

Cifrari a blocchi: Advanced Encryption Standard

a.a. 2019/20

Alfredo De Santis

Dipartimento di Informatica
Università di Salerno

ads@unisa.it

<http://www.di-srv.unisa.it/~ads>



Marzo 2020

Advanced Encryption Standard

- Il *National Institute of Standard and Technology* (NIST) propose il DES come standard nel 1977 ...
- DES riaffermato nel 1993 fino a Dicembre 1998
- Critiche al DES:
 - chiave di soli 56 bit
 - blocchi di 64 bit
 - criteri costruttivi non chiari (ci sono trapdoor nelle S-box?)
 - lento nell'implementazione software (3-DES ancora di più)
- Obiettivo del NIST:
 - nuovo cifrario a blocchi per uso commerciale e governativo più **sicuro** ed **efficiente** di DES triplo

Processo di Selezione AES

- 12 Settembre 1997: il NIST indice un concorso pubblico per la nomina dell'AES
- Pubblico scrutinio (<http://www.nist.gov/AES>)
- Prima conferenza AES, 20-23 agosto 1998
(presentazione di 15 candidature)
- Pubblico scrutinio
- Seconda conferenza AES, 22-23 marzo 1999
(presentazione analisi e testing)
- 9 Agosto 1999: annuncio dei 5 finalisti
(MARS, RC6, RINJDAEL, SERPENT, TWOFISH)
- Pubblico scrutinio
- Terza conferenza AES, 13-14 aprile 2000
(presentazione analisi e testing)

Processo di Selezione AES

- 2 ottobre 2000: Scelta del finalista: **RINJDAEL**
- 28 febbraio 2001: Pubblicazione di un Draft di *Federal Information Processing Standard* (FIPS)
- **Pubblico scrutinio di 90 giorni**
- Approvato come FIPS 197, 26 novembre 2001
 - Effettivo a partire dal 26 maggio 2002

Requisiti e Selezione per l'AES

Requisiti richiesti dal NIST:

- Cifrario a blocchi
- Lunghezza chiave: 128, 192, o 256 bit
- Lunghezza blocco: 128 bit
- Permette l'implementazione su smart-card
- Royalty-free

Piattaforma del NIST per l'analisi dei candidati:

- PC IBM-compatibile, Pentium Pro 200MHz, 64MB RAM, WINDOWS 95
- Compilatori Borland C++ 5.0 ed il Java Development Kit (JDK) 1.1

Selezione del NIST basata su:

- Sicurezza
- Efficienza implementazioni hardware e software
- Grandezza codice e memoria utilizzata

Documentazione dei Candidati

- Descrizione algoritmo
- Analisi algoritmo (vantaggi e limiti)
- Stima dell'efficienza computazionale
- Analisi dell'algoritmo rispetto agli attacchi di crittoanalisi più conosciuti (ad esempio known o chosen plaintext)
- Implementazione di riferimento in ANSI C
- Implementazione ottimizzata dell'algoritmo implementata in ANSI C e Java

AES: Finalisti e candidati

RIJNDAEL **Joan Daemen, Vincent Rijmen**

MARS IBM

RC6 RSA Laboratories

SERPENT R. Anderson, E. Biham, L. Knudsen

TWOFISH B.Schneider, J.Kelsey, D.Whiting, D.Wagner, C.Hall, N.Ferguson

CAST-256 Entrust Technologies, INC.

CRYPTON Future System, INC.

DEAL R. Outerbridge, L.Knudsen

DFC CNRS

E2 Nippon Telegraph and Telephone Corp.

FROG TecApro Internacional S.A.

HPC L.Brown, J.Pieprzyk, J.Seberry

LOKI97 L.Brown, J.Pieprzyk, J.Seberry

MAGENTA Deutsche Telekom AG

SAFER+ Cylink Corp.

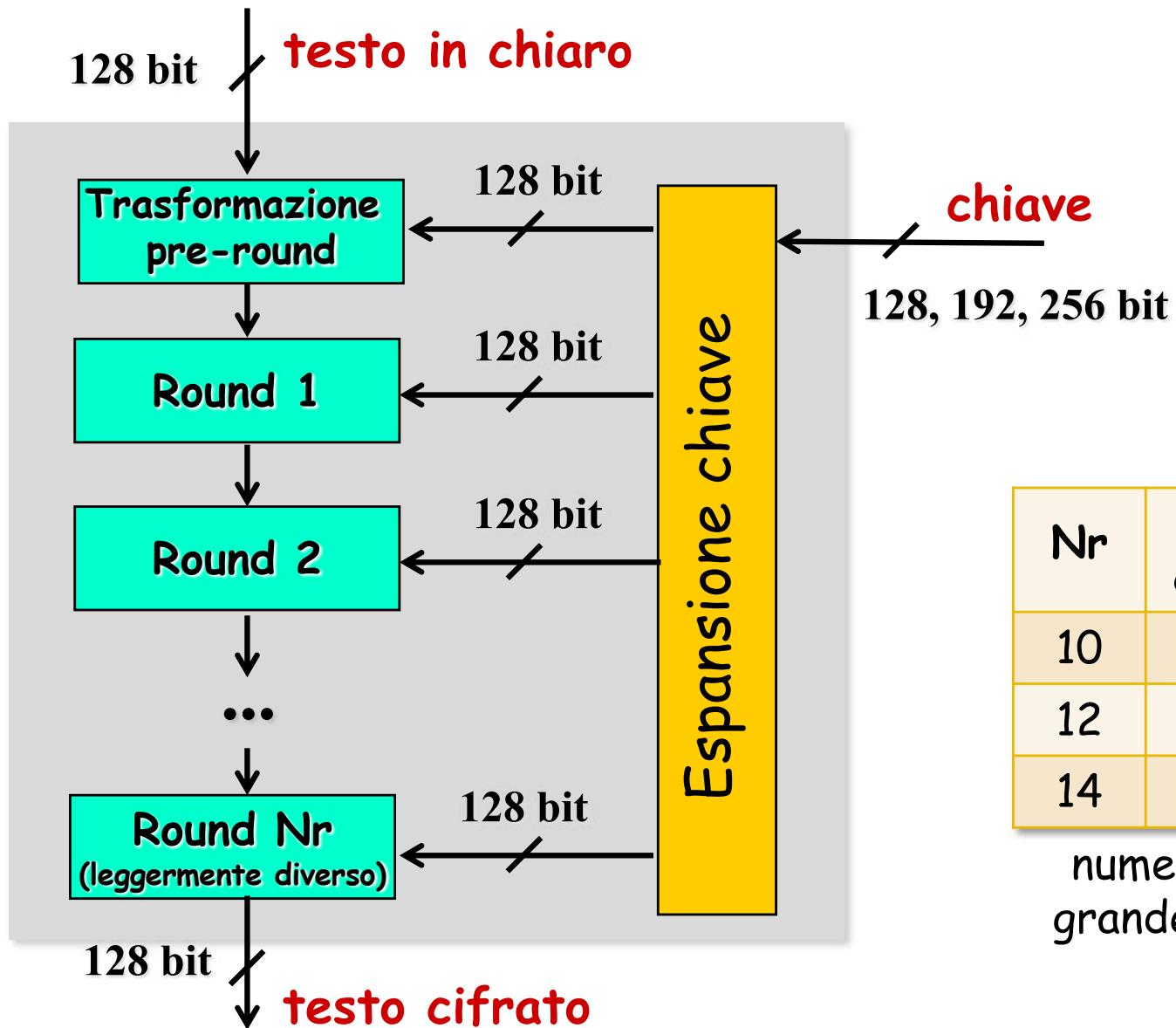
Pronuncia: Reign Dahl,
Rain Doll, Rhine Dahl

Sicurezza contro attacchi “Brute Force”

Grandezza chiavi (bits)	Numero di possibili chiavi	Tempo medio necessario con 1 decifratura/ μs	Tempo medio necessario con 10^6 decifrature/ μs
32	$2^{32} \approx 4,3 \times 10^9$	$2^{31} \mu\text{s} = 35,8$ minuti	2,15 millisecondi
56	$2^{56} \approx 7,2 \times 10^{16}$	$2^{55} \mu\text{s} = 1142$ anni	10,01 ore
112	$2^{112} \approx 5,19 \times 10^{33}$	$2^{111} \mu\text{s} = 8,2 \times 10^{19}$ anni	$8,2 \times 10^{13}$ anni
128	$2^{128} \approx 3,4 \times 10^{38}$	$2^{127} \mu\text{s} = 5,4 \times 10^{24}$ anni	$5,4 \times 10^{18}$ anni
168	$2^{168} \approx 3,7 \times 10^{50}$	$2^{167} \mu\text{s} = 5,9 \times 10^{36}$ anni	$5,9 \times 10^{30}$ anni
192	$2^{192} \approx 6,2 \times 10^{57}$	$2^{167} \mu\text{s} = 9,9 \times 10^{43}$ anni	$9,9 \times 10^{37}$ anni
256	$2^{256} \approx 1,15 \times 10^{77}$	$2^{255} \mu\text{s} = 1,83 \times 10^{63}$ anni	$1,83 \times 10^{57}$ anni

Stima età universo $13,798 \pm 0,037$ miliardi anni $\approx 4 \times 10^{17}$ secondi
 (atre stime 13-15 miliardi anni) $\approx 4 \times 10^{23}$ microsecondi

Struttura AES



Nr	Grandezza chiave in bit
10	128
12	192
14	256

numero round e
grandezza chiave

AES: Rijndael

- Non è un cifrario di Feistel
 - Lavora sull'intero blocco in input
- Usa operazioni facilmente ed efficientemente implementabili
 - Sia su architetture ad 8 bit che a 32 bit

State

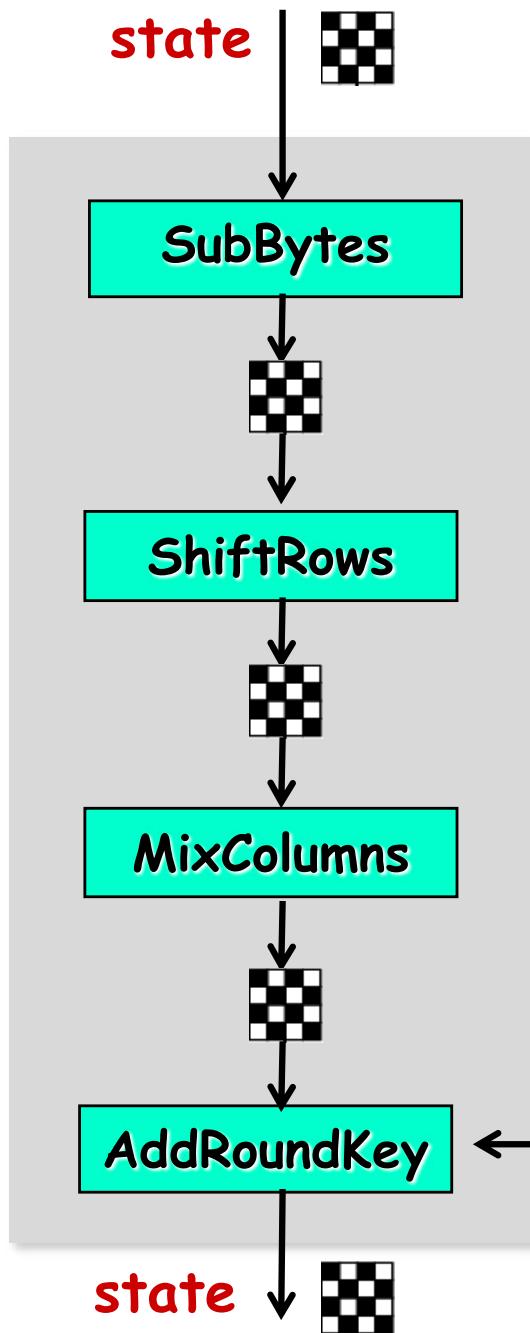
Operazioni effettuate su una matrice di byte, detta **state**

- 4 righe
- 4 colonne, costituite da word a 32 bit
- $S_{r,c}$ byte in riga r e colonna c

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

state

Struttura round



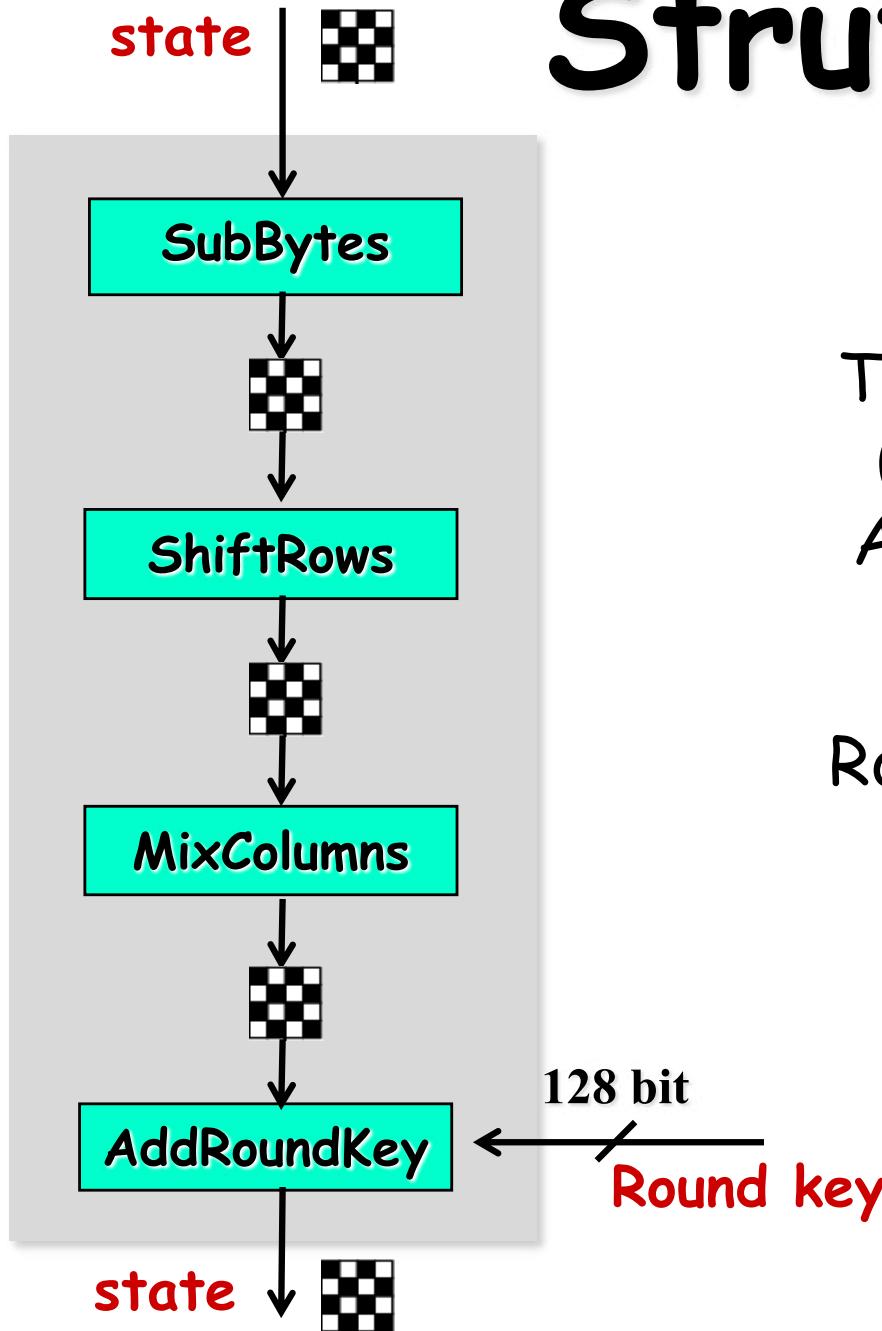
sostituzione di ognuno dei
16 byte mediante S-box

