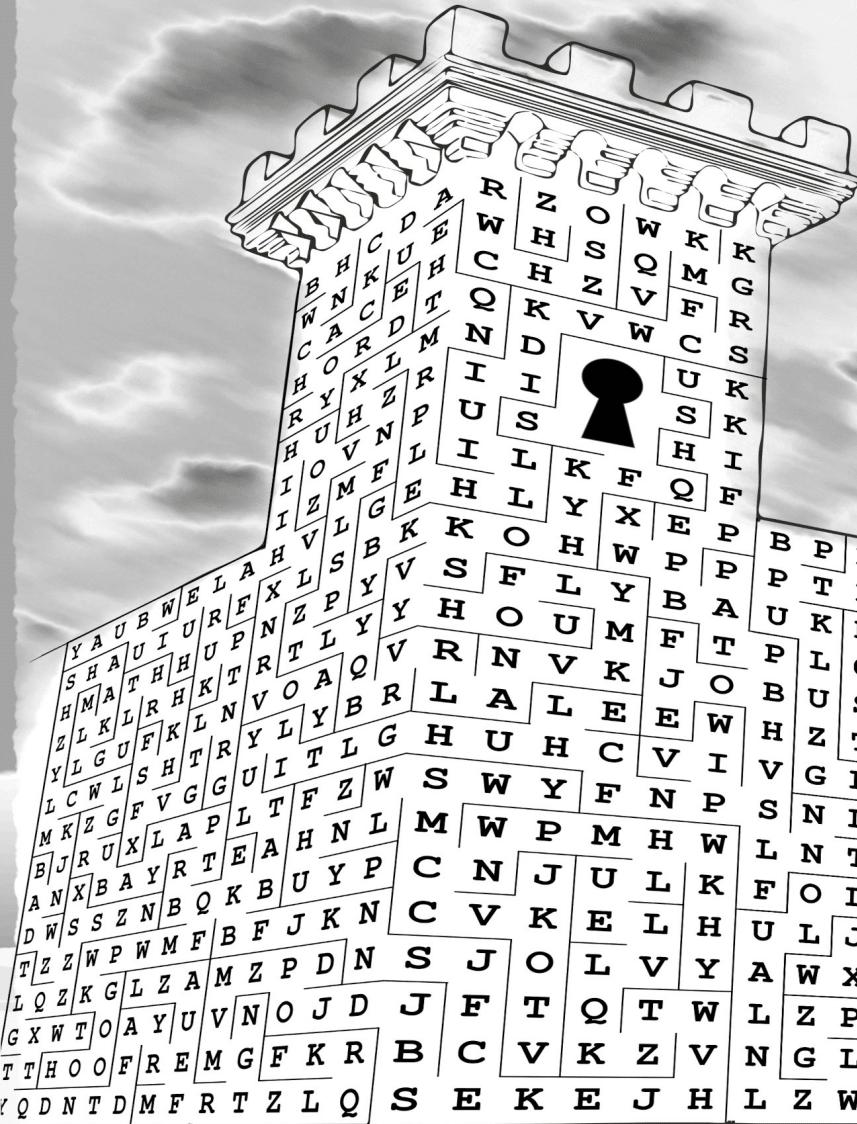


Szimmetrikus kulcsú rendszerek

Az AES (Advanced Encryption Standard)



AES (Advanced Encryption Standard)

- 1997-ben az amerikai szabványügyi hivatal (NIST) pályázatot írt ki a DES-t felváltó, következő titkosítási szabványra.
- A beérkezett 15 pályázatból 5 jutott a második fordulóba (MARS, RC6, **Rijndael**, Serpent és Twofish).
- 2000 októberében bejelentették, hogy az AES szabvány (Advanced Encryption Standard) a 128, a 192 és a 256 bites kulcsú Rijndael algoritmus lesz.
- Az algoritmus szerzői Joan Daemen és Vincent Rijmen belga kriptográfusok.

