SWINBURNE UNIVERSITY OF TECHNOLOGY

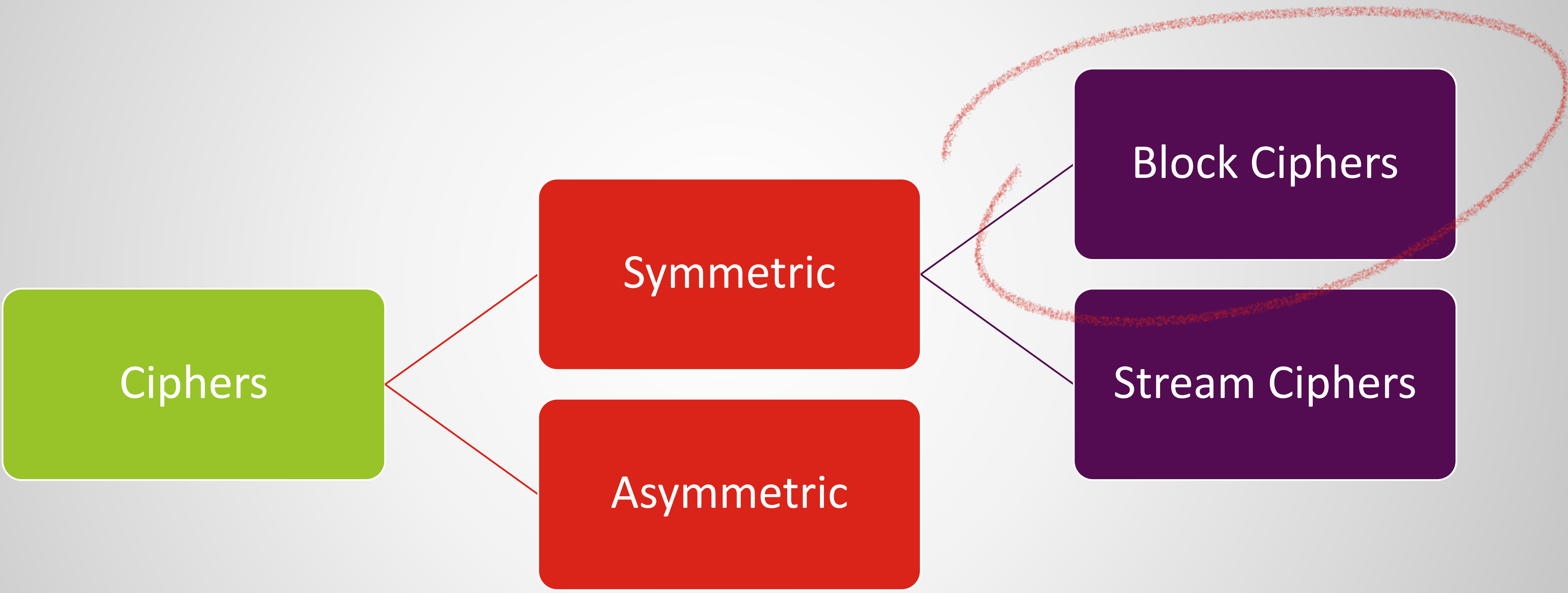
Theory of Blockchain

Session 2:

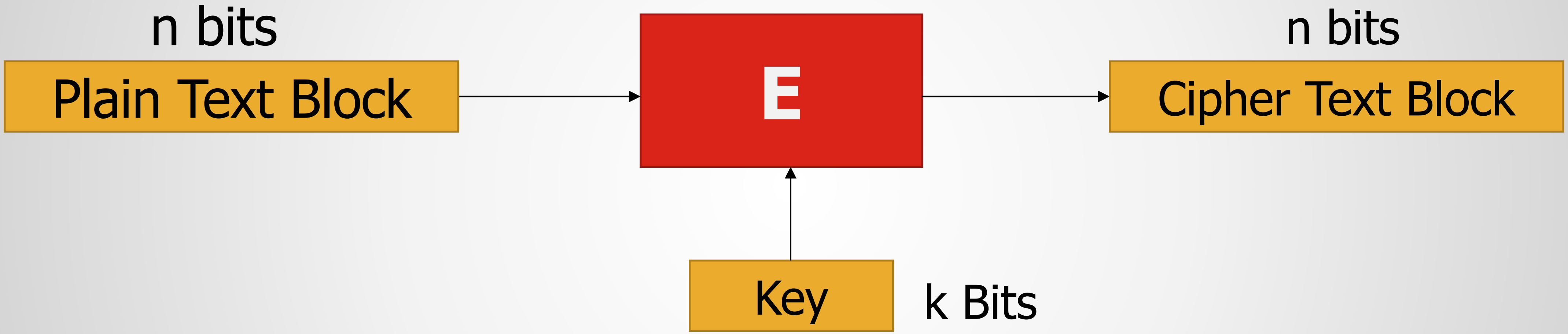
Symmetric Cryptography

Module 2 - Symmetric Encryption Algorithms

Classification of Ciphers



Block Ciphers Model



Example:

