Chapitre 5

Cryptographie symétrique: Etude de l'algorithme AES

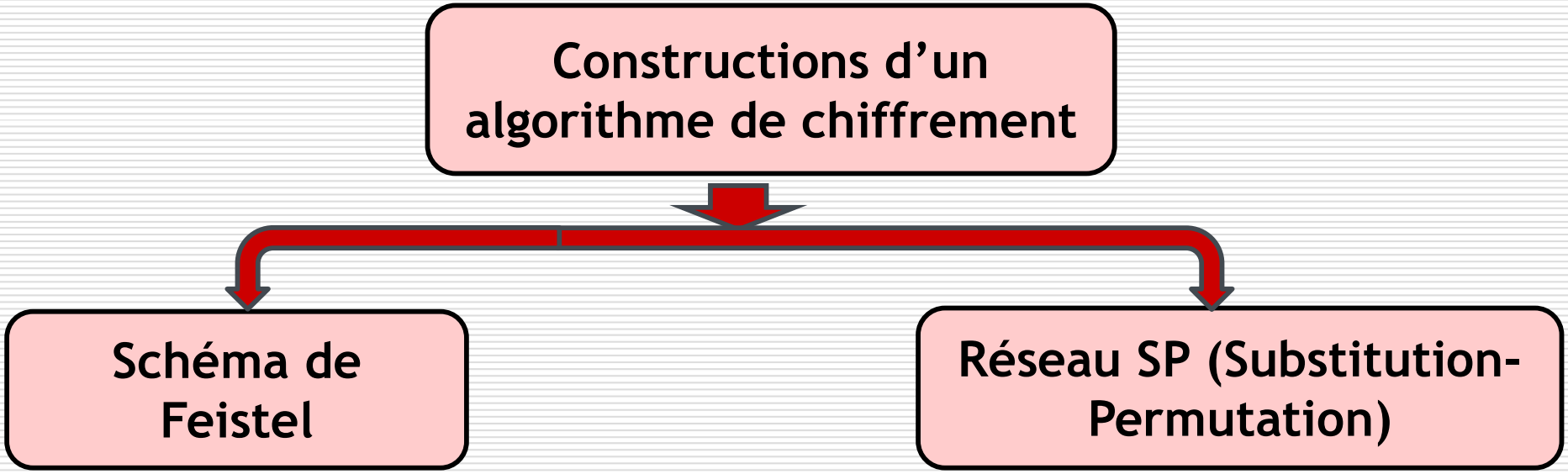
Dr.Wafa Berrayana | AU: 2025-2026

2025

Sécurité

- Deux principaux paramètres de sécurité
  - **La taille du bloc** (e.g. n = 64 ou 128 bits). Les modes opératoires permettent généralement des attaques quand plus de 2n/2 blocs sont chiffrés avec une même clé
  - **La taille de clé** (e.g. k = 128 bits). Pour un bon algorithme, la meilleure attaque doit coûter 2k opérations (recherche exhaustive)

Constructions classiques d'un algorithme de chiffrement



- DES
- 3DES

- AES

Présentation de AES

- AES : Advanced Encryption Standard)
- AES devint en 2001 le nouveau standard de chiffrement pour les organisations du gouvernement des États-Unis
- Il a été approuvé par la NSA (National Security Agency)

- **Utilisations courantes du chiffrement AES**
  - Sécurité Internet : HTTPS, TLS, SSH, VPN, transactions bancaires
  - Stockage de données : disques chiffrés, fichiers sensibles
  - Applications mobiles et bases de données
  - Wi-Fi (WPA2 / WPA3)

