Cifrari a chiave simmetrica

- Cifrari a Blocchi
  - Il testo in chiaro è suddiviso in blocchi di lunghezza fissa
  - La cifratura è fatta blocco per blocco.

- Cifrari a Flusso
  - Data una stringa del testo in chiaro producono una stringa del testo cifrato usando un keystream
  - Caso speciale di un cifrario a blocchi di lunghezza 1

Ricerca esaustiva della chiave

Key Size (bits)	Number of Alternative Keys	Time required at 1 encryption/μs	Time required at 10^6 encryptions/μs
32	$2^{32} = 4.3 \times 10^9$	2^{31} μs = 35.8 minutes	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	2^{55} μs = 1142 years	10.01 hours
128	$2^{128} = 3.4 \times 10^{38}$	2^{127} μs = 5.4×10^{24} years	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	2^{167} μs = 5.9×10^{36} years	5.9×10^{30} years
26 characters (permutation)	$26! = 4 \times 10^{26}$	2×10^{26} μs = 6.4×10^{12} years	6.4×10^{6} years

E' un attacco con forza bruta con un gran numero di processori in parallelo

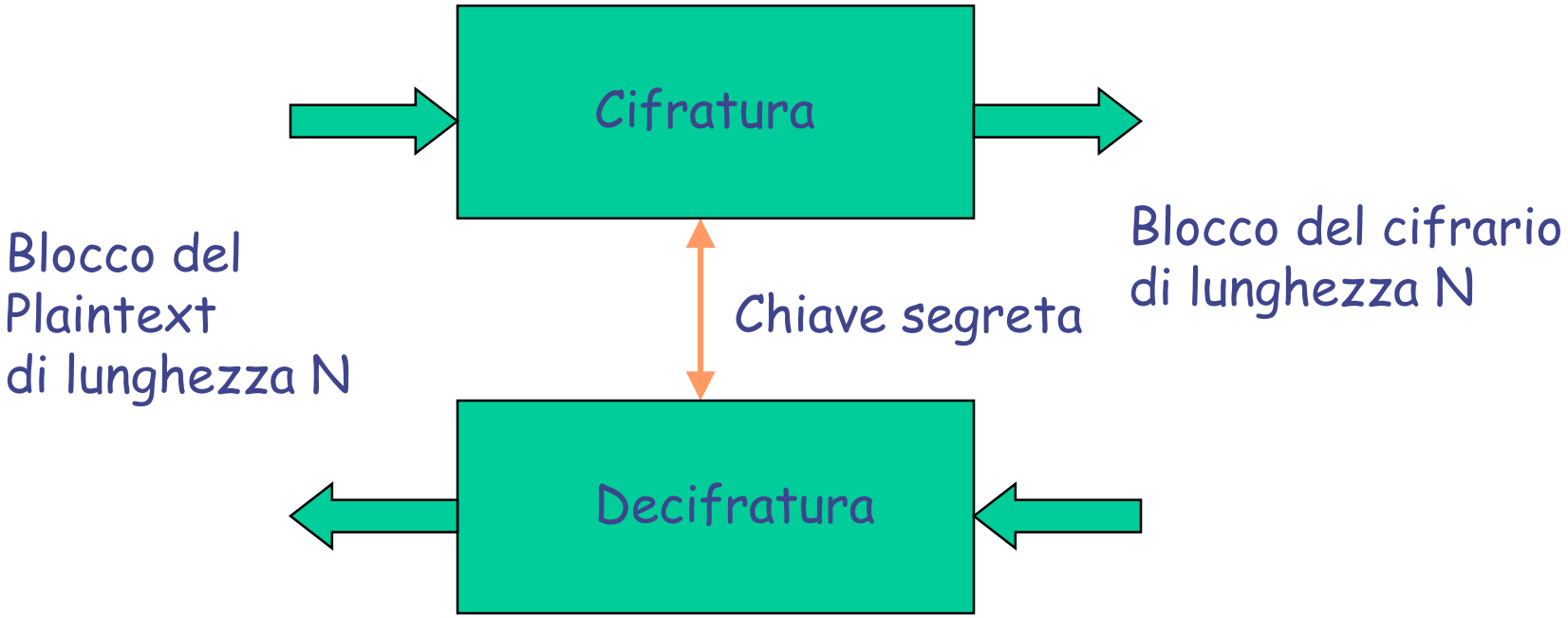
Cifratura del 20° secolo

- I malavitosi degli anni '20 e 30 usavano ampiamente la crittografia
- L'FBI crea un dipartimento per rompere i codici
- Giapponesi: **Purple Machine**
- Tedeschi: **Enigma Machine**

Cifratura di un blocco generico

- Trasforma un blocco in un altro: 1-1
- Abbastanza lungo da evitare l'attacco known-plaintext
  - tipicamente a 64 bit, bene per RISC
- Output dovrebbe apparire casuale
  - Nessuna correlazione tra plaintext e ciphertext
  - Dispersione di bit
- ciclo: combinazione di blocchi per la sostituzione e per la permutazione spesso fanno sì che il cambiamento di un bit abbia impatto su ogni bit di output
  - Quanti cicli occorrono? Non pochi, ma neppure molti

Modello del cifrario a blocchi



Struttura del cifrario di Feistel

- Horst Feistel (IBM), 1973
- L'input è un blocco di plaintext di lunghezza *2w* bits (in genere 64) ed una chiave *K*
- Il blocco è suddiviso in due parti, L_0 ed R_0
- Ad ogni ciclo *i* gli inputs sono L_{i-1} ed R_{i-1}, che sono ottenuti a partire dal ciclo precedente, per mezzo della chiave K_i
- La sostituzione è applicata nella parte a sinistra dei dati
- Ad ogni ciclo la funzione *F* è applicata sulla parte a destra dei dati ed il risultato è passato in XOR assieme alla parte sinistra dei dati.