1. DES: n= 64 bits, k = 56 bits
2. AES: n=128 bits, k = 128, 192, 256 bits

The Era of Standard Ciphers

- Data Encryption Standard (DES) - 1974

No. of keys = 2^{56} , block size = 64 bits

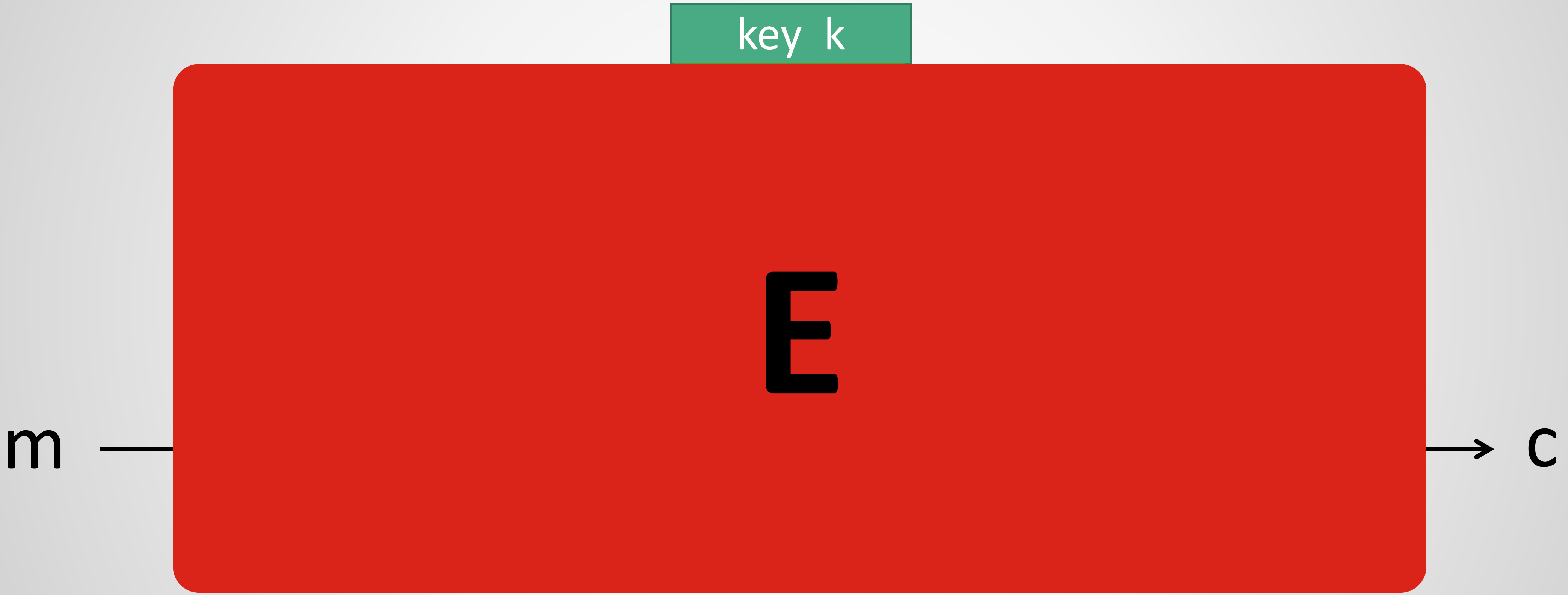
Bigger block does not (necessarily) mean more security.
Security is determined by the key length.

Still in Service (today): AES - 2001

No. of keys = $2^{128} \sim 2^{256}$, block size = 128~256

How are Block Ciphers Usually Constructed?