shift ciclico dei 4 byte di ogni riga

sostituzione che usa aritmetica
sul campo finito $GF(2^8)$

XOR bit a bit con chiave
espansa

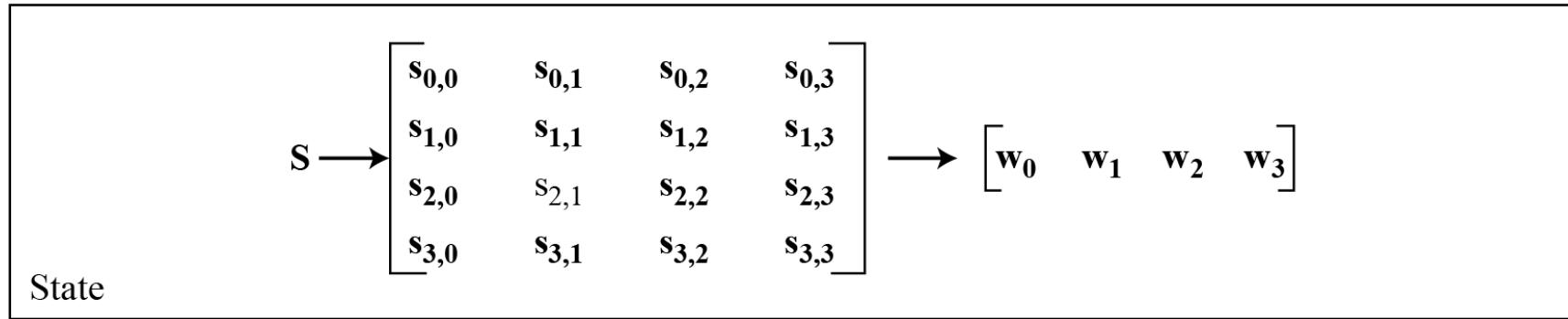
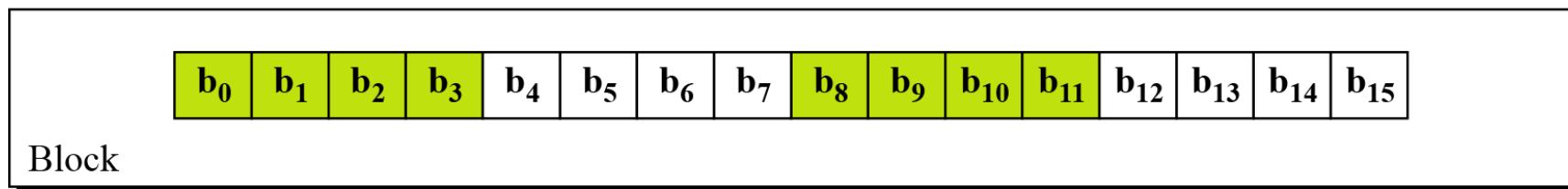
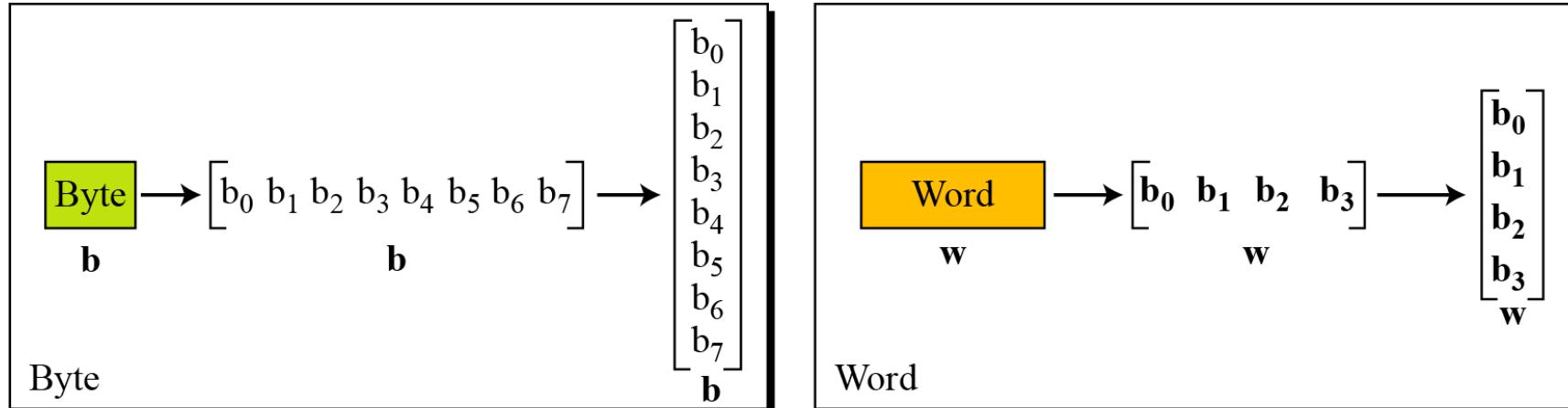
Struttura round



Trasformazione pre-round
(prima del primo round):
AddRoundKeyMixColumns

Round Nr (ultimo round):
manca MixColumns

Unità dati in AES



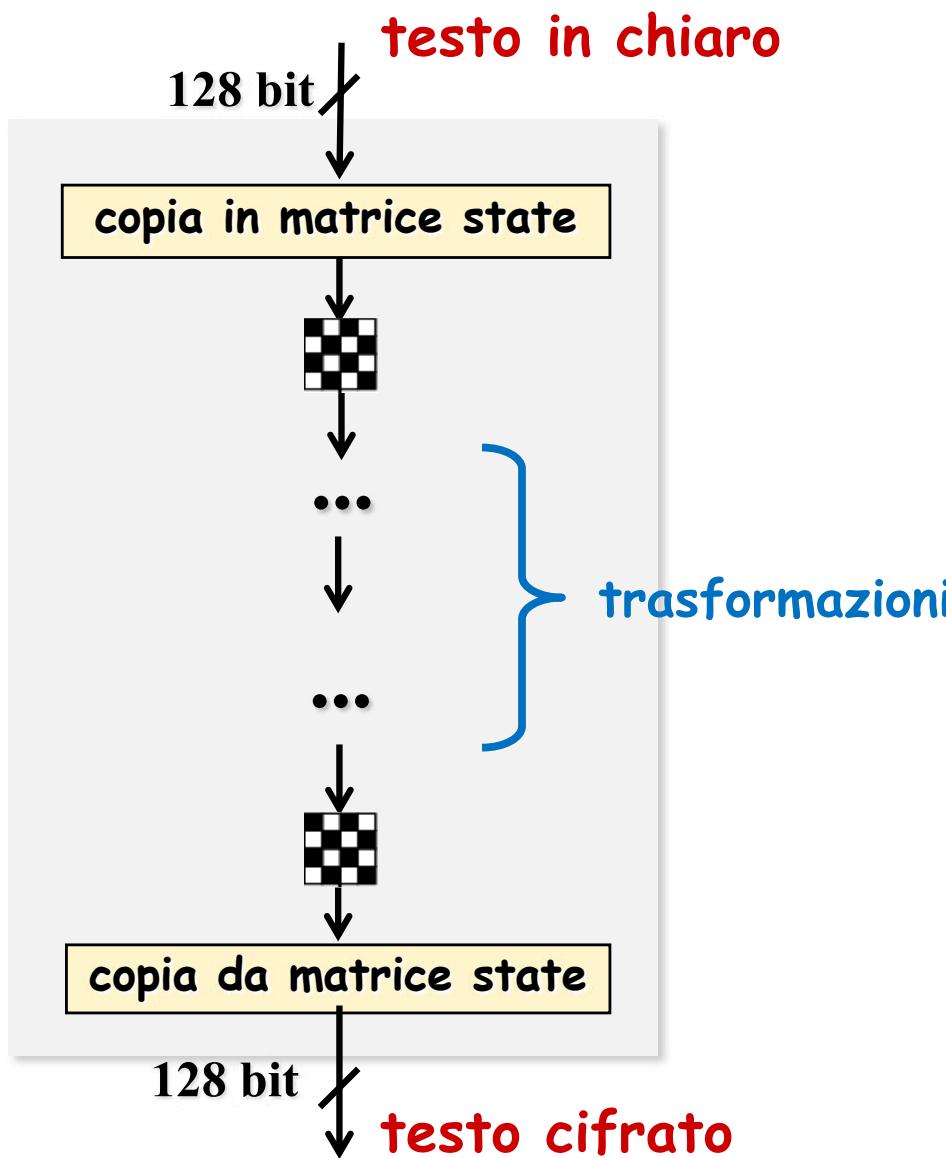
Chiave

Chiave di lunghezza variabile (128, 192, 256 bit)

- rappresentata come una **matrice di byte** con 4 righe e N_k colonne, $N_k = \text{lungh. chiave} / 32$
 - chiave 128 bit = 16 byte $\rightarrow N_k=4$
 - chiave 192 bit = 24 byte $\rightarrow N_k=6$
 - chiave 256 bit = 32 byte $\rightarrow N_k=8$

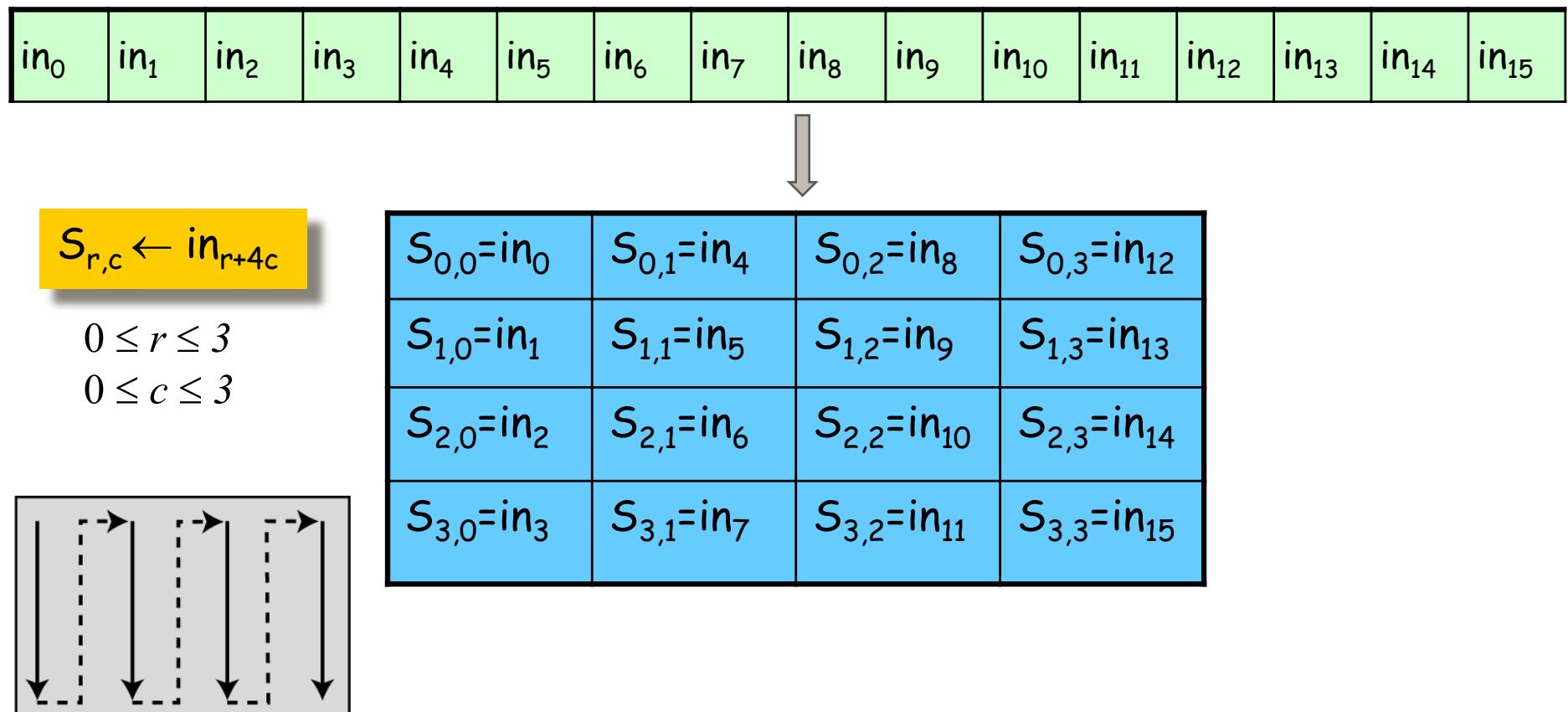
$K_{0,0}$	$K_{0,1}$	$K_{0,2}$	$K_{0,3}$
$K_{1,0}$	$K_{1,1}$	$K_{1,2}$	$K_{1,3}$
$K_{2,0}$	$K_{2,1}$	$K_{2,2}$	$K_{2,3}$
$K_{3,0}$	$K_{3,1}$	$K_{3,2}$	$K_{3,3}$

Computazioni su matrice state



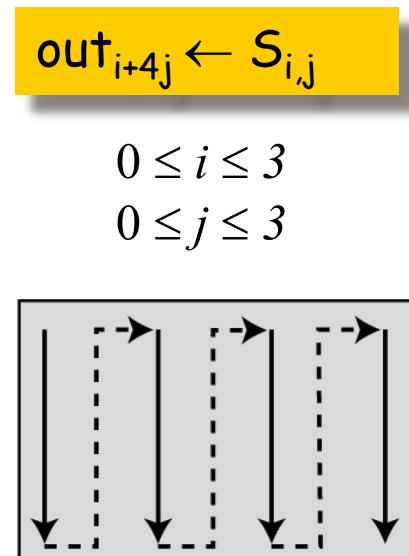
State

Matrice di byte in input copiata nella matrice **state**



State

Al termine, matrice **state** copiata nella matrice output



$S_{0,0} = \text{out}_0$	$S_{0,1} = \text{out}_4$	$S_{0,2} = \text{out}_8$	$S_{0,3} = \text{out}_{12}$
$S_{1,0} = \text{out}_1$	$S_{1,1} = \text{out}_5$	$S_{1,2} = \text{out}_9$	$S_{1,3} = \text{out}_{13}$
$S_{2,0} = \text{out}_2$	$S_{2,1} = \text{out}_6$	$S_{2,2} = \text{out}_{10}$	$S_{2,3} = \text{out}_{14}$
$S_{3,0} = \text{out}_3$	$S_{3,1} = \text{out}_7$	$S_{3,2} = \text{out}_{11}$	$S_{3,3} = \text{out}_{15}$

Diagram illustrating the mapping from state matrix indices to output array indices. A large downward-pointing arrow indicates the flow of data from the state matrix to the output array.

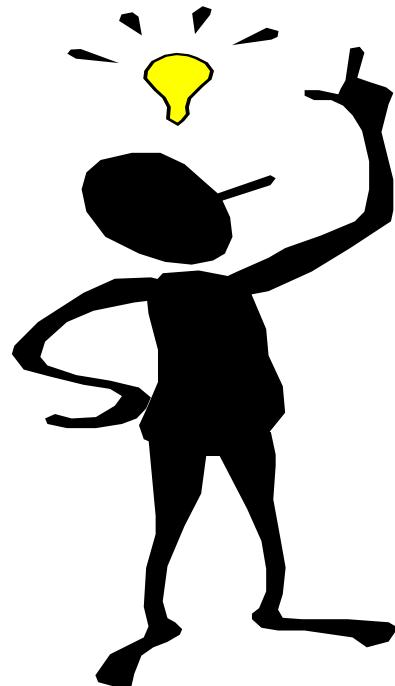
out_0	out_1	out_2	out_3	out_4	out_5	out_6	out_7	out_8	out_9	out_{10}	out_{11}	out_{12}	out_{13}	out_{14}	out_{15}
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------

Preliminari matematici

- Operazioni sui byte
 - Addizione e moltiplicazione
 - Efficienti computazionalmente
 - Elegante struttura matematica
- Operazioni sulle word di 32 bit
 - Addizione e moltiplicazione
 - Efficienti computazionalmente
 - Elegante struttura matematica

Preliminari

Il byte è l'unità di base nella computazione dell'AES



- Rappresentazione dei byte
- Operazioni sui byte
Addizione e moltiplicazione
- Struttura di $GF(2^8)$
Campo finito con 256 elementi

Byte

I valori dei byte sono rappresentati in notazione esadecimale

- Due cifre esadecimali per ciascun byte

$$\{ \underbrace{11}_{\text{d4}} \underbrace{01}_{\text{d4}} \underbrace{01}_{\text{d4}} \underbrace{00}_{\text{d4}} \} \rightarrow \text{d4}$$

Ciascun byte è interpretato come un elemento del campo finito $GF(2^8)$

$$\{b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0\} \rightarrow b_7 x^7 + b_6 x^6 + b_5 x^5 + b_4 x^4 + b_3 x^3 + b_2 x^2 + b_1 x^1 + b_0$$

$$\{11010100\} \rightarrow x^7 + x^6 + x^4 + x^2$$

Addizione su byte

Addizione in $GF(2^8)$: somma corrisponde al polinomio i cui coefficienti sono la somma modulo 2 dei coefficienti dei due polinomi

$$(x^6 + x^4 + x^2 + x + 1) + (x^7 + x + 1) = x^7 + x^6 + x^4 + x^2$$

$$\{01010111\} \oplus \{10000011\} = \{11010100\}$$

$$\{57\} \oplus \{83\} = \{d4\}$$

Addizione su byte

Addizione in $GF(2^8)$: somma corrisponde al polinomio i cui coefficienti sono la somma modulo 2 dei coefficienti dei due polinomi

$$(x^6 + x^4 + x^2 + x + 1) + (x^7 + x + 1) = x^7 + x^6 + x^4 + x^2$$

$$\{01010111\} \oplus \{10000011\} = \{11010100\}$$

$$\{57\} \oplus \{83\} = \{d4\}$$

Algoritmo per somma di 2 byte:
xor dei byte



Esempio moltiplicazione

$$\{01010111\} \bullet \{10000011\}$$

$$\{57\} \bullet \{83\}$$

$$\begin{aligned}(x^6 + x^4 + x^2 + x + 1) (x^7 + x + 1) &= x^{13} + x^{11} + x^9 + x^8 + x^7 + \\&\quad x^7 + x^5 + x^3 + x^2 + x + \\&\quad x^6 + x^4 + x^2 + x + 1 \\&= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1\end{aligned}$$

