METODE CRIPTOGRAFICE DE PROTECȚIE A INFORMAȚIEI

Tema: Cifruri bloc moderne

2.3.5.2.2 Specificarea algoritmului AES

În continuare, sunt expuse detaliile de construcție și de implementare ale cifrului bloc cu cheie simetrică AES. În mare parte, acesta constă din algoritmul de criptare, algoritmul de decriptare și procedura de expandare a cheii Rijndael.

2.3.5.2.2.1 Tabloul de stare, cheia secretă și numărul de runde

2.3.5.2.2.1.1 Tabloul de stare

În standardul FIPS 197 [14], operațiile din cadrul algoritmului AES sunt definite sub formă de operații la nivel de matrice, unde atât cheia, cât și blocul de date, sunt scrise în două tablouri bidimensionale. La începutul rulării cifrului, blocul de text (de 16 octeți în varianta standardizată) este copiat într-un tablou bidimensional numit „tablou de stare", primii 4 octeți în prima coloană, apoi următorii 4 octeți – în a doua coloană, și tot așa până la completarea tabloului (a se vedea tabloul „*input bytes*" în *Figura*

2.3.18). Astfel, tabloul de stare constă din 4 linii și *Nb* coloane de octeți, unde *Nb* este lungimea blocului divizată la 32 (numărul de cuvinte pe 32 biți ale blocului de text), iar în varianta standardizată *Nb* este egal cu 4. Pe parcursul rundelor algoritmul modifică tabloul de stare și îl furnizează la ieșire (tabloul „output bytes" în *Figura 2.3.18*).

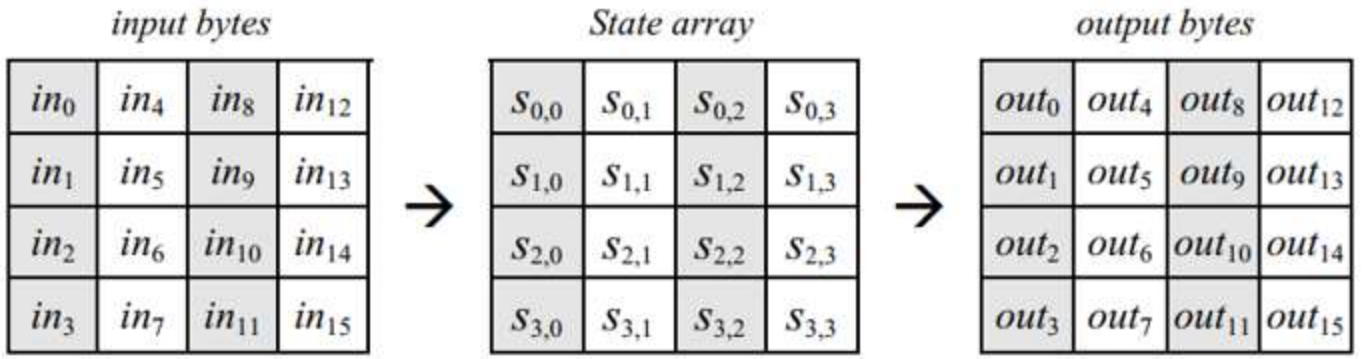


Figura 2.3.18. Prelucrarea tabloului de stare

SISTEMUL DE CRIPTARE AES

În tabloul de stare s, fiecare octet individual are doi indici: indicele de linie $r \in [0,3]$ și indicele de coloană $c \in [0, Nb-1]$. Astfel, fiecare octet al tabloului de stare este referit sub forma $s_{r,c}$. La inițializarea procedurii de criptare sau de decriptare, tabloul inițial in, format din octeții $in_0, in_1, \ldots, in_{15}$, este copiat în tabloul de stare, folosind relația:

$$s_{r,c} := in_{r+4c},\ r = \overline{0,3},\ c = \overline{0, Nb-1}.$$

După efectuarea pașilor algoritmului, rezultatul final din tabloul de stare este copiat în tabloul out, format din octeții $out_0, out_1, \ldots, out_{15}$, în baza relației:

$$out_{r+4c} := s_{r,c},\ r = \overline{0,3},\ c = \overline{0, Nb-1}.$$

SISTEMUL DE CRIPTARE AES

Cei 4 octeți din fiecare coloană a tabloului de stare formează, de fapt, cuvinte pe 32 biți, unde indicele de linie r indică locația celor 4 octeți ce formează cuvântul. Astfel, tabloul de stare poate fi interpretat ca un tablou unidimensional de cuvinte pe 32 biți (coloanele) $w_0 \ldots w_3$, în care indicele coloanei c stabilește indicele elementelor. De exemplu, tabloul de stare ilustrat în *Figura 2.3.18* poate fi considerat ca un tablou din 4 cuvinte $w_0 w_1 w_2 w_3$, unde

$$w_0 = s_{0,0} s_{1,0} s_{2,0} s_{3,0},\ w_1 = s_{0,1} s_{1,1} s_{2,1} s_{3,1},\ w_2 = s_{0,2} s_{1,2} s_{2,2} s_{3,2},\ w_3 = s_{0,3} s_{1,3} s_{2,3} s_{3,3}.$$

2.3.5.2.2.1.2 Lungimea cheii secrete și numărul de runde

În algoritmul AES lungimea blocului de text clar și a blocului de text criptat este de 128 biți, iar numărul de coloane în tabloul de stare este $Nb = 4$. Lungimea cheii secrete K este de 128, 192 sau 256 biți.

Tradițional pentru cifrurile bloc, cheia secretă K este utilizată pentru generarea cheilor de rundă în cadrul procedurii de expandare a cheii. Prin analogie cu tabloul de stare, cheia secretă se reprezintă ca un tablou bidimensional cu 4 linii și Nk coloane (numărul de cuvinte pe 32 biți ale cheii) cu elementele octeți. Pentru standardul AES se consideră Nk egal cu 4, 6 sau 8 în dependență de lungimea cheii.

SISTEMUL DE CRIPTARE AES

Numărul de runde efectuate în cadrul algoritmului AES depinde de lungimea cheii. Astfel, numărul de runde *Nr* este 10, 12 sau 14, după cum este ilustrat în *Tabelul 2.3.13*.

	Lungimea cheii *Nk* cuvinte pe 4 octeți	Lungimea blocului *Nb* cuvinte pe 4 octeți	Numărul de runde *Nr* runde
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14
Rijndael	4	6	12
Rijndael	6	6	12
Rijndael	8	6	14
Rijndael	4	8	14
Rijndael	6	8	14
Rijndael	8	8	14

Tabelul 2.3.13. Parametrii ce definesc algoritmii Rijndael

SISTEMUL DE CRIPTARE AES

Spre deosebire de algoritmul DES, AES este construit pe o rețea de tip substituție-permutare. Rețeaua Feistel pe care este bazat DES criptează doar o jumătate din biții blocului (32 biți) la o rundă, pe când AES criptează toți cei 128 biți la o rundă. Acest fapt contribuie la un număr ceva mai redus de runde în AES.