Struttura del cifrario di Feistel

L'operazione di base è la seguente:
Suddividere un blocco del plaintext in due parti uguali, (L_0, R_0)
Ad ogni ciclo i=1,…,n, calcolare

$L_i = R_{i-1}$

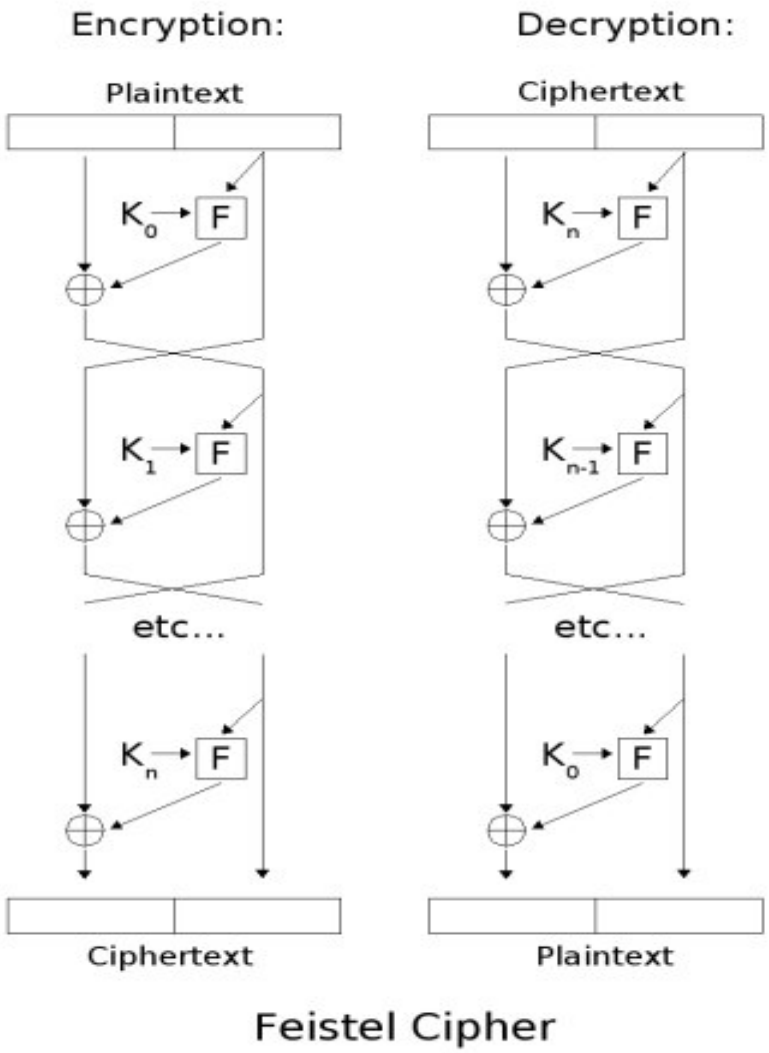
$R_i = L_{i-1}$ xor $f(R_{i-1}, K_i)$

In cui f è la round function e K_i la sub-key.
Quindi, il ciphertext è (L_n, R_n).
A prescindere dalla funzione f, la decifratura è ottenuta con

$R_{i-1} = L_i$

$L_{i-1} = R_i$ xor $f(R_{i-1}, K_i)$

Struttura del cifrario di Feistel



Feistel Cipher

Elementi importanti:

-Dimensione blocco (64)
-Lunghezza Chiave (128)
-# di cicli (16)
-Generazione della SubKey
-Round function

Data Encryption Standard (DES)

- Le aziende finanziarie (banche) avevano bisogno di un algoritmo crittografico che avrebbero avuto la benedizione dal governo USA (=NSA)
- Prima richiesta a Maggio 73, seguita da una successiva richiesta ad Agosto 74
- Non ci fuono molte sottomissioni (Perchè?)
  - IBM sottomette Lucifer
- NSA lavora insieme a IBM alla ri-progettazione dell'algoritmo

Data Encryption Standard (DES)

- Adottato nel 1977, approvato per altri 5 anni fino al 1994, da NBS/NIST
- Plaintext di 64 bits (o blocchi di 64 bits), chiave a 56 bits
- Il plaintext è passato in input ad un ciclo di iterazione di 16 passi; ad ogni passo è prodotto un valore intermedio che fa da input per l'iterazione successiva.
- Oggi il DES nella sua versione originale è facilissimo da rompere e non è più utilizzato come algoritmo di cifratura

DES

- DES diventa uno standard federale nel Novembre 76
  - NBS (NIST) standard hardware nel Jennaio 77
  - ANSI X3.92-1981 (hardware + software)
  - ANSI X3.106-1983 (modi delle operazioni)
  - Australia AS2805.5-1985
- Usato nella maggior parte di EFT and EFTPOS dalle banche
  - Per ben due volte è stato riapprovato come standard per 5 anni
  - Da allora, 3-DES è stato raccomandato, oggi AES

