OS Project presentation 2023

MATTEO CARNEVALE
ALBERTO CASTRONOVO
GIUSEPPE CUTRERA
LUIGI PALMISANO

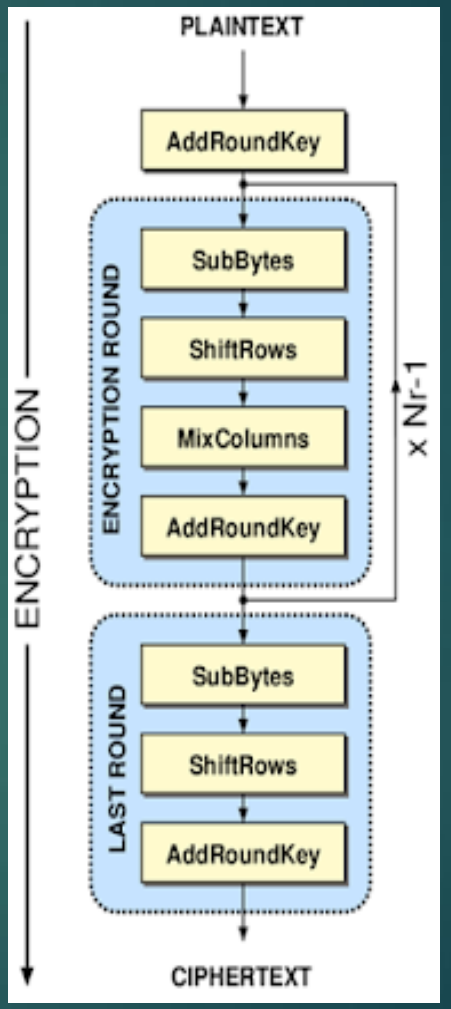
License

What is AES-256?

- Block cipher tecnhique substituting DES
- Plain and Cipher text of same size
- Input key
- Plain text size, Key lenght, Cipher text size
- Rounds: 10 for 128-bit keys,

Plenty of rounds…

- Substitution of bytes
- Shift rows
- Mix columns
- Add round key



SubBytes (ENC)

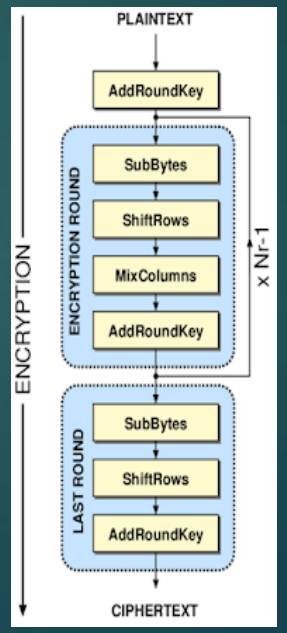
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	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

For example, EA → 87

EA	04	65	85
83	45	5D	96
5C	33	98	B0
F0	2D	AD	C5

↓

87	F2	4D	97
EC	6E	4C	90
4A	C3	46	E7
8C	D8	95	A6



SubBytes (DEC)