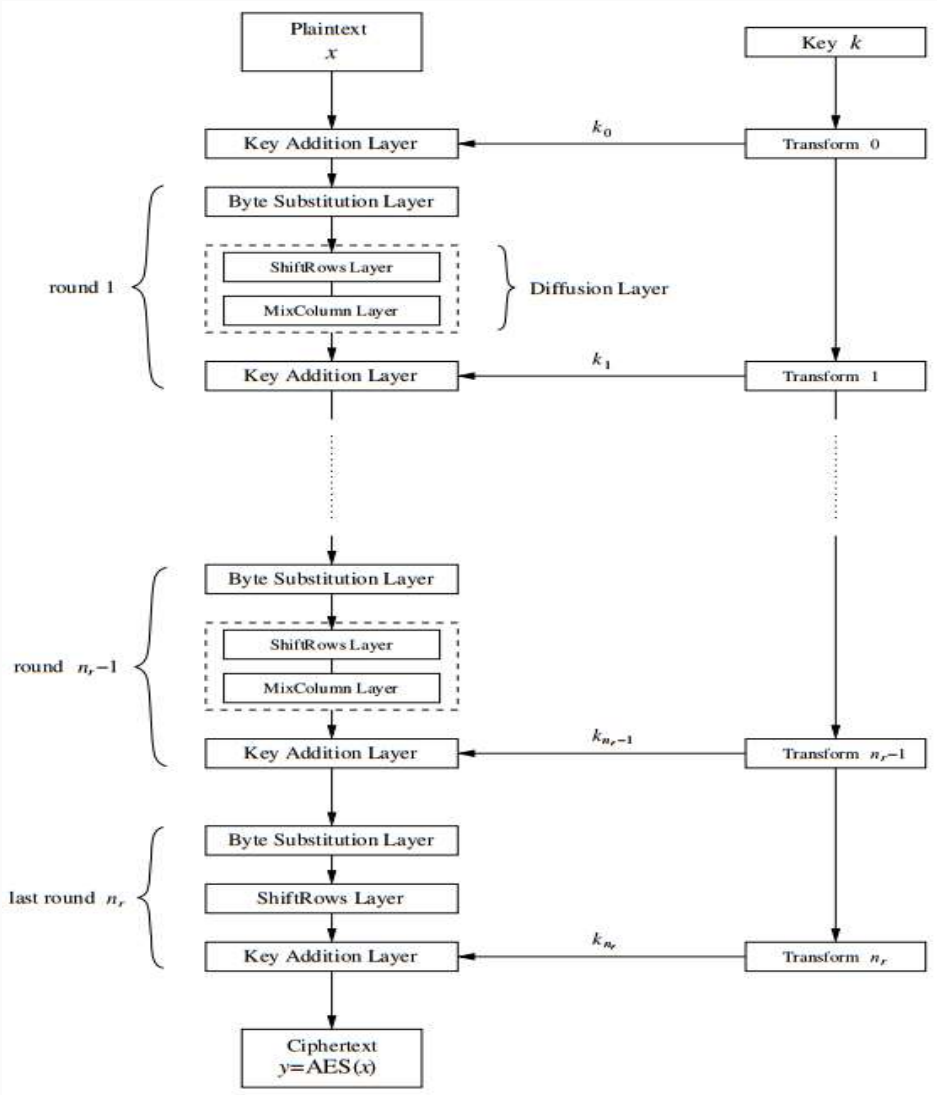
Două runde ale algoritmului AES asigură o difuzie „completă", în sensul că fiecare bit al tabloului de stare depinde de toți biții tabloului de stare de la 2 runde anterioare. La fel, o modificare a unui bit al tabloului de stare afectează aproximativ jumătate din biții tabloului de stare pentru următoarele 2 runde. Gradul mare al difuziei rundei AES este datorat structurii uniforme a acesteia.

SISTEMUL DE CRIPTARE AES

2.3.5.2.2.2 Algoritmul de criptare AES

La inițializarea procedurii de criptare, datele de intrare sunt scrise în tabloul de stare. După o mixare inițială cu cheia de rundă, sunt criptate datele din tabloul de stare în cadrul a 9, 11 sau 13 runde (în dependență de lungimea cheii), după care urmează runda finală, realizată separat, deoarece aceasta diferă de primele $Nr-1$ runde prin aceea că se omite una din cele 4 transformări utilizate în cadrul funcției de rundă. Tabloul de stare final (bidimensional) este transformat la ieșire într-un tablou unidimensional.

SISTEMUL DE CRIPTARE AES



SISTEMUL DE CRIPTARE AES

Algoritmul de criptare AES

Date de intrare:

Nb - numărul de cuvinte pe 4 octeți ale blocului de text clar

Nr - numărul de runde ale cifrului

in - tablou ale cărui elemente sunt octeți, numărul de elemente fiind $4Nb$

w - tablou cu elementele cuvinte pe 4 octeți, numărul de elemente fiind $Nb(Nr+1)$

Date de ieșire:

out - tablou ale cărui elemente sunt octeți, numărul de elemente fiind $4Nb$

SISTEMUL DE CRIPTARE AES

state - tabloul de stare bidimensional de dimensiune $4 \times Nb$, elementele fiind octeți

$state := in$

$AddRoundKey(state, w_{0,Nb-1})$

Pentru $round = \overline{1, Nr-1}$ execută

{

$SubBytes(state)$

$ShiftRows(state)$

$MixColumns(state)$

$AddRoundKey(state, w_{round*Nb,(round+1)*Nb-1})$

}

$SubBytes(state)$

$ShiftRows(state)$

$AddRoundKey(state, w_{Nr*Nb,(Nr+1)*Nb-1})$

$out := state$

2.3.5.2.2.2.1 Funcția de rundă

Cifrul AES folosește o funcție de rundă formată din patru transformări ce acționează la nivel de octeți:

1) Transformarea de substituție a octeților în baza unui tabel de substituție (S-boxă) - *SubBytes*;
2) Transformarea de rotație ciclică la stânga a octeților din liniile tabloului de stare - *ShiftRows*;
3) Transformarea de combinare a octeților din fiecare coloană a tabloului de stare - *MixColumns*;
4) Transformarea de mixare a tabloului de stare cu cheia de rundă - *AddRoundKey*.

Aceste transformări criptografice sunt descrise în continuare.

