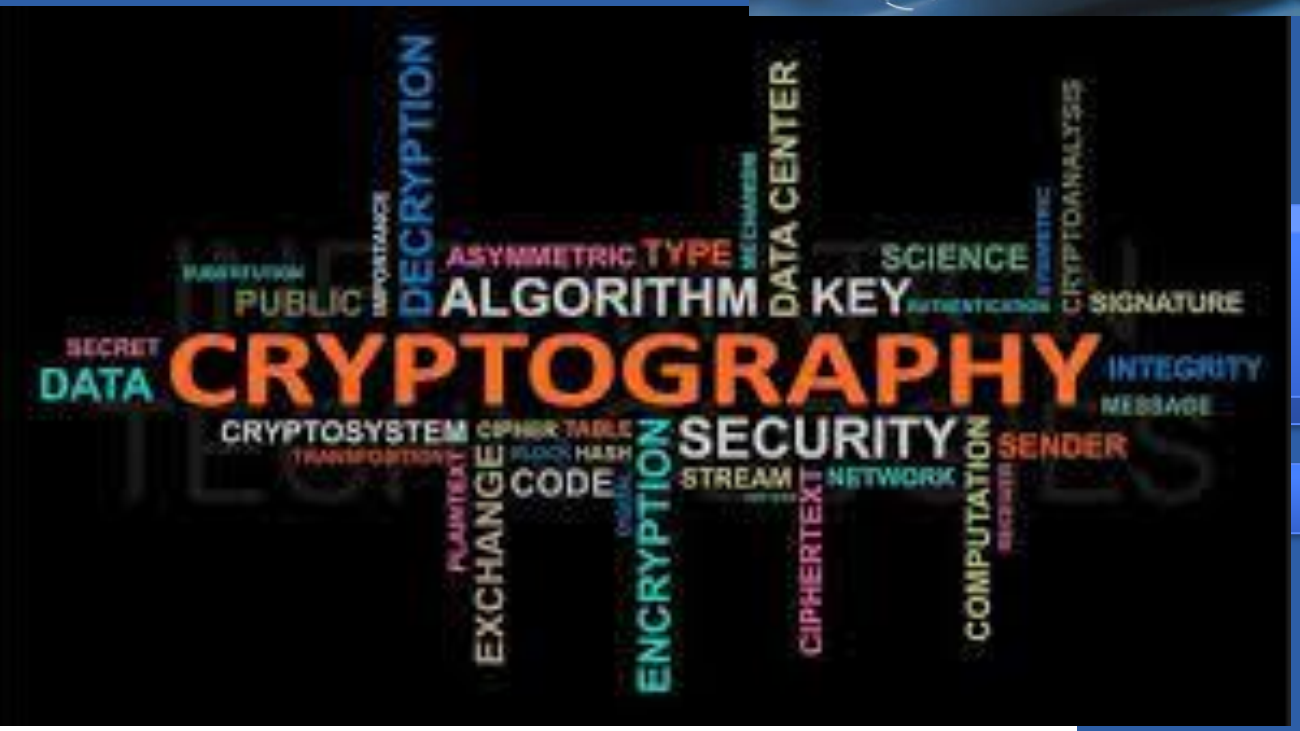


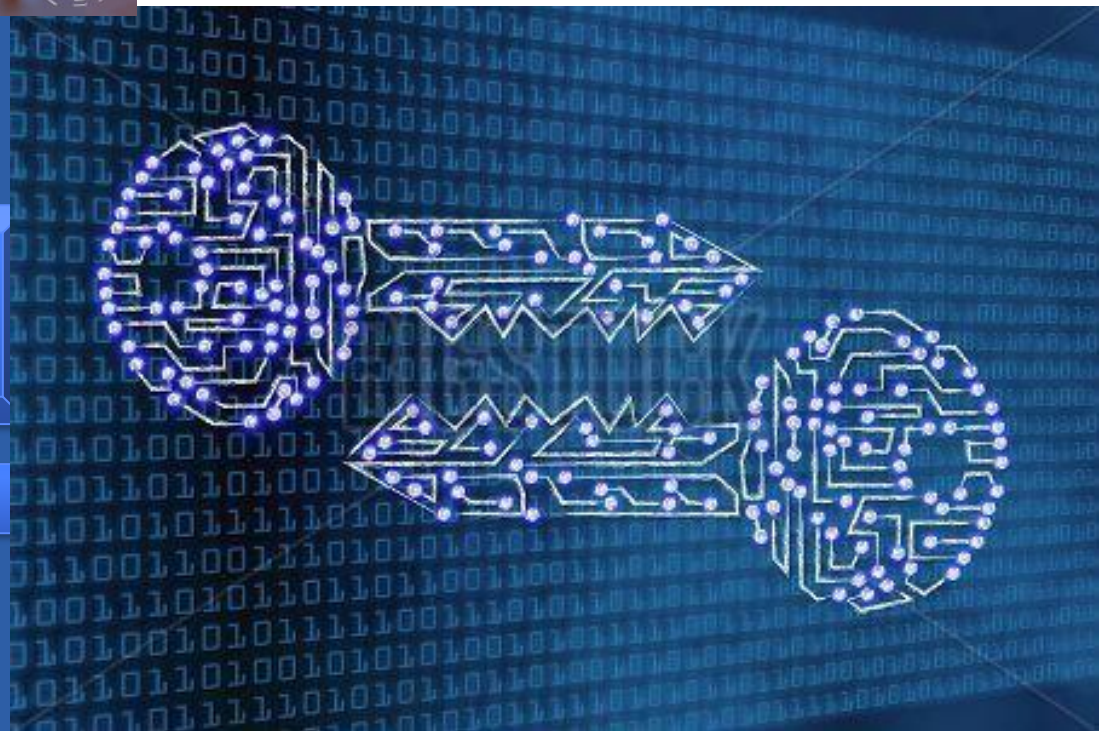
2º) DEM: Protegemos los datos con la clave simétrica (p.ej. AES)

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 4.6
PKI





Sistemas de Clave Pública

- Introducción
- Distribución de claves
- Cifrado asimétrico
- Firma Digital
- Modelos Híbridos
- **Certificados Digitales y PKI**

Necesidad de algo más que algoritmos

- Los sistemas de clave asimétrica no garantizar la vinculación entre el usuario y las claves públicas asociadas
- Confidencialidad ¿Cómo podemos garantizar que la clave pública del receptor realmente corresponde a ese usuario?
- Firma digital ¿Cómo podemos garantizar que la clave pública del firmante realmente corresponde a ese usuario?

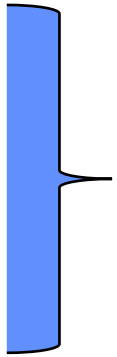
- Necesitamos algo más que meros algoritmos para garantizar el funcionamiento adecuado:
  - **Infraestructura de Clave Pública (PKI)**

Certificados Digitales

- Vinculan la identidad de un sujeto, con su clave pública
- Además, permiten determinar información adicional:
  - Fecha de validez
  - Parámetros públicos de la clave pública
    - Algoritmo
    - Parámetros comunes
    - Uso esperado (para cifrar, para firma personal, para firma delegada…)
- Van firmados digitalmente por el emisor del certificado
  - E indican los parámetros para poder verificar dicha firma

Necesidad de algo más que algoritmos: PKI

- La **RFC 2828** (*Internet Security Glossary*) define la **Infraestructura de Clave Pública (PKI)** como el **conjunto de:**
  - HW
  - SW
  - Personas
  - Políticas
  - y Procedimientos
- **Necesarios para:**
  - Crear
  - Gestionar
  - Almacenar
  - Distribuir y Revocar



Certificados digitales basados en criptografía asimétrica.