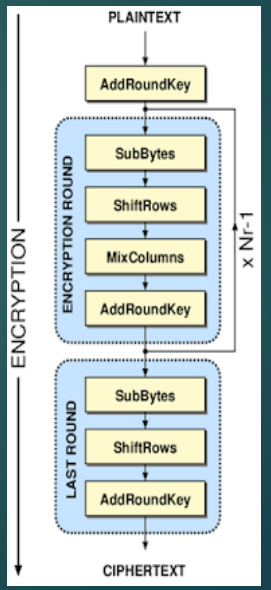
For example, 87 → EA

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

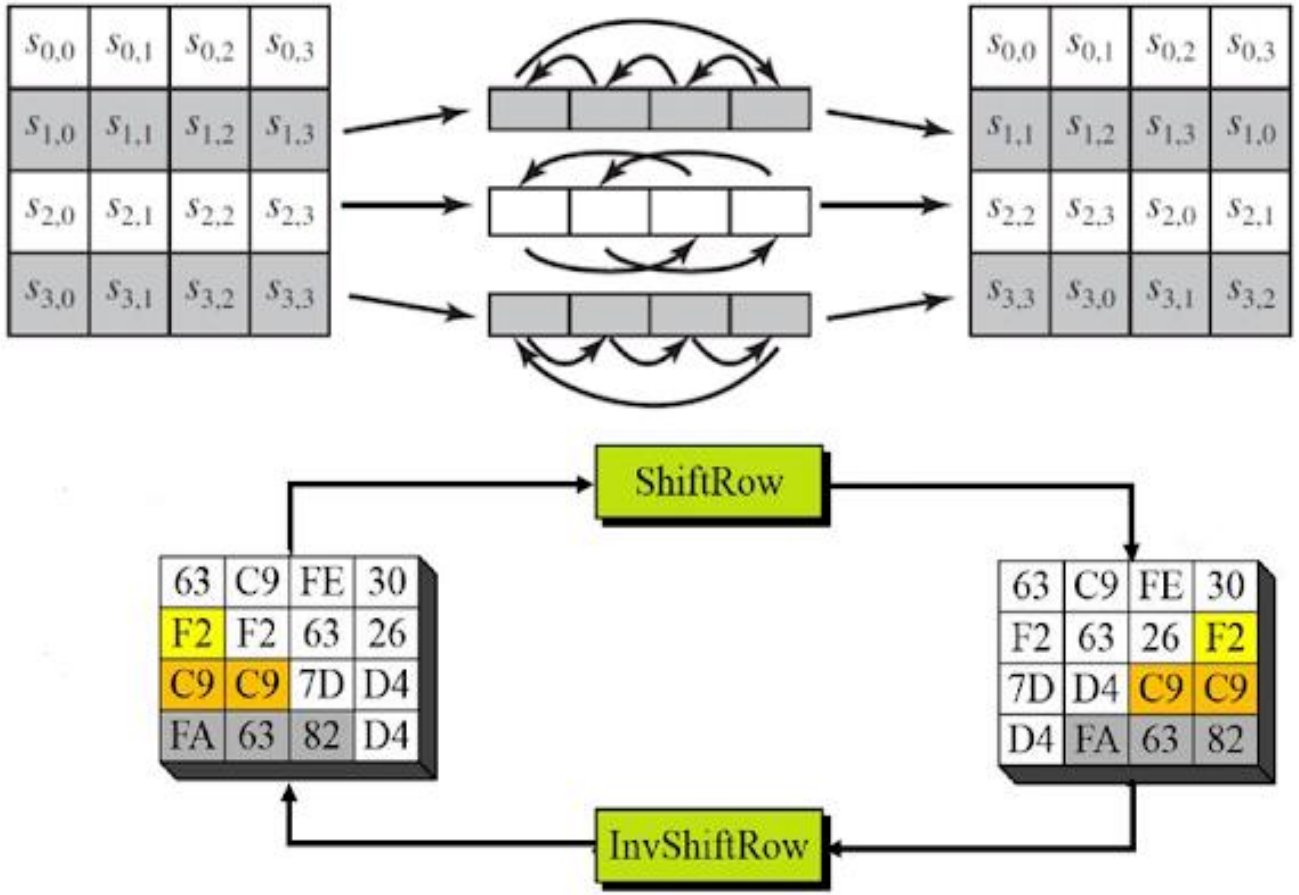
87	F2	4D	97
EC	6E	4C	90
4A	C3	46	E7
8C	D8	95	A6

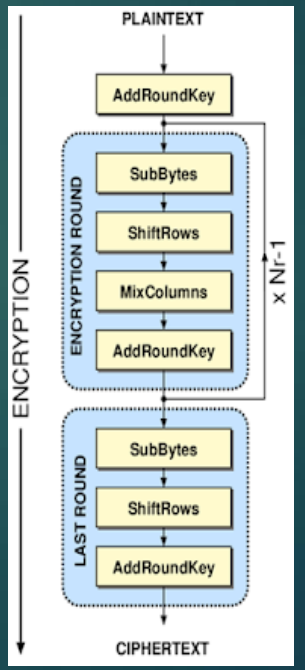
↓

EA	04	65	85
83	45	5D	96
5C	33	98	B0
F0	2D	AD	C5



ShiftRows


$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,1}$ $s_{1,2}$ $s_{1,3}$ $s_{1,0}$
$s_{2,2}$ $s_{2,3}$ $s_{2,0}$ $s_{2,1}$
$s_{3,3}$ $s_{3,0}$ $s_{3,1}$ $s_{3,2}$
ShiftRow
63 C9 FE 30
F2 F2 63 26
C9 C9 7D D4
FA 63 82 D4
63 C9 FE 30
F2 63 26 F2
7D D4 C9 C9
D4 FA 63 82
InvShiftRow