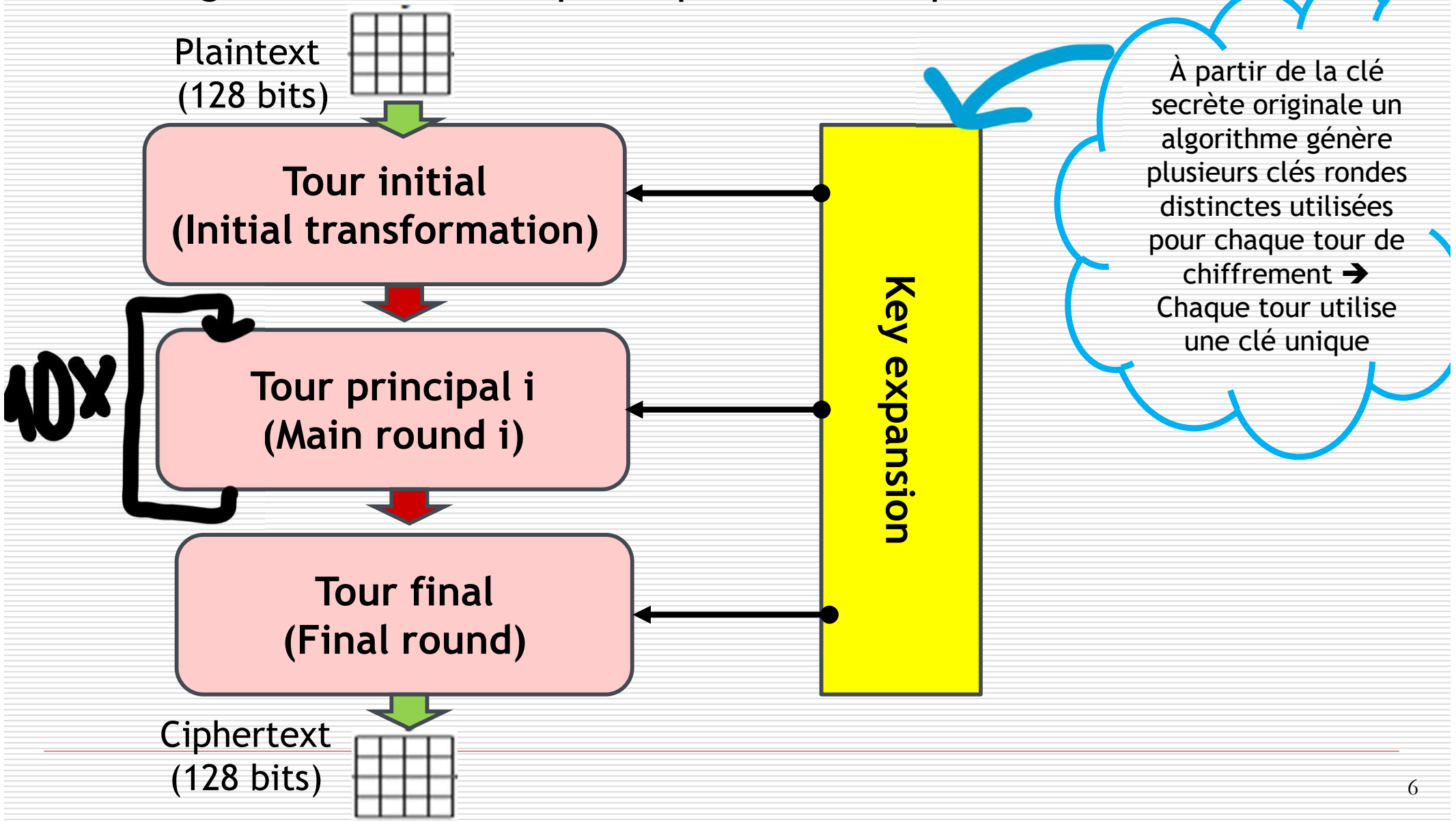
Caractéristiques clés de l'AES

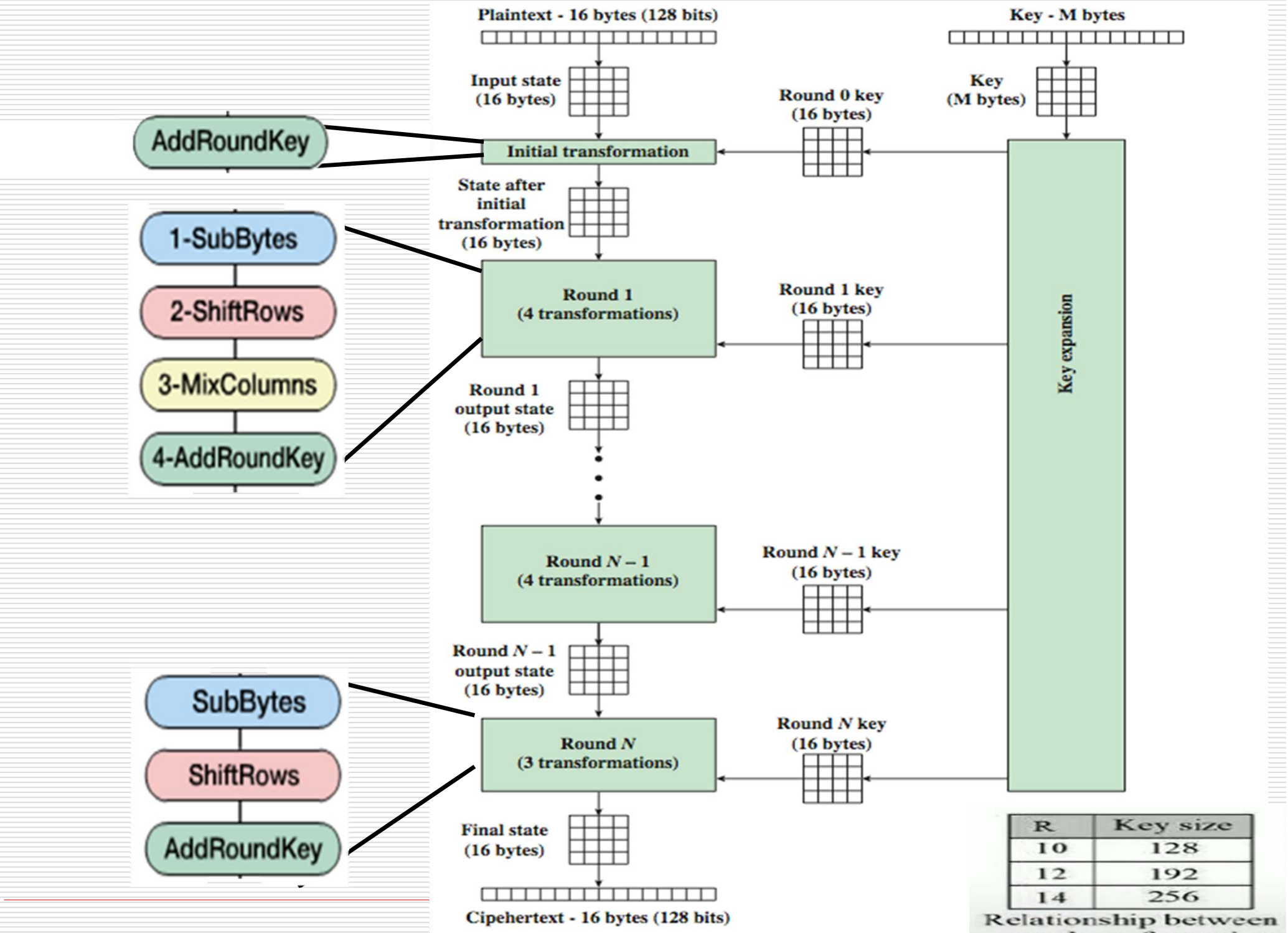
- l'AES s'appuie sur un **réseau de substitutions et de permutations (SP-network)**
- AES est un algorithme de chiffrement par blocs
- Chaque bloc de 128 bits est traité sous forme d'une **matrice 4×4 d'octets** appelée **State.**
- AES peut utiliser trois tailles de clé :

Variante	Taille de clé	Nombre de tours	Niveau de sécurité
AES-128	128 bits	10	Élevé
AES-192	192 bits	12	Très élevé
AES-256	256 bits	14	Très élevé (utilisé pour données sensibles ou militaires)