https://www.rfc-editor.org/rfc/pdfrfc/rfc2828.txt.pdf

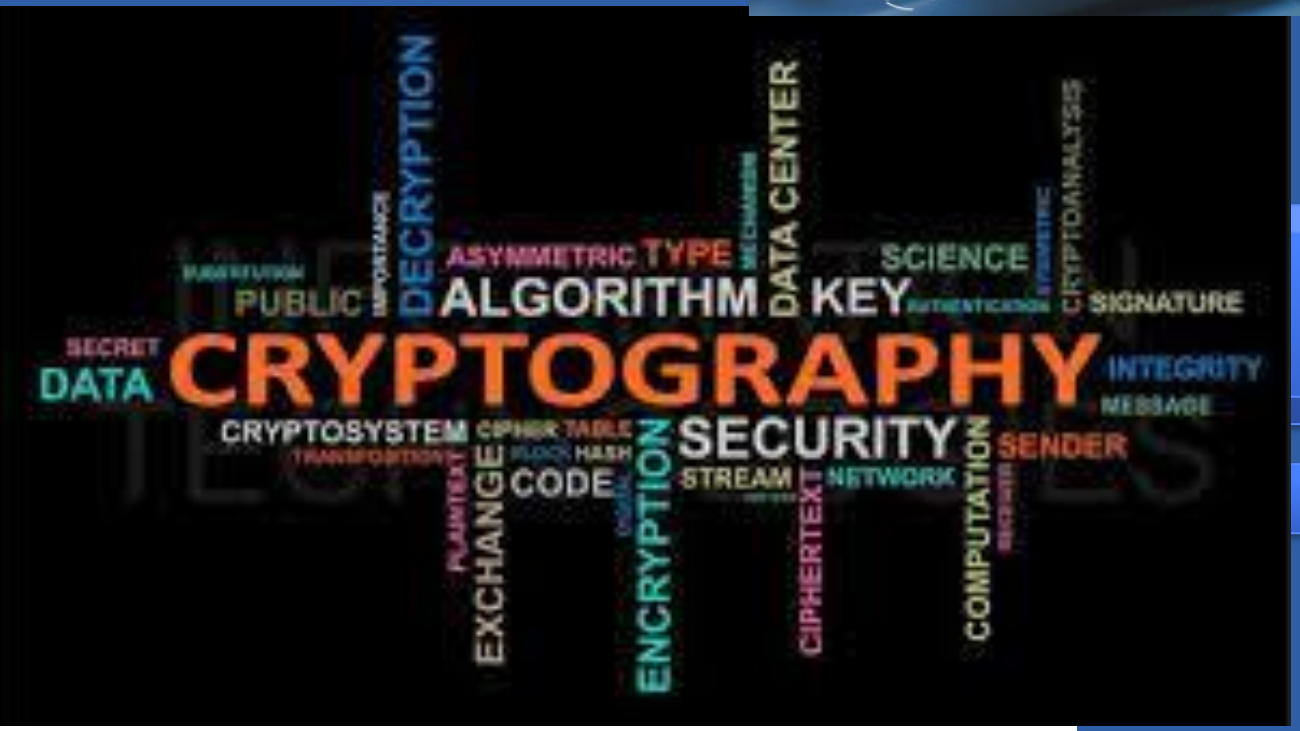
Necesidad de algo más que algoritmos: PKI