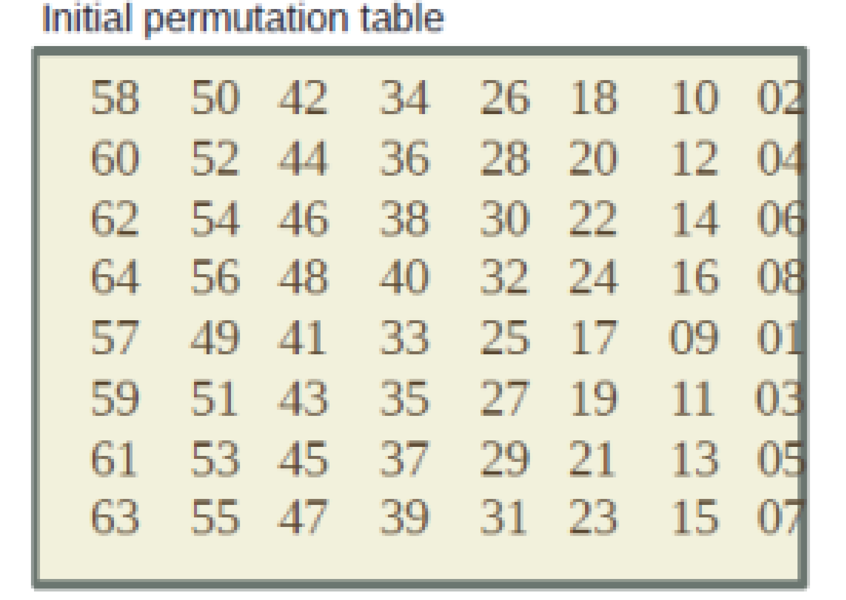
Answer: By repetition



R(k,m) is called the round function

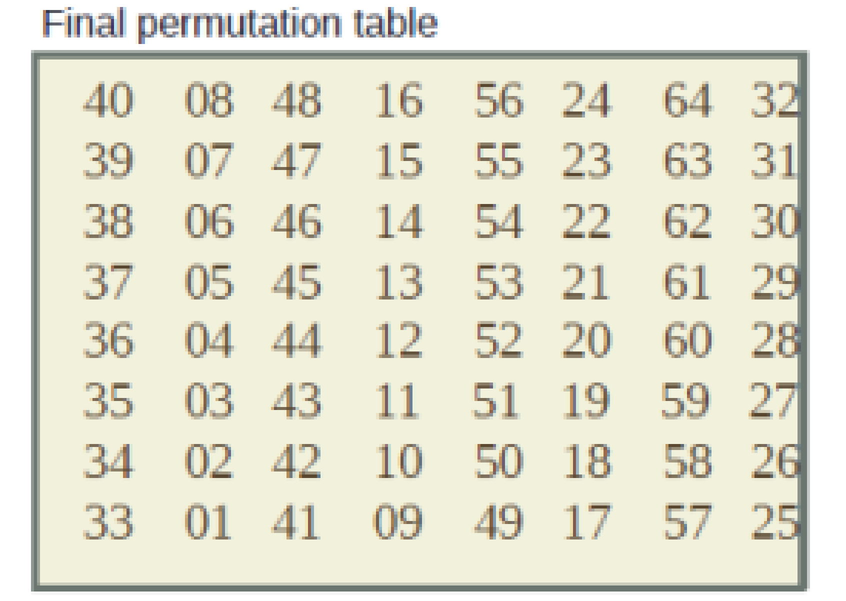
DES (d=16), AES-128 (d=10)

DES (initially Lucifer)