➤ Non va in un byte, occorrono più bit



Esempio moltiplicazione

$\{01010111\} \bullet \{10000011\}$

$\{57\} \bullet \{83\}$

$$\begin{aligned}(x^6 + x^4 + x^2 + x + 1) (x^7 + x + 1) &= x^{13} + x^{11} + x^9 + x^8 + x^7 + \\&\quad x^7 + x^5 + x^3 + x^2 + x + \\&\quad x^6 + x^4 + x^2 + x + 1 \\&= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1\end{aligned}$$

➤ Non va in un byte, occorrono più bit



➤ Riduzione modulo il polinomio $m(x) = x^8 + x^4 + x^3 + x + 1$

Esempio moltiplicazione

$$\{01010111\} \bullet \{10000011\} = \{11000001\}$$

$$\{57\} \bullet \{83\} = \{c1\}$$

$$\begin{aligned}(x^6 + x^4 + x^2 + x + 1) (x^7 + x + 1) &= x^{13} + x^{11} + x^9 + x^8 + x^7 + \\&\quad x^7 + x^5 + x^3 + x^2 + x + \\&\quad x^6 + x^4 + x^2 + x + 1 \\&= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1\end{aligned}$$

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{modulo } x^8 + x^4 + x^3 + x + 1$$

$$= x^7 + x^6 + 1$$

dividiamo per $m(x)$ e
teniamo il resto

Divisione polinomi

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{diviso} \quad x^8 + x^4 + x^3 + x + 1$$

$$\begin{array}{c} x^8 + x^4 + x^3 + x + 1 \end{array} \overline{\left. \begin{array}{l} x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \end{array} \right.}$$

Divisione polinomi

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{diviso} \quad x^8 + x^4 + x^3 + x + 1$$

$$\begin{array}{c} x^{13} \text{ diviso } x^8 \\ \hline x^8 + x^4 + x^3 + x + 1 \quad \overline{x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1} \\ x^5 \end{array}$$

Divisione polinomi

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{diviso} \quad x^8 + x^4 + x^3 + x + 1$$

$$\begin{array}{r} x^5 \\ \hline x^8 + x^4 + x^3 + x + 1 \quad \sqrt{x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1} \\ x^{13} + x^9 + x^8 + x^6 + x^5 \end{array}$$

$x^8 + x^4 + x^3 + x + 1$
per x^5

Divisione polinomi

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{diviso} \quad x^8 + x^4 + x^3 + x + 1$$

$$\begin{array}{r} x^5 \\ \hline x^8 + x^4 + x^3 + x + 1 \sqrt{x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1} \\ x^{13} + x^9 + x^8 + x^6 + x^5 \\ \hline x^{11} + x^4 + x^3 + 1 \end{array}$$

sottrazione

Divisione polinomi

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{diviso} \quad x^8 + x^4 + x^3 + x + 1$$

$$\begin{array}{r} x^5 \\ \hline x^8 + x^4 + x^3 + x + 1 \sqrt{x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1} \\ x^{13} + x^9 + x^8 + x^6 + x^5 \\ \hline x^{11} + x^4 + x^3 + 1 \end{array}$$

sottrazione

Divisione polinomi

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{diviso} \quad x^8 + x^4 + x^3 + x + 1$$

$$\begin{array}{r} x^5 \\ \hline x^8 + x^4 + x^3 + x + 1 \quad \sqrt{x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1} \\ \underline{-x^{13} - x^{11} - x^9 - x^8 - x^6 - x^5 - x^4 - x^3} \\ x^{11} + x^4 + x^3 + 1 \end{array}$$

Quindi:

$$\begin{aligned} & x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \\ & = (x^8 + x^4 + x^3 + x + 1)(x^5) + (x^{11} + x^4 + x^3 + 1) \end{aligned}$$

Divisione polinomi

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{diviso} \quad x^8 + x^4 + x^3 + x + 1$$

$$\begin{array}{r} x^5 + x^3 \\ \hline x^8 + x^4 + x^3 + x + 1 \quad \sqrt{x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1} \\ x^{13} + x^9 + x^8 + x^6 + x^5 \\ \hline x^{11} + x^4 + x^3 + 1 \\ x^{11} + x^7 + x^6 + x^4 + x^3 \\ \hline x^7 + x^6 + 1 \end{array}$$

Quindi:

$$\begin{aligned} & x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \\ & = (x^8 + x^4 + x^3 + x + 1)(x^5 + x^3) + (x^7 + x^6 + 1) \end{aligned}$$

Divisione polinomi

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{diviso} \quad x^8 + x^4 + x^3 + x + 1$$

$$\begin{array}{r} x^5 + x^3 \\ \hline x^8 + x^4 + x^3 + x + 1 \quad \swarrow \\ x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \\ x^{13} + x^9 + x^8 + x^6 + x^5 \\ \hline x^{11} + x^4 + x^3 + 1 \\ x^{11} + x^7 + x^6 + x^4 + x^3 \\ \hline x^7 + x^6 + 1 \end{array}$$

Quoziente della divisione

Resto della divisione

Quindi:

$$\begin{aligned} & x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \\ & = (x^8 + x^4 + x^3 + x + 1)(x^5 + x^3) + (x^7 + x^6 + 1) \end{aligned}$$

Esempio moltiplicazione

$$\{01010111\} \bullet \{10000011\} = \{11000001\}$$

$$\{57\} \bullet \{83\} = \{c1\}$$

$$\begin{aligned}(x^6 + x^4 + x^2 + x + 1) (x^7 + x + 1) &= x^{13} + x^{11} + x^9 + x^8 + x^7 + \\&\quad x^7 + x^5 + x^3 + x^2 + x + \\&\quad x^6 + x^4 + x^2 + x + 1 \\&= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1\end{aligned}$$

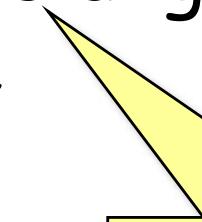
$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \quad \text{modulo } x^8 + x^4 + x^3 + x + 1$$

$$= x^7 + x^6 + 1$$

dividiamo per $m(x)$ e
teniamo il resto

Moltiplicazione su byte

- Moltiplicazione in $GF(2^8)$ (denotata da \bullet) corrisponde alla moltiplicazione di polinomi modulo un polinomio irriducibile di grado 8
- Il risultato è un polinomio di grado ≤ 7



unici divisori:
1 e se stesso

Moltiplicazione su byte

- Moltiplicazione in $GF(2^8)$ (denotata da \bullet) corrisponde alla moltiplicazione di polinomi modulo un polinomio irriducibile di grado 8
 - Il risultato è un polinomio di grado ≤ 7
- Polinomio irriducibile per AES:
 $m(x) = x^8 + x^4 + x^3 + x + 1$
- E' solo uno dei 30 polinomi irriducibili di grado 8

unici divisori:
1 e se stesso

Polinomi irriducibili

Numero polinomi irriducibili grado n, coefficienti GF(2)

n	1	2	3	4	5	6	7	8	9	10	11	12
#poly	2	1	2	3	6	9	18	30	56	99	186	335

n	irreducible polynomials
1	$1 + x, x$
2	$1 + x + x^2$
3	$1 + x + x^3, 1 + x^2 + x^3$
4	$1 + x + x^4, 1 + x + x^2 + x^3 + x^4, 1 + x^3 + x^4$
5	$1 + x^2 + x^5, 1 + x + x^2 + x^3 + x^5, 1 + x^3 + x^5, 1 + x + x^3 + x^4 + x^5, 1 + x^2 + x^3 + x^4 + x^5, 1 + x + x^2 + x^4 + x^5$

Proprietà

➤ Addizione

- Associativa
- Commutativa
- Identità {00}
- Esiste inverso $-a(x)$ per ogni $a(x)$

$$\begin{aligned}(a(x)+b(x))+c(x) &= a(x)+(b(x)+c(x)) \\ a(x)+b(x) &= b(x)+a(x) \\ a(x)+0 &= 0+a(x) = a(x) \\ -a(x)+a(x) &= 0\end{aligned}$$

➤ Moltiplicazione

- Associativa
 - Commutativa
 - Identità {01}
 - Esiste inverso $a^{-1}(x)$ per ogni $a(x) \neq 0$
- $a(x) \bullet (b(x) + c(x)) = a(x) \bullet b(x) + a(x) \bullet c(x)$

$$\begin{aligned}(a(x) \bullet b(x)) \bullet c(x) &= a(x) \bullet (b(x) \bullet c(x)) \\ a(x) \bullet b(x) &= b(x) \bullet a(x) \\ a(x) \bullet 1 &= 1 \bullet a(x) = a(x) \\ a^{-1}(x) \bullet a(x) &= 1\end{aligned}$$

Proprietà

- gruppo abeliano
- Addizione
 - Associativa $(a(x)+b(x))+c(x) = a(x)+(b(x)+c(x))$
 - Commutativa $a(x)+b(x) = b(x)+a(x)$
 - Identità {00} $a(x)+0 = 0+a(x) = a(x)$
 - Esiste inverso $-a(x)$ per ogni $a(x)$ $-a(x)+a(x) = 0$
 - Moltiplicazione
 - Associativa $(a(x)\bullet b(x))\bullet c(x) = a(x)\bullet(b(x)\bullet c(x))$
 - Commutativa $a(x)\bullet b(x) = b(x)\bullet a(x)$
 - Identità {01} $a(x)\bullet 1 = 1\bullet a(x) = a(x)$
 - Esiste inverso $a^{-1}(x)$ per ogni $a(x) \neq 0$ $a^{-1}(x)\bullet a(x) = 1$
 - $a(x)\bullet(b(x) + c(x)) = a(x)\bullet b(x) + a(x)\bullet c(x)$
 - Struttura del campo finito $GF(2^8)$

Calcolo della moltiplicazione

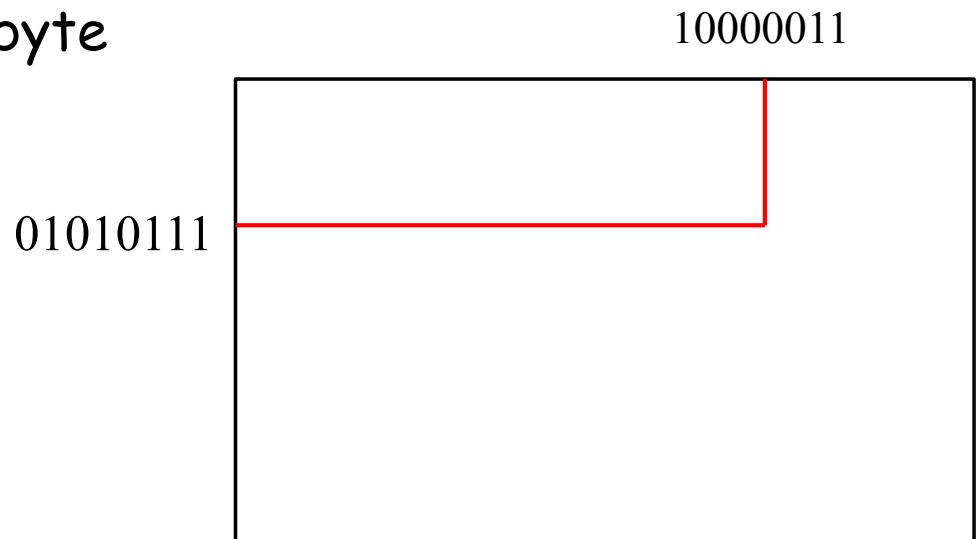
Calcoliamo $\{01010111\} \cdot \{10000011\}$

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\}$

Primo metodo

- Costruiamo una tabella 255×255 (evitiamo "0")
- Ogni entrata è il byte corrispondente al prodotto
- Grandezza tabella 65.025 byte



Calcolo della moltiplicazione

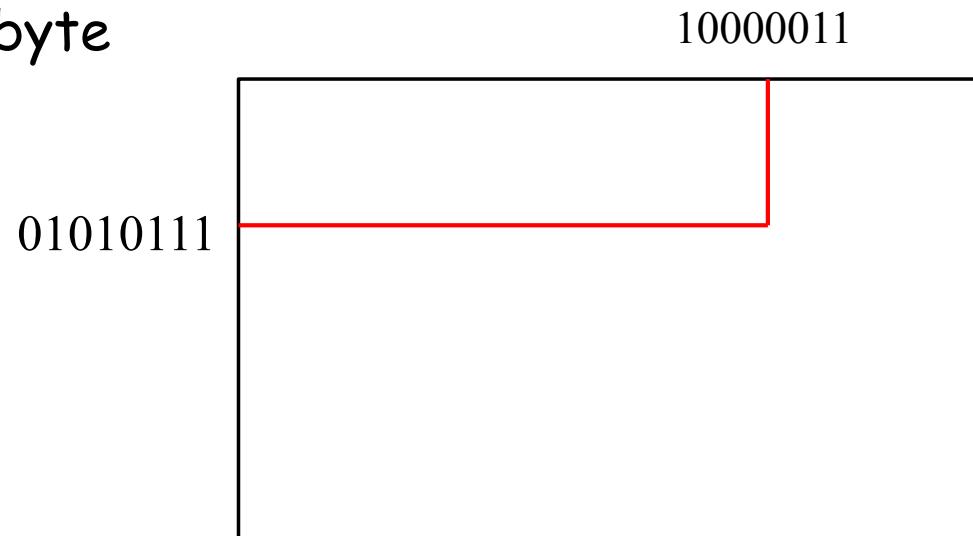
Calcoliamo $\{01010111\} \cdot \{10000011\}$

Primo metodo

- Costruiamo una tabella 255×255 (evitiamo "0")
- Ogni entrata è il byte corrispondente al prodotto
- Grandezza tabella 65.025 byte

Vediamo un secondo metodo

- non richiede tabelle!
- solo calcoli
- senza divisioni tra polinomi
- efficiente



Calcolo della moltiplicazione secondo metodo

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\begin{aligned}\{01010111\} \cdot \{10000011\} \\ = \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\})\end{aligned}$$


$$\begin{array}{r} 10000000 \oplus \\ 00000010 \oplus \\ \hline 00000001 \\ \hline 10000011 \end{array}$$

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\begin{aligned}\{01010111\} \cdot \{10000011\} \\&= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\}) \\&= \{01010111\} \cdot \{10000000\} \\&\quad + \{01010111\} \cdot \{00000010\} \\&\quad + \{01010111\} \cdot \{00000001\}\end{aligned}$$

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\{01010111\} \cdot \{10000011\}$$

$$= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\})$$

$$= \{01010111\} \cdot \{10000000\}$$

$$+ \{01010111\} \cdot \{00000010\}$$

$$+ \{01010111\} \cdot \{00000001\}$$

$$01010111 \oplus 00000001 = 01010111$$

Perché 00000001 è l'elemento neutro

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\{01010111\} \cdot \{10000011\}$$

$$= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\})$$

$$= \{01010111\} \cdot \{10000000\}$$

$$+ \{01010111\} \cdot \{00000010\}$$

$$+ \{01010111\}$$

$$01010111 \oplus 00000010$$

$$\begin{aligned} \text{Cioè } & (x^6+x^4+x^2+x+1)x \\ & = x^7+x^5+x^3+x^2+x \end{aligned}$$