DES

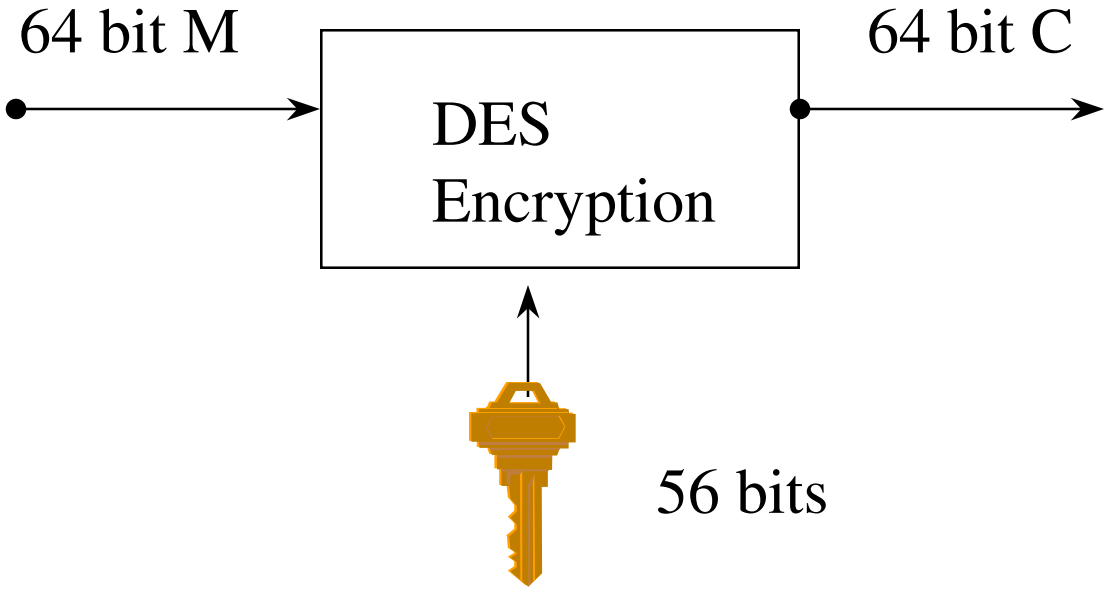
- Usa il principio di Feistel
- Molti punti di contatto con Lucifer
- Lo standard è pubblico, le specifiche di progettazione sono classificate
- Uno dei principali dibattiti riguarda la lunghezza della chiave (56 bits)
  - W Diffie, M Hellman "Exhaustive Cryptanalysis of the NBS Data Encryption Standard" IEEE Computer 10(6), June 1977, pp74-84
  - M Hellman "DES will be totally insecure within ten years" IEEE Spectrum 16(7), Jul 1979, pp 31-41
- Altro dibattito: esiste una trapdoor?

DES

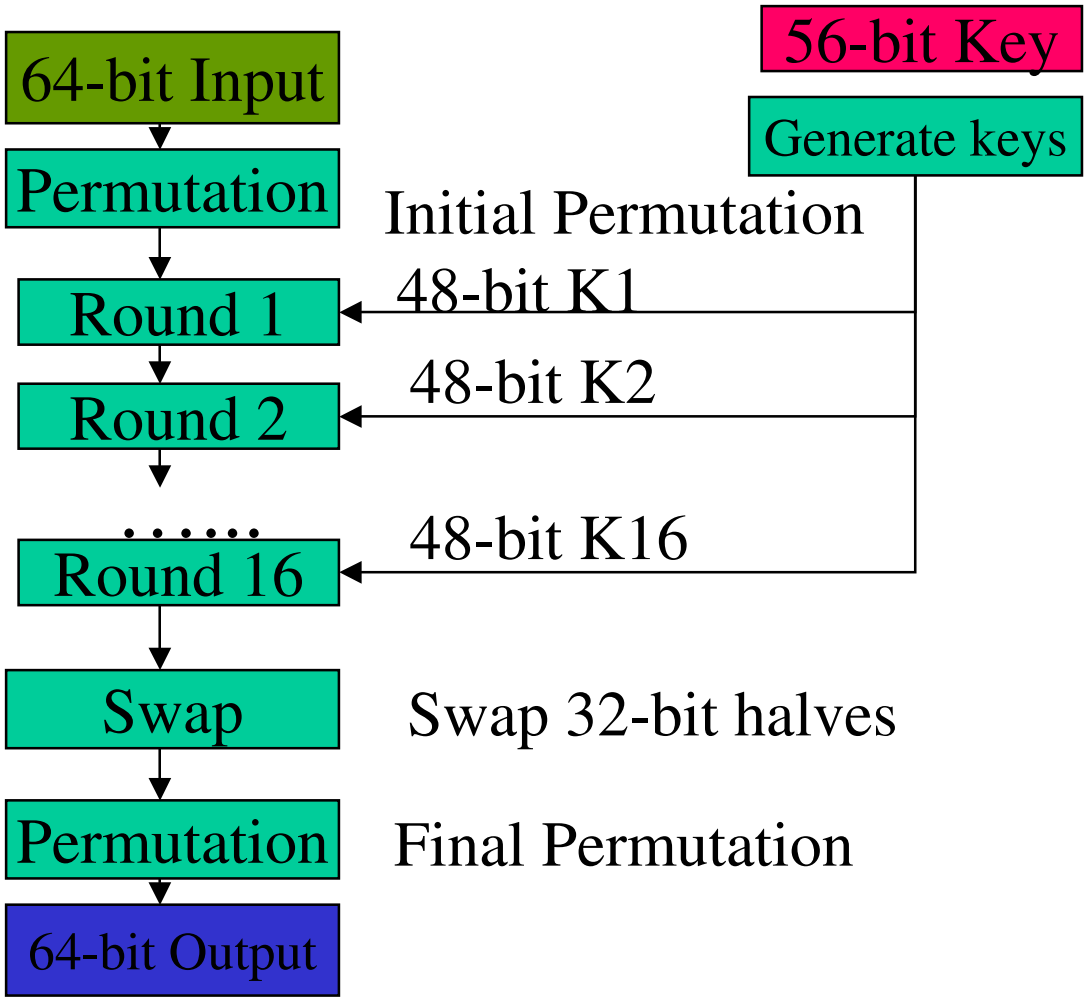
- E' stato dimostrato che DES ha un codice ben costruito
- E' stato provato che 56 bits sono pochi
  - Con circa $200,000 il sitema può essere rotto
  - Aumentare la chiave a 112 bits?
- Il modo migliore (dopo la forza bruta) che si conosce per fare la crittoanalisi su DES consiste nell'analisi differenziale
  - NSA lo sapeva già fin dall'inizio??