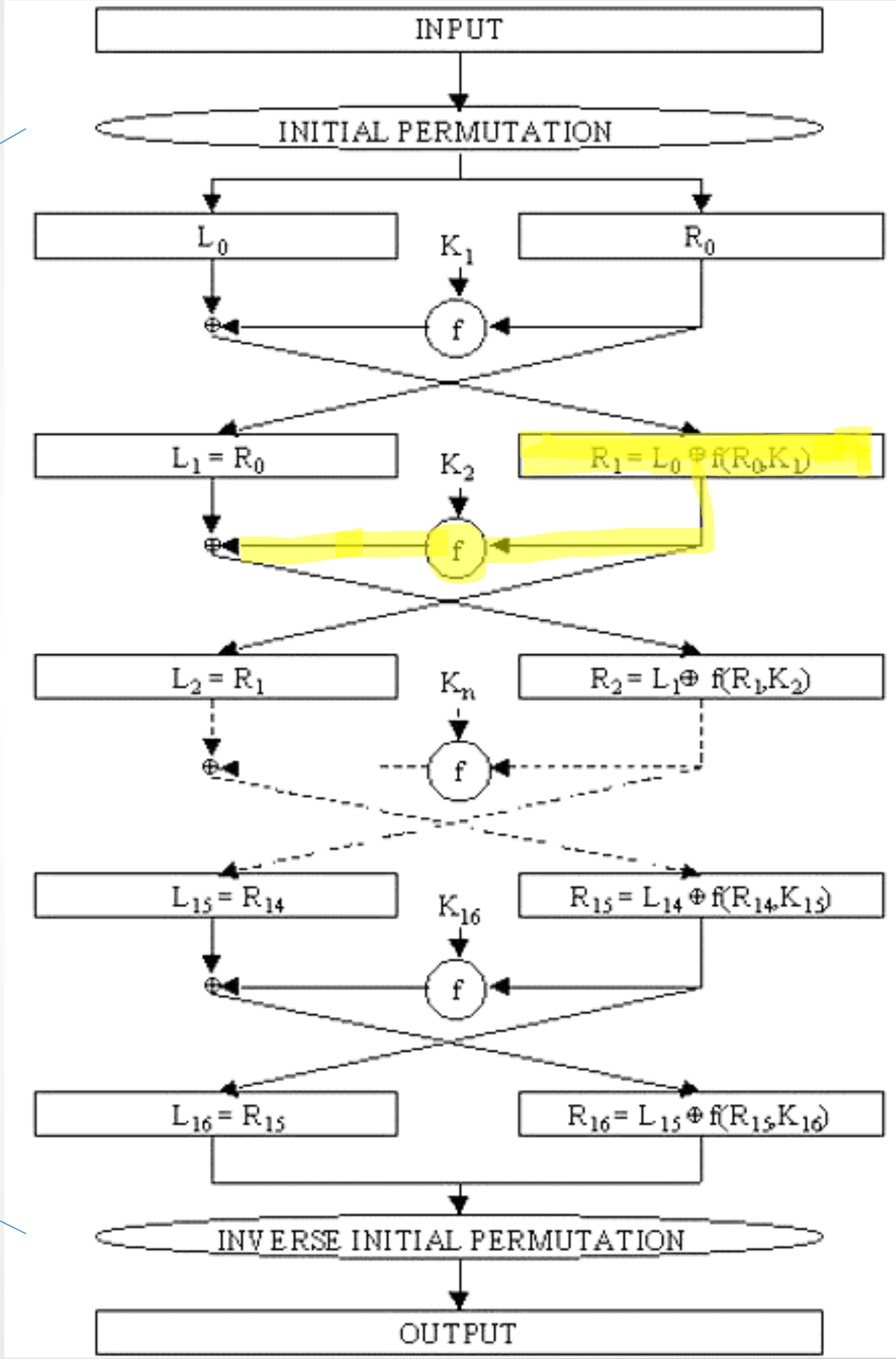
Initial permutation table

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

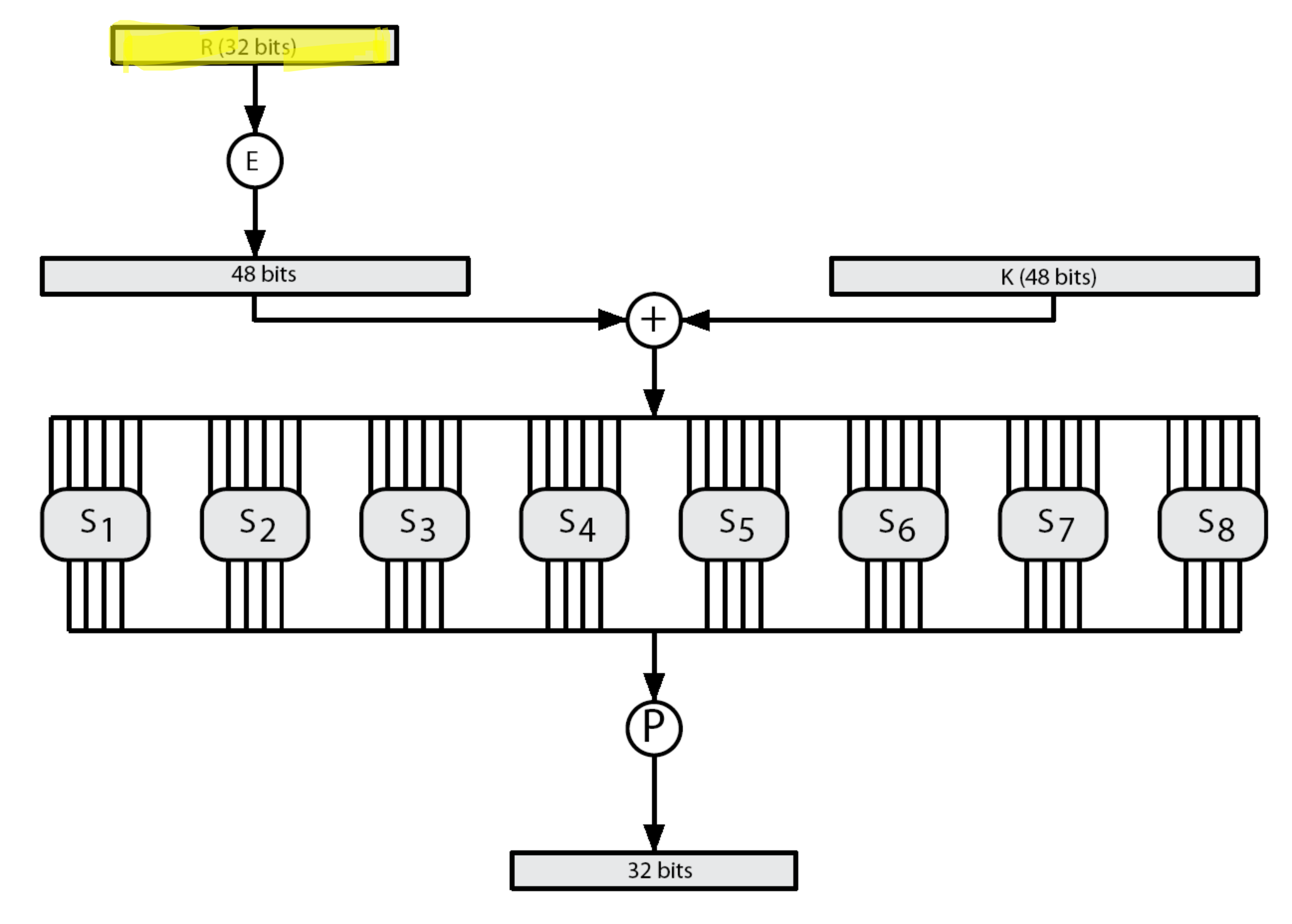


Final permutation table

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

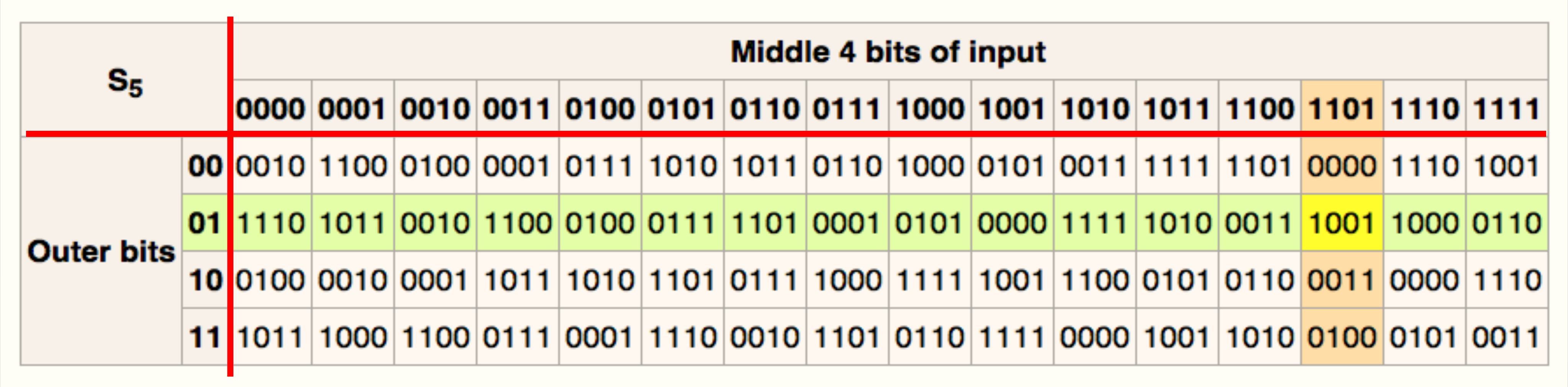

INPUT
INITIAL PERMUTATION
L_0
K_1
R_0
f
$L_1 = R_0$
K_2
$R_1 = L_0 \oplus f(R_0,K_1)$
f
$L_2 = R_1$
K_n
$R_2 = L_1 \oplus f(R_1,K_2)$
f
$L_{15} = R_{14}$
K_{16}
$R_{15} = L_{14} \oplus f(R_{14},K_{15})$
f
$L_{16} = R_{15}$
$R_{16} = L_{15} \oplus f(R_{15},K_{16})$
INVERSE INITIAL PERMUTATION
OUTPUT

The function $f(k_i, x)$



The S-boxes

$$S_i: \{0,1\}^6 \longrightarrow \{0,1\}^4$$



S_5		Middle 4 bits of input															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Outer bits	**00**	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