PLAINTEXT
AddRoundKey
SubBytes
ShiftRows
MixColumns
AddRoundKey
ENCRYPTION ROUND
x Nr-1
SubBytes
ShiftRows
AddRoundKey
LAST ROUND
ENCRYPTION
CIPHERTEXT

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

$$\begin{aligned} s'_{0,j} &= (2 \bullet s_{0,j}) \oplus (3 \bullet s_{1,j}) \oplus s_{2,j} \oplus s_{3,j} \\ s'_{1,j} &= s_{0,j} \oplus (2 \bullet s_{1,j}) \oplus (3 \bullet s_{2,j}) \oplus s_{3,j} \\ s'_{2,j} &= s_{0,j} \oplus s_{1,j} \oplus (2 \bullet s_{2,j}) \oplus (3 \bullet s_{3,j}) \\ s'_{3,j} &= (3 \bullet s_{0,j}) \oplus s_{1,j} \oplus s_{2,j} \oplus (2 \bullet s_{3,j}) \end{aligned}$$

Predefine Matrix | State Array | New State Array

2	3	1	1
1	2	3	1
1	1	2	3
3	1	1	2

*

87	F2	4D	97
6E	4C	90	EC
46	E7	4A	C3
A6	8C	D8	95

$\{02\} \cdot \{87\} \oplus \{03\} \cdot \{6E\} \oplus \{46\} \oplus \{A6\} = \{47\}$

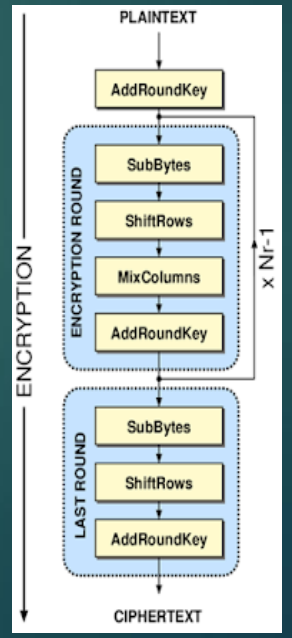
$\{87\} \oplus \{02\} \cdot \{6E\} \oplus \{03\} \cdot \{46\} \oplus \{A6\} = \{37\}$

$\{87\} \oplus \{6E\} \oplus \{02\} \cdot \{46\} \oplus \{03\} \cdot \{A6\} = \{94\}$

$\{03\} \cdot \{87\} \oplus \{6E\} \oplus \{46\} \oplus \{02\} \cdot \{A6\} = \{ED\}$