2.3.5.2.2.2.1.1 Transformarea SubBytes

SubBytes reprezintă o transformare neliniară de substituție a octeților (un cifru cu substituție), care prelucrează independent fiecare octet al tabloului de stare, generând confuzie în datele criptate. Substituția de octeți este realizată, folosind un tabel de substituție, numit S-boxă Rijndael (a se vedea *Tabelul 2.3.14*). S-boxa Rijndael reprezintă o transformare inversabilă și este construită prin compoziția a două transformări aplicate fiecărui octet din cei 256 ai lui $GF\left(2^8\right)$, în modul următor:

1. Pentru fiecare element din $GF\left(2^8\right)$ se calculează inversul multiplicativ în $GF\left(2^8\right)$. Elementul $\{00\}_{16}$ (care nu admite un invers multiplicativ în $GF\left(2^8\right)$) este aplicat în $\{00\}_{16}$ (a se vedea Tabelul 2.3.12).

2. La fiecare invers multiplicativ obținut (un octet) este aplicată o transformare afină (se înmulțește o matrice la un vector, după care, la rezultat se adună un vector) peste $GF(2)$:

$$b_i' := b_i \oplus b_{\text{mod}(i+4,8)} \oplus b_{\text{mod}(i+5,8)} \oplus b_{\text{mod}(i+6,8)} \oplus b_{\text{mod}(i+7,8)} \oplus c_i,\ i = \overline{0,7},$$

unde b_i este bitul de pe locul i (indexarea se face de la dreapta la stânga, începând cu indicele 0) din cadrul octetului examinat, iar c_i este, corespunzător, bitul de pe locul i al octetului $c = \{63\}_{16}$, care în binar se reprezintă astfel $\{01100011\}_2$. Reprezentarea matricială a transformării afine menționate este următoarea:

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

S-boxa Rijndael nu conține puncte fixe și nici puncte anti-fixe, adică avem $S\left(s_{i,j}\right) \neq s_{i,j}$ și $S\left(s_{i,j}\right) \oplus s_{i,j} \neq \{ff\}_{16}$.

Figura 2.3.20 ilustrează modul în care acţionează transformarea *SubBytes* asupra elementelor tabloului de stare.

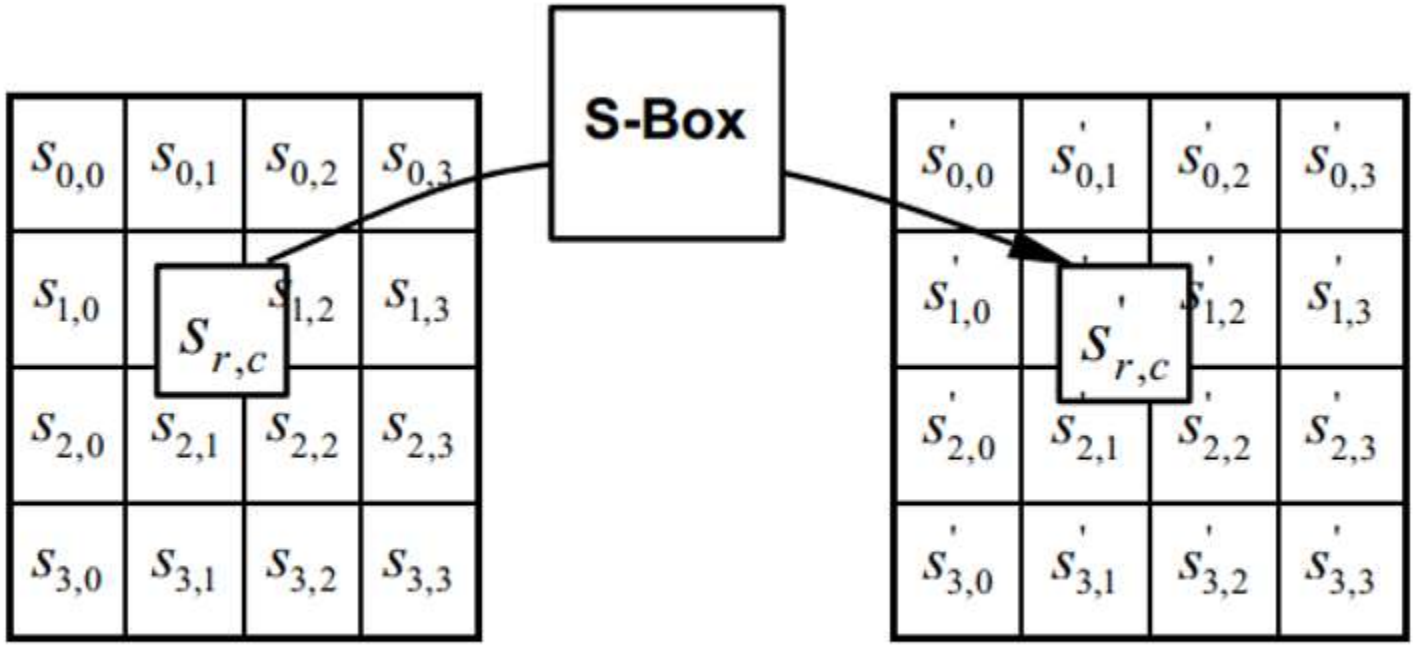


Figura 2.3.20. SubBytes aplică transformarea de substituție la fiecare octet al tabloului de stare

SISTEMUL DE CRIPTARE AES

În *Tabelul 2.3.14* este prezentată S-boxa Rijndael (elementele sunt în reprezentare hexazecimală).

		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

Tabelul 2.3.14. S-boxa: octetul xy este substituit prin octetul de la intersecția liniei x și a coloanei y

De exemplu, dacă $s_{1,1} = \{53\}_{16}$, atunci acesta este substituit prin octetul de la intersecția liniei cu indicele '5' și a coloanei cu indicele '3' din *Tabelul 2.3.14*. În rezultat, se obține $s'_{1,1} = \{ed\}_{16}$.

2.3.5.2.2.2.1.2 Transformarea ShiftRows

Transpoziția *ShiftRows* realizează rotația ciclică la stânga cu un anumit număr de poziții a octeților din liniile tabloului de stare. Octeții din linia întâi, de indice $r = 0$, nu sunt antrenați în rotația ciclică la stânga; octeții din linia a doua – sunt rotiți ciclic la stânga cu o poziție; octeții din linia a treia - cu două poziții, iar octeții din linia a patra – cu trei poziții. Astfel, transformarea *ShiftRows* acționează în modul următor:

$$s'_{r,c} := s_{r,\mathrm{mod}(c+shift(r,Nb),Nb)},\ r = \overline{1,3},\ c = \overline{0,Nb-1},$$

unde valoarea rotației ciclice la stânga $shift(r,Nb)$ este definită în dependență de numărul liniei r astfel (amintim că $Nb = 4$):