AES (Advanced Encryption Standard)

- Was approved by NIST in 2001 and named AES (it’s original name was Rijndael (Dutch pronunciation: [ˈrɛindaːl]).

- It’s made by repetition/rounds, similar to DES. But it has a different structure.

- The easiest way to show how it works is by animation.

(credit: Enrique Zabala, Cryptool.org)

AES Animation goes here

https://www.cryptool.org/en/cto/aes-animation

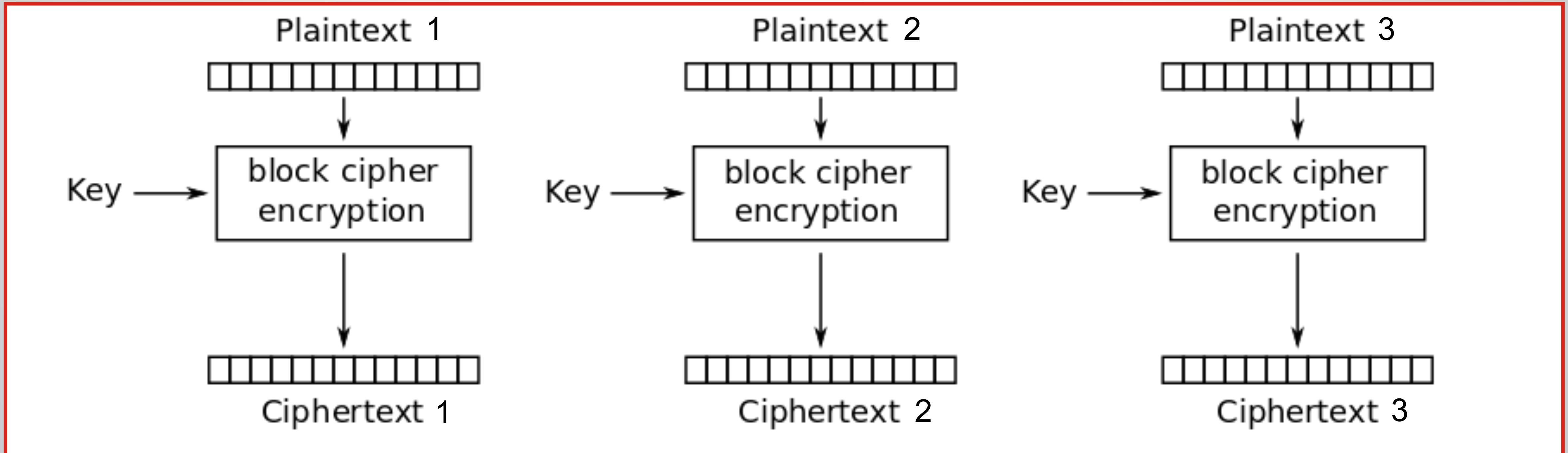
Blockciphers Modes of Operation

- Regardless of the internal architecture of “E” and “D” boxes/algorithms, we can use them in different configurations.

- Each configuration has a specific attribute, which makes it suitable for a specific family of applications.