AES (Advanced Encryption Standard)

Kulcs: egy 128, 192 vagy 256 bites bináris adattömb

Bináris tömb (bit-index)	0	1	2	3	4	4	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	...
Bájt-index	0								1							2								...	
Bájton belüli bit-index	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	...

Bájtok és bitek indexei az AES leírásában

Az AES változatai

- Jelölések:
 - Nk : a kulcs szavakban mért hossza (word length), ahol **1 szó = 4 bájt = 32 bit**
 - Nb : a nyílt szövegblokk szavakban mért hossza
 - Nr : iterációk száma
- Megjegyzés: változattól függetlenül az AES minden **128 bites (4 szó hosszúságú)** tömböt kódol

Az AES változatai

Kulcshossz (Nk szó)	Bemenet hossza (Nb szó)	Menetek száma (Nr)
AES-128	4	4
AES-192	6	4
AES-256	8	4

AES

Kódolás:

$$P = p_0 \parallel p_1 \parallel \cdots \parallel p_{15}$$

a nyílt szöveg (P) bájtjai

p_0	p_4	p_8	p_{12}
p_1	p_5	p_9	p_{13}
p_2	p_6	p_{10}	p_{14}
p_3	p_7	p_{11}	p_{15}

állapottábla (S)

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

a kódolt szöveg (C) bájtjai

c_0	c_4	c_8	c_{12}
c_1	c_5	c_9	c_{13}
c_2	c_6	c_{10}	c_{14}
c_3	c_7	c_{11}	c_{15}

$$C = c_0 \parallel c_1 \parallel \cdots \parallel c_{15}$$

Algorithm AES (E_k):

Input: P, Nk, k

Output: C

$S := P$

$K := KeyExpansion(Nk, k)$

$S := AddRoundKey(S, K[0 \dots Nb - 1])$

for $n := 1, \dots, Nr - 1$ **do**

$S := SubBytes(S)$

$S := ShiftRows(S)$

$S := MixColumns(S)$

$S := AddRoundKey(S, K[n \cdot Nb \dots (n + 1) \cdot Nb - 1])$

end for

$S := SubBytes(S)$

$S := ShiftRows(S)$

$S := AddRoundKey(S, K[Nr \cdot Nb \dots (Nr + 1) \cdot Nb - 1])$

$C := S$

return C

end Algorithm

AES – KeyExpansion(...)

- $SubWord([a_0, a_1, a_2, a_3]) = [SubBytes(a_0), SubBytes(a_1), SubBytes(a_2), SubBytes(a_3)]$
- $RotWord([a_0, a_1, a_2, a_3]) = [a_1, a_2, a_3, a_0]$
- $Rcon(i) = [r_i, 0, 0, 0], \quad r_i = X^{i-1} \in \mathbb{F}_{2^8} \cong \mathbb{Z}_2[X]/(f)$

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
r_i	01	02	04	08	10	20	40	80	1B	36	6C	D8	AB	4D	9A

Algorithm *KeyExpansion()*:

Input: Nk, k

Output: K

$i := 0$

while $i < Nk$ **do**

$K[i] := k_{4i} \parallel k_{4i+1} \parallel k_{4i+2} \parallel k_{4i+3}$

$i := i + 1$

end while

$i := Nk$

while $i < Nb(Nr + 1)$ **do**

$temp := K[i - 1]$

if $i \bmod Nk = 0$ **then**

$temp := SubWord(RotWord(temp)) \oplus Rcon[i/Nk]$

else

if $(Nk > 6)$ and $(i \bmod Nk = 4)$ **then**

$temp := SubWord(temp)$

end if

end if

$K[i] := K[i - Nk] \oplus temp$

$i := i + 1$

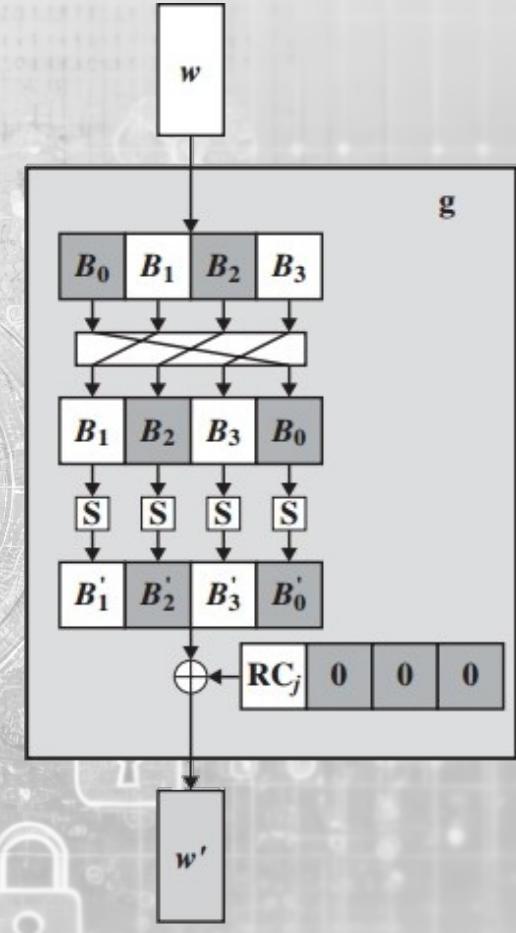
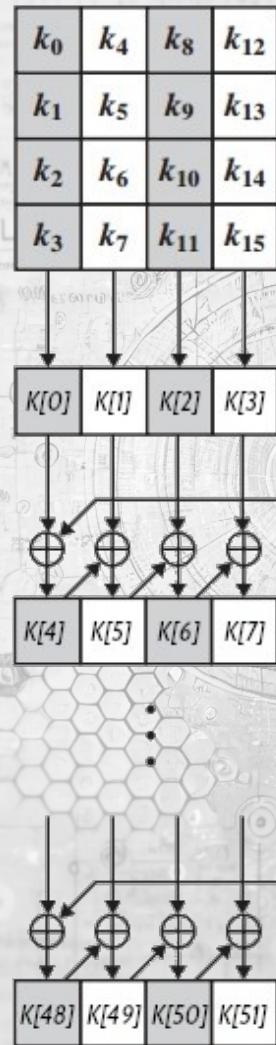
end while

return K

end Algorithm

AES – *KeyExpansion(...)*

- a k kulcsból legenerálja a K segédkulcs-tömböt tartalmazó tömböt
- a kimeneti K tömb mérete $Nb(Nr + 1)$ szó (pl. az AES-192 esetén $4(12+1) = 52$ szó)
- a kódoló eljárás során $Nr+1$ -szer adódik hozzá a segédkulcs-tömb az állapottáblához az *AddRoundKey(...)* segítségével



AES – *SubBytes(...)*

- egy nemlineáris függvény, amely az állapottábla bájtjait egyenként (egymástól függetlenül) helyettesíti új értékekkel
- két transzformáció (T_1 és T_2) összetevéséből áll
- $s'_{i,j} = T_2(T_1(s_{i,j}))$

AES – *SubBytes(...)*

- legyen $f = X^8 + X^4 + X^3 + X + 1$ irreducibilis főpolinom $\mathbb{Z}_2[X]$ -ben
- tekintsük az $s_{i,j} = (\beta_7, \beta_6, \beta_5, \beta_4, \beta_3, \beta_2, \beta_1, \beta_0)_2$ bájtot egy elemként a $\mathbb{F}_{2^8} \cong \mathbb{Z}_2[X]/(f)$ véges testben:

$$g = \beta_7 X^7 + \beta_6 X^6 + \beta_5 X^5 + \beta_4 X^4 + \beta_3 X^3 + \beta_2 X^2 + \beta_1 X + \beta_0 = \sum_{i=0}^7 \beta_i X^i$$