AES : Vue globale

- L'algorithme AES comporte plusieurs étapes essentielles

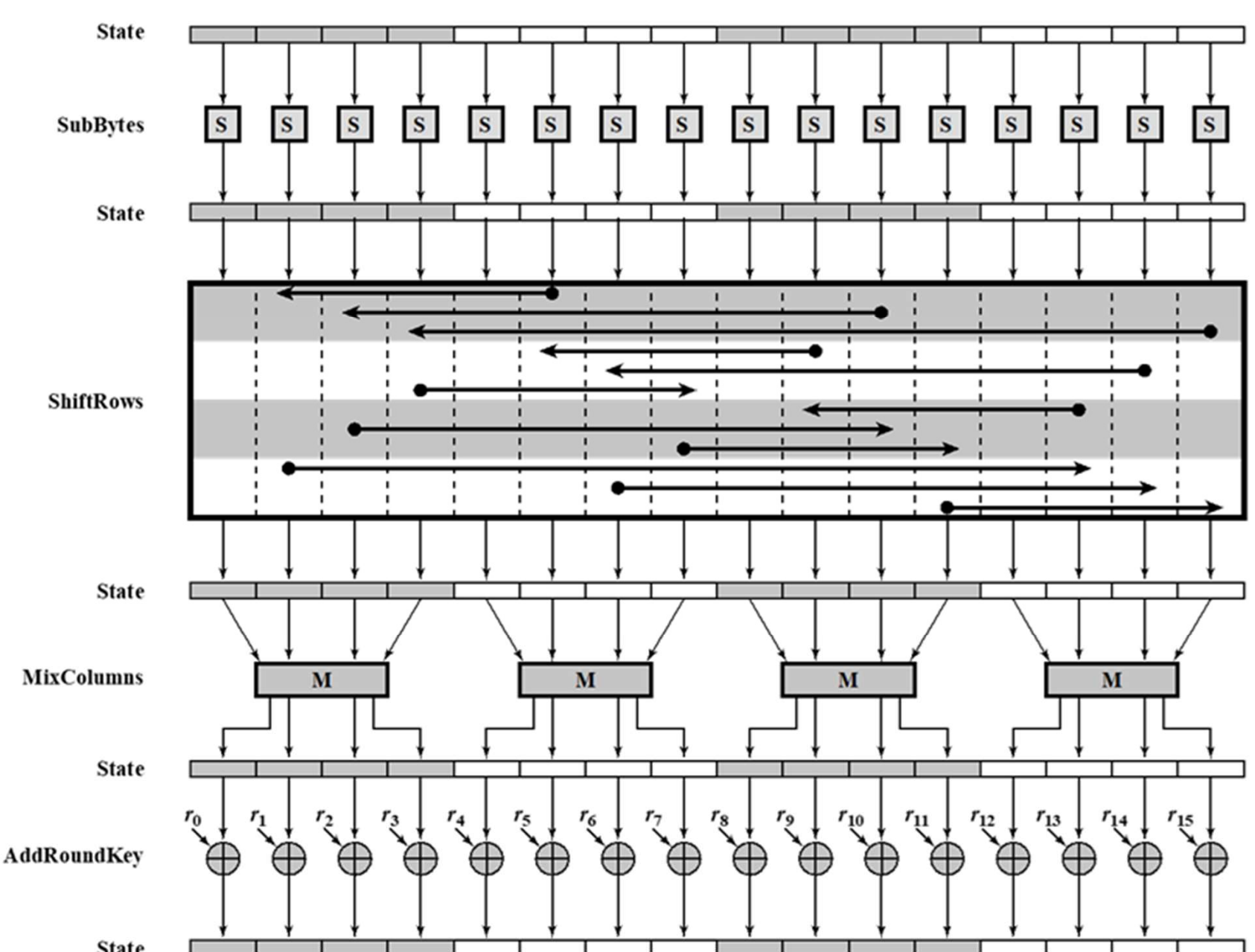




R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

AES encryption round: une autre perspective



Les fonctions de AES

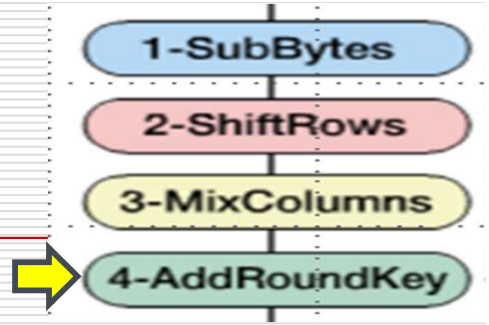
Étape	Fonction	Rôle principal
AddRoundKey	XOR entre le bloc et une sous-clé dérivée	Introduit la clé dans le processus
SubBytes	Chaque octet est remplacé via une table (S-box)	Introduit de la non-linéarité (confusion)
ShiftRows	Décale les lignes de la matrice d'octets	Mélange les données (diffusion)
MixColumns	Combine les octets de chaque colonne (opérations GF(2^8))	Mélange plus profondément les bits
AddRoundKey (final)	Applique la dernière sous-clé	Termine le chiffrement

Prétraitement des bits du texte à chiffrer

- **Taille du bloc:**
  - AES travaille toujours sur des blocs de **128 bits = 16 octets**, quelle que soit la taille de la clé (128, 192 ou 256 bits)
- Avant qu'AES ne commence à chiffrer les données, il doit **convertir le texte en clair (plaintext)** en une **structure de données appelée "état" (state)**
- Cette structure est une **matrice 4×4 d'octets**, qui sert de base à toutes les transformations de l'algorithme
- **Exemple :** Plaintext (en hexadécimal) :
  - 32 88 31 e0 43 5a 31 37 f6 30 98 07 a8 8d a2 34
  - Formation de la matrice d'état

Colonne 0	Colonne 1	Colonne 2	Colonne 3
32	43	f6	a8
88	5a	30	8d
31	31	98	a2
e0	37	07	34

Déroulement de AES (1)



- **AddRoundKey (1/3)** est **la plus essentielle** du chiffrement AES, car c'est **là que la clé intervient directement**
  - Lors de cette étape, l'**état** (la matrice 4×4 d'octets représentant le bloc de données) est **combiné** avec une **clé de ronde** (une sous-clé dérivée de la clé principale) à l'aide de l'opération **XOR** (ou exclusif, « $\oplus$ »).
  - **Opération**
    - Pour chaque octet de l'état :

$$\text{État}[i][j] = \text{État}[i][j] \oplus \text{CléDeRonde}[i][j]$$

    - où :
      - $\oplus$ = opération XOR bit à bit
      - **État** = bloc courant de données (16 octets)
      - **CléDeRonde** = sous-clé correspondant à la ronde actuelle (16 octets)