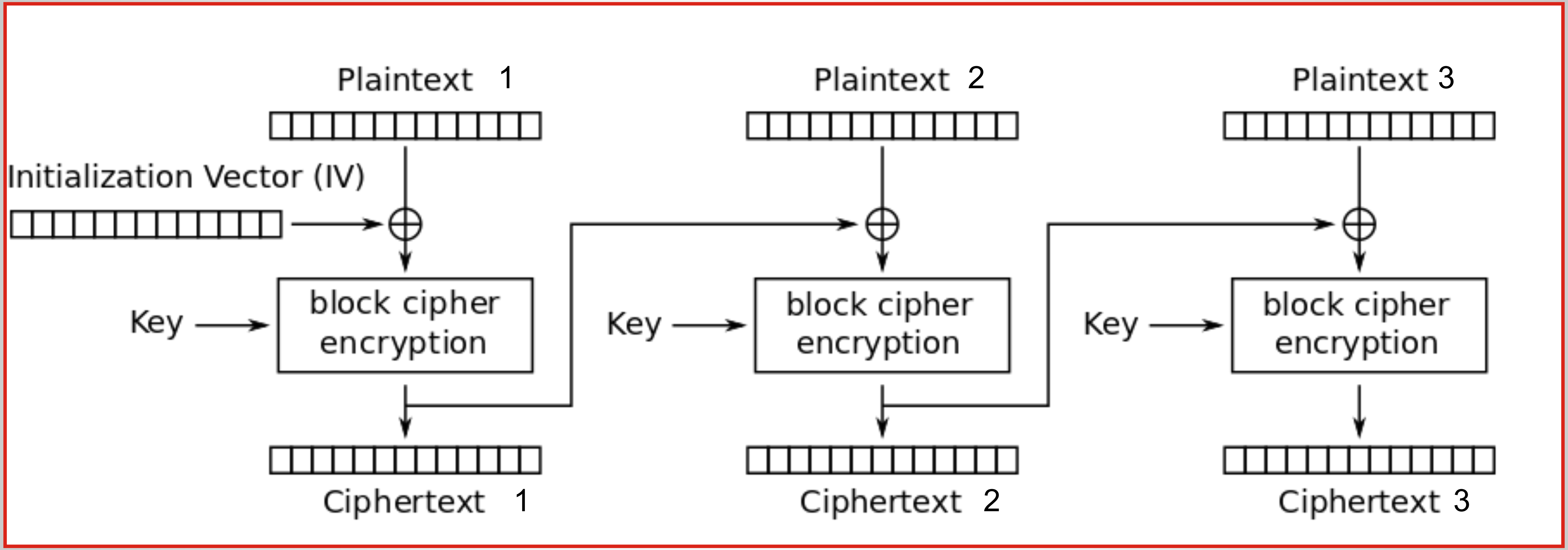
Electronic Code Book Mode

The plaintext is broken into block-size pieces and encrypted one by one. For a fixed key, this is like a look up table.