47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

MixColumns



47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

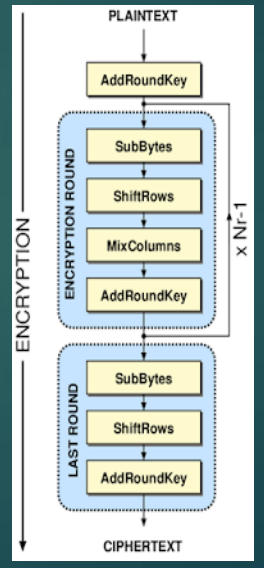
$\oplus$

AC	19	28	57
77	FA	D1	5C
66	DC	29	00
F3	21	41	6A

=

EB	59	8B	1B
40	2E	A1	C3
F2	38	13	42
1E	84	E7	D2

AddRoundKey

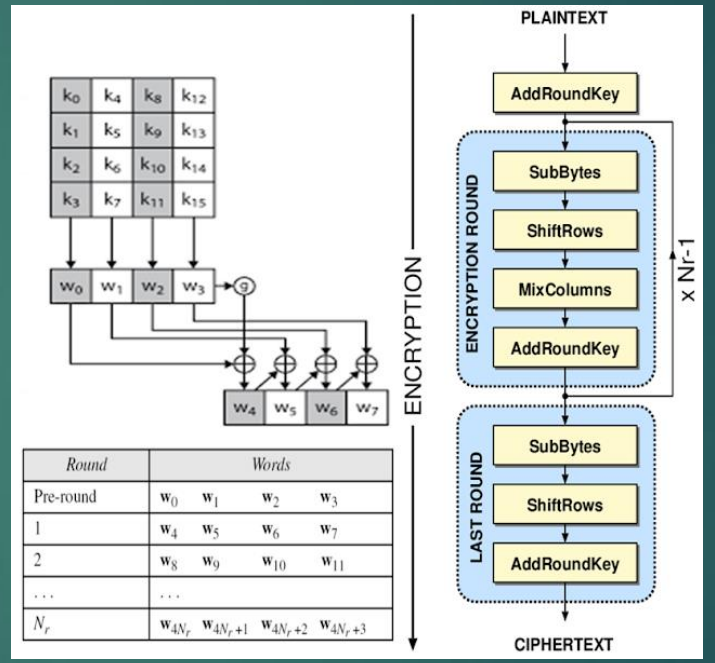


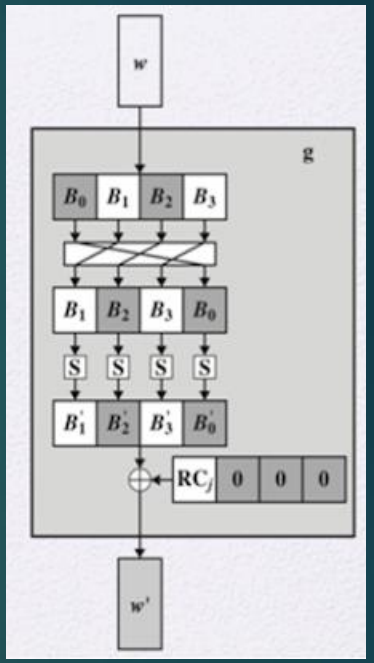
…but what is a RoundKey?

- Output in the key expansion process

- G function applied:

shifts, rotations and substitutions





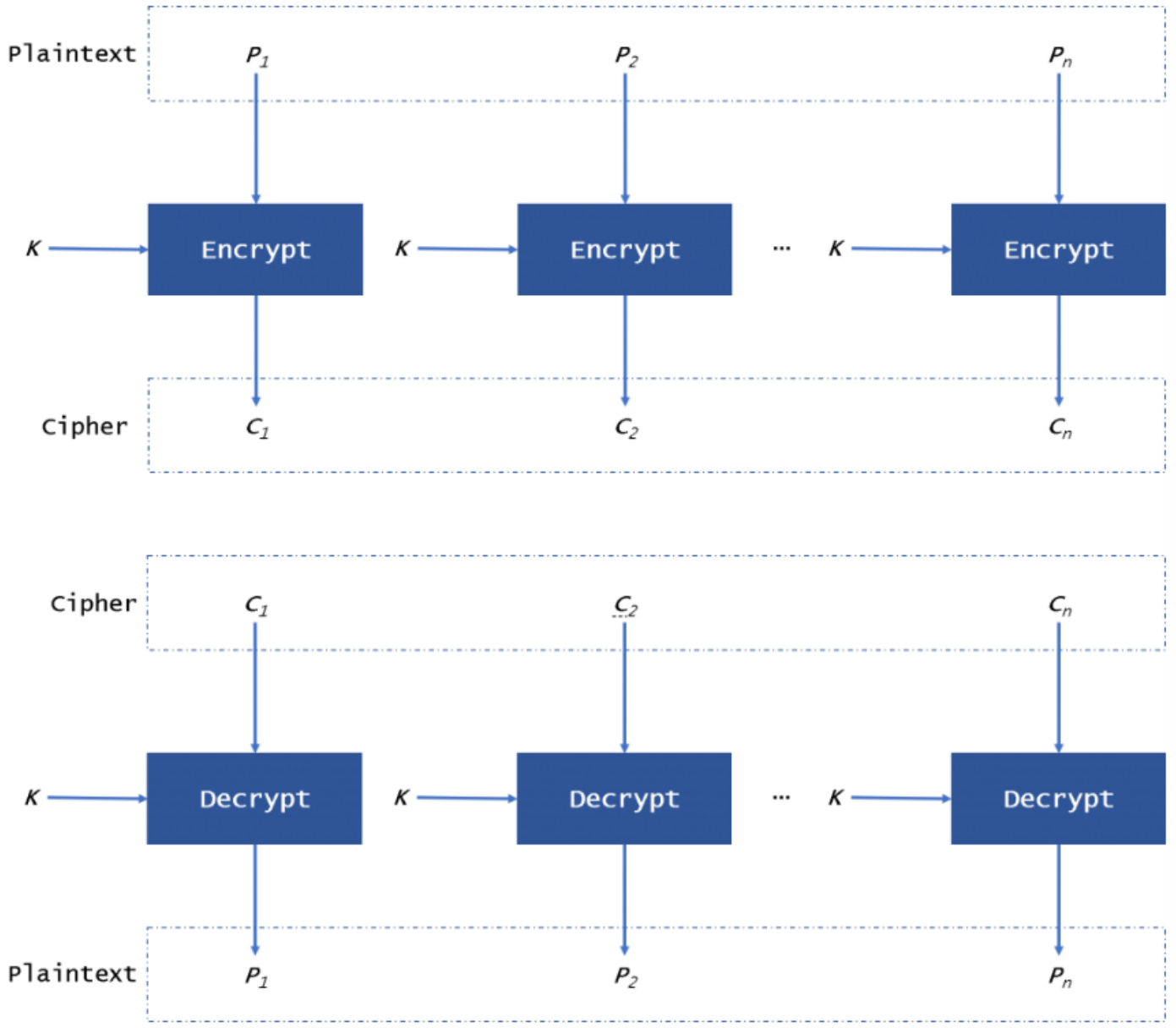
Cryptocore vulnerabilities

General attacks: an overview

- Side-Channel attacks: exploit runtime metrics (timing, power consumption, …) to gain information about the secret key
- Weak Key Derivation Function: a non-cryptographically-secure key may expose the system to preimage or collision attacks
- Key storage flaws: the key/IV should not be stored in plaintext, as an attacker could access it on the local file system
- Birthday attacks: two plaintexts could be encrypted with the same ciphertext (collision), which exposes digital signature vulnerabilities