Quindi, il prodotto è 10101110

Ovvero, uno shift a sx

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\{01010111\} \cdot \{10000011\}$$

$$= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\})$$

$$= \{01010111\} \cdot \{10000000\}$$

$$+ \{10101110\}$$

$$+ \{01010111\}$$

$$01010111 \oplus 10000000$$

$$\text{Cioè } (x^6 + x^4 + x^2 + x + 1) x^7$$

$$= (x^7 + x^5 + x^3 + x^2 + x) x^6$$

$$= (x^8 + x^6 + x^4 + x^3 + x^2) x^5$$

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\{01010111\} \cdot \{10000011\}$$

$$= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\})$$

$$= \{01010111\} \cdot \{10000000\}$$

$$+ \{10101110\}$$

$$+ \{01010111\}$$

$$01010111 \oplus 10000000$$

$$\text{Cioè } (x^6 + x^4 + x^2 + x + 1) x^7$$

$$= (x^7 + x^5 + x^3 + x^2 + x) x^6$$

$$= (x^8 + x^6 + x^4 + x^3 + x^2) x^5$$

$$= (x^8 + x^6 + x^4 + x^3 + x^2 + x^8 + x^4 + x^3 + x + 1) x^5$$

$$\text{Perché mod } (x^8 + x^4 + x^3 + x + 1)$$

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\{01010111\} \cdot \{10000011\}$$

$$= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\})$$

$$= \{01010111\} \cdot \{10000000\}$$

$$+ \{10101110\}$$

$$+ \{01010111\}$$

$$x^8 + x^4 + x^3 + x + 1$$

$$01010111 \oplus 10000000$$

$$\text{Cioè } (x^6 + x^4 + x^2 + x + 1) x^7$$

$$= (x^7 + x^5 + x^3 + x^2 + x) x^6$$

$$= (x^8 + x^6 + x^4 + x^3 + x^2) x^5$$

$$= (x^6 + x^2 + x + 1) x^5$$

$$= (x^7 + x^3 + x^2 + x) x^4$$

$$= (x^8 + x^4 + x^3 + x^2) x^3$$

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\begin{aligned}\{01010111\} \cdot \{10000011\} &= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\}) \\ &= \{01010111\} \cdot \{10000000\} \\ &\quad + \{10101110\} \\ &\quad + \{01010111\}\end{aligned}$$

$x^8 + x^4 + x^3 + x + 1$

$$\begin{aligned}01010111 \oplus 10000000 \\ \text{Cioè } (x^6+x^4+x^2+x+1) x^7 \\ = (x^7+x^5+x^3+x^2+x) x^6 \\ = (x^8+x^6+x^4+x^3+x^2) x^5 \\ = (x^6+x^2+x+1) x^5 \\ = (x^7+x^3+x^2+x) x^4 \\ = (x^8+x^4+x^3+x^2) x^3 \\ = (x^8+x^4+x^3+x^2+x^8+x^4+x^3+x+1) x^3\end{aligned}$$

Perché mod $(x^8+x^4+x^3+x+1)$

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\begin{aligned}\{01010111\} \cdot \{10000011\} &= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\}) \\ &= \{01010111\} \cdot \{10000000\} \\ &\quad + \{10101110\} \\ &\quad + \{01010111\}\end{aligned}$$

$x^8 + x^4 + x^3 + x + 1$

$$01010111 \oplus 10000000$$

$$\begin{aligned}\text{Cioè } (x^6+x^4+x^2+x+1) x^7 &= (x^7+x^5+x^3+x^2+x) x^6 \\ &= (x^8+x^6+x^4+x^3+x^2) x^5 \\ &= (x^6+x^2+x+1) x^5 \\ &= (x^7+x^3+x^2+x) x^4 \\ &= (x^8+x^4+x^3+x^2) x^3 \\ &= (x^8+x^4+x^3+x^2+x^8+x^4+x^3+x+1) x^3 \\ &= (x^2+x+1) x^3\end{aligned}$$

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = ?$

$$\{01010111\} \cdot \{10000011\}$$

$$= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\})$$

$$= \{00111000\}$$

$$+ \{10101110\}$$

$$+ \{01010111\}$$

$$01010111 \oplus 10000000$$

$$\text{Cioè } (x^6+x^4+x^2+x+1)x^7$$

$$= (x^7+x^5+x^3+x^2+x) x^6$$

$$= (x^6+x^2+x+1) x^5$$

$$= (x^7+x^3+x^2+x) x^4$$

$$= (x^8+x^4+x^3+x^2) x^3$$

$$= (x^2+x+1) x^3$$

$$= (x^5+x^4+x^3)$$

Quindi, il prodotto è 00111000

Calcolo della moltiplicazione

Calcoliamo $\{01010111\} \cdot \{10000011\} = \{11000001\}$

$$\begin{aligned}\{01010111\} \cdot \{10000011\} \\&= \{01010111\} \cdot (\{10000000\} + \{00000010\} + \{00000001\}) \\&= \quad \{00111000\} \\&\quad + \{10101110\} \\&\quad + \{01010111\} \\&= \{11000001\}\end{aligned}$$

Calcolo della moltiplicazione

- Abbiamo usato solo
 - \oplus
 - Moltiplicazioni per x , cioè shift a sx con eventualmente una somma di $x^8+x^4+x^3+x+1$
- Non abbiamo usato divisioni tra polinomi!
- Esercizio: Descrivere l'algoritmo per la moltiplicazione tra polinomi

Preliminari matematici

- Operazioni sui byte
 - Addizione e moltiplicazione
 - Efficienti computazionalmente
 - Elegante struttura matematica
- Operazioni sulle word di 32 bit
 - Addizione e moltiplicazione
 - Efficienti computazionalmente
 - Elegante struttura matematica

Rappresentazione word

Word $[a_0, a_1, a_2, a_3] \rightarrow$ polinomio $a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$

Word di 32 bit, cioè, 4 byte

- I coefficienti sono byte
- Elementi di $GF(2^8)$

Esempio:

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$$

Polinomi con coefficienti in $GF(2^8)$

Word $[a_0, a_1, a_2, a_3] \rightarrow$ polinomio $a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$

Addizione

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$a(x) + b(x) = a_3x^3 + a_2x^2 + a_1x + a_0 + b_3x^3 + b_2x^2 + b_1x + b_0$$

Polinomi con coefficienti in $GF(2^8)$

Word $[a_0, a_1, a_2, a_3] \rightarrow$ polinomio $a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

Addizione

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$\begin{aligned} a(x) + b(x) &= a_3x^3 + a_2x^2 + a_1x + a_0 + b_3x^3 + b_2x^2 + b_1x + b_0 \\ &= (a_3 \oplus b_3)x^3 + (a_2 \oplus b_2)x^2 + (a_1 \oplus b_1)x + (a_0 \oplus b_0) \end{aligned}$$

Somma in $GF(2^8)$

Somma in $GF(2^8)$

Somma in $GF(2^8)$

Somma in $GF(2^8)$

Polinomi con coefficienti in $GF(2^8)$

Word $[a_0, a_1, a_2, a_3] \rightarrow$ polinomio $a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$

Addizione

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$\begin{aligned} a(x) + b(x) &= a_3x^3 + a_2x^2 + a_1x + a_0 + b_3x^3 + b_2x^2 + b_1x + b_0 \\ &= (a_3 \oplus b_3)x^3 + (a_2 \oplus b_2)x^2 + (a_1 \oplus b_1)x + (a_0 \oplus b_0) \end{aligned}$$

Algoritmo per somma di 2 word:
xor delle intere stringhe binarie



Polinomi con coefficienti in GF(2⁸)

Moltiplicazione

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$c(x) = c_6x^6 + c_5x^5 + c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0$$

$$c_0 = a_0 \bullet b_0$$

$$c_4 = a_3 \bullet b_1 \oplus a_2 \bullet b_2 \oplus a_1 \bullet b_3$$

$$c_1 = a_1 \bullet b_0 \oplus a_0 \bullet b_1$$

$$c_5 = a_3 \bullet b_2 \oplus a_2 \bullet b_3$$

$$c_2 = a_2 \bullet b_0 \oplus a_1 \bullet b_1 \oplus a_0 \bullet b_2$$

$$c_6 = a_3 \bullet b_3$$

$$c_3 = a_3 \bullet b_0 \oplus a_2 \bullet b_1 \oplus a_1 \bullet b_2 \oplus a_0 \bullet b_3$$

Polinomi con coefficienti in GF(2⁸)

Moltiplicazione

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$c(x) = c_6x^6 + c_5x^5 + c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0$$

$$c_0 = a_0 \bullet b_0$$

$$c_4 = a_3 \bullet b_1 \oplus a_2 \bullet b_2 \oplus a_1 \bullet b_3$$

$$c_1 = a_1 \bullet b_0 \oplus a_0 \bullet b_1$$

$$c_5 = a_3 \bullet b_2 \oplus a_2 \bullet b_3$$

$$c_2 = a_2 \bullet b_0 \oplus a_1 \bullet b_1 \oplus a_0 \bullet b_2$$

$$c_6 = a_3 \bullet b_3$$

$$c_3 = a_3 \bullet b_0 \oplus a_2 \bullet b_1 \oplus a_1 \bullet b_2 \oplus a_0 \bullet b_3$$

- Non va in una word, occorrono più bit



Polinomi con coefficienti in GF(2⁸)

Moltiplicazione

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$c(x) = c_6x^6 + c_5x^5 + c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0$$

$$c_0 = a_0 \bullet b_0$$

$$c_4 = a_3 \bullet b_1 \oplus a_2 \bullet b_2 \oplus a_1 \bullet b_3$$

$$c_1 = a_1 \bullet b_0 \oplus a_0 \bullet b_1$$

$$c_5 = a_3 \bullet b_2 \oplus a_2 \bullet b_3$$

$$c_2 = a_2 \bullet b_0 \oplus a_1 \bullet b_1 \oplus a_0 \bullet b_2$$

$$c_6 = a_3 \bullet b_3$$

$$c_3 = a_3 \bullet b_0 \oplus a_2 \bullet b_1 \oplus a_1 \bullet b_2 \oplus a_0 \bullet b_3$$

- Non va in una word, occorrono più bit
- Riduzione modulo il polinomio x^4+1



Potenze di x mod x^4+1

$$x^4 \equiv -1 \equiv 1 \pmod{x^4+1}$$

$$x^5 \equiv -x \equiv x \pmod{x^4+1}$$

$$x^6 \equiv -x^2 \equiv x^2 \pmod{x^4+1}$$

...

$$x^i \equiv x^{i \bmod 4} \pmod{x^4+1} \quad \text{in generale}$$

Questa proprietà facilita il calcolo della moltiplicazione mod x^4+1
(evitando la divisione tra polinomi!)

Polinomi con coefficienti in GF(2⁸)

Moltiplicazione

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$c(x) = c_6x^6 + c_5x^5 + c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0$$

$$c_0 = a_0 \bullet b_0$$

$$c_4 = a_3 \bullet b_1 \oplus a_2 \bullet b_2 \oplus a_1 \bullet b_3$$

$$c_1 = a_1 \bullet b_0 \oplus a_0 \bullet b_1$$

$$c_5 = a_3 \bullet b_2 \oplus a_2 \bullet b_3$$

$$c_2 = a_2 \bullet b_0 \oplus a_1 \bullet b_1 \oplus a_0 \bullet b_2$$

$$c_6 = a_3 \bullet b_3$$

$$c_3 = a_3 \bullet b_0 \oplus a_2 \bullet b_1 \oplus a_1 \bullet b_2 \oplus a_0 \bullet b_3$$

Sostituiamo $\begin{cases} x^4 \equiv 1 \text{ mod } (x^4+1) \\ x^5 \equiv x \text{ mod } (x^4+1) \\ x^6 \equiv x^2 \text{ mod } (x^4+1) \end{cases}$

Polinomi con coefficienti in GF(2⁸)

Moltiplicazione

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$c(x) = c_6x^6 + c_5x^5 + c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0$$

$$c_0 = a_0 \bullet b_0$$

$$c_4 = a_3 \bullet b_1 \oplus a_2 \bullet b_2 \oplus a_1 \bullet b_3$$

$$c_1 = a_1 \bullet b_0 \oplus a_0 \bullet b_1$$

$$c_5 = a_3 \bullet b_2 \oplus a_2 \bullet b_3$$

$$c_2 = a_2 \bullet b_0 \oplus a_1 \bullet b_1 \oplus a_0 \bullet b_2$$

$$c_6 = a_3 \bullet b_3$$

$$c_3 = a_3 \bullet b_0 \oplus a_2 \bullet b_1 \oplus a_1 \bullet b_2 \oplus a_0 \bullet b_3$$

Sostituiamo $\begin{cases} x^4 \equiv 1 \text{ mod } (x^4+1) \\ x^5 \equiv x \text{ mod } (x^4+1) \\ x^6 \equiv x^2 \text{ mod } (x^4+1) \end{cases}$

Otteniamo

$$c(x) = c_3x^3 + (c_6 \oplus c_2)x^2 + (c_5 \oplus c_1)x + (c_4 \oplus c_0)$$

Polinomi con coefficienti in $GF(2^8)$

Prodotto mod x^4+1 cioè $d(x) = d_3x^3 + d_2x^2 + d_1x + d_0$

$$\left\{ \begin{array}{l} d_0 = (a_0 \bullet b_0) \oplus (a_3 \bullet b_1) \oplus (a_2 \bullet b_2) \oplus (a_1 \bullet b_3) \\ \qquad \qquad \qquad = c_4 \oplus c_0 \\ d_1 = (a_1 \bullet b_0) \oplus (a_0 \bullet b_1) \oplus (a_3 \bullet b_2) \oplus (a_2 \bullet b_3) \\ \qquad \qquad \qquad = c_5 \oplus c_1 \\ d_2 = (a_2 \bullet b_0) \oplus (a_1 \bullet b_1) \oplus (a_0 \bullet b_2) \oplus (a_3 \bullet b_3) \\ \qquad \qquad \qquad = c_6 \oplus c_2 \\ d_3 = (a_3 \bullet b_0) \oplus (a_2 \bullet b_1) \oplus (a_1 \bullet b_2) \oplus (a_0 \bullet b_3) \\ \qquad \qquad \qquad = c_3 \end{array} \right.$$

Polinomi con coefficienti in $GF(2^8)$

Prodotto mod x^4+1 cioè $d(x) = d_3x^3 + d_2x^2 + d_1x + d_0$

$$\begin{cases} d_0 = (a_0 \bullet b_0) \oplus (a_3 \bullet b_1) \oplus (a_2 \bullet b_2) \oplus (a_1 \bullet b_3) \\ d_1 = (a_1 \bullet b_0) \oplus (a_0 \bullet b_1) \oplus (a_3 \bullet b_2) \oplus (a_2 \bullet b_3) \\ d_2 = (a_2 \bullet b_0) \oplus (a_1 \bullet b_1) \oplus (a_0 \bullet b_2) \oplus (a_3 \bullet b_3) \\ d_3 = (a_3 \bullet b_0) \oplus (a_2 \bullet b_1) \oplus (a_1 \bullet b_2) \oplus (a_0 \bullet b_3) \end{cases} \quad \begin{aligned} &= c_4 \oplus c_0 \\ &= c_5 \oplus c_1 \\ &= c_6 \oplus c_2 \\ &= c_3 \end{aligned}$$

In forma matriciale

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} a_0 & a_3 & a_2 & a_1 \\ a_1 & a_0 & a_3 & a_2 \\ a_2 & a_1 & a_0 & a_3 \\ a_3 & a_2 & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Polinomi con coefficienti in $GF(2^8)$

- x^4+1 non è irriducibile su $GF(2^8)$
- Non tutti i polinomi hanno inverso mod x^4+1
- AES usa $a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$
 $a^{-1}(x) = \{0b\}x^3 + \{0d\}x^2 + \{09\}x + \{0e\}$

Moltiplicazione in AES

Moltiplicazione mod x^4+1 di un polinomio

$$p(x) = p_3x^3 + p_2x^2 + p_1x + p_0$$

per il polinomio

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$$

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

per il polinomio

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

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

Pseudocodice per l'AES

Cipher (byte in[4 · Nb], byte out[4 · Nb], word w[Nb · (Nr + 1)])

byte state[4, Nb]

state \leftarrow in

AddRoundKey (state, w)

for round = 1 to Nr - 1

SubBytes (state)

ShiftRows (state)

MixColumns (state)

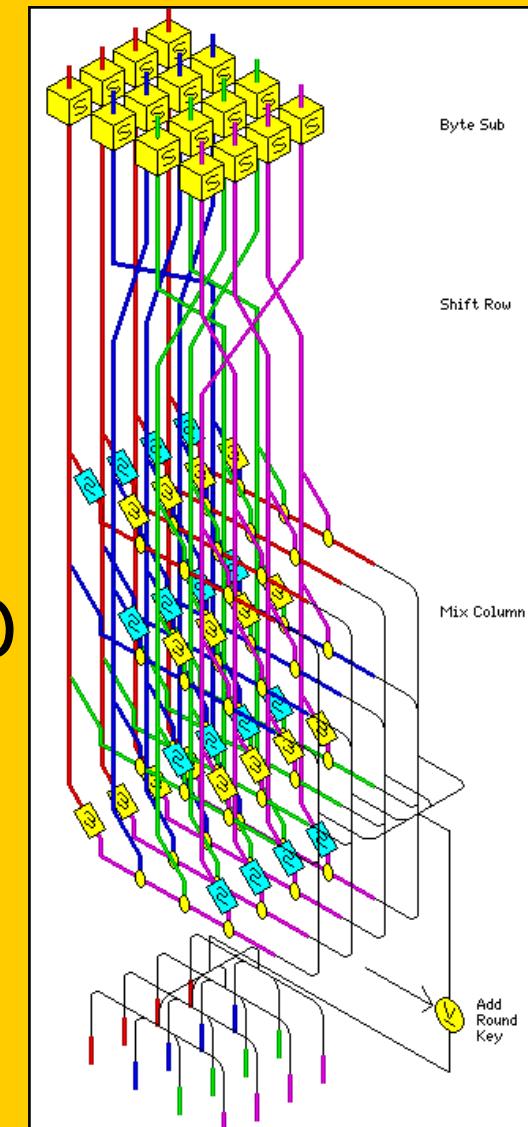
AddRoundKey (state, w + round · Nb)