DES (Data Encryption Standard)

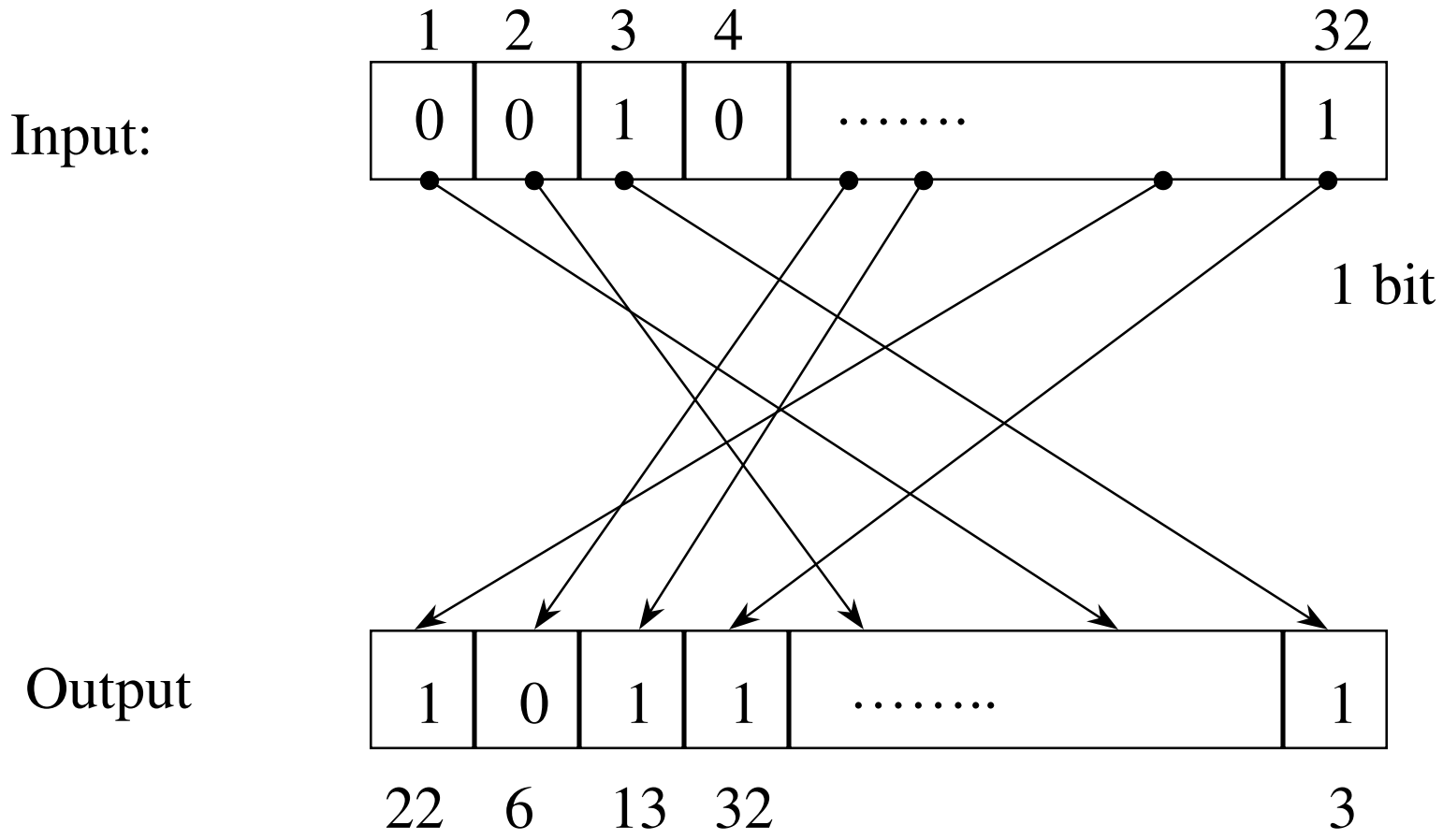
- Chiave: 64 bit = chiave di 56 bits + 8 bits per la parità del blocco
  - L' 8° è un bit di parità.
- input di 64 bit, output di 64 bit.