$$shift(1,4) = 1,\ shift(2,4) = 2,\ shift(3,4) = 3.$$

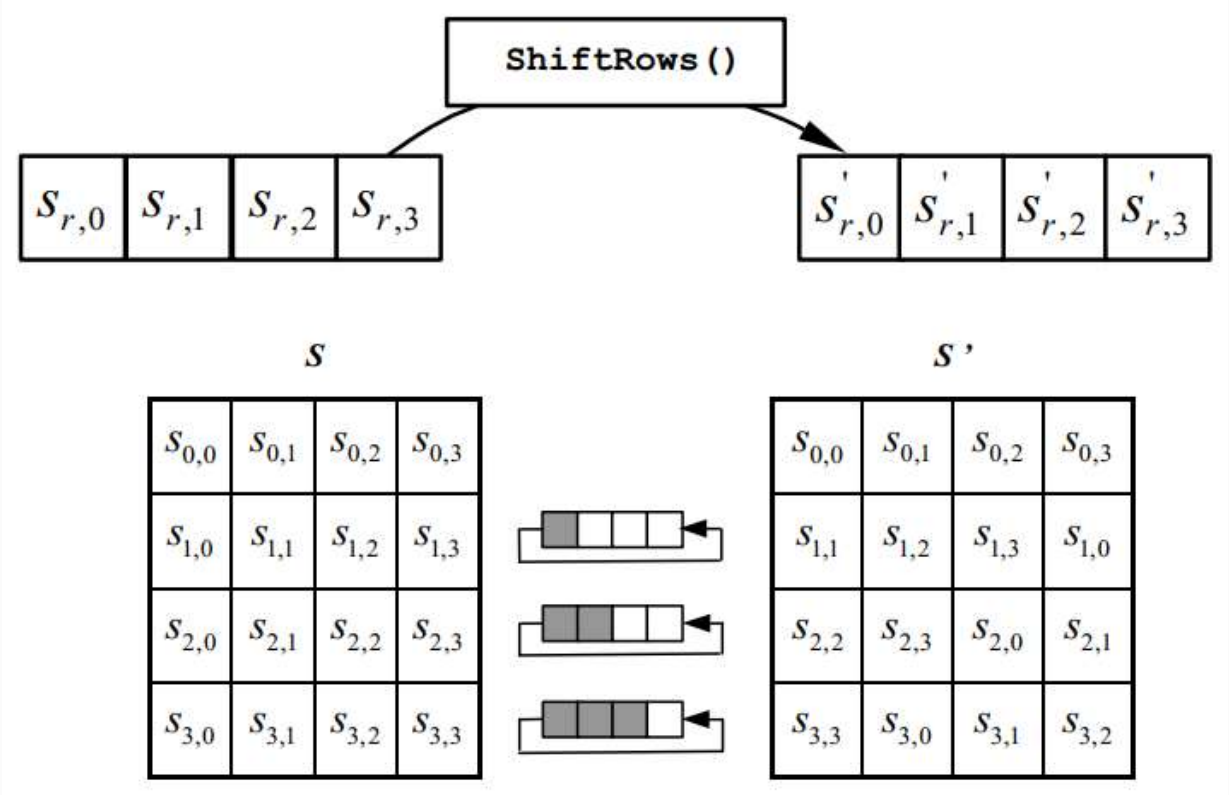


Figura 2.3.21. Ilustrarea transformării ShiftRows

Remarca 2.3.1. *În algoritmul Rijndael cu* $Nb = 6$ *avem* $shift(1,6) = 1$, $shift(2,6) = 2$, $shift(3,6) = 3$, *iar în versiunea cu* $Nb = 8$ *avem* $shift(1,8) = 1$, $shift(2,8) = 3$, $shift(3,8) = 4$.

În urma aplicării transformării *ShiftRows* fiecare coloană din tabloul de stare rezultat este formată din octeți ai fiecărei coloane a tabloului de stare de la runda precedentă. Acest aspect este important, deoarece tabloul de stare este scris inițial pe coloane, iar la pașii ulteriori operațiile se aplică, la fel, la nivel de coloane. Transformarea *ShiftRows* permite să se evită criptarea independentă a coloanelor, caz în care AES ar degenera în 4 cifruri bloc independente.

2.3.5.2.2.2.1.3 Transformarea MixColumns

Transformarea *MixColumns* acționează la nivelul coloanelor tabloului de stare, combinând cei 4 octeți ai coloanei în baza unei transformări liniare inversabile. Funcția *MixColumns* preia la intrare șirul de 4 octeți $\left(s_{0,c}, s_{1,c}, s_{2,c}, s_{3,c}\right)^T$ ai coloanei c și întoarce șirul de 4 octeți $\left(s'_{0,c}, s'_{1,c}, s'_{2,c}, s'_{3,c}\right)^T$ în coloana respectivă.

Fiecărei coloane i se asociază un polinom de gradul 3 cu coeficienții octeți din $GF\left(2^8\right)$, de forma menționată la secțiunea 2.3.5.2.1.4. Polinomul asociat coloanei este înmulțit modulo $\{01\}_{16}x^4+\{01\}_{16}$ în $GF\left(2^8\right)$ cu polinomul $a(x)$, definit astfel $a(x)=\{03\}_{16}x^3+\{01\}_{16}x^2+\{01\}_{16}x+\{02\}_{16}$. Polinomul $a(x)$ este coprim cu $\{01\}_{16}x^4+\{01\}_{16}$ și, prin urmare, este inversabil. Ținând cont de cele relatate la secțiunea 2.3.5.2.1.4, operația de înmulțire modulară în $GF\left(2^8\right)$, notată prin $s'(x)=a(x)\otimes s(x)$, se poate scrie sub următoarea formă matricială (a se vedea relația (2.3.2)):

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{pmatrix}, \quad c=\overline{0,Nb-1}.$$

Valorile $s'_{i,c}$ sunt elementele coloanei c rezultate după înmulțire, iar $s_{i,c}$ - elementele aceleași coloane, dar înainte de aplicarea transformării. Efectuând operația de înmulțire matricială a octeților (a se vedea relația (2.3.1)), se obțin patru octeți noi ai coloanei c:

$$s'_{0,c} = \left(\{02\}_{16} \cdot s_{0,c}\right) \oplus \left(\{03\}_{16} \cdot s_{1,c}\right) \oplus s_{2,c} \oplus s_{3,c},$$
$$s'_{1,c} = s_{0,c} \oplus \left(\{02\}_{16} \cdot s_{1,c}\right) \oplus \left(\{03\}_{16} \cdot s_{2,c}\right) \oplus s_{3,c},$$
$$s'_{2,c} = s_{0,c} \oplus s_{1,c} \oplus \left(\{02\}_{16} \cdot s_{2,c}\right) \oplus \left(\{03\}_{16} \cdot s_{3,c}\right),$$
$$s'_{3,c} = \left(\{03\}_{16} \cdot s_{0,c}\right) \oplus s_{1,c} \oplus s_{2,c} \oplus \left(\{02\}_{16} \cdot s_{3,c}\right),$$

unde prin $\oplus$ este notată operația XOR a octeților, iar prin „$\cdot$” – operația de înmulțire modulo polinomul ireductibil $x^8 + x^4 + x^3 + x + 1$ a octeților considerați ca și coeficienți ai unui polinom de grad mai mic sau egal cu 7.