AES – *SubBytes(...)*

- legyen $g^{-1} = b_7X^7 + b_6X^6 + b_5X^5 + b_4X^4 + b_3X^3 + b_2X^2 + b_1X^1 + b_0X^0 \text{ mod } f$
- a T_1 transzformáció a következő:

$$T_1(s_{i,j}) = \begin{cases} (b_7b_6b_5b_4b_3b_2b_1b_0)_2 & \text{ha } s_{i,j} \neq (00000000)_2 \\ (00000000)_2 & \text{ha } s_{i,j} = (00000000)_2 \end{cases}$$

AES – *SubBytes(...)*

- a T_2 transzformáció a következő:

$$T_2((b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0)_2) = (b'_7 b'_6 b'_5 b'_4 b'_3 b'_2 b'_1 b'_0)_2, \text{ ahol}$$

$$b'_i = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i$$

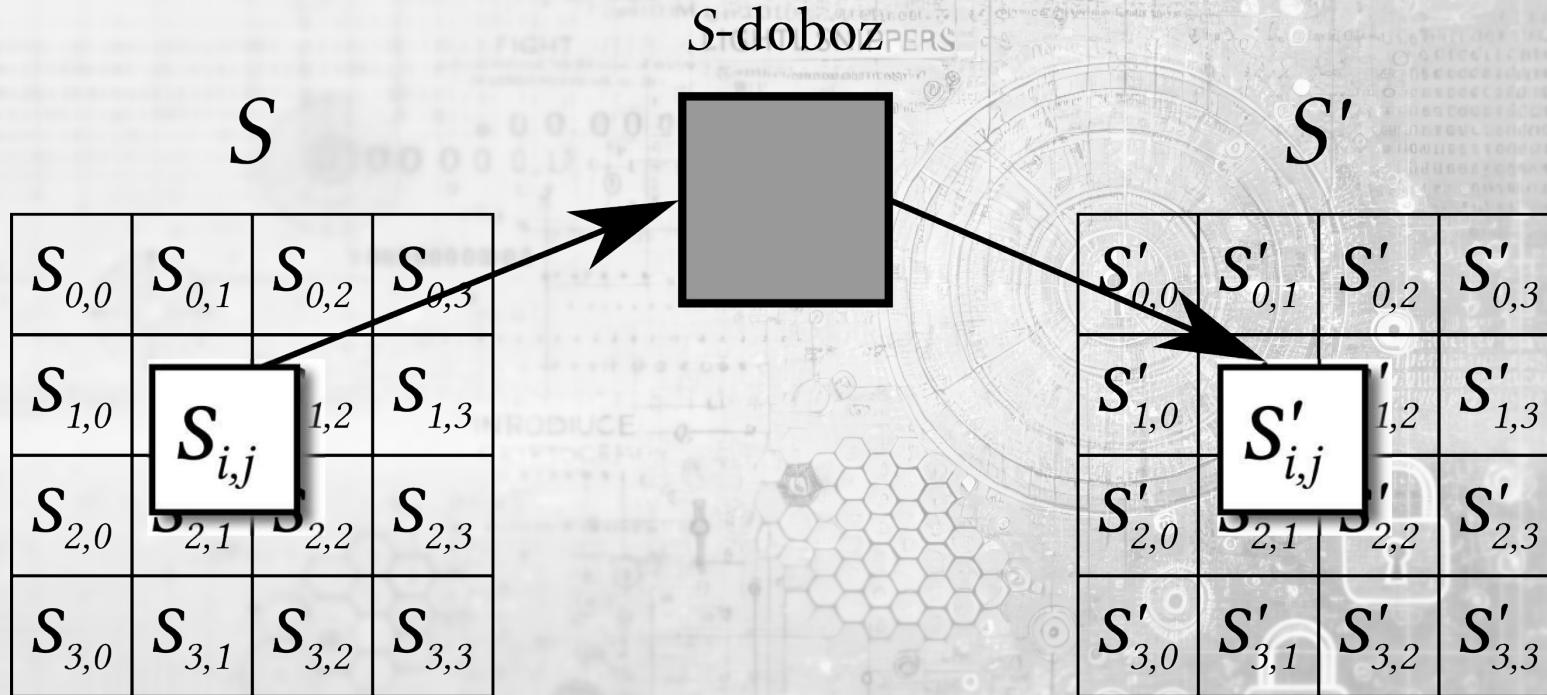
$$\text{bármely } 0 \leq i < 8\text{-ra és } c = (c_7 c_6 c_5 c_4 c_3 c_2 c_1 c_0)_2 = (01100011)_2$$