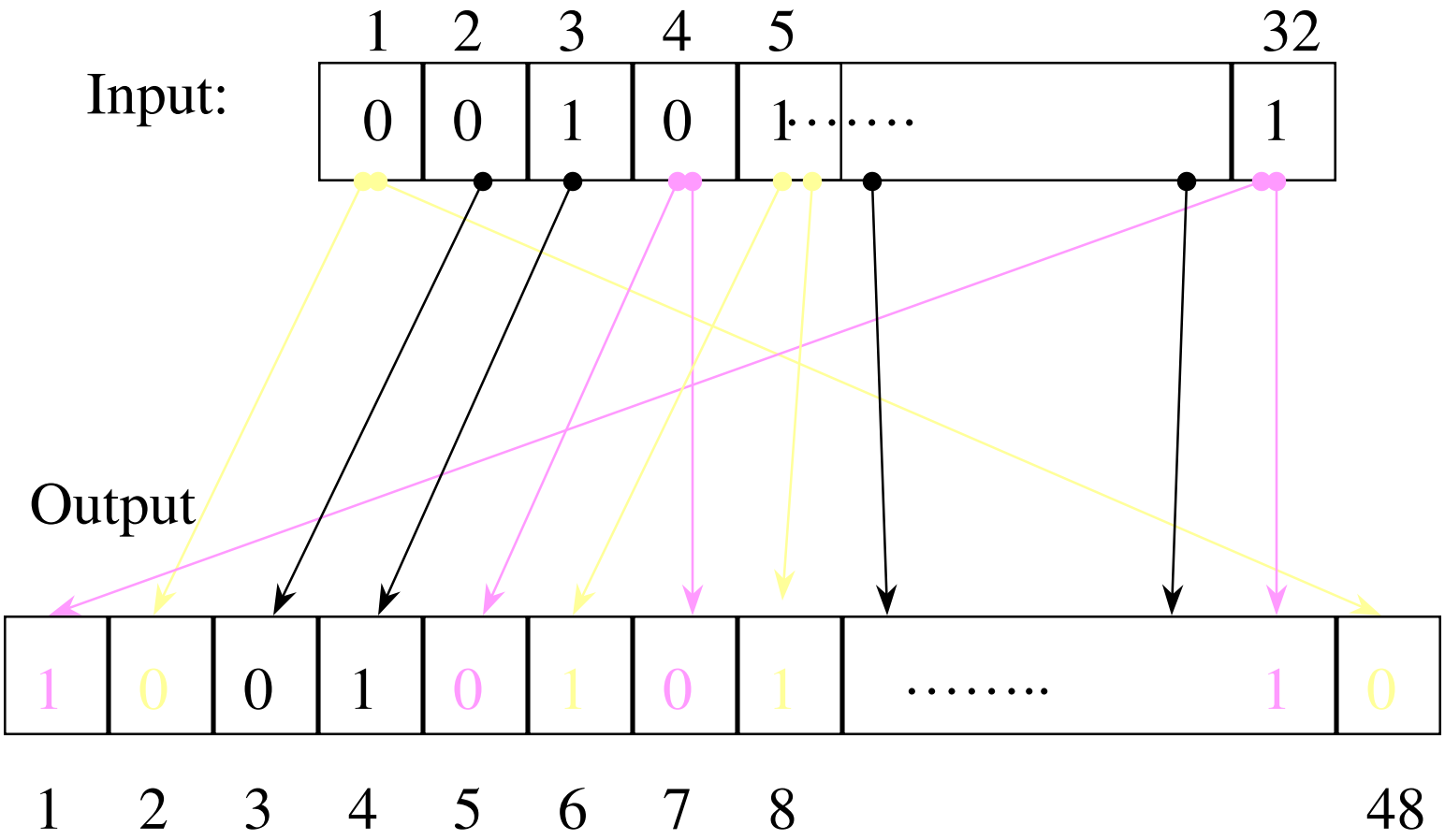
DES Top View


56-bit Key
64-bit Input
Generate keys
Permutation
Initial Permutation
Round 1
48-bit K1
Round 2
48-bit K2
.....
Round 16
48-bit K16
Swap
Swap 32-bit halves
Permutation
Final Permutation
64-bit Output

Permutazione di bit (1-to-1)



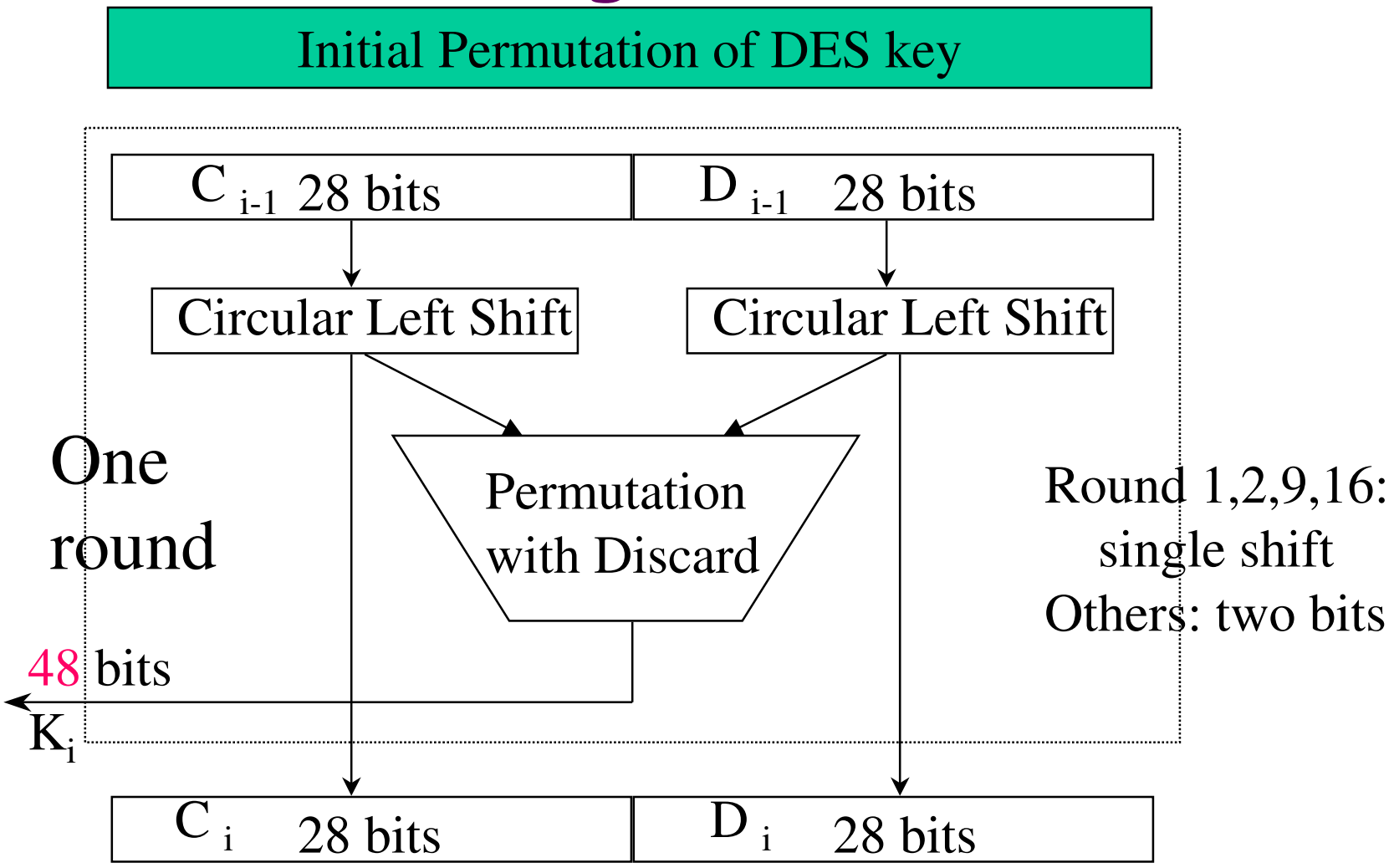
Bits Expansion (1-to-m)