Transformarea *MixColumns* are proprietatea că fiecare din cei patru octeți de la intrare afectează toți cei patru octeți de la ieșire. În combinație cu *ShiftRows*, această transformare asigură că după câteva runde, fiecare octet al tabloului de stare actual depinde de fiecare octet al tabloului de stare inițial (format din octeții textului clar). Transformările *MixColumns* și *ShiftRows* sunt principala sursă de difuzie în algoritmul AES.

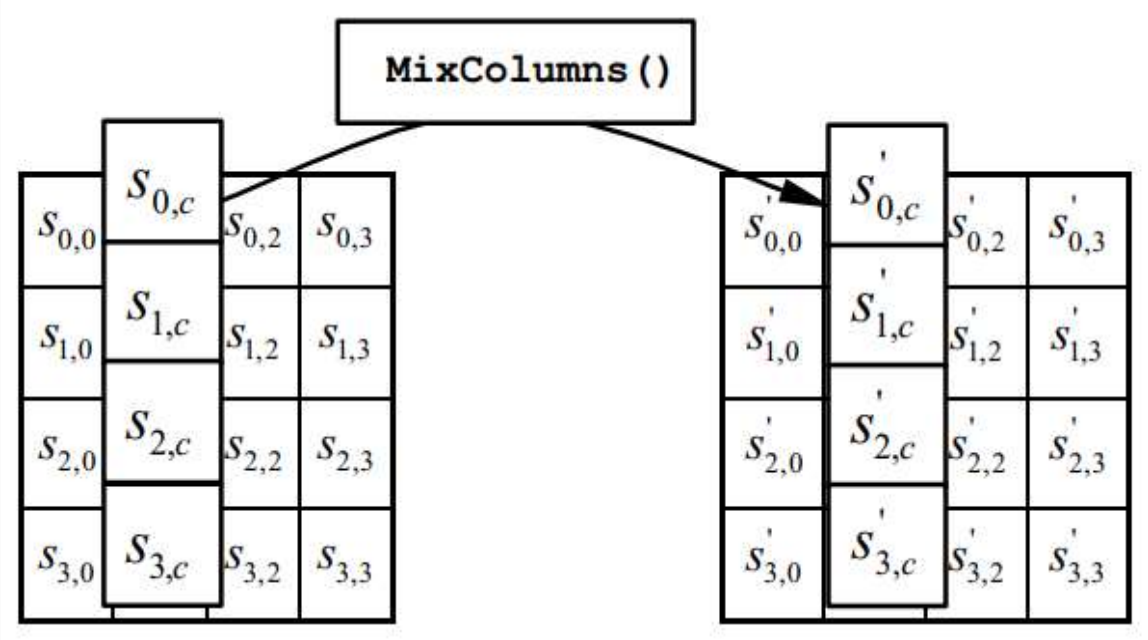


Figura 2.3.22. Ilustrarea transformării MixColumns

2.3.5.2.2.2.1.4 Transformarea AddRoundKey

Această transformare combină prin XOR subcheia de rundă cu tabloul de stare. Fiecare subcheie de rundă constă din Nb cuvinte pe patru octeți (este obținută prin aplicarea algoritmului de expandare a cheii ce va fi descris la secțiunea următoare), care se combină prin XOR cu tabloul de stare, în modul următor (se ia XOR-ul octeților corespunzători):

$$\left[s'_{0,c}, s'_{1,c}, s'_{2,c}, s'_{3,c}\right] = \left[s_{0,c}, s_{1,c}, s_{2,c}, s_{3,c}\right] \oplus \left[w_{round*Nb+c}\right], \; c = \overline{0, Nb-1},$$

unde $\left[w_i\right]$ este cuvântul i al cheii de rundă, iar $round$ este un întreg din intervalul $\left[0, Nr\right]$. În *Algoritmul de criptare AES* are loc combinarea tabloului de stare cu subcheia de rundă înainte de prima aplicare a funcției de rundă (pentru $round = 0$), iar apoi în cadrul a Nr runde ale algoritmului de criptare (pentru $round = \overline{1, Nr}$).

Transformarea *AddRoundKey* este ilustrată în *Figura 2.3.23*, în care avem $l = round * Nb$.