Déroulement de AES (2)

- AddRoundKey (2/3)
  - But
    - Cette étape ajoute le secret de la clé directement dans les données.Elle assure la confusion — une propriété essentielle pour la sécurité du chiffrement : chaque bit de la clé influence les bits du texte chiffré
  - Exemple (1)

C'est la clé de la ronde actuelle

$$\begin{pmatrix} 54 & 4F & 4E & 20 \\ 77 & 6E & 69 & 54 \\ 6F & 65 & 6E & 77 \\ 20 & 20 & 65 & 6F \end{pmatrix} \oplus \begin{pmatrix} 54 & 73 & 20 & 67 \\ 68 & 20 & 4B & 20 \\ 61 & 6D & 75 & 46 \\ 74 & 79 & 6E & 75 \end{pmatrix} = \begin{pmatrix} 00 & 3C & 6E & 47 \\ 1F & 4E & 22 & 74 \\ 0E & 08 & 1B & 31 \\ 54 & 59 & 0B & 1A \end{pmatrix}$$

C'est l'état de la ronde actuelle

C'est le nouvel état après l'étape AddRoundKey

Déroulement de AES (3)

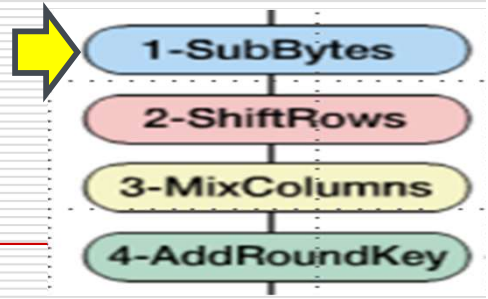
- AddRoundKey (3/3)
  - Exemple (2) : une autre représentation
    - On applique XOR octet par octet :

```
État :       [19 a0 9a e9 3d f4 c6 f8 e3 e2 8d 48 be 2b 2a 08]
CléDeRonde : [a0 88 23 2a fa 54 a3 6c fe 2c 39 76 17 b1 39 05]
```

    - C'est le nouvel **état** après l'étape **AddRoundKey**.

```
Résultat :   [b9 28 b9 c3 c7 a0 65 94 1d ce b4 3e a9 9a 13 0d]
```

Déroulement de AES (4)



- **SubBytes (chiffrement) /InvSubBytes (déchiffrement)**
  - **Opération:** Chaque octet d'état individuel est mappé dans un nouvel octet de la manière suivante:
    - les 4 bits à gauche de l'octet représentent la **valeur de ligne**. Les 4 bits les plus à droite sont représentent **valeur de colonne**
    - Ces valeurs de ligne et de colonne servent d'index dans le **S-box** pour sélectionner une valeur de sortie 8 bits unique
  - **But:**
    - Introduire de la non-linéarité → rend le chiffrement résistant aux attaques linéaires et différentielles.
    - Mélanger les bits de chaque octet de manière complexe.
    - Garantir qu'un petit changement dans un octet d'entrée entraîne un grand changement dans le résultat.
  - **Exemple :** la valeur hexadécimale [95] fait référence à la ligne 9, colonne 5

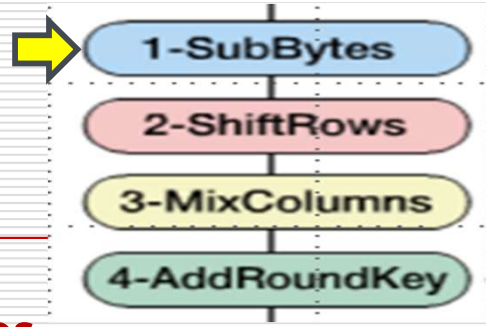
Déroulement de AES (5)

- **SubBytes**
  - **Exemple** Lire à partir de la matrice S-Box la cellule correspondante à

3C

ligne = 3
Colonne = C

$$\text{state} = \begin{pmatrix} 00 & 3C & 6E & 47 \\ 1F & 4E & 22 & 74 \\ 0E & 08 & 1B & 31 \\ 54 & 59 & 0B & 1A \end{pmatrix} \Rightarrow \text{S\_box(State)} = \begin{pmatrix} 63 & EB & 9F & A0 \\ C0 & 2F & 93 & 92 \\ AB & 30 & AF & C7 \\ 20 & CB & 2B & A2 \end{pmatrix}$$