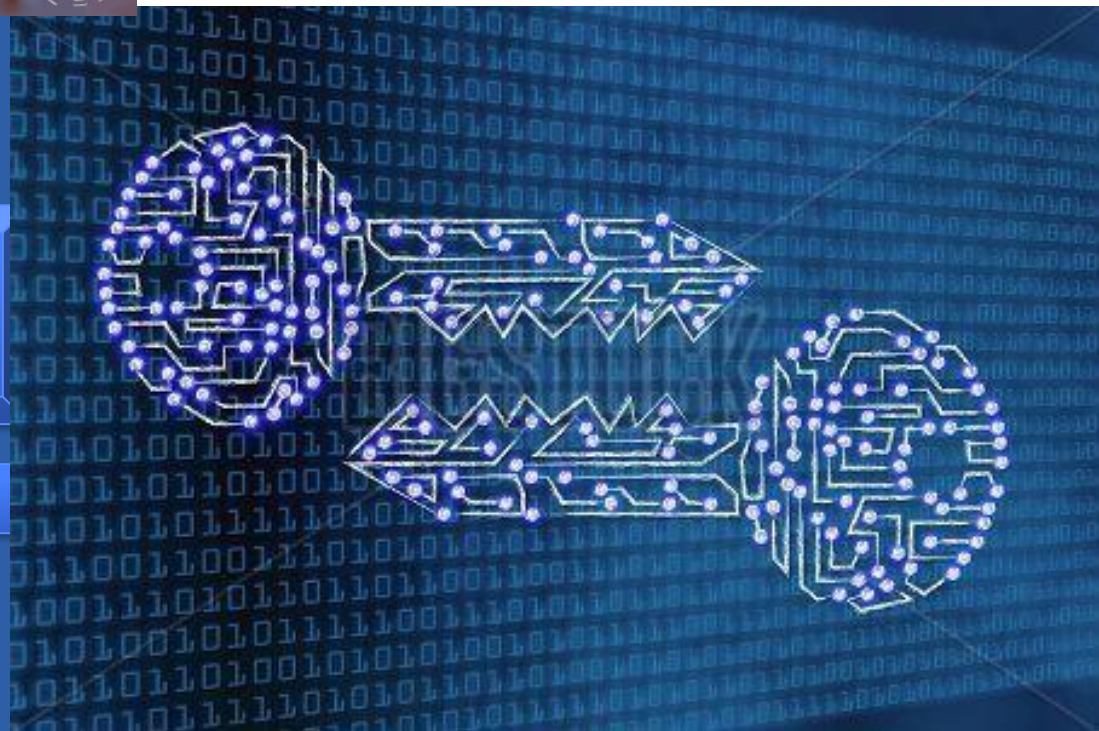
- El principal objetivo para desarrollar una **Infraestructura de Clave Pública (PKI)** es permitir la adquisición y distribución de claves públicas de una manera:
  - Segura,
  - Adecuada, y
  - Eficiente.

Criptografía

Universidad Francisco de Vitoria
UFV Madrid

Tema 5
Aplicaciones de la Criptografía





Comunicaciones Seguras: SSH

- **SSH™** (o Secure SHell) es un protocolo que facilita las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor y que permite a los usuarios **conectarse a un host remotamente**.

- A diferencia de otros protocolos de comunicación remota tales como FTP o Telnet, **SSH encripta la sesión de conexión**, haciendo imposible que alguien pueda obtener contraseñas no encriptadas.

- SSH está diseñado para **reemplazar** los métodos más viejos y menos seguros para registrarse remotamente en otro sistema a través de la shell de comando, tales como **telnet o rsh**.

- El uso de métodos seguros para registrarse remotamente a otros sistemas **reduce los riesgos** de seguridad tanto para el sistema cliente como para el sistema remoto.

Comunicaciones Seguras: SSH

- El protocolo SSH proporciona los siguientes tipos de protección:
  - Después de la conexión inicial, el cliente puede **verificar** que se está conectando al mismo servidor al que se conectó anteriormente.
  - El cliente transmite su **información de autenticación** al servidor usando una **encriptación** robusta de 128 bits.
  - Todos los **datos enviados y recibidos** durante la sesión se transfieren por medio de **encriptación** de 128 bits, lo cual los hacen extremamente difícil de descifrar y leer.
- Ya que el protocolo SSH encripta todo lo que envía y recibe, se puede usar para **asegurar protocolos inseguros**. El servidor SSH puede convertirse en un conducto para convertir en seguros los protocolos inseguros mediante el uso de una técnica llamada **reenvío por puerto**, como por ejemplo POP, incrementando la seguridad del sistema en general y de los datos.

Comunicaciones Seguras: SSL, TLS y HTTPS

- **SSL** es el acrónimo de **Secure Sockets Layer** (capa de sockets seguros), la tecnología estándar para mantener segura una **conexión** a Internet, así como para proteger cualquier **información confidencial** que se envía entre dos sistemas e impedir que los delincuentes lean y modifiquen cualquier dato que se transfiera, incluida información que pudiera considerarse personal.