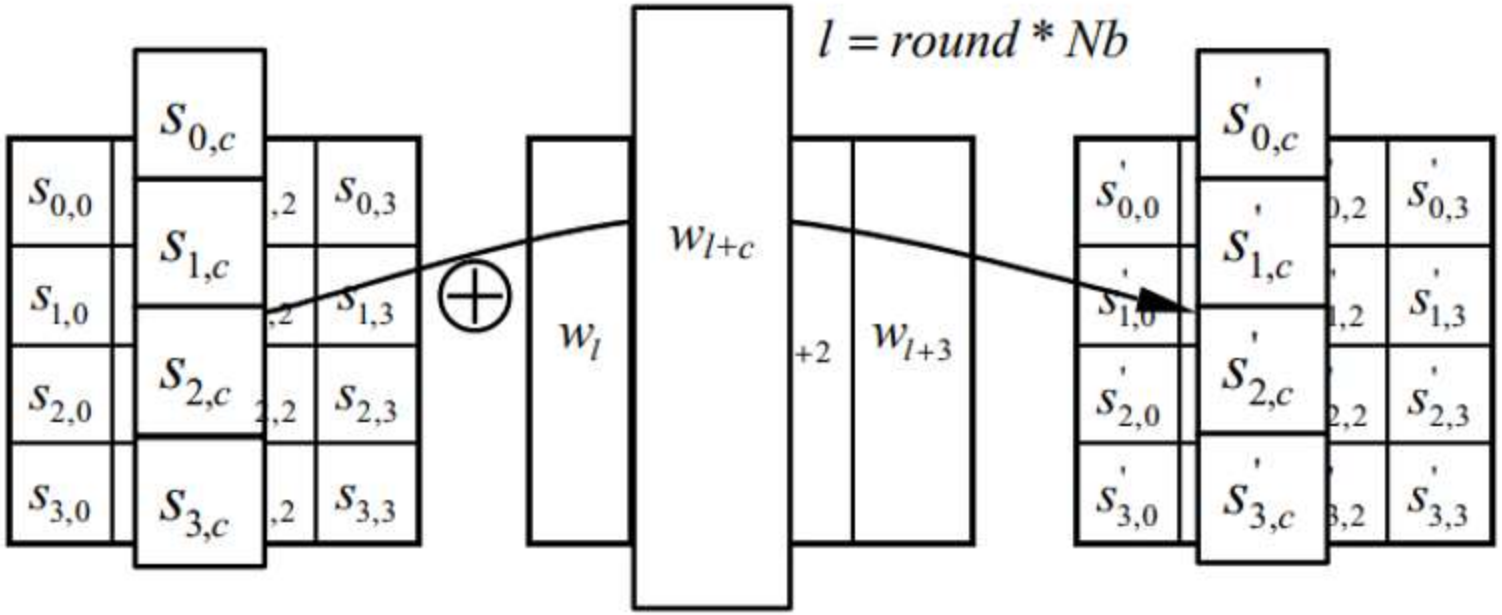


Figura 2.3.23. Ilustrarea transformării AddRoundKey

2.3.5.2.2.3 Algoritmul de expandare a cheii Rijndael

Procedura de expandare a cheii pornește de la cheia secretă K de lungime Nk cuvinte pe 4 octeți și generează un tablou de $Nb(Nr+1)$ cuvinte pe 4 octeți ce formează subcheile de rundă. Algoritmul de criptare AES necesită în faza inițială primele Nb cuvinte ale cheii expandate, după care, fiecare din cele Nr runde necesită încă câte Nb cuvinte succesive (următoarele chei de rundă). Algoritmul ce urmează generează mulțimea cheilor de rundă, stocate într-un tablou liniar de cuvinte pe 4 octeți, notat cu $[w_i]$, $i=\overline{0,Nb(Nr+1)-1}$. Primele Nk cuvinte ale tabloului w conțin cheia secretă.

SISTEMUL DE CRIPTARE AES

Algoritmul de expandare a cheii AES

Date de intrare:

Nb - numărul de cuvinte pe 32 de biți ale blocului de text

Nr - numărul de runde (10, 12 sau 14 în dependență de lungimea cheii)

Nk - numărul de cuvinte pe 32 biți din care este formată cheia secretă K (Nk este 4,6 sau 8)

key - tablou de octeți ce conține $4Nk$ elemente

w - tablou de cuvinte ce conține $Nb(Nr+1)$ elemente

Date de ieșire:

w - tablou de cuvinte ce conține $Nb(Nr+1)$ elemente

SISTEMUL DE CRIPTARE AES

$temp$ - cuvânt intermediar

Pentru $i := \overline{0, Nk-1}$ execută

{

$w_i := key_{4i} \,||\, key_{4i+1} \,||\, key_{4i+2} \,||\, key_{4i+3}$ (cuvânt pe 4 octeți)

}

Pentru $i := \overline{Nk, Nb(Nr+1)-1}$ execută

{

$temp := w_{i-1}$

Dacă $\mathrm{mod}(i, Nk) = 0$ atunci { $temp := SubWord(RotWord(temp)) \oplus Rcon_{i/Nk}$ }

Altfel dacă $(Nk > 6) \, and \, (\mathrm{mod}(i, Nk) = 4)$ { $temp := SubWord(temp)$ }

$w_i := w_{i-Nk} \oplus temp$

}

SISTEMUL DE CRIPTARE AES

În *„Algoritmul de expandare a cheii AES”* primele Nk cuvinte ale cheii expandate sunt cuvintele cheii secrete. Fiecare dintre cuvintele ce urmează w_i se obține recursiv printr-un XOR al cuvântului precedent w_{i-1} și al cuvântului obținut Nk poziții anterioare - w_{i-Nk}. Pentru cuvinte de pe poziții ce sunt un multiplu a lui Nk, mai întâi sunt aplicate două transformări la w_{i-1}, după care urmează o combinare printr-un XOR cu o constantă de rundă $Rcon_{i/Nk}$. Cele două transformări reprezintă permutarea ciclică la stânga a octeților cuvântului $RotWord(\ )$, urmată de aplicarea tabelului de substituție la fiecare din cei 4 octeți ai cuvântului $SubWord(\ )$.

De menționat că procedura de expandare a cheii pentru chei secrete pe 256 biți ($Nk=8$) este ceva diferită de cea pentru chei pe 128 sau 192 biți. Dacă $Nk=8$ și $i-4$ este un multiplu al lui Nk, atunci se aplică transformarea $SubWord(\ )$ la w_{i-1} înainte de XOR.

SISTEMUL DE CRIPTARE AES

Algoritmul de expandare a cheii folosește transformarea $SubWord(\ )$, care este o funcție ce preia la intrare un cuvânt pe 4 octeți și aplică S-boxa Rijndael (a se vedea *Tabelul 1.3.15*) la fiecare dintre cei 4 octeți ai cuvântului, generând un alt cuvânt la ieșire. Funcția $RotWord(\ )$ preia cuvântul $[a_0, a_1, a_2, a_3]$ la intrare și realizează o permutare ciclică la stânga a octeților acestuia, returnând cuvântul $[a_1, a_2, a_3, a_0]$. Tabloul constantelor de rundă $Rcon_i$ este independent de Nk și este definit astfel: $Rcon_i = [x^{i-1}, \{00\}_{16}, \{00\}_{16}, \{00\}_{16}]$ (i începe de la 1), unde x^{i-1} sunt puteri ale lui x (x este notație pentru $\{02\}_{16}$) în câmpul $GF(2^8)$.

În *Tabelul 1.3.16* sunt prezentate valorile în formă hexazecimală pentru $Rcon_i$.

01	00	00	00
02	00	00	00
04	00	00	00
08	00	00	00
10	00	00	00
20	00	00	00
40	00	00	00
80	00	00	00
1b	00	00	00
36	00	00	00

Tabelul 1.3.16. Valorile constantelor de rundă Rcon

SISTEMUL DE CRIPTARE AES

Funcția de rundă a algoritmului de criptare AES (Rijndael) nu este o rețea Feistel. Amintim că un avantaj al rețelei Feistel este că algoritmul de decriptare este aproape identic cu cel de criptare. Deoarece funcția de rundă a rețelei Feistel este o involuție, doar ordinea cheilor de rundă este inversată. Pentru AES aceasta nu este aplicabil. Decriptarea AES este realizată prin utilizarea inverselor transformărilor din componența algoritmului de criptare AES, considerate în ordine inversă.

Procedura de decriptare AES folosește transformări specifice precum $InvShiftRows(\ )$, $InvSubBytes(\ )$, $InvMixColumns(\ )$.

SISTEMUL DE CRIPTARE AES

Algoritmul de decriptare AES
Date de intrare: *in* - tablou de lungime $4Nb$ cu elementele octeți *w* - tablou de de lungime $Nb(Nr+1)$ cu elementele cuvinte (tabloul conține expandarea cheii) Date de ieșire: *out* - tablou de lungime $4Nb$ cu elementele octeți
state - tablou bidimensional de dimensiune $4 \times Nb$

Transformarea $AddRoundKey(\ )$ coincide cu inversa sa, deoarece aceasta implică doar operația XOR.