SubBytes (state)

ShiftRows (state)

AddRoundKey (state, w + Nr · Nb)

out \leftarrow state

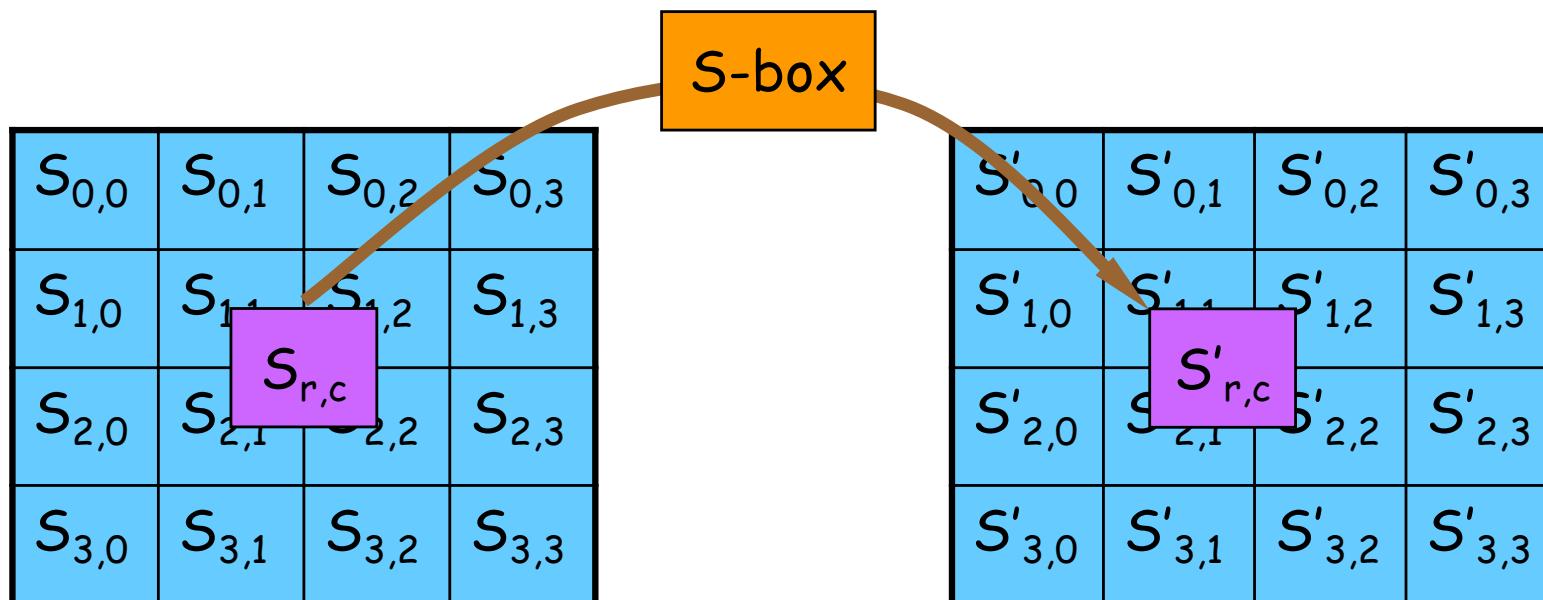


SubBytes Transformation

Byte trasformati mediante una S-box non lineare ma invertibile

$$S'_{r,c} \leftarrow S\text{-box}(S_{r,c})$$

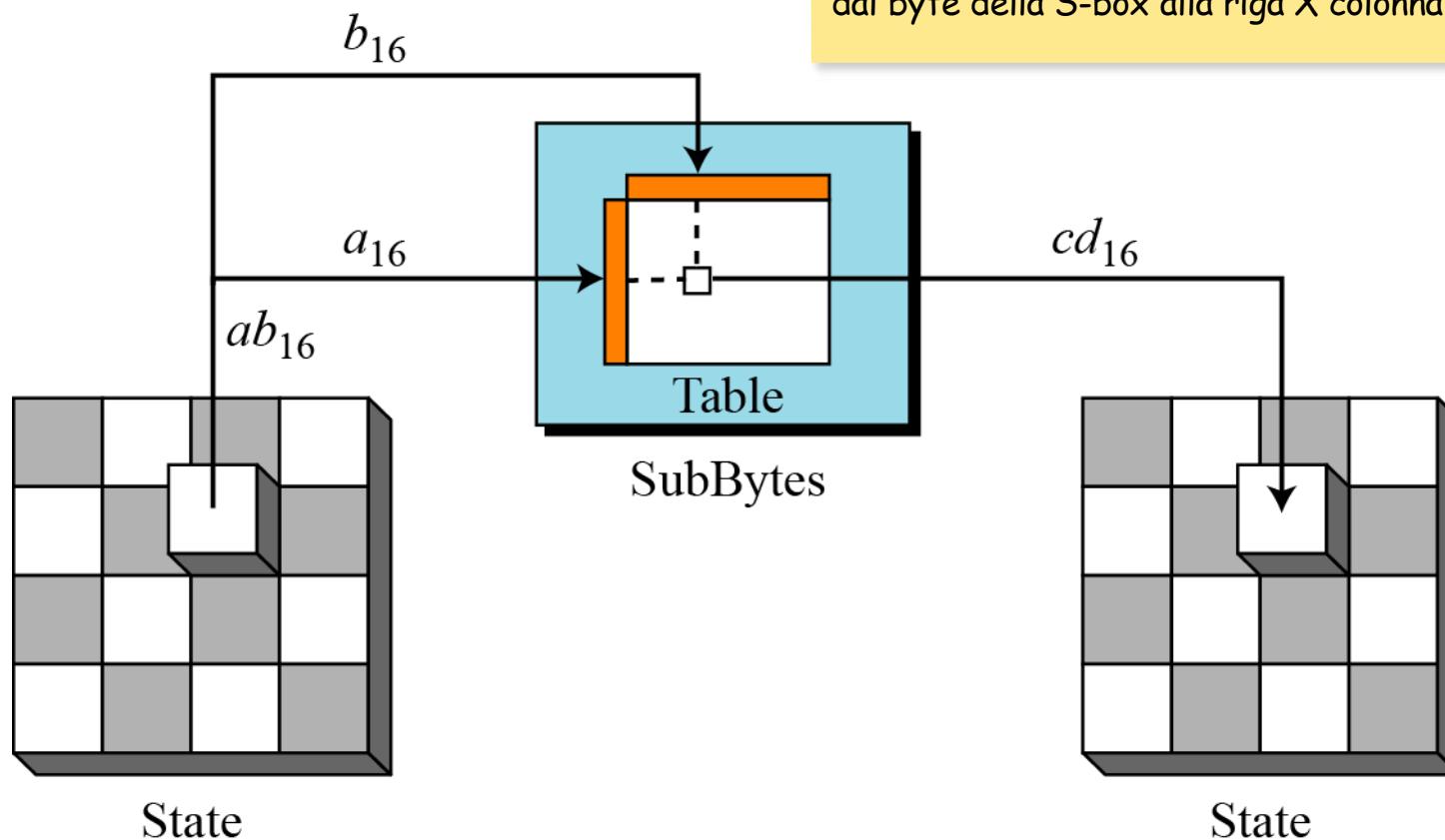
$$0 \leq r \leq 3 \quad 0 \leq c \leq 3$$



SubBytes Transformation

S-box: matrice 16×16
Permutazione di tutti i 256 valori ad 8 bit

Ogni singolo byte della matrice di stato
➤ Sostituito utilizzando la S-box
➤ Byte matrice di stato: XY in esadecimale viene sostituito dal byte della S-box alla riga X colonna Y



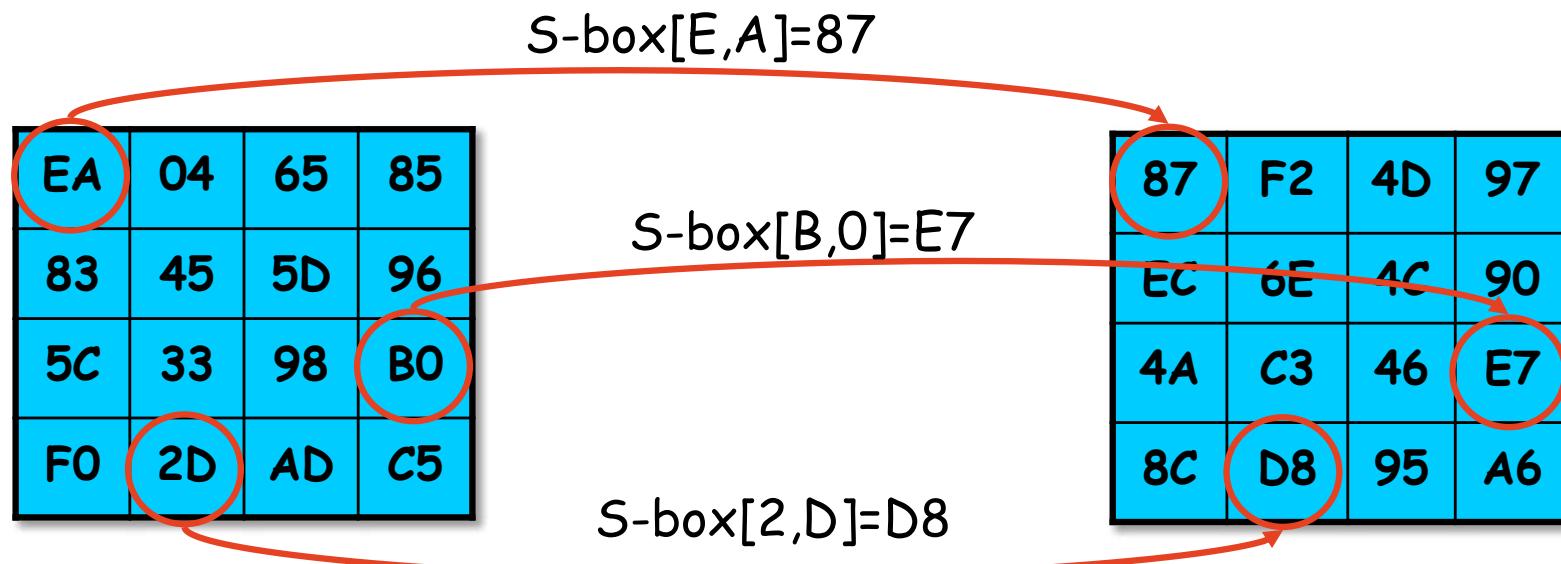
Esempio: 3C viene sostituito da S-box[3,C]

S-box

	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

SubBytes Transformation

Esempio di utilizzo



Costruzione S-box

- Inizializzare la S-box con i valori dei byte in ordine ascendente riga per riga
 - Prima riga: {00}, {01}, ..., {0F},
 - Seconda riga: {10}, {11}, ..., {1F},
 - ...

Costruzione S-box

- Inizializzare la S-box con i valori dei byte in ordine ascendente riga per riga
 - Prima riga: {00}, {01}, ..., {0F},
 - Seconda riga: {10}, {11}, ..., {1F},
 - ...
- Sostituire ciascun byte con il suo inverso moltiplicativo in $GF(2^8)$
 - {00} resta {00}

Costruzione S-box

- Inizializzare la S-box con i valori dei byte in ordine ascendente riga per riga
 - Prima riga: {00}, {01}, ..., {0F},
 - Seconda riga: {10}, {11}, ..., {1F},
 - ...
- Sostituire ciascun byte con il suo inverso moltiplicativo in $GF(2^8)$
 - {00} resta {00}
- Applicare una trasformazione affine in $GF(2^8)$

$$b'_i \leftarrow b_i \oplus b_{(i+4)\bmod 8} \oplus b_{(i+5)\bmod 8} \oplus b_{(i+6)\bmod 8} \oplus b_{(i+7)\bmod 8} \oplus \{01100011\}_i$$

i-esimo bit
del byte {63}

Trasformazione affine

$(b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0)$ diventa $(b'_7, b'_6, b'_5, b'_4, b'_3, b'_2, b'_1, b'_0)$

- $b'_0 = b_0 \oplus b_4 \oplus b_5 \oplus b_6 \oplus b_7 \oplus 0$
- $b'_1 = b_1 \oplus b_5 \oplus b_6 \oplus b_7 \oplus b_0 \oplus 1$
- $b'_2 = b_2 \oplus b_6 \oplus b_7 \oplus b_0 \oplus b_1 \oplus 1$
- $b'_3 = b_3 \oplus b_7 \oplus b_0 \oplus b_1 \oplus b_2 \oplus 0$
- $b'_4 = b_4 \oplus b_0 \oplus b_1 \oplus b_2 \oplus b_3 \oplus 0$
- $b'_5 = b_5 \oplus b_1 \oplus b_2 \oplus b_3 \oplus b_1 \oplus 0$
- $b'_6 = b_6 \oplus b_2 \oplus b_3 \oplus b_4 \oplus b_5 \oplus 1$
- $b'_7 = b_7 \oplus b_3 \oplus b_4 \oplus b_5 \oplus b_6 \oplus 1$

Trasformazione affine: forma matriciale

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} \xleftarrow{\quad} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Proprietà S-box

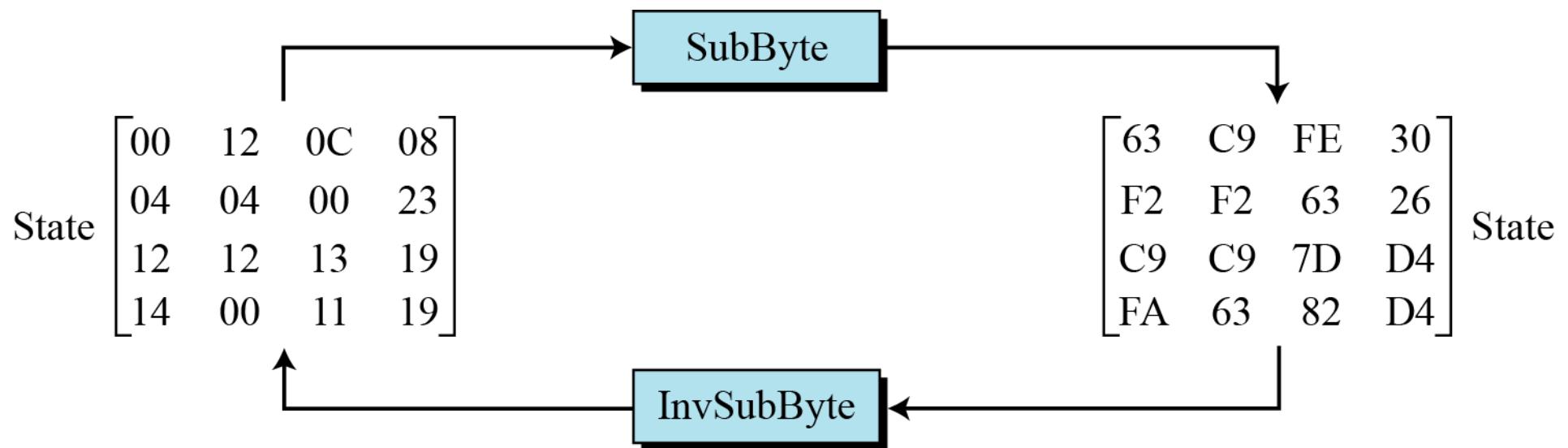
- L'output non è una funzione lineare dell'input
- Non ha punti fissi diretti né opposti
 - $S\text{-box}(a) \neq a$ e $S\text{-box}(\bar{a}) \neq \bar{a}$
- È invertibile
 - $\text{Inverse_S-box}(S\text{-box}(a)) = a$
- Non è self-invertibile
 - $S\text{-box}(a) \neq \text{Inverse_S-box}(a)$
- Progettata per resistere ad attacchi crittoanalitici noti