AES – *SubBytes(...)*

- a $T_2 \circ T_1$ affin transzformáció mátrixos alakban:

$$\begin{pmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

AES – *SubBytes(...)*



AES – *SubBytes(...)*, S-doboznal

Az S-doboz a *SubBytes(...)* függvény eredményét tárolja az összes lehetséges bemenetre (256 lehetőség).

Példa:

- $s_{1,1} = (5B)_{16}$ (sor: 5, oszlop: B)
- $s'_{1,1} = T_2(T_1(s_{1,1}))$
 $= (39)_{16} = (00111001)_2$
 $= 57$

	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

A *SubBytes()* eljárás S-doboza

AES – *ShiftRows(...)*

ShiftRows()

S

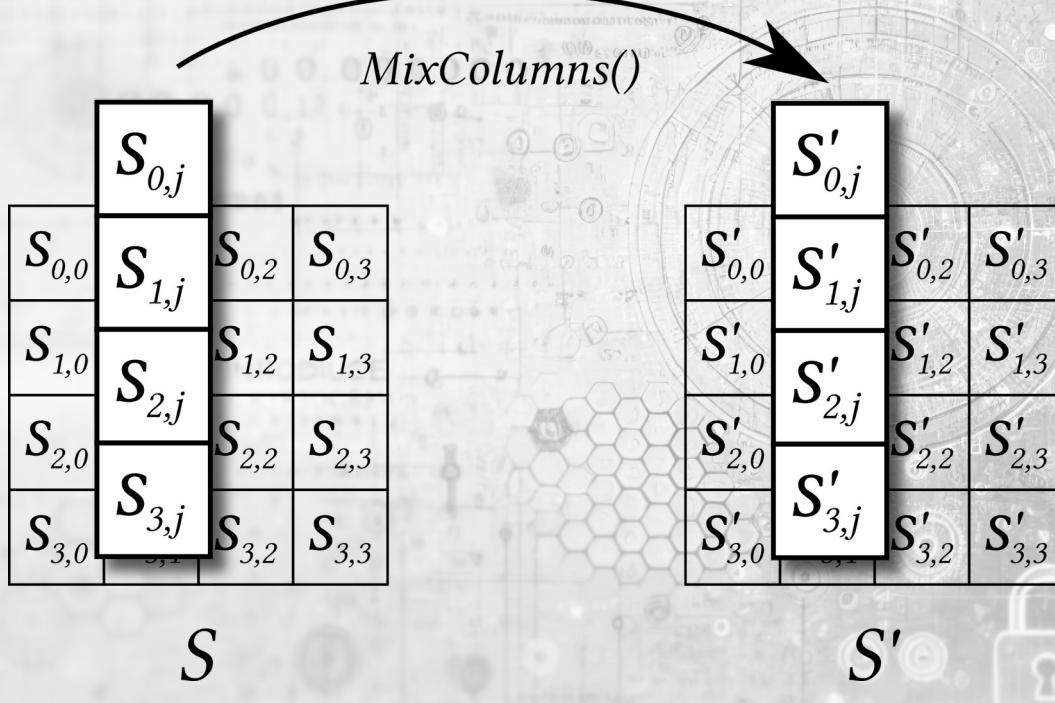
$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$



S'