General attacks: safe practices

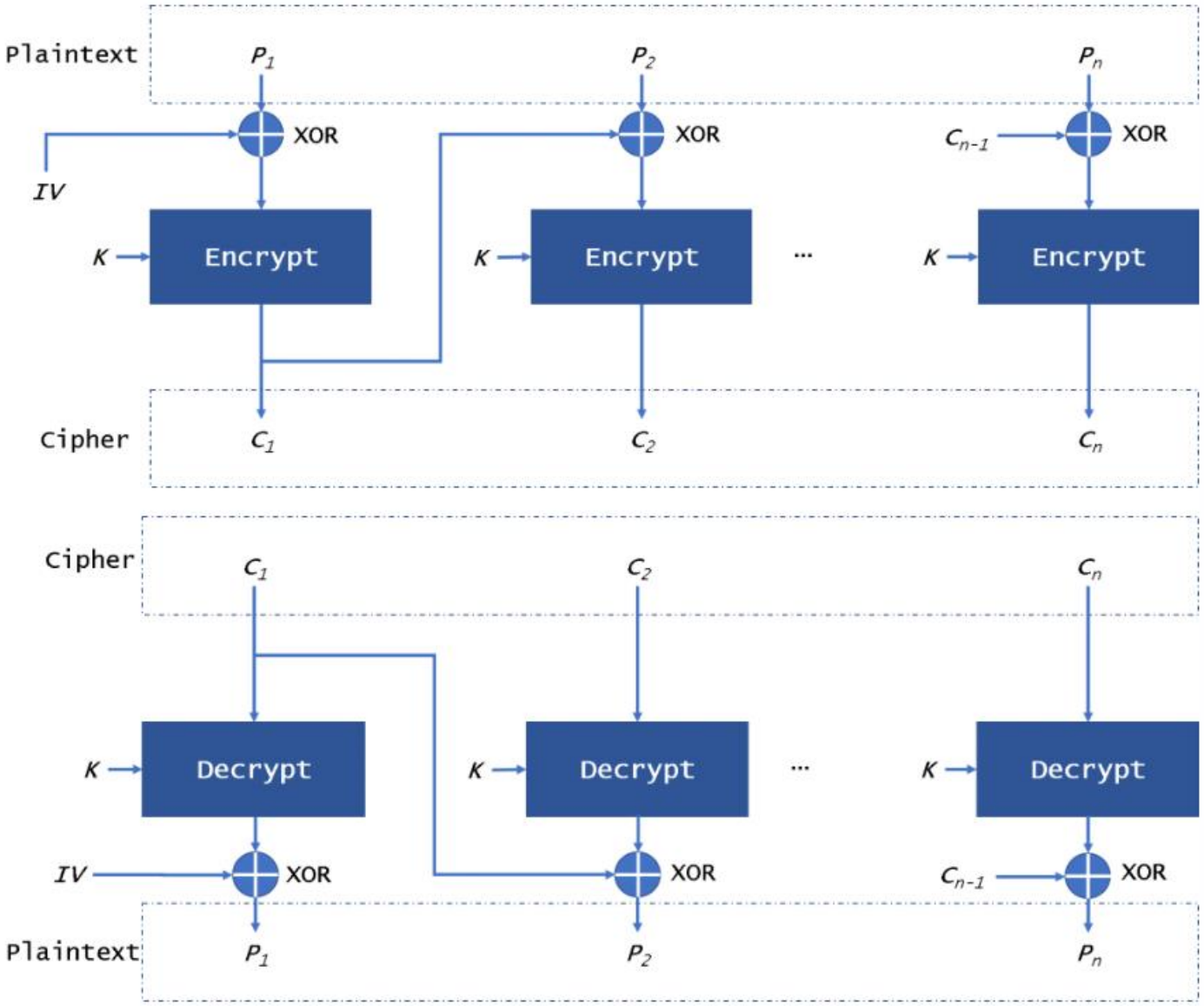
- Side-Channel attacks: implement time-constant encoding/decoding
- Weak KDF: ensure to use a cryptographically secure generator, change the key at least every 2^(blocks/2) blocks to avoid birthday attacks
- Key storage flaws: only exchange the key with secure exchange protocols. If possible, use HSMs
- Birthday attacks: use larger block sizes and/or change key and IV more frequently


Plaintext
P_1
P_2
P_n
K
Encrypt
K
Encrypt
...
K
Encrypt
Cipher
C_1
C_2
C_n
Cipher
C_1
C_2
C_n
K
Decrypt
K
Decrypt
...
K
Decrypt
Plaintext
P_1
P_2
P_n