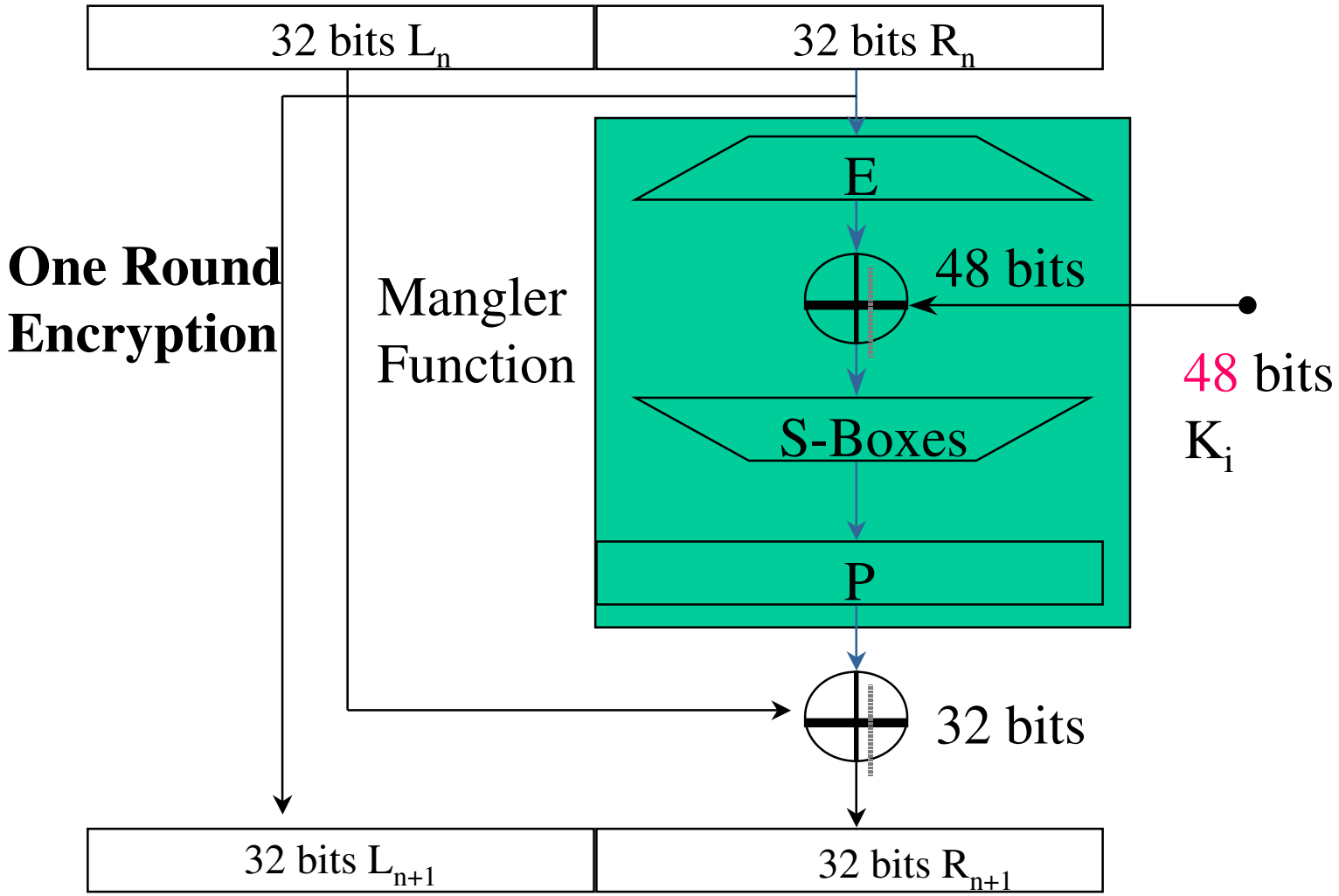
Permutazioni iniziale e finale

- Initial permutation (*IP*)
- L'input è trattato come una matrice
  M: 8-byte X 8-bit
- Trasforma M in M1 in due passi
  - La riga *x* diventa la colonna (9-*x*), 0<*x*<9
  - Applica la permutazione di righe:
    - La colonna *y* è trasformata nella riga *y*/2
    - La colonna di ordine dispari *y*, è trasformata nella riga (5+*y*/2)
- Final permutation $FP = IP^{-1}$