Electronic Codebook (ECB) mode encryption

Cipher Block Chaining Mode

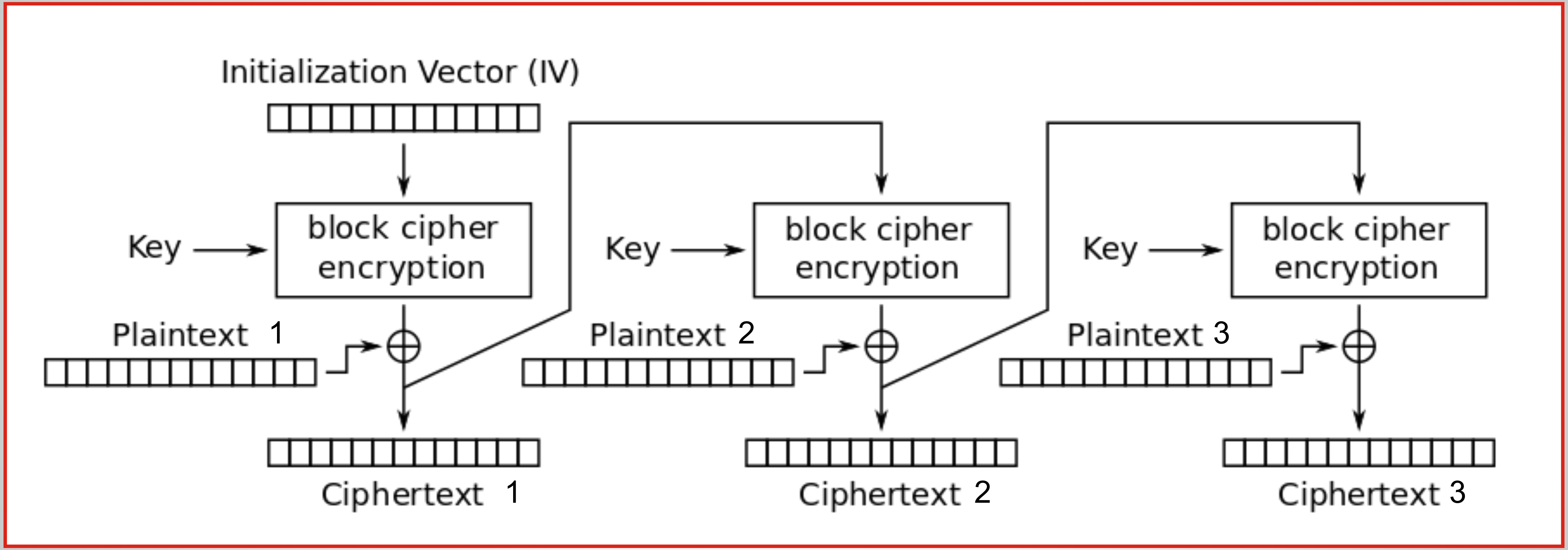


Cipher Block Chaining (CBC) mode encryption

$$C_i = E_K(P_i \oplus C_{i-1}), C_0 = IV$$

$$P_i = D_K(C_i) \oplus C_{i-1}, C_0 = IV.$$

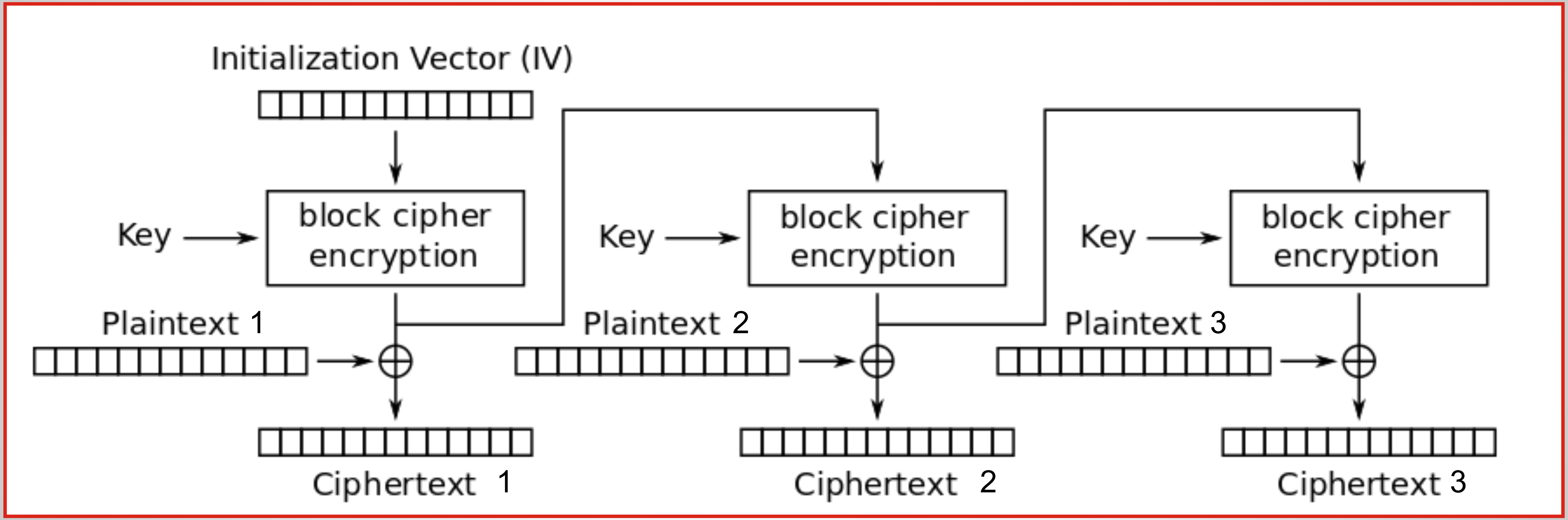
Cipher Feedback Mode



Cipher Feedback (CFB) mode encryption

$$C_0 = \mathrm{IV}$$
$$C_i = E_K(C_{i-1}) \oplus P_i$$
$$P_i = E_K(C_{i-1}) \oplus C_i$$

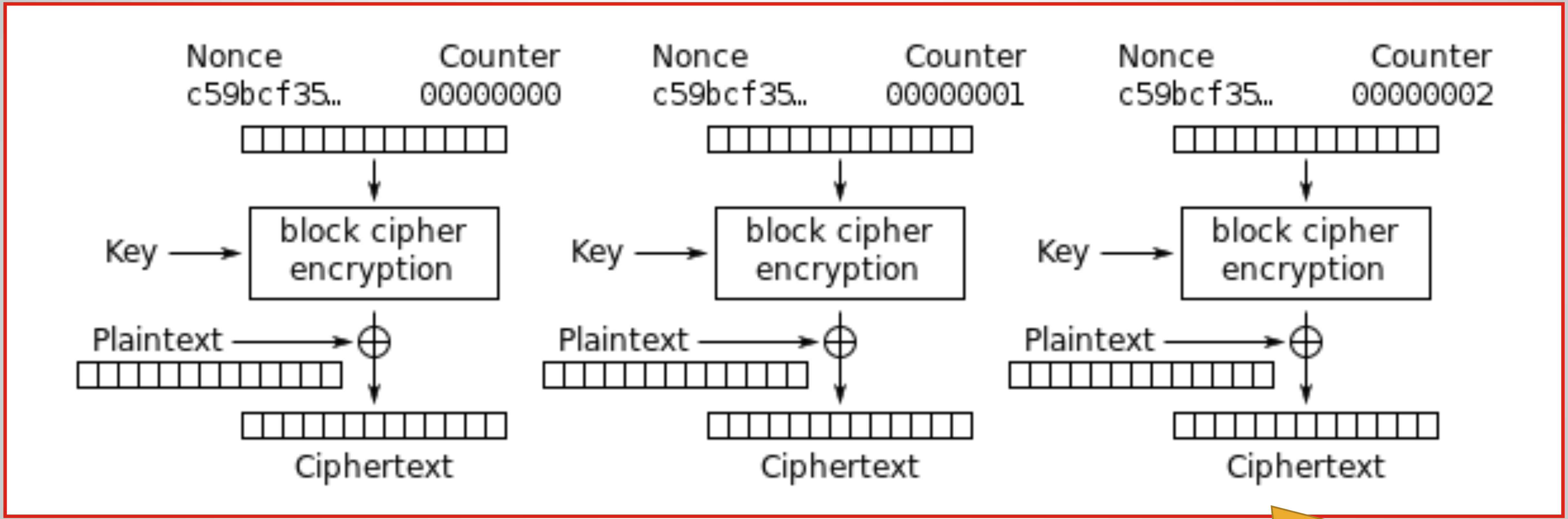
Output Feedback Mode



$$C_j = P_j \oplus O_j$$
$$P_j = C_j \oplus O_j$$
$$O_j = E_K(I_j)$$
$$I_j = O_{j-1}$$
$$I_0 = \mathrm{IV}$$