ECB Mode

ECB Mode

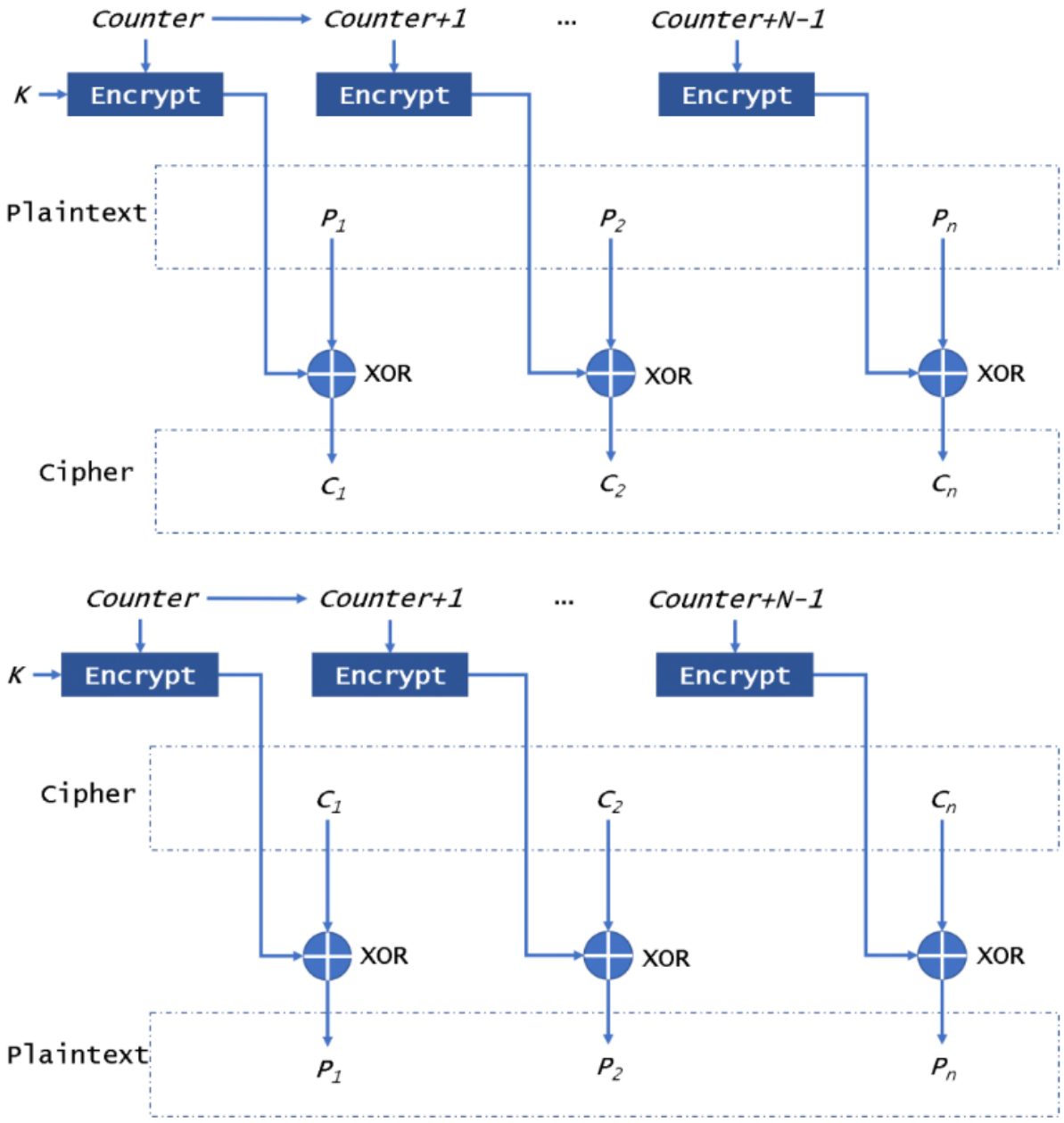
- Core vulnerability: 1-on-1 correspondence between plaintex and ciphertext
- If two ciphertext blocks are the same, so are their plaintexts, regardless of their order
- Frequency analysis attacks are an issue
- Needs padding → Padding Oracle attacks, Adaptive Chosen Plaintext attacks


Plaintext
P_1
P_2
P_n
XOR
XOR
XOR
C_{n-1}
IV
K
Encrypt
K
Encrypt
...
K
Encrypt
Cipher
C_1
C_2
C_n
Cipher
C_1
C_2
C_n
K
Decrypt
K
Decrypt
...
K
Decrypt
IV
XOR
XOR
C_{n-1}
XOR
Plaintext
P_1
P_2
P_n

CBC Mode

CBC Mode

- No 1-on-1 correspondence (order of blocks matters)
- Still needs padding (vulnerable to padding oracle attacks, if oracles are available)
- Problem: if a block is corrupted, so will all following blocks
- Vulnerable to bit-flipping attacks (ciphertext needed)


Counter
Counter+1
...
Counter+N-1
K
Encrypt
Encrypt
Encrypt
Plaintext
P_1
P_2
P_n
XOR
XOR
XOR
Cipher
C_1
C_2
C_n
Counter
Counter+1
...
Counter+N-1
K
Encrypt
Encrypt
Encrypt
Cipher
C_1
C_2
C_n
XOR
XOR
XOR
Plaintext
P_1
P_2
P_n

CTR Mode

CTR Mode

- Solved the problem of subsequent corruption (no propagation)
- More vulnerable to bit-flipping attacks (ciphertext needed), as only an XOR operation is performed on the plaintext