Generazione delle chiavi ad ogni ciclo



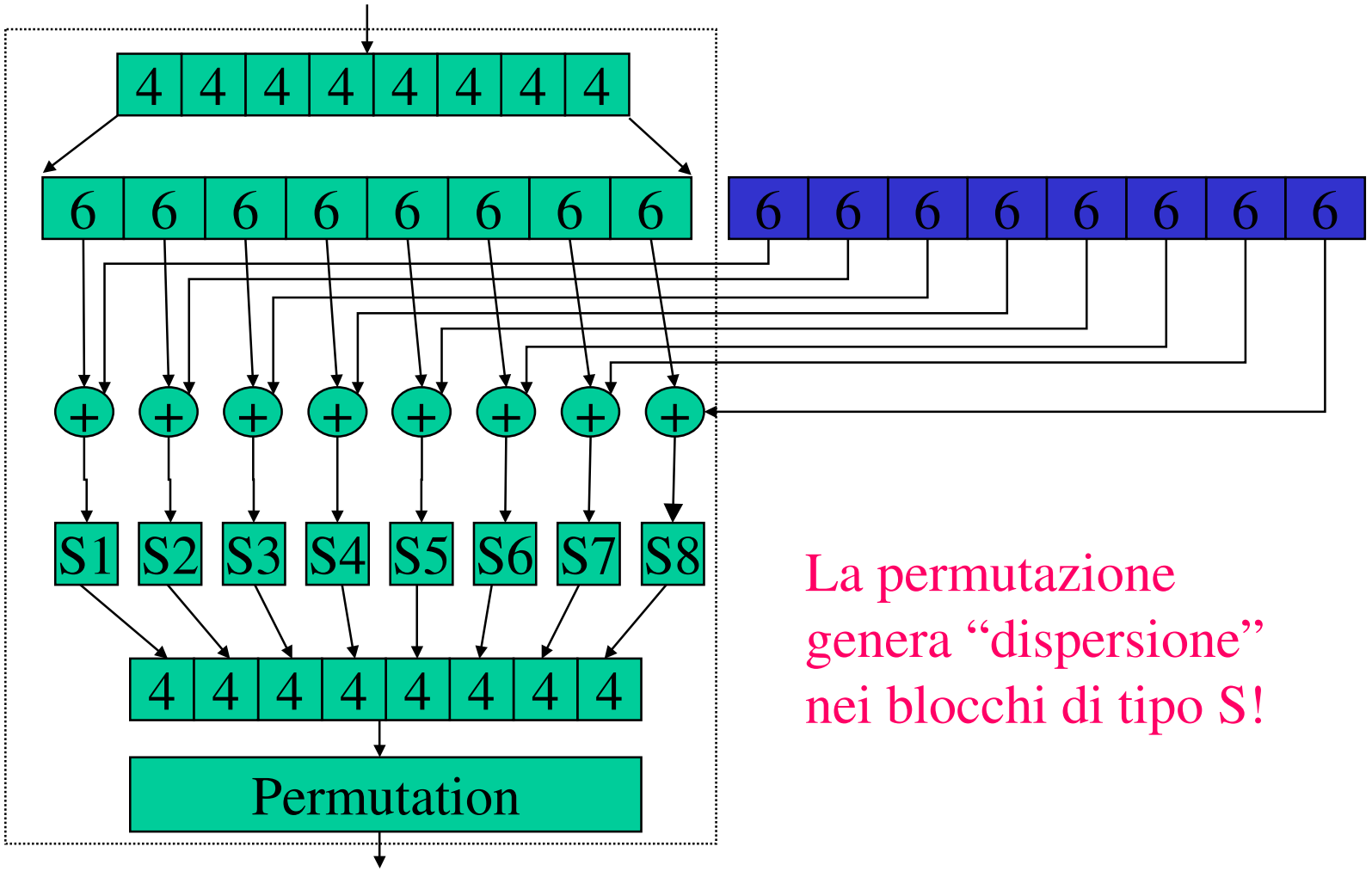
Un ciclo DES


32 bits L_n
32 bits R_n
One Round Encryption
Mangler Function
E
48 bits
S-Boxes
P
48 bits
K_i
32 bits
32 bits L_{n+1}
32 bits R_{n+1}

Decifratura

- Ad ogni ciclo, applica le stesse operazioni con la stessa chiave K_i:
  - Input: $R_{n+1}|L_{n+1}$
    - Per via dell'operazione di "swap"
  - Output: $R_n|L_n$
    - Lo swap finale genera il risultato corretto: $L|R$

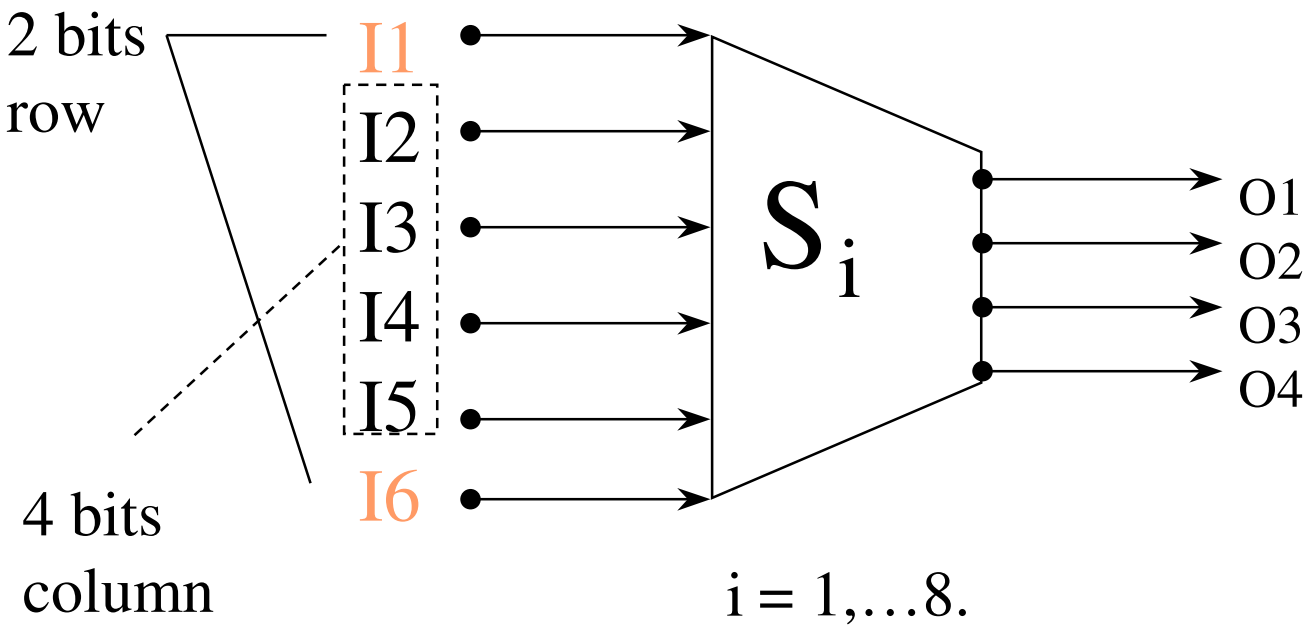
Funzione di Mangler



La permutazione genera “dispersione” nei blocchi di tipo S!

Blocchi di tipo S (Substitute and Shrink)

- 48 bits ==> 32 bits. (8*6 ==> 8 *4)
- 2 bits sono usati per scegliere 4 bits nelle 4 permutazioni



S1: (p. 70)

Righe e colonne contengono elementi differenti.

	0	1	2	3	4	5	6	7	8	9.... 15
0	14	4	13	1	2	15	11	8	3	
1	0	15	7	4	14	2	13	1	10	
2	4	1	14	8	13	6	2	11	15	
3	15	12	8	2	4	9	1	7	5	

Esempio: input: 100110 output: ???

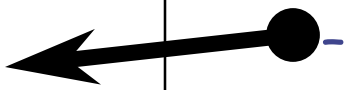
8 Blocchi di tipo S

- La logica che sta dietro alla scelta dei blocchi di tipo S è segreta e non viene resa pubblica
- Sarebbe una buona idea tecnicamente renderla pubblica?