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,1}$	$S_{1,2}$	$S_{1,3}$	$S_{1,0}$
$S_{2,2}$	$S_{2,3}$	$S_{2,0}$	$S_{2,1}$
$S_{3,3}$	$S_{3,0}$	$S_{3,1}$	$S_{3,2}$

AES – *MixColumns(...)*



AES – *MixColumns(...)*

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

$$\begin{aligned} s'_{0,j} &= ((02)_{16} \bullet s_{0,j}) \oplus ((03)_{16} \bullet s_{1,j}) \oplus s_{2,j} \oplus s_{3,j} \\ s'_{1,j} &= s_{0,j} \oplus ((02)_{16} \bullet s_{1,j}) \oplus ((03)_{16} \bullet s_{2,j}) \oplus s_{3,j} \\ s'_{2,j} &= s_{0,j} \oplus s_{1,j} \oplus ((02)_{16} \bullet s_{2,j}) \oplus ((03)_{16} \bullet s_{3,j}) \\ s'_{3,j} &= ((03)_{16} \bullet s_{0,j}) \oplus s_{1,j} \oplus s_{2,j} \oplus ((02)_{16} \bullet s_{3,j}) \end{aligned}$$

- A „ \bullet ” művelet az $\mathbb{F}_{2^8} \cong \mathbb{Z}_2[X]/(f)$ testbeli szorzás (vagyis polinomok szorzása modulo $f = X^8 + X^4 + X^3 + X + 1$).
- A „ \oplus ” az XOR műveletet jelöli.

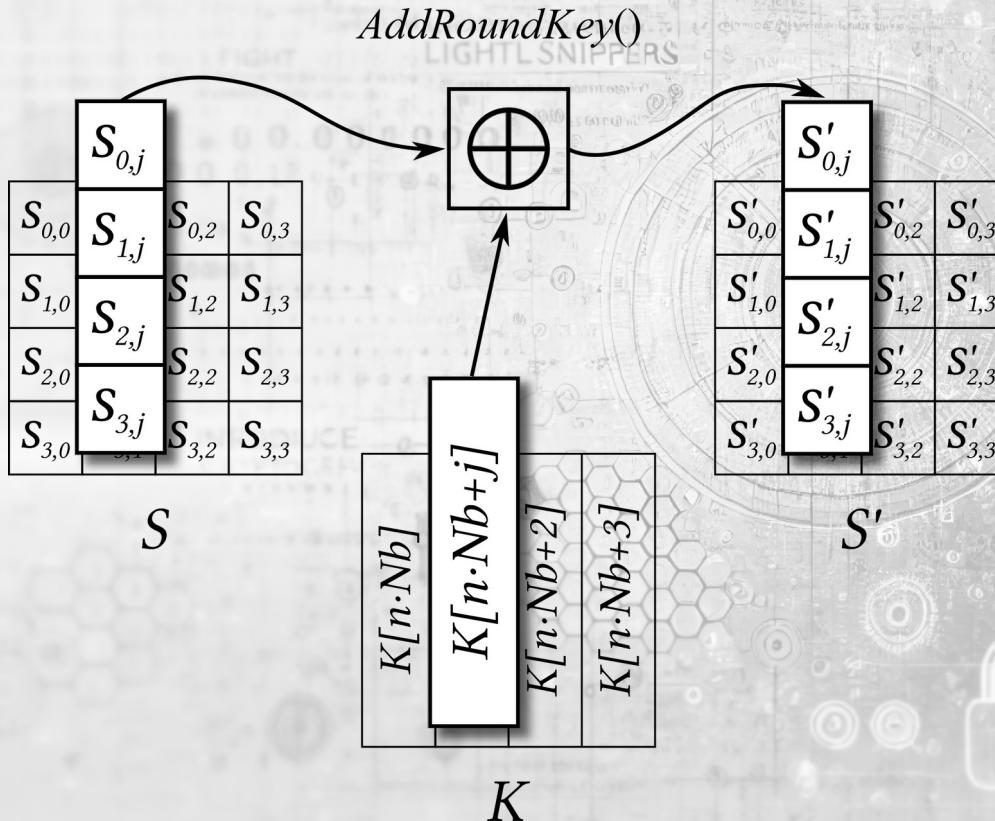
AES – *AddRoundKey(...)*

- minden egyes menet alkalmával hozzáadja (XOR műveettel) a menetnek megfelelő segédkulcsot az állapottábla bájtjaihoz:

$$\begin{pmatrix} s'_{0,j} \\ s'_{1,j} \\ s'_{2,j} \\ s'_{3,j} \end{pmatrix} = \begin{pmatrix} s_{0,j} \\ s_{1,j} \\ s_{2,j} \\ s_{3,j} \end{pmatrix} \oplus \begin{pmatrix} K'[j]_0 \\ K'[j]_1 \\ K'[j]_2 \\ K'[j]_3 \end{pmatrix} = \begin{pmatrix} s_{0,j} \\ s_{1,j} \\ s_{2,j} \\ s_{3,j} \end{pmatrix} \oplus \begin{pmatrix} K[n \cdot Nb + j]_0 \\ K[n \cdot Nb + j]_1 \\ K[n \cdot Nb + j]_2 \\ K[n \cdot Nb + j]_3 \end{pmatrix} \quad 0 \leq j < Nb,$$

ahol $K'[j]_i$ a K' tömb j indexű szavának i indexű bájtját jelöli.

AES - *AddRoundKey(...)*



AES

Dekódolás:

$C = c_0 \parallel c_1 \parallel \dots \parallel c_{15}$
a kódolt szöveg (C) bájtjai

c_0	c_4	c_8	c_{12}
c_1	c_5	c_9	c_{13}
c_2	c_6	c_{10}	c_{14}
c_3	c_7	c_{11}	c_{15}

állapottábla (S)

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

a nyílt szöveg (P) bájtjai

p_0	p_4	p_8	p_{12}
p_1	p_5	p_9	p_{13}
p_2	p_6	p_{10}	p_{14}
p_3	p_7	p_{11}	p_{15}

$P = p_0 \parallel p_1 \parallel \dots \parallel p_{15}$

Algorithm AES (D_k):