- Los dos sistemas pueden ser un servidor y un cliente (por ejemplo, un sitio web de compras y un navegador) o de servidor a servidor (por ejemplo, una aplicación con información que puede identificarse como personal o con datos de nóminas).

- Esto lo lleva a cabo asegurándose de que todos los datos que se transfieren entre usuarios y sitios web o entre dos sistemas sean imposibles de leer. Utiliza algoritmos de **cifrado** para codificar los datos que se transmiten.

Comunicaciones Seguras: SSL, TLS y HTTPS

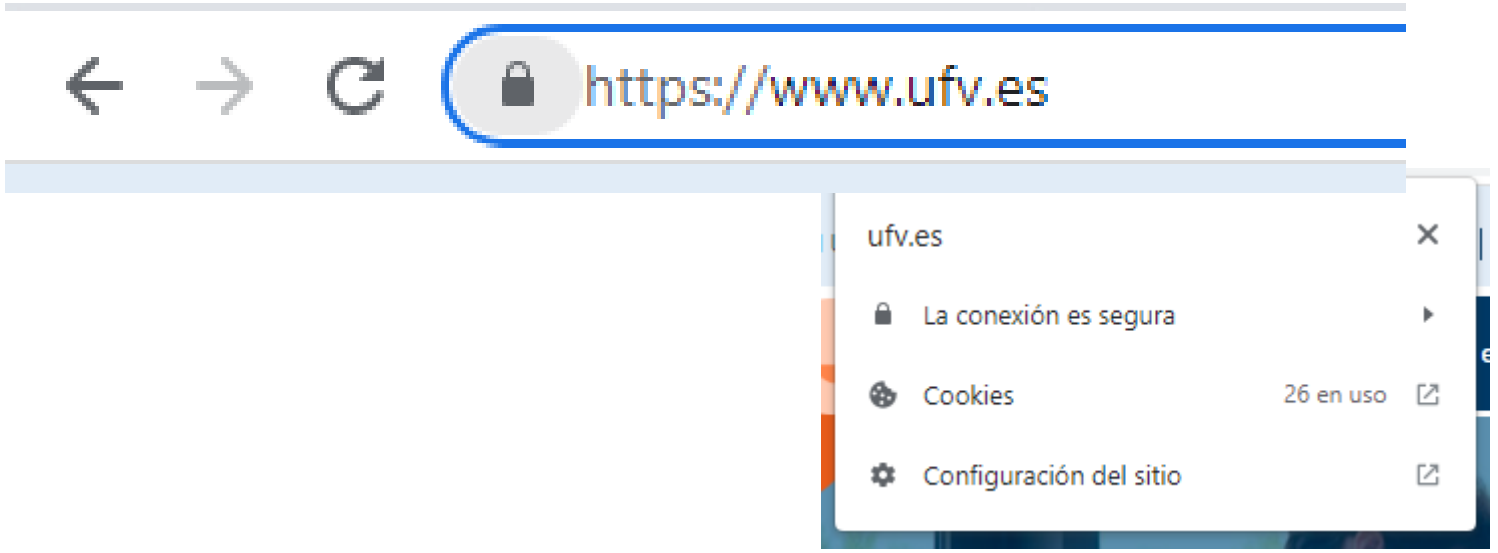
- El protocolo **TLS** (***Transport Layer Security***, seguridad de la capa de transporte) es solo una versión **actualizada y más segura de SSL**.

- Si bien aún denominamos a algunos certificados de seguridad SSL porque es un término más común, al comprar certificados SSL, en realidad se compran los certificados TLS más actualizados con la opción de cifrado ECC, RSA o DSA.

Comunicaciones Seguras: SSL, TLS y HTTPS

- **HTTPS** (Hyper Text Transfer Protocol Secure o protocolo seguro de transferencia de hipertexto) aparece en la dirección URL cuando un sitio web está protegido por un certificado SSL.