SISTEMUL DE CRIPTARE AES

$state := in$

$AddRoundKey\left(state, w_{Nr*Nb,(Nr+1)Nb-1}\right)$

Pentru $round = \overline{Nr-1:-1:1}$ execută

{

  $InvShiftRows\left(state\right)$

  $InvSubBytes\left(state\right)$

  $AddRoundKey\left(state, w_{round*Nb,(round+1)*Nb-1}\right)$

  $InvMixColumns\left(state\right)$

}

$InvShiftRows\left(state\right)$

$InvSubBytes\left(state\right)$

$AddRoundKey\left(state, w_{0,Nb-1}\right)$

$out := state$

1.3.6.2.2.4.1 Transformarea InvShiftRows()

$InvShiftRows(\ )$ este inversa transformării $ShiftRows(\ )$. Aceasta reprezintă o permutare ciclică la dreapta aplicată octeților tabloului de stare. Octeții tabloului de stare sunt permutați ciclic cu un număr diferit de poziții. În linia întâi, $r=0$, nu este aplicată operația de permutare. La ultimele 3 linii este aplicată operația de permutare la dreapta cu $Nb-shift(r,Nb)$ octeți, unde valoarea permutării $shift(r,Nb)$ depinde de numărul liniei:

$$shift(1,4)=1,\ shift(2,4)=2,\ shift(3,4)=3.$$

Astfel, octetul de pe poziția s în linia r trece în poziția $\mathrm{mod}(s+Nb-shift(r,4),Nb)$. Mai exact, transformarea $InvShiftRows(\ )$ acționează astfel:

$$s'_{r,\mathrm{mod}(c+shift(r,Nb),Nb)} := s_{r,c},\ r=\overline{1,3}, c=\overline{0,Nb-1}.$$

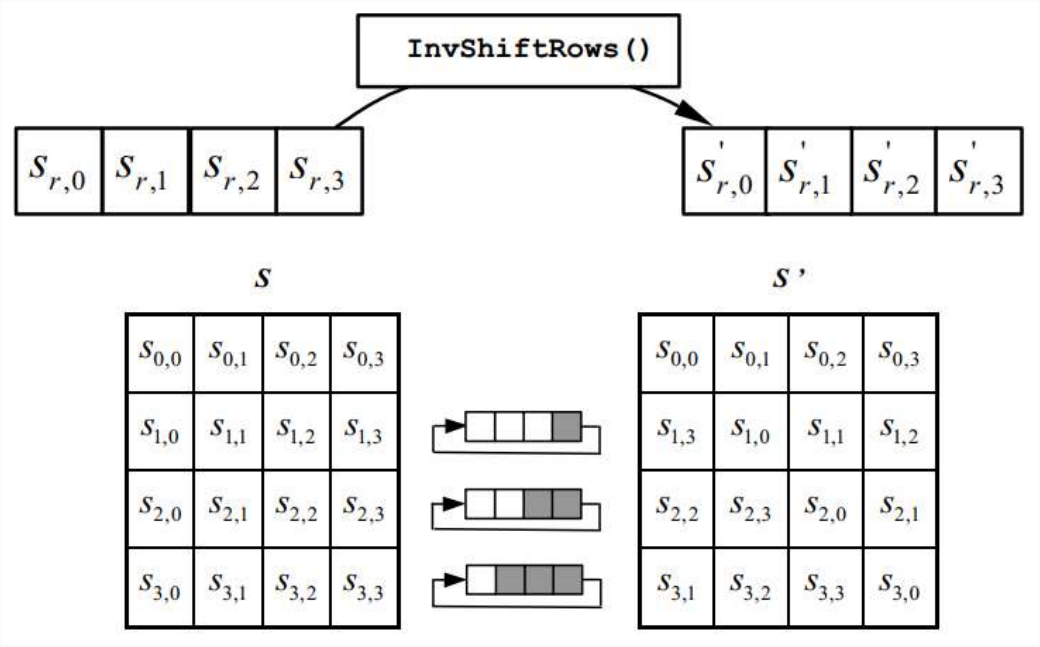


Figura 1.3.23. Transformarea InvShiftRows

Transformarea InvSubBytes()

$InvSubBytes(\ )$ este inversa transformării $SubBytes(\ )$ și este la fel o transformare de substituție a octeților, în care este aplicată inversa S-boxei la fiecare octet al tabelului de stare. Aceasta se obține prin inversarea transformării afine $b_i' := b_i \oplus b_{\text{mod}(i+4,8)} \oplus b_{\text{mod}(i+5,8)} \oplus b_{\text{mod}(i+6,8)} \oplus b_{\text{mod}(i+7,8)} \oplus c_i$, $i = \overline{0,7}$ (semnificațiile lui b_i și c_i le găsiți la secțiunea 1.3.6.2.2.2.1.1), urmată de calculul inversului multiplicativ în $GF\left(2^8\right)$ (a se vedea Tabelul 1.3.13).

Transformarea afină inversă este definită astfel:

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

Tabelul 1.3.17. Inversa S-boxei: valorile de substituție pentru octetul xy

SISTEMUL DE CRIPTARE AES

		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

1.3.6.2.2.4.3 Transformarea InvMixColumns()

$InvMixColumns(\ )$ este inversa transformării $MixColumns(\ )$ și acționează în mod similar asupra coloanelor tabelului de stare, considerând pe baza fiecărei coloane un polinom de gradul 3 peste $GF\left(2^8\right)$. Fiecare coloană este transformată prin multiplicarea acesteia modulo x^4+1 cu un polinom dat $a^{-1}(x)$, definit astfel încât $a(x)\otimes a^{-1}(x)=\{01\}_{16}$. Astfel, $a^{-1}(x)$ este de forma

$$a^{-1}(x)=\{0b\}_{16}x^3+\{0d\}_{16}x^2+\{09\}_{16}x+\{0e\}_{16}.$$

După cum a fost menționat anterior, această transformare poate fi scrisă sub forma unei operații de înmulțire matricială. Dacă $s'(x) = a^{-1}(x) \otimes s(x)$, atunci

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{pmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{pmatrix} \begin{pmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{pmatrix}, \quad c = \overline{0, Nb-1}.$$