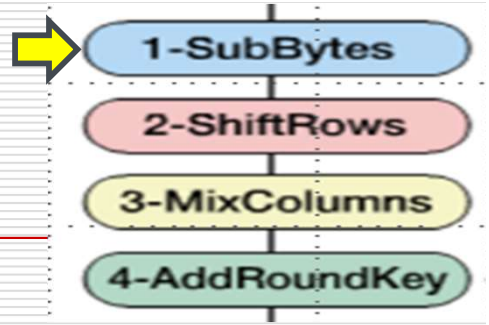
Déroulement de AES (6)

- The **S-box** est utilisé au moment du chiffrement par **SubBytes**

		y															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
x	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

(a) S-box

Déroulement de AES (7)

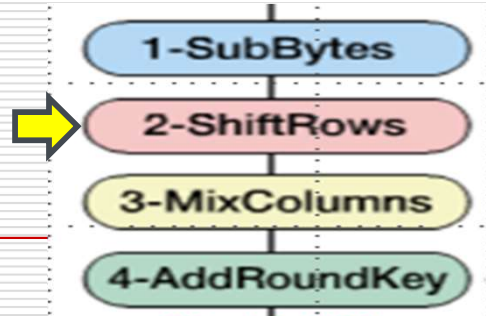


- The **Inverse S-box** est utilisé au moment du déchiffrement par **InvSubBytes**

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

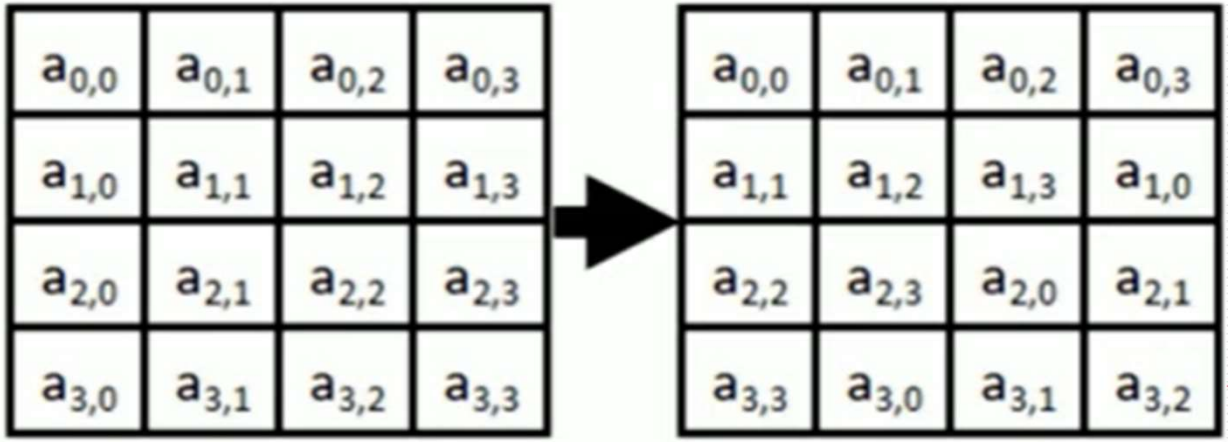
(b) Inverse S-box

Déroulement de AES (8)

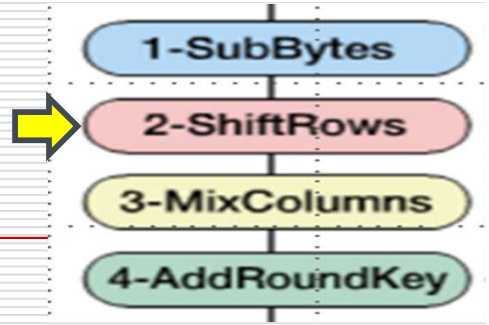


ShiftRows (chiffrement)/Inv ShiftRows (déchiffrement) (1/2)

- On travaille toujours sur la matrice état de 4×4 octets
- La **1re ligne** reste inchangée.
- La **2e ligne** est décalée d'un octet vers la gauche.
- La **3e ligne** est décalée de deux octets.
- La **4e ligne** est décalée de trois octets.
- **But :** Ce décalage **casse la structure** de la matrice pour augmenter la **diffusion** des données (mélange des bits sur plusieurs colonnes)