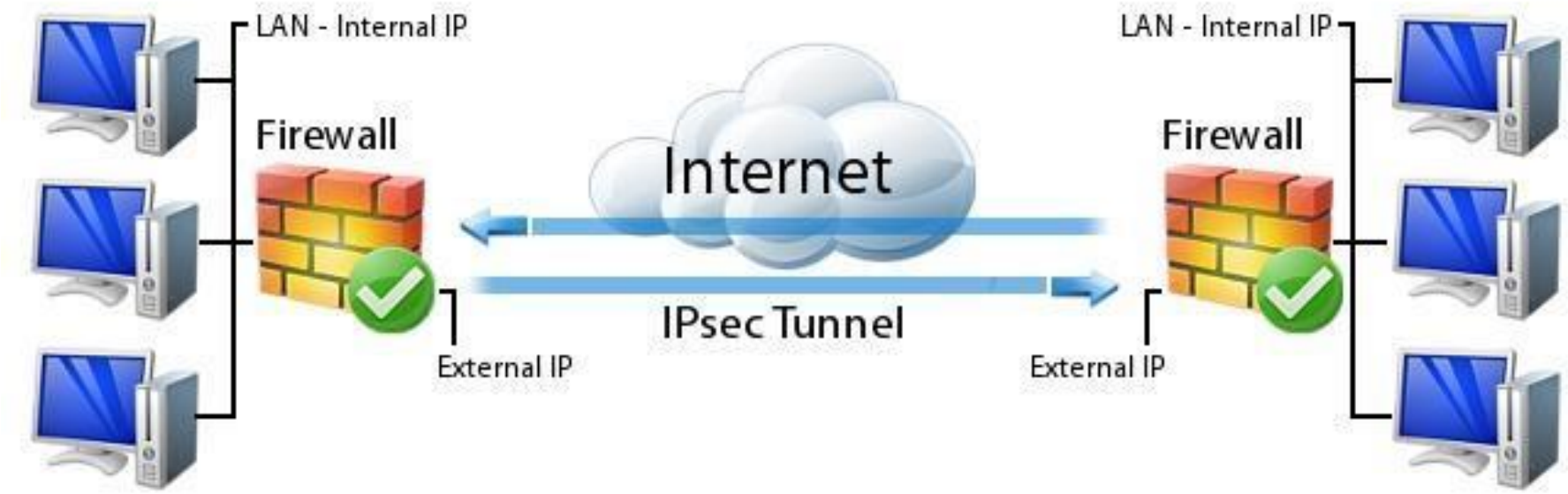
- Los detalles del certificado, por ejemplo la entidad emisora y el nombre corporativo del propietario del sitio web, se pueden ver haciendo clic en el símbolo de candado de la barra del navegador.



Comunicaciones Seguras: IPSEC

- **IPsec** (abreviatura de Internet Protocol security) es un conjunto de protocolos cuya función es asegurar las comunicaciones sobre el Protocolo de Internet (IP) autenticando y/o cifrando cada paquete IP en un flujo de datos.

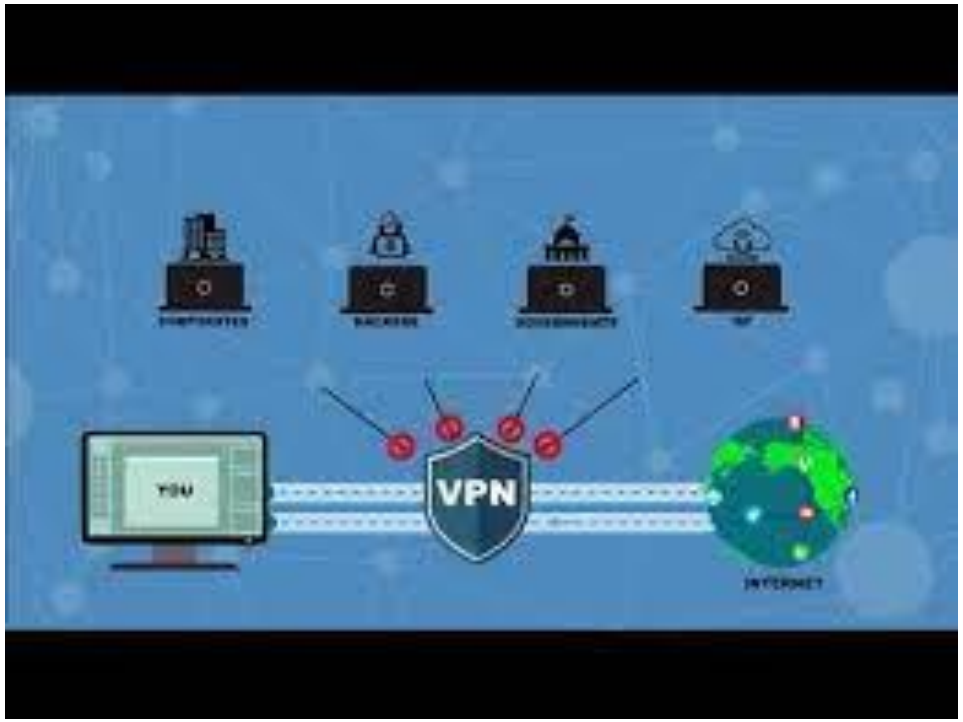
- IPsec también incluye protocolos para el establecimiento de claves de **cifrado**.