Ca și rezultat al acestei operații de înmulțire matricială, cei 4 octeți ai coloanei examinate sunt înlocuiți prin următorii:

$$s'_{0,c} = \left(\{0e\}_{16} \cdot s_{0,c}\right) \oplus \left(\{0b\}_{16} \cdot s_{1,c}\right) \oplus \left(\{0d\}_{16} \cdot s_{2,c}\right) \oplus \left(\{09\}_{16} \cdot s_{3,c}\right),$$

$$s'_{1,c} = \left(\{09\}_{16} \cdot s_{0,c}\right) \oplus \left(\{0e\}_{16} \cdot s_{1,c}\right) \oplus \left(\{0b\}_{16} \cdot s_{2,c}\right) \oplus \left(\{0d\}_{16} \cdot s_{3,c}\right),$$

$$s'_{2,c} = \left(\{0d\}_{16} \cdot s_{0,c}\right) \oplus \left(\{09\}_{16} \cdot s_{1,c}\right) \oplus \left(\{0e\}_{16} \cdot s_{2,c}\right) \oplus \left(\{0b\}_{16} \cdot s_{3,c}\right),$$

$$s'_{3,c} = \left(\{0b\}_{16} \cdot s_{0,c}\right) \oplus \left(\{0d\}_{16} \cdot s_{1,c}\right) \oplus \left(\{09\}_{16} \cdot s_{2,c}\right) \oplus \left(\{0e\}_{16} \cdot s_{3,c}\right).$$

Procedura echivalentă de decriptare

În procedura de decriptare prezentată mai sus, ordinea în care sunt aplicate transformările diferă de cea de la procedura de criptare, pe când cheile de rundă rămân aceleași, atât pentru algoritmul criptare, cât și pentru algoritmul de decriptare. Cu toate acestea, unele proprietăți ale transformărilor utilizate în algoritmul AES, permit elaborarea unei proceduri echivalente de decriptare, care să aibă aceeași ordine de execuție a secvenței de transformări ca și procedura de criptare, doar că transformările componente ale rundei sunt înlocuite cu inversele lor. Cheile de rundă în această reprezentare sunt diferite de cheile de rundă utilizate în algoritmul de criptare. Scopul urmărit poate fi atins dacă se efectuează o modificare în procedura de expandare a cheii.

Prezentăm două proprietăţi ale transformărilor implicate, care permit formularea unei proceduri echivalente de decriptare:

1. Transformările $SubBytes(\ )$ şi $ShiftRows(\ )$ sunt comutative în sensul că aplicarea transformării $SubBytes(\ )$, urmată imediat de $ShiftRows(\ )$ este echivalentă cu aplicarea transformării $ShiftRows(\ )$, urmată imediat de $SubBytes(\ )$. $ShiftRows(\ )$ transpune octeţii cuvântului şi nu are efect asupra valorilor acestora, pe când $SubBytes(\ )$ lucrează pe octeţi independent de poziţia acestora. Acelaşi lucru este adevărat pentru inversele acestor transformări, $InvSubBytes(\ )$ şi $InvShiftRows(\ )$.
2. Operaţiile de mixare a coloanelor - $MixColumns(\ )$ şi $InvMixColumns(\ )$ - sunt liniare în raport cu datele de intrare, adică
$$InvMixColumns(state \oplus Round\ Key) = InvMixColumns(state) \oplus InvMixColumns(Round\ Key).$$

Cele două proprietăți permit să se inverseze ordinea de aplicare a transformărilor $InvSubBytes(\ )$ și $InvShiftRows(\ )$, precum și ordinea transformărilor $AddRoundKey(\ )$ și $InvMixColumns(\ )$.

Procedura echivalentă de decriptare este definită în baza *„Algoritmului de decriptare AES”*, în care este inversată ordinea de aplicare a transformărilor $InvSubBytes(\ )$ și $InvShiftRows(\ )$ și a transformărilor $AddRoundKey(\ )$ și $InvMixColumns(\ )$. Procedura de expandare a cheii se va modifica pentru $round = \overline{1, Nr-1}$ (cu excepția primei și a ultimei runde), folosind transformarea $InvMixColumns(\ )$.

SISTEMUL DE CRIPTARE AES

Algoritmul echivalent de decriptare AES

Date de intrare:

Nb - numărul de cuvinte pe 32 de biți ale blocului de text

Nr - numărul de runde (10, 12 sau 14 în dependență de lungimea cheii)

in - tablou de lungime $4Nb$ cu elementele octeți

dw - tablou de lungime $Nb(Nr+1)$ cu elementele cuvinte (tabloul expandării modificate a cheii)

Date de ieșire:

out - tablou de lungime $4Nb$ cu elementele octeți

state - tablou bidimensional de dimensiune $4 \times Nb$

SISTEMUL DE CRIPTARE AES

$state := in$

$AddRoundKey\left(state, dw_{Nr*Nb,(Nr+1)Nb-1}\right)$

Pentru $round = \overline{Nr-1:-1:1}$ execută

{

$InvSubBytes\left(state\right)$

$InvShiftRows\left(state\right)$

$InvMixColumns\left(state\right)$

$AddRoundKey\left(state, dw_{round*Nb,(round+1)*Nb-1}\right)$

}

$InvSubBytes\left(state\right)$

$InvShiftRows\left(state\right)$

$AddRoundKey\left(state, dw_{0,Nb-1}\right)$

$out := state$

Elementele matricei corespunzătoare operației inverse a transformării *MixColumns* sunt mai mari ca cele ale matricei corespunzătoare din algoritmul de criptare AES (acolo avem valorile 1,2 și 3). De aceea, procedura de înmulțire a matricei la vector va necesita mai mult timp de execuție.

Algoritmul modificat de expandare a cheii AES
Date de intrare: *Nb* - numărul de cuvinte pe 32 de biți ale blocului de text *Nr* - numărul de runde (10, 12 sau 14 în dependență de lungimea cheii) *Nk* - numărul de cuvinte pe 32 biți din care este formată cheia secretă *key* - tablou de octeți ce conține $4Nk$ elemente *w* - tablou de cuvinte ce conține $Nb(Nr+1)$ elemente Date de ieșire: *dw* - tablou de cuvinte ce conține $Nb(Nr+1)$ elemente
temp - cuvânt intermediar

Pentru $i := \overline{0, Nk-1}$ execută
{
$w_i := key_{4i} \,||\, key_{4i+1} \,||\, key_{4i+2} \,||\, key_{4i+3}$
}
Pentru $i := \overline{Nk, Nb(Nr+1)-1}$ execută
{
$temp := w_{i-1}$
Dacă $\mathrm{mod}(i, Nk) = 0$ atunci { $temp := SubWord(RotWord(temp)) \oplus Rcon_{i/Nk}$ }
Altfel dacă $(Nk > 6) and (\mathrm{mod}(i, Nk) = 4)$ { $temp := SubWord(temp)$ }
$w_i := w_{i-Nk} \oplus temp$
}

Pentru $i := \overline{0, (Nr+1)Nb-1}$ execută { $dw_i := w_i$ }
Pentru $round := \overline{1, Nr-1}$ execută { $InvMixColumns(dw_{round*Nb, (round+1)*Nb-1})$ }
% Se modifică tipul intrării pentru $InvMixColumns()$ din tablou unidimensional în bidimensional