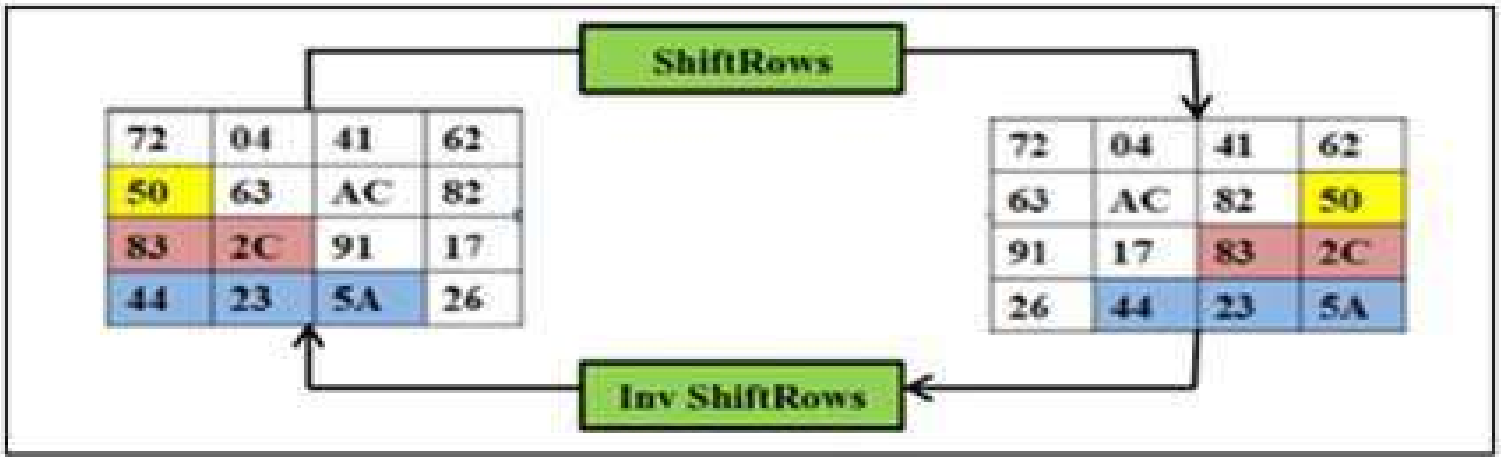
Déroulement de AES (9)



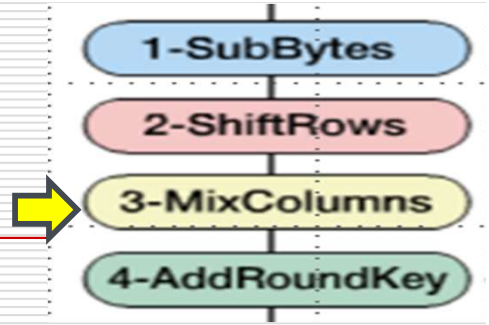
- ShiftRows (chiffrement)/Inv ShiftRows (déchiffrement) (2/2)
  - Exemple (1)

$$\begin{pmatrix} 63 & EB & 9F & A0 \\ C0 & 2F & 93 & 92 \\ AB & 30 & AF & C7 \\ 20 & CB & 2B & A2 \end{pmatrix} \Rightarrow \begin{pmatrix} 63 & EB & 9F & A0 \\ 2F & 93 & 92 & C0 \\ AF & C7 & AB & 30 \\ A2 & 20 & CB & 2B \end{pmatrix}$$

  - Exemple (2)

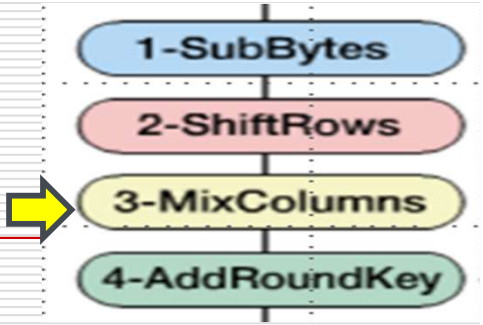


Déroulement de AES (10)



- **MixColumns (chiffrement)/InvMixColumns (déchiffrement) (1/5)** est une étape cruciale du chiffrement **AES**, car c'est elle qui assure une **forte diffusion** entre les octets du bloc
- **Buts**
  - **Diffusion :** mélanger les octets de chaque colonne de la **matrice d'état** pour rendre la relation entre le texte clair et le texte chiffré beaucoup plus complexe.
  - **Propagation :** Une variation d'un seul octet affecte toute la colonne
  - **Renforcement :** Rend le chiffrement plus résistant aux attaques différentielles et linéaires
- **Opération**
  - **Définition mathématique**
    - Chaque colonne (composée de 4 octets) est multipliée par une **matrice fixe** dans le corps fini $GF(2^8)$.