Comunicaciones Seguras: VPN

- Una **VPN** significa 'red privada virtual' (***virtual private network***). Es una herramienta digital que redirige tu tráfico de internet a través de un túnel seguro, **ocultando tu dirección IP y encriptando tus datos**.

Así es como una VPN mantiene tus datos privados y te protege frente a potenciales. ciberataques



Blockchain

Blockchain es un libro mayor (**ledger**) compartido e inmutable que facilita el proceso de registro de **transacciones** y de seguimiento de **activos** en una red de negocios.

Un *activo* puede ser tangible (una casa, un auto, dinero en efectivo, terrenos) o intangible (propiedad intelectual, patentes, derechos de autor, marcas).

Prácticamente cualquier cosa de valor puede ser rastreada y comercializada en una red de blockchain, reduciendo el riesgo y los costos para todos los involucrados.

Blockchain: Elementos Clave

- Tecnología de libro mayor distribuido

Todos los participantes de la red tienen acceso al libro mayor distribuido y a su registro inmutable de transacciones. Con este libro mayor compartido, las transacciones se registran solo una vez, eliminando la duplicación del esfuerzo que es típico de las redes de negocios tradicionales.

- Registros inalterables

Ningún participante puede cambiar o falsificar una transacción una vez grabada en el libro mayor compartido. Si el registro de una transacción incluye un error, se debe añadir una nueva transacción para revertir el error, pero ambas transacciones serán visibles.

- Contratos inteligentes

Para acelerar las transacciones, un conjunto de reglas, llamado contrato inteligente, se almacena en el blockchain y se ejecuta automáticamente. Un contrato inteligente puede definir las condiciones para las transferencias de bonos corporativos, incluir los términos de un seguro de viaje que se pagará y mucho más.

Blockchain: Cómo funciona

- A medida que se produce una transacción, se registra como un "**bloque**" de datos

Estas transacciones muestran el **movimiento de un activo**, el cual puede ser tangible (un producto) o intangible (intelectual). El bloque de datos puede registrar la información de su elección: quién, qué, cuándo, dónde, cuánto e incluso la condición, como la temperatura de un envío de alimentos.

- Cada bloque está **conectado** al bloque anterior y al bloque posterior

Estos bloques forman una cadena de datos a medida que un activo se mueve de un lugar a otro o cambia de dueño. Los bloques confirman tanto el tiempo exacto como la secuencia de las transacciones y se unen de forma segura para evitar que se alteren o se inserten entre dos bloques existentes.

- Las transacciones se unen y forman una **cadena irreversible**: un blockchain

Cada bloque adicional refuerza la verificación del bloque anterior y, por lo tanto, de todo el blockchain. Esto hace que dicha cadena sea a prueba de manipulaciones, lo que constituye la ventaja principal de la inalterabilidad. Esto evita que alguien malintencionado modifique la cadena y crea un libro mayor distribuido de transacciones en la que usted y otros miembros de la red pueden confiar.