Inverse_S-box

	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

SubBytes e InvSubBytes Transformation

SubByte e InvSubByte sono una
l'inversa dell'altra: esempio



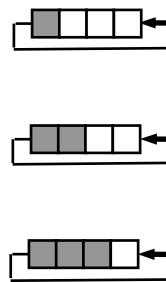
ShiftRows Transformation

Righe shiftate di posizioni differenti (shift ciclico a sx)

$$S'_{r,c} \leftarrow S_{r,(c+\text{shift}(r,\text{Nb}))\bmod\text{Nb}}$$

$$\begin{array}{ll} 0 \leq r \leq 3 & 0 \leq c \leq 3 \\ \text{shift}(1,4) = 1 & \text{shift}(2,4) = 2 & \text{shift}(3,4) = 3 \end{array}$$

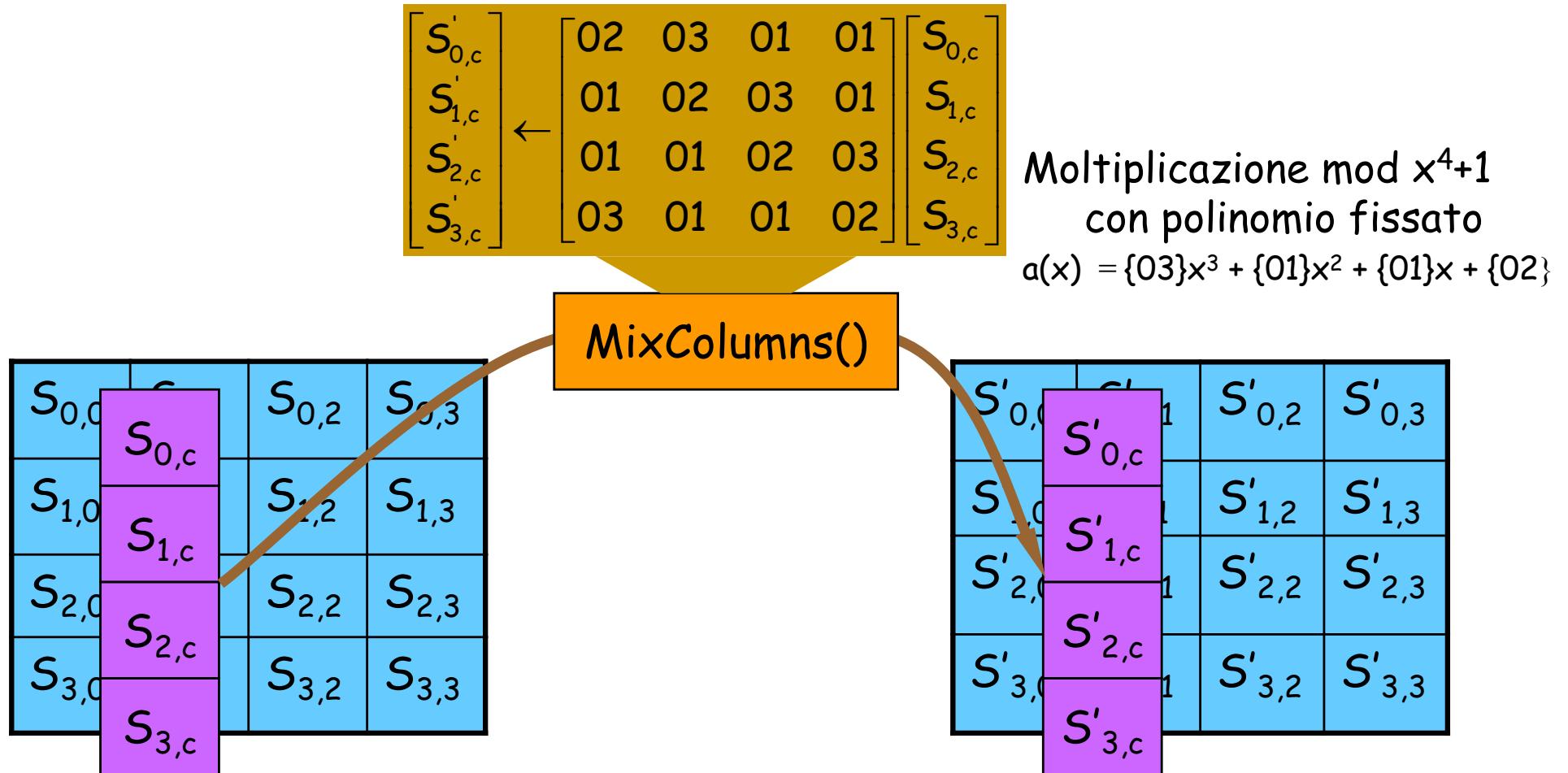
$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$



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

I 4 byte di una colonna sono spostati su colonne differenti

MixColumns Transformation

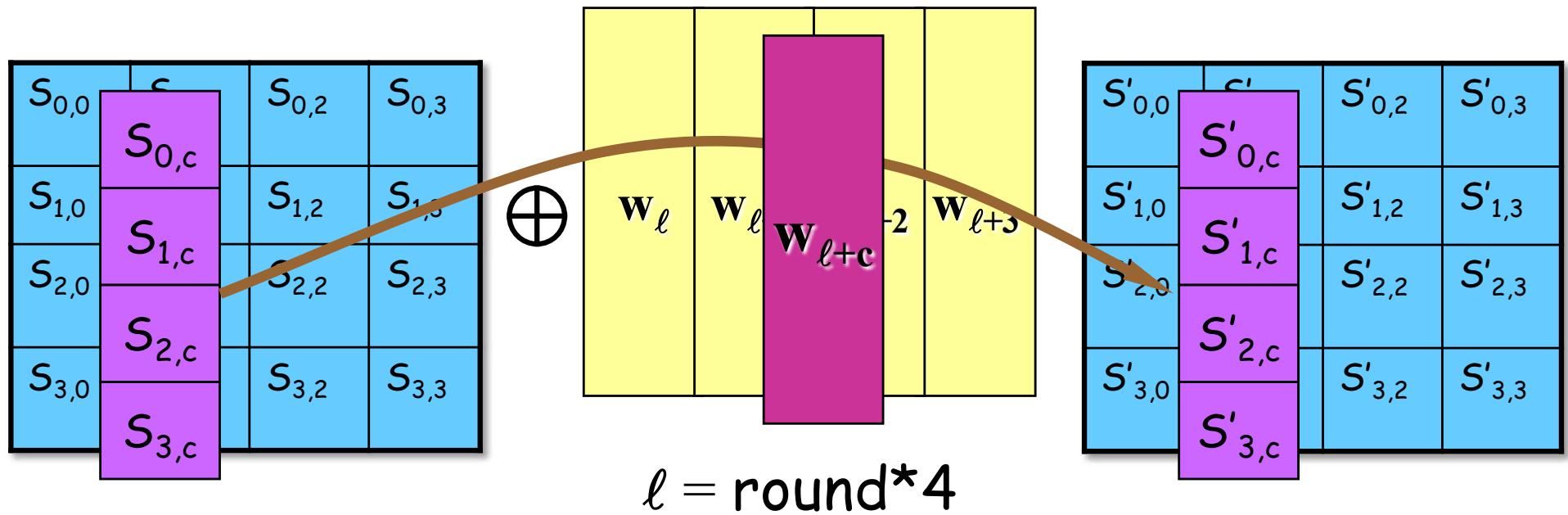


Byte nelle colonne combinati linearmente

AddRoundKey Transformation

round key (word)

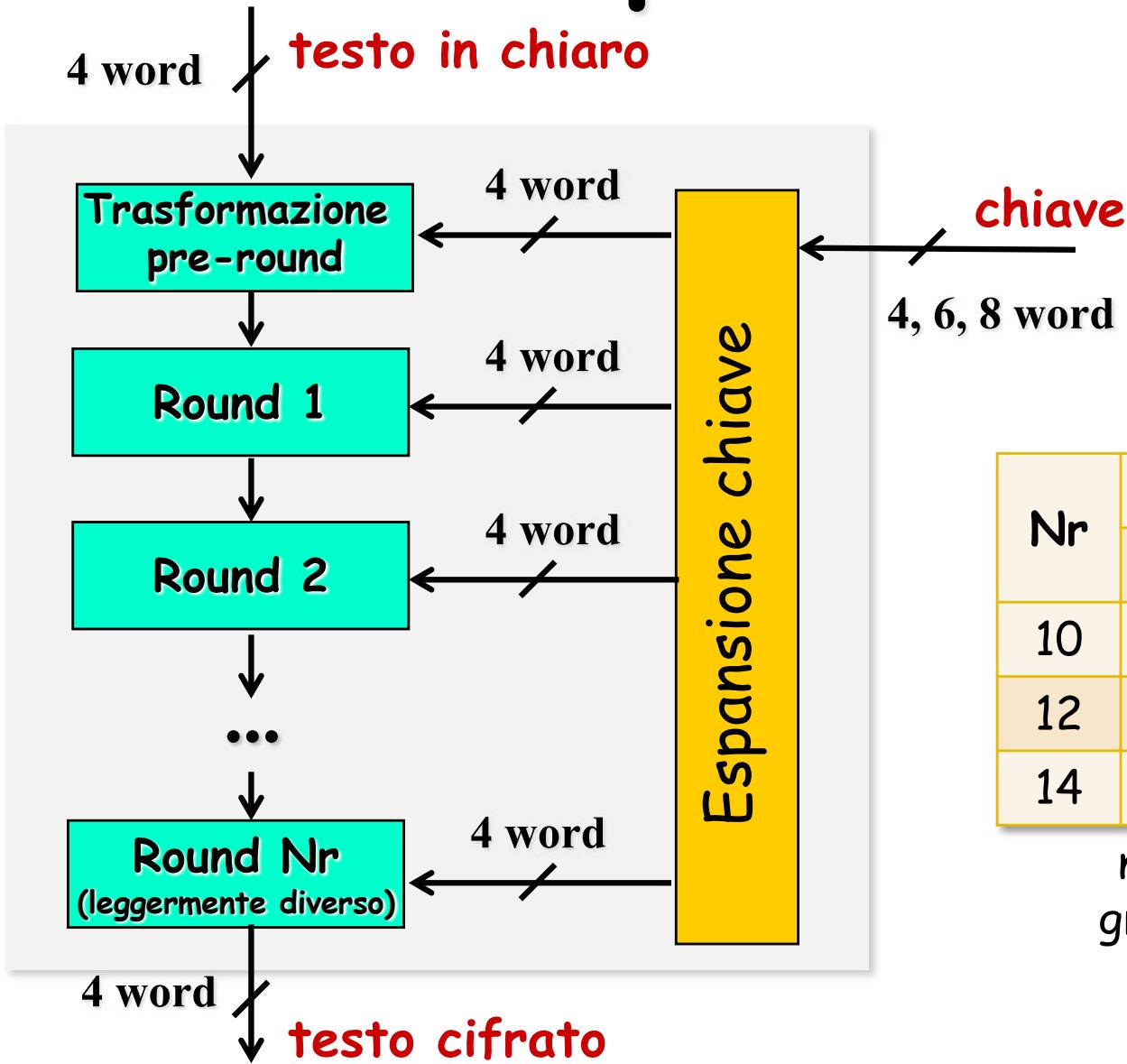
$$[S'_{0,c}, S'_{1,c}, S'_{2,c}, S'_{3,c}] \leftarrow [S_{0,c}, S_{1,c}, S_{2,c}, S_{3,c}] \oplus [w_{\text{round}*Nb+c}] \quad 0 \leq c < 4$$



Espansione chiave

- A partire dalla chiave (matrice **key** di $4 \times Nk$ byte), genera le chiavi schedulate (array **w** di $4 \times (Nr+1)$ word)
 - 4 word per AddRoundKey iniziale
 - 4 word per AddRoundKey in ciascun round
- Se $Nk=4$ (chiave a 128 bit), sono prodotte 44 word
 - 4 per AddRoundKey iniziale
 - 4 per ognuno dei 10 round