Déroulement de AES (11)



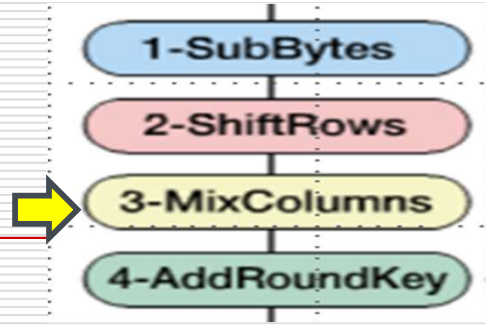
- MixColumns (chiffrement)/InvMixColumns (déchiffrement) (2/5)
  - La matrice fixe

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

$$\begin{bmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \cdot \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

  - Toutes les opérations se font dans le corps fini $GF(2^8)$,défini par le **polynôme irréductible :** $x^8 + x^4 + x^3 + x + 1$

Déroulement de AES (12)



- MixColumns (chiffrement)/InvMixColumns (déchiffrement) (3/5)
  - **Détail du calcul**
    - Pour une colonne donnée $[b_0, b_1, b_2, b_3]$,les nouveaux octets sont :

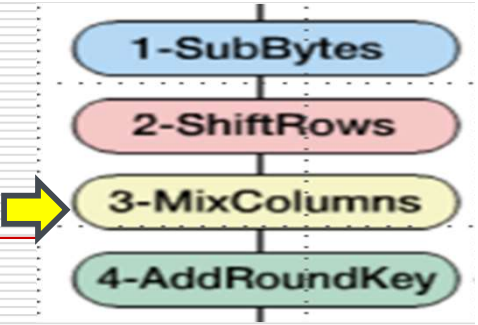
$$b'_0 = (02 \cdot b_0) \oplus (03 \cdot b_1) \oplus (01 \cdot b_2) \oplus (01 \cdot b_3)$$
$$b'_1 = (01 \cdot b_0) \oplus (02 \cdot b_1) \oplus (03 \cdot b_2) \oplus (01 \cdot b_3)$$
$$b'_2 = (01 \cdot b_0) \oplus (01 \cdot b_1) \oplus (02 \cdot b_2) \oplus (03 \cdot b_3)$$
$$b'_3 = (03 \cdot b_0) \oplus (01 \cdot b_1) \oplus (01 \cdot b_2) \oplus (02 \cdot b_3)$$

  - **Exemple (1)**
    - Prenons la colonne suivante : [**DB, 13, 53, 45**]

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

    - Valeurs de départ
      - $b0$=DB, $b1$=13,$b2$=53,$b3$=45b0 =DB,b1 =13,b2 =53,b3 =45

Déroulement de AES (13)

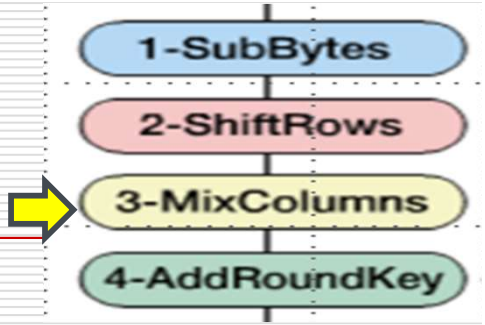


MixColumns (chiffrement)/InvMixColumns (déchiffrement) (4/5)

- Règles de multiplication dans $GF(2^8)$
  - Dans AES :
    - **×1** → identique
    - **×2** → décalage à gauche d'un bit (modifié par **1B** si le bit le plus haut était 1)
    - **×3** → (×2 ⊕ valeur)
    - **Note:** Si un octet dépasse 0xFF, on fait un XOR avec 0x1B (le polynôme réducteur)
- **Exemple (2)**

$$\begin{pmatrix} 02\,03\,01\,01 \\ 01\,02\,03\,01 \\ 01\,01\,02\,03 \\ 03\,01\,01\,02 \end{pmatrix} \begin{pmatrix} 63 & EB & 9F & A0 \\ 2F & 93 & 92 & C0 \\ AF & C7 & AB & 30 \\ A2 & 20 & CB & 2B \end{pmatrix} = \begin{pmatrix} BA & 84 & E8 & 1B \\ 75 & A4 & 8D & 40 \\ F4 & 8D & 06 & 7D \\ 7A & 32 & 0E & 5D \end{pmatrix}$$

Déroulement de AES (14)



- MixColumns (chiffrement)/InvMixColumns (déchiffrement) (5/5)
  - Exemple (2)

$$\begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} 63 & EB & 9F & A0 \\ 2F & 93 & 92 & C0 \\ AF & C7 & AB & 30 \\ A2 & 20 & CB & 2B \end{pmatrix} = \begin{pmatrix} BA & 84 & E8 & 1B \\ 75 & A4 & 8D & 40 \\ F4 & 8D & 06 & 7D \\ 7A & 32 & 0E & 5D \end{pmatrix}$$