Input: C, Nk, k

Output: P

$S := C$

$K := KeyExpansion(Nk, k)$

$S := AddRoundKey(S, K[Nr \cdot Nb \dots (Nr + 1) \cdot Nb - 1])$

for $n := Nr - 1, \dots, 1$ **do**

$S := InvShiftRows(S)$

$S := InvSubBytes(S)$

$S := AddRoundKey(S, K[n \cdot Nb \dots (n + 1) \cdot Nb - 1])$

$S := InvMixColumns(S)$

end for

$S := InvShiftRows(S)$

$S := InvSubBytes(S)$

$S := AddRoundKey(S, K[0 \dots Nb - 1])$

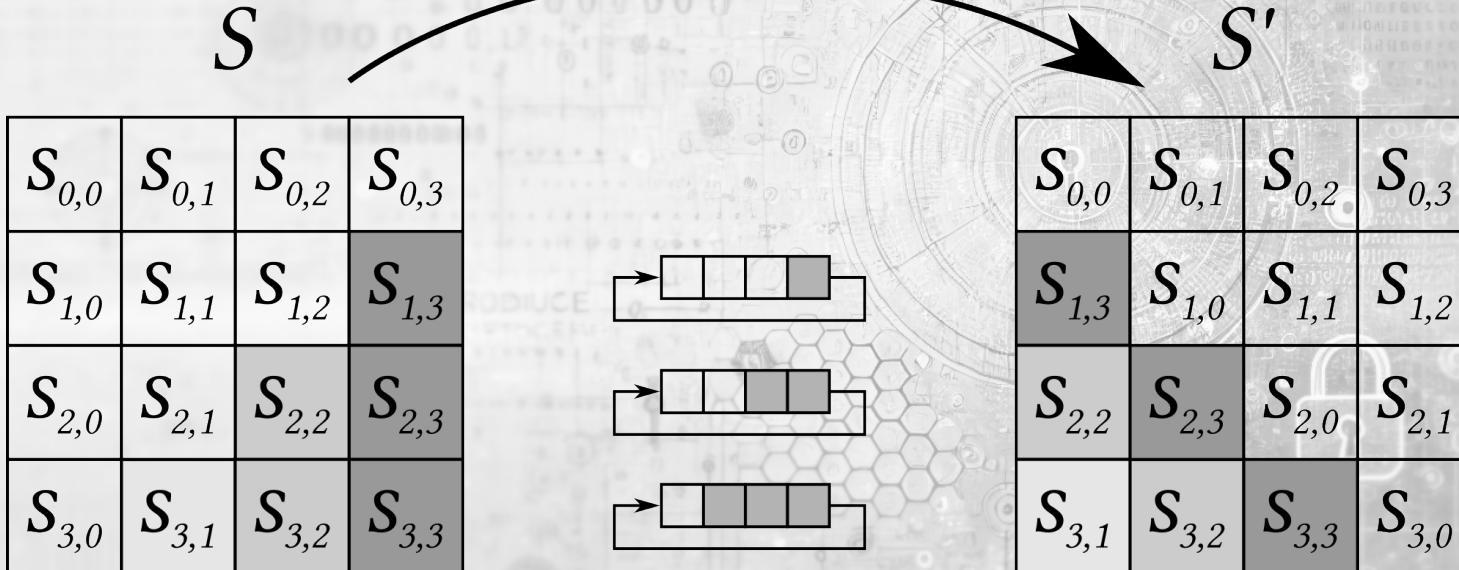
$P := S$

return P

end Algorithm

AES – *InvShiftRows(...)*

InvShiftRows()



AES – *InvSubBytes(...)*, S-doboznal

Az S-doboz a *SubBytes(...)* függvény inverzánek eredményét tárolja az összes lehetséges bemenetre (256 lehetőség).

Példa:

- $s'_{1,1} = (39)_{16}$ (sor: 3, oszlop: 9)
- $s_{1,1} = T_1^{-1}(T_2^{-1}(s'_{1,1}))$
 $= (5B)_{16}$

	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

Az *InvSubBytes()* eljárás S-doboza

AES – *InvMixColumns(...)*

$$\begin{pmatrix} s'_{0,j} \\ s'_{1,j} \\ s'_{2,j} \\ s'_{3,j} \end{pmatrix} = \begin{pmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0d & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{pmatrix} \begin{pmatrix} s_{0,j} \\ s_{1,j} \\ s_{2,j} \\ s_{3,j} \end{pmatrix},$$

$$s'_{0,j} = ((0E)_{16} \bullet s_{0,j}) \oplus ((0B)_{16} \bullet s_{1,j}) \oplus ((0D)_{16} \bullet s_{2,j}) \oplus ((09)_{16} \bullet s_{3,j})$$

$$s'_{1,j} = ((09)_{16} \bullet s_{0,j}) \oplus ((0E)_{16} \bullet s_{1,j}) \oplus ((0B)_{16} \bullet s_{2,j}) \oplus ((0D)_{16} \bullet s_{3,j})$$

$$s'_{2,j} = ((0D)_{16} \bullet s_{0,j}) \oplus ((09)_{16} \bullet s_{1,j}) \oplus ((0E)_{16} \bullet s_{2,j}) \oplus ((0B)_{16} \bullet s_{3,j})$$

$$s'_{3,j} = ((0B)_{16} \bullet s_{0,j}) \oplus ((0D)_{16} \bullet s_{1,j}) \oplus ((09)_{16} \bullet s_{2,j}) \oplus ((0E)_{16} \bullet s_{3,j})$$