Output Feedback (OFB) mode encryption

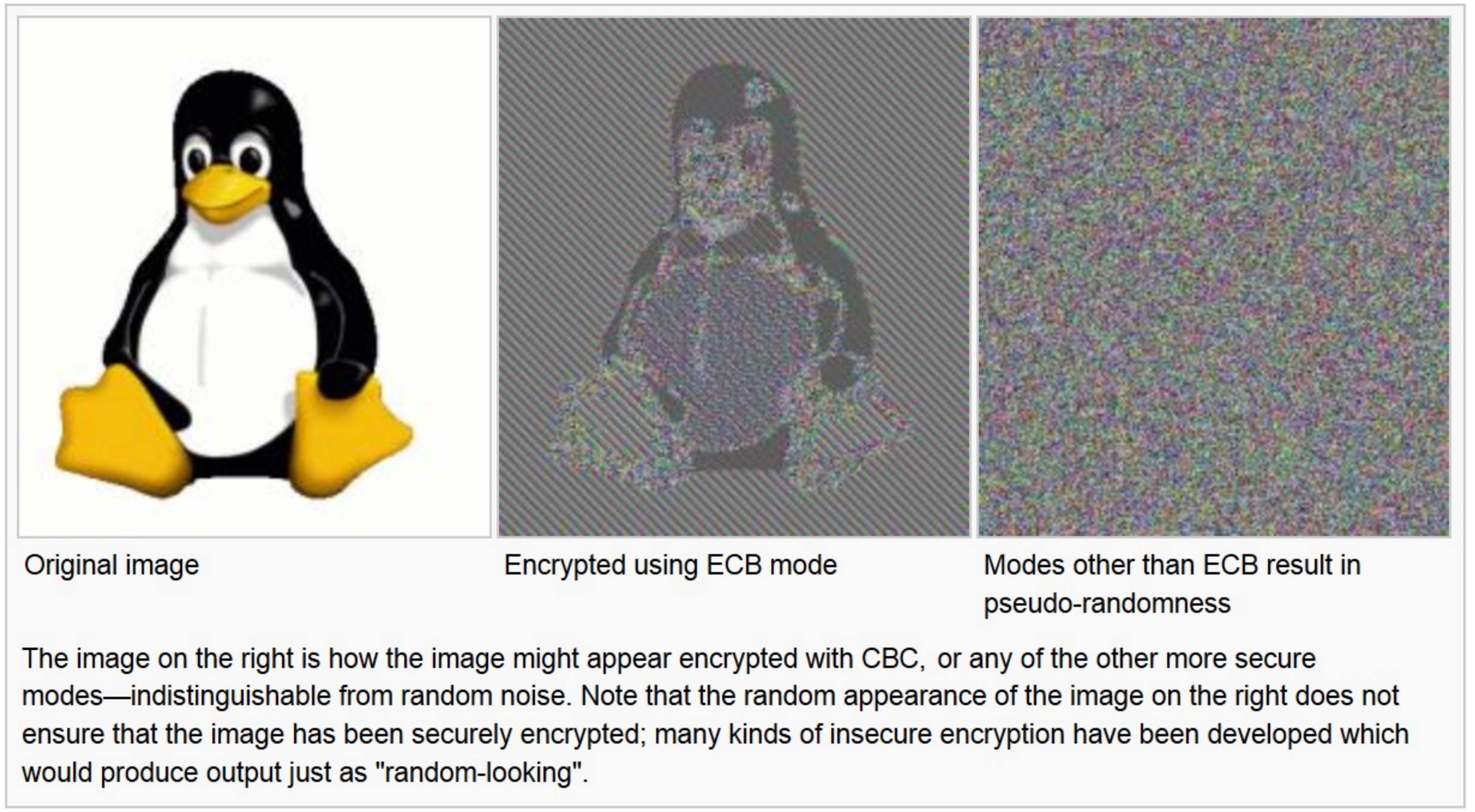
Counter Mode



Counter (CTR) mode encryption

Used in WiFi
(WPA2,3)

Difference of Outputs in an Application Scenario



Original image

Encrypted using ECB mode

Modes other than ECB result in pseudo-randomness

The image on the right is how the image might appear encrypted with CBC, or any of the other more secure modes—indistinguishable from random noise. Note that the random appearance of the image on the right does not ensure that the image has been securely encrypted; many kinds of insecure encryption have been developed which would produce output just as "random-looking".

(Wikipedia)

What Comes Next …

- We learned about the symmetric encryption algorithms and how they are made.

- We also learned how these algorithms are used in different modes.

- In the next video, we introduce a relevant yet different subject, i.e. hash functions, AKA digestion functions.

See you in the next video …