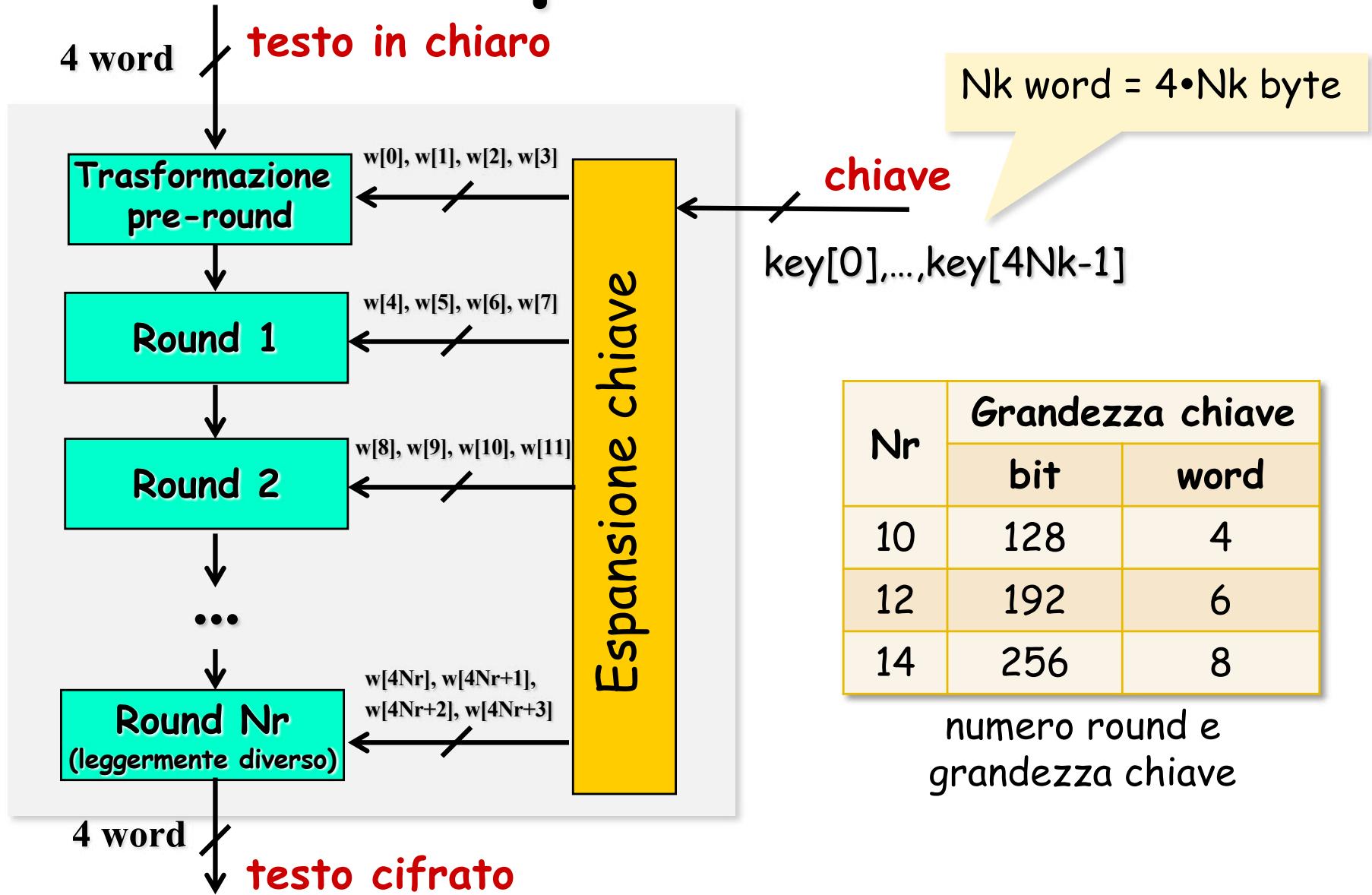
Espansione chiave



Nr	Grandezza chiave	
	bit	word
10	128	4
12	192	6
14	256	8

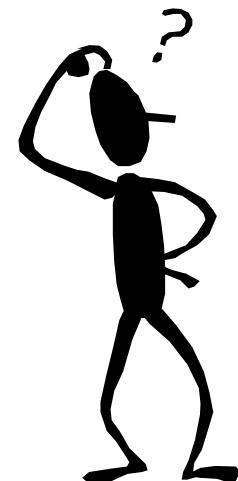
numero round e
grandezza chiave

Espansione chiave

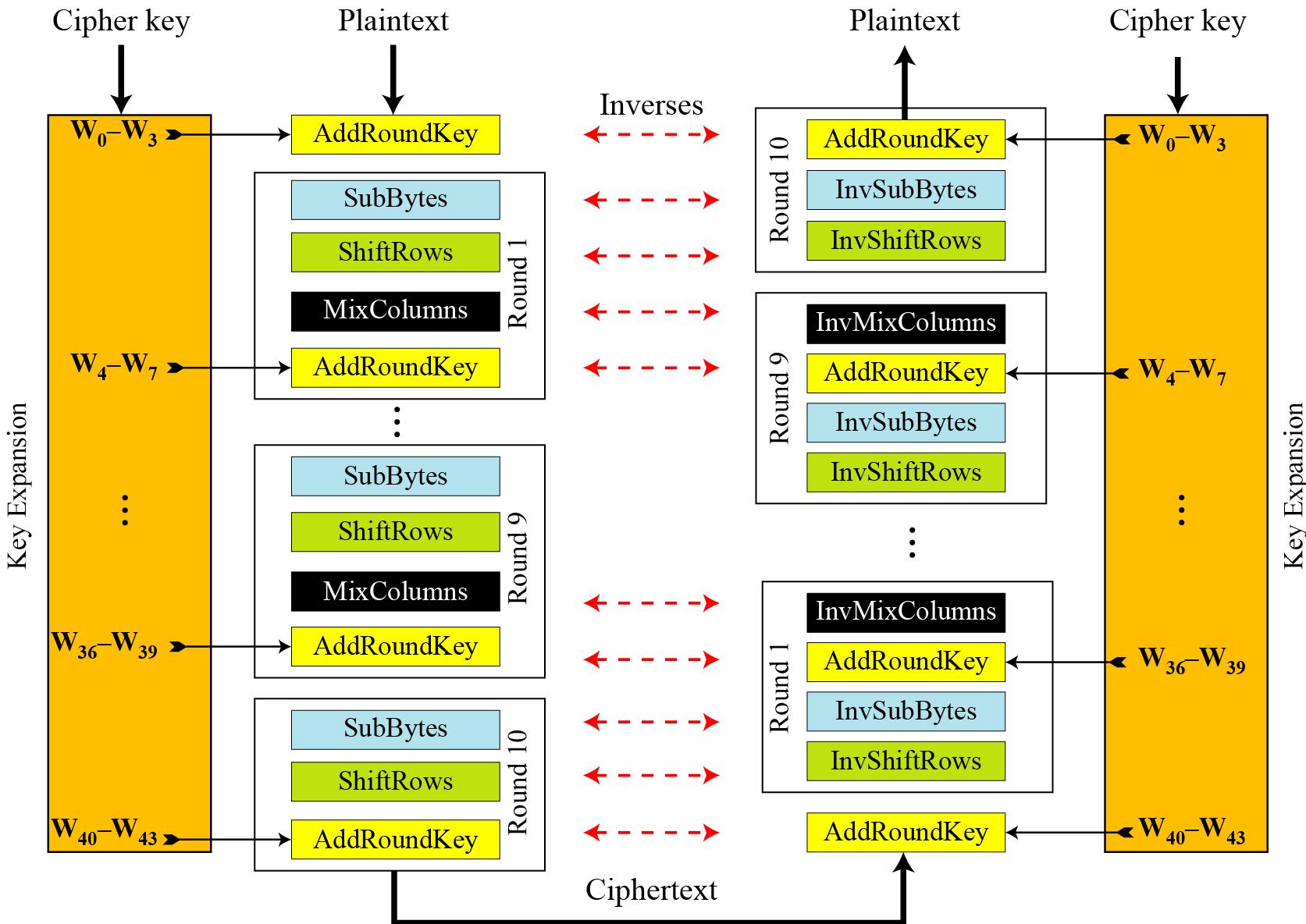


Decifratura AES

- L'algoritmo di decifratura non è lo stesso della cifratura
 - Usa una diversa sequenza di trasformazioni
 - Usa le trasformazioni inverse
 - Svantaggio: necessaria una doppia implementazione
- Esiste anche un algoritmo di decifratura che ha la stessa struttura di quello di cifratura
 - Stessa sequenza di trasformazioni
 - Usa le trasformazioni inverse
 - Necessita di un cambiamento nella schedulazione della chiave



Cifratura e Decifratura



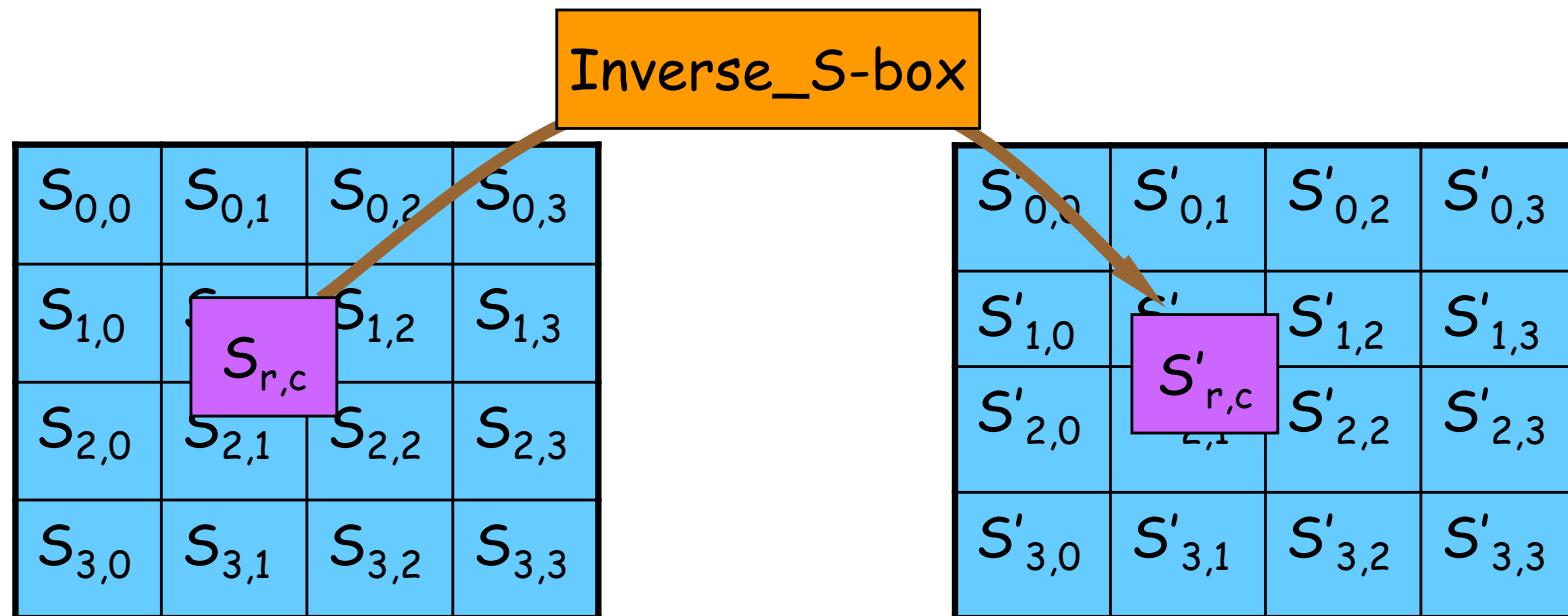
Decifratura AES

```
InvCipher(byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])  
byte state[4,Nb]  
state ← in  
AddRoundKey(state, w + Nr * Nb)  
for round = Nr - 1 step -1 to 1  
    InvShiftRows(state)  
    InvSubBytes(state)  
    AddRoundKey(state, w + round * Nb)  
    InvMixColumns(state)  
InvShiftRows(state)  
InvSubBytes(state)  
AddRoundKey(state, w)  
out ← state
```

InvSubBytes Transformation

$S'_{r,c} \leftarrow \text{Inverse_S-box}(S_{r,c})$

$$0 \leq r < 4 \quad 0 \leq c < Nb$$

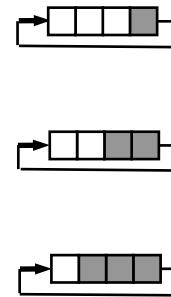


InvShiftRows Transformation

$S'_{r,(c+\text{shift}(r,\text{Nb}))\text{modNb}} \leftarrow S_{r,c}$

$0 \leq r \leq 3 \quad 0 \leq c \leq 3$
 $\text{shift}(1,4)=1 \quad \text{shift}(2,4)=2 \quad \text{shift}(3,4)=3$

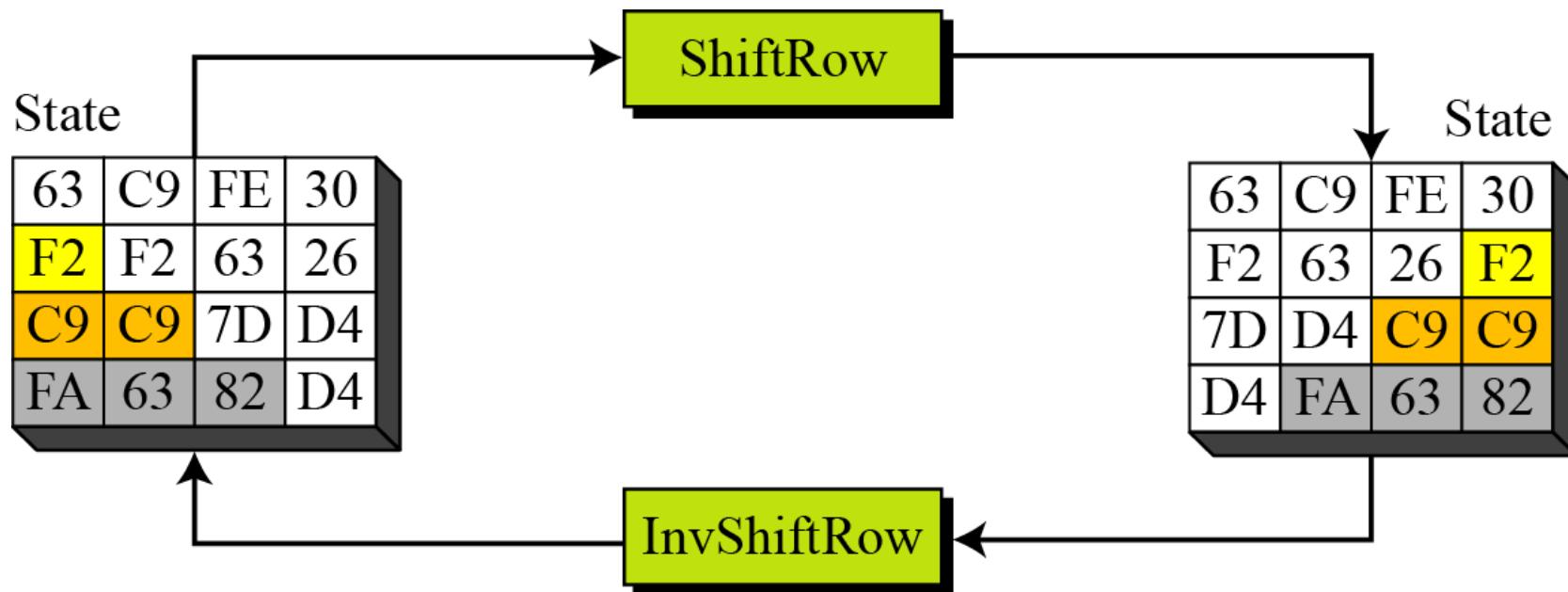
$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$



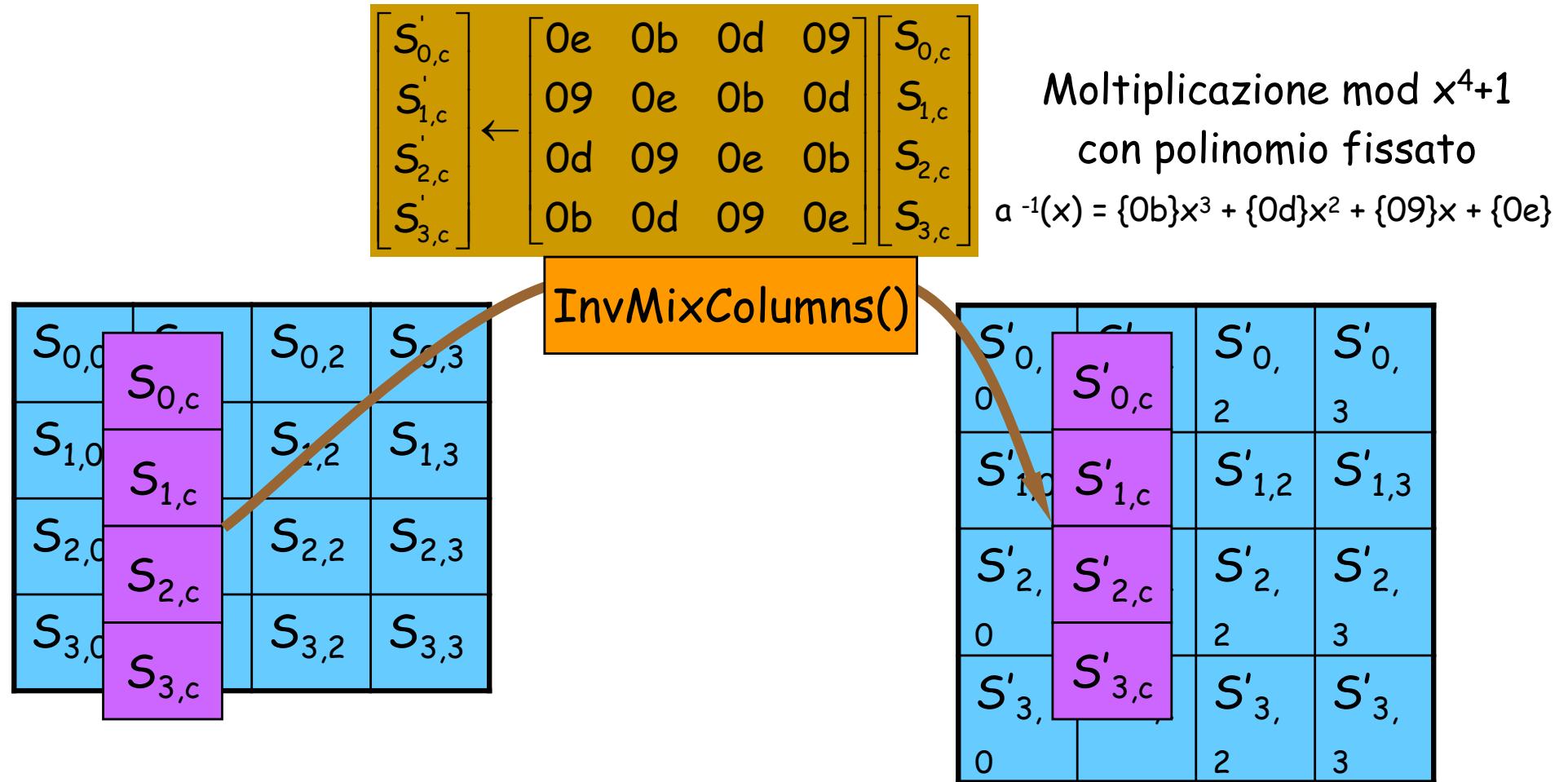
$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,1}$	$S_{1,2}$	$S_{1,3}$	$S_{1,0}$
$S_{2,2}$	$S_{2,3}$	$S_{2,0}$	$S_{2,1}$
$S_{3,3}$	$S_{3,0}$	$S_{3,1}$	$S_{3,2}$

ShiftRow e InvShiftRow Transformation

ShiftRow e InvShiftRow sono una
l'inversa dell'altra: esempio

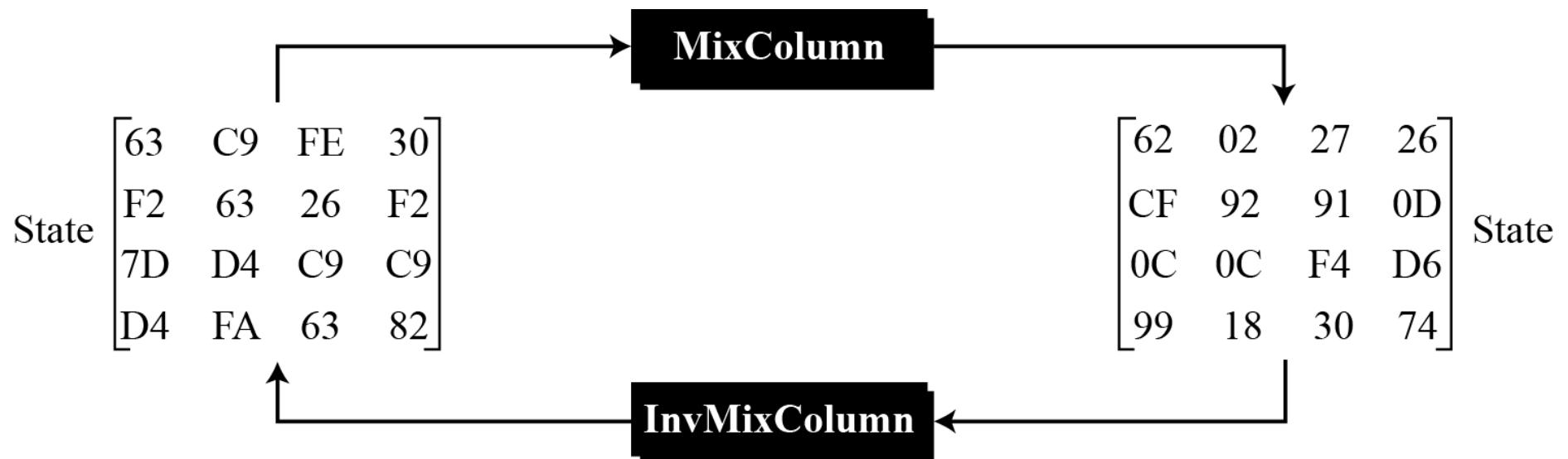


InvMixColumns Transformation



MixColumn e InvMixColumn Transformation

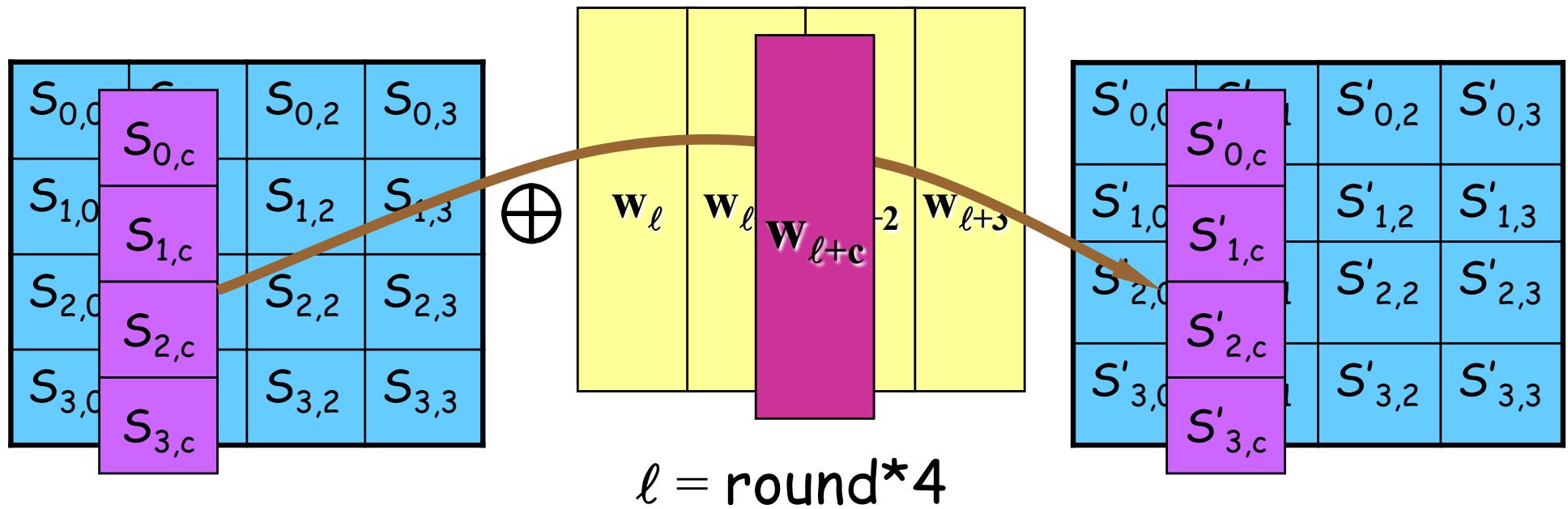
MixColumn e InvMixColumn sono
una l'inversa dell'altra: esempio



AddRoundKey Transformation

È l'inversa di se stessa!

$$[S'_{0,c}, S'_{1,c}, S'_{2,c}, S'_{3,c}] \leftarrow [S_{0,c}, S_{1,c}, S_{2,c}, S_{3,c}] \oplus [w_{\text{round}*Nb+c}] \quad 0 \leq c < 4$$



Avalanche effect

Un piccolo cambiamento del testo in chiaro oppure della chiave produce un grande cambiamento del testo cifrato

Esempio:

Cipher Key:	24	75	A2	B3	34	75	56	88	31	E2	12	00	13	AA	54	87
--------------------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Plaintext 1:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
--------------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Plaintext 2:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	<u>01</u>
--------------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----------

Ciphertext 1:	63	2C	D4	5E	5D	56	ED	B5	62	04	01	A0	AA	9C	2D	8D
---------------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Ciphertext 2:	26	F3	9B	BC	A1	9C	0F	B7	C7	2E	7E	30	63	92	73	13
---------------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

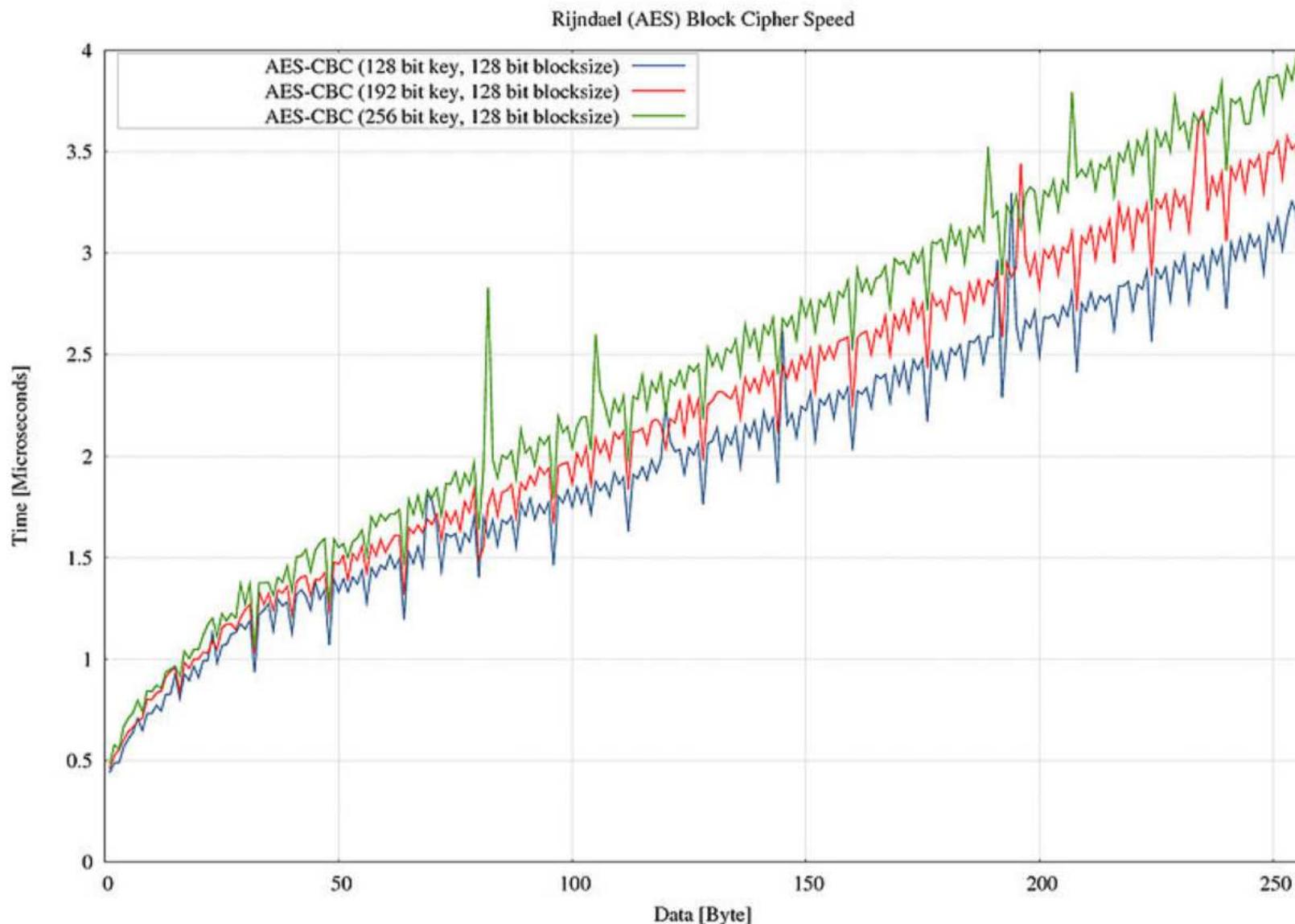
Security lifetime

Table 4: Recommended algorithms and minimum key sizes

Algorithm security lifetimes	Symmetric key algorithms (Encryption & MAC)
Through 2010 (min. of 80 bits of strength)	2TDEA ²³ 3TDEA AES-128 AES-192 AES-256
Through 2030 (min. of 112 bits of strength)	3TDEA AES-128 AES-192 AES-256
Beyond 2030 (min. of 128 bits of strength)	AES-128 AES-192 AES-256

NIST SP 800-57 ,
"Recommendation for Key
Management - Part 1: General",
March 2007

AES speed



Benefici di AES

NIST

Search NIST  NIST MENU

NEWS

NIST's Encryption Standard Has Minimum \$250 Billion Economic Benefit, According to New Study

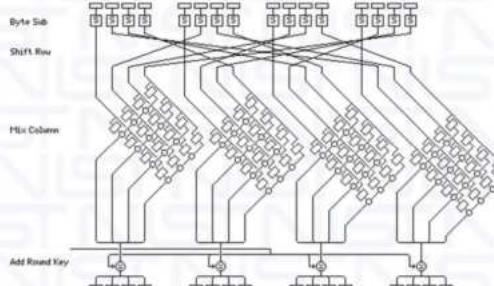
An international competition led to the voluntary standard that today protects millions of IT systems.

September 19, 2018

NIST GCR 18-017

The Economic Impacts of the Advanced Encryption Standard, 1996 - 2017



David P. Leech
Stacey Ferris, CPA
John T. Scott, Ph.D.
September 2018

NIST
National Institute of
Standards and Technology
U.S. Department of Commerce

This publication is available free of charge from: <https://doi.org/10.6028/NIST.GCR.18-017>

<https://nvlpubs.nist.gov/nistpubs/gcr/2018/NIST.GCR.18-017.pdf>

Leggi e regolamenti più importanti negli US 1996 - 2004, settore privato

Figure 2-4. Laws and Regulations Requiring Use of Encryption by the Private Sector

1996 - Health Insurance Portability & Accountability Act (HIPAA)

- Requires all industries to use encryption to protect health data.

1997 - FDA Title 21 CFR Part 11

- Drug-makers and all FDA regulated industries must use encryption to protect data at rest and in transit.

1999 - Gramm-Leach-Bliley Act

- Banking and financial services required to use encryption to protect personal information.

2002 - e-Government Act (Federal Information Security Management Act, FISMA)

- All Federal Agencies required to implement cybersecurity measures, including use of FIPS-140 approved encryption.

2002 - Sarbanes-Oxley Act

- All industries required to use encryption technology to protect sensitive financial data storage and transmission.

2004 - Payment Card Industry Data Security Standard (DSS); amended 2016

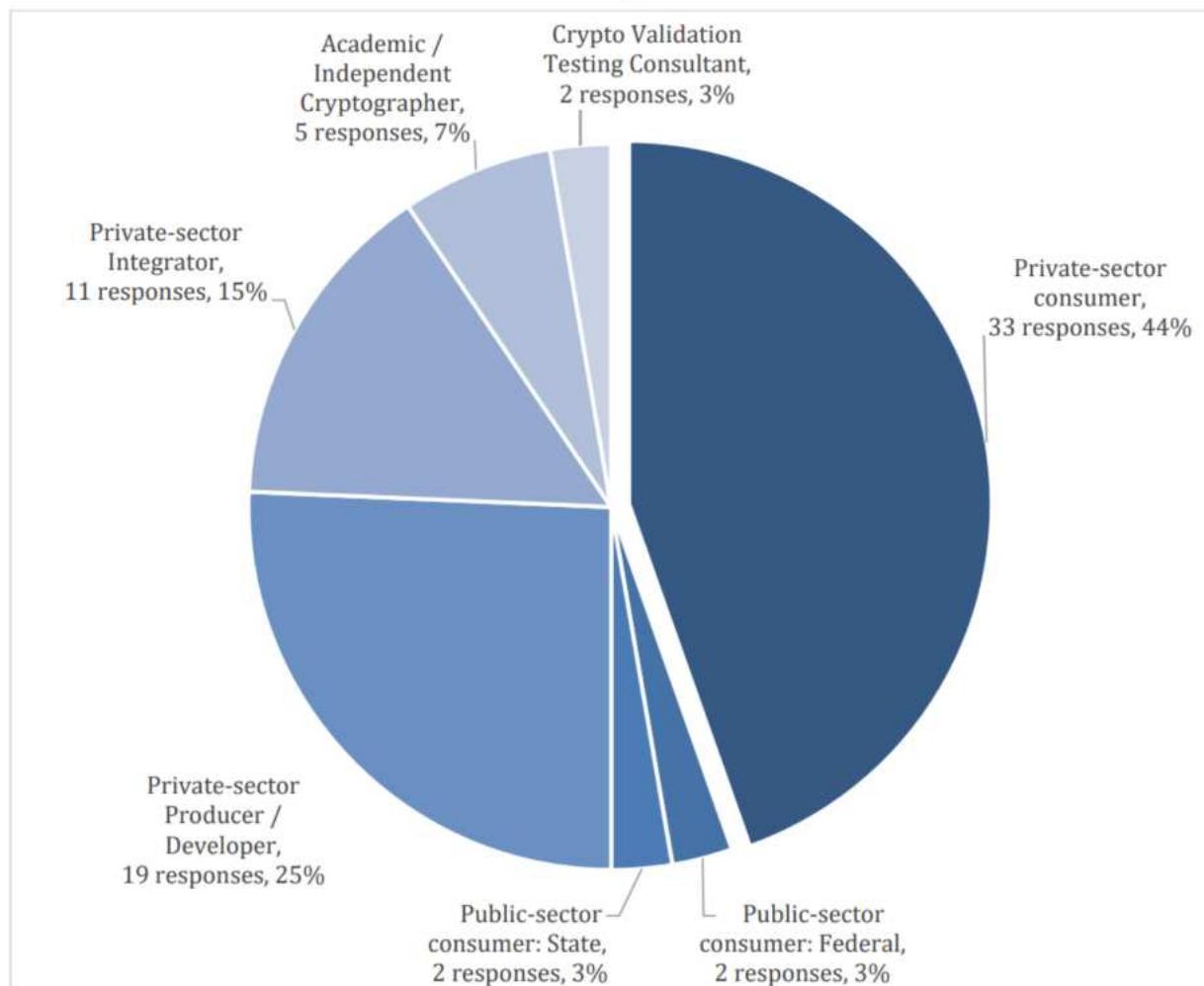
- Card processors (Visa, MasterCard, American Express, etc.) must use encryption to protect data storage and transmission.

Benefici da costi evitati

- 1) Avoided costs of slower processing speed
- 2) Interoperability costs avoided
- 3) Breach costs avoided
- 4) Pre-acquisition costs avoided
- 5) Standards development costs avoided
- 6) Lost sales and profits avoided
- 7) Hardware and software module quality degradation avoided

Survey

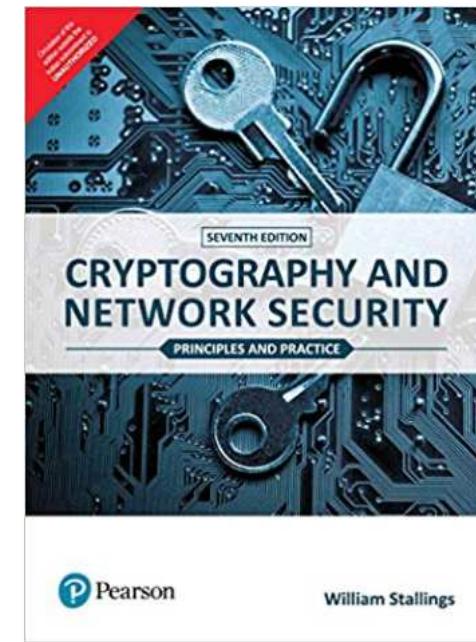
Figure 5-1. Survey Results: Seventy-Four (74) Respondents Reporting Quantifiable Benefits



Bibliografia

Cryptography and Network Security:
Principles and Practices
Prentice-Hall (7/Ed)
by W. Stallings, 2016

- cap. 5 (AES) + appendice 5.A
- cap. 4, par. 4.6
(proprietà di $GF(2^8)$)



Domande?