Standard DES

- Iterazione del Cifrario :
  - Input: 64 bits
  - Key: 48 bits
  - Output: 64 bits

- Blocco di Generazione della chiave :
  - Input: 56 bits
  - Output: 48 bits



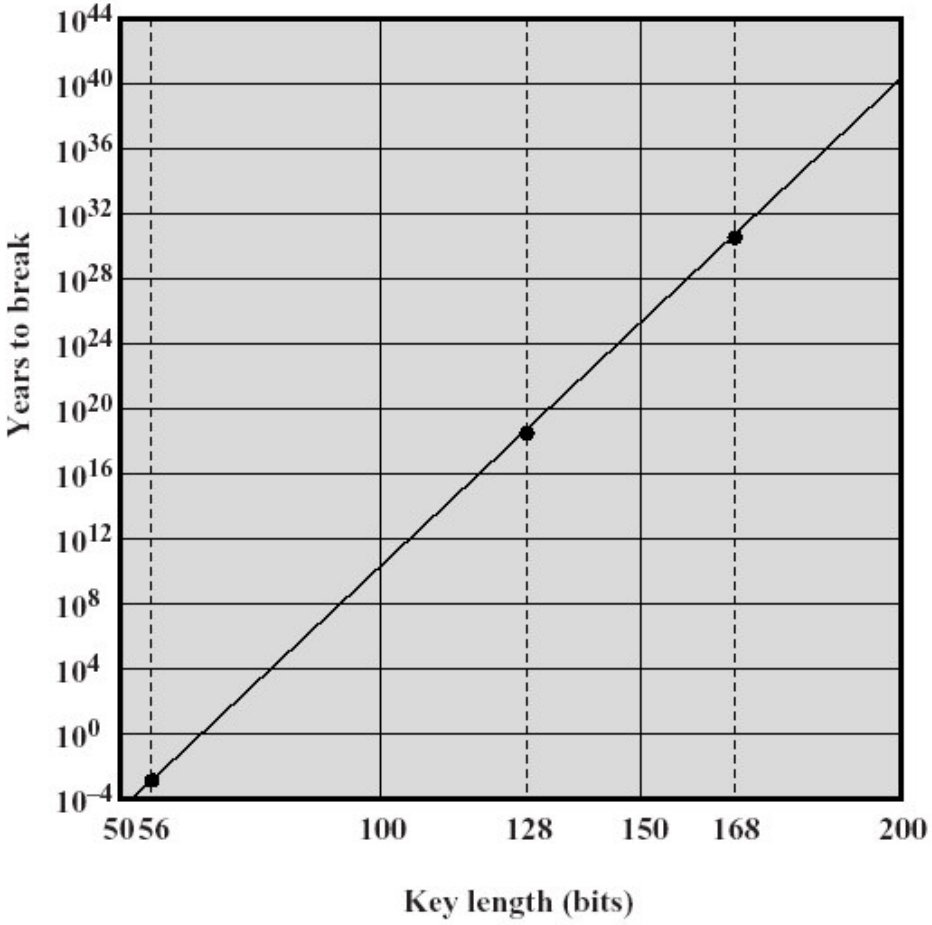
1 Ciclo (Complessivamente 16 rounds)

Riepilogo sul blocco DES

- Semplice e facile da implementare:
  - Hardware/gigabits/second, software/megabits/second
- La chiave DES a 56 bits può essere accettabile per applicazioni non critiche; piuttosto il triple DES (DES3) dovrebbe essere sicuro per la maggiorparte delle applicazioni odierne
- Supporta molte modalità operative: ECB CBC, OFB, CFB

Forza del DES

- Aspetti riguardanti l'algoritmo stesso
- Aspetti riguardanti la chiave a 56 bits - questo è il più grande difetto

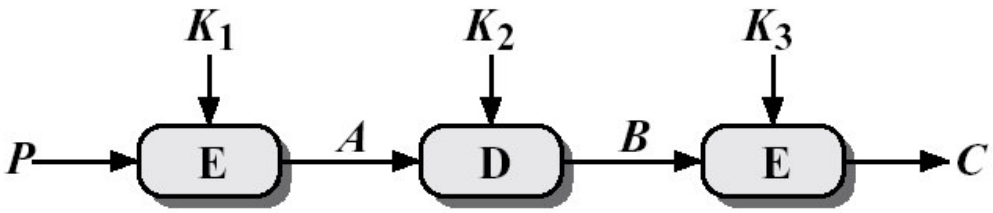


Forza del DES

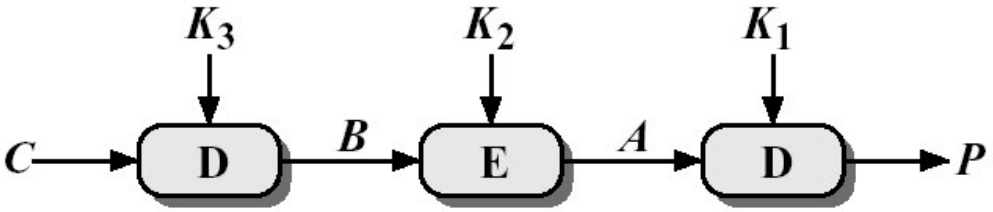
- Il DES è l'algoritmo di cifratura che è stato più ampiamente studiato in assoluto
- Nessuno è mai riuscito a scoprire una forte debolezza
- 1998, DES cracker costruito da Electronic Frontier Foundation per $250,000
- Soluzione: Uso di una chiave più lunga

Triple DES

$C = E_{K_3}[D_{K_2}[E_{K_1}[P]]]$



(a) Encryption



(b) Decryption

Triple DES

- La cifratura tripla compensa la lunghezza troppo corta della chiave DES di base
  - Maggiore sicurezza
- Standard de facto: $E(K_3, D(K_2, E(K_1, P)))$
- $K_1=K_3$ dà luogo ad un DES equivalente a 112-bit che fornisce uno spazio di chiavi sufficiente (cosa succede con $K_2=K_3$ o $K_1=K_2$?)
- La scelta di K_1, K_2, K_3 diverse dà luogo ad un DES a 168-bit che risulta anche più sicuro

3-DES non è la scelta migliore...

- Ci sono chiavi più lunghe da inizializzare!
- 3-DES ha un tempo di esecuzione che è il triplo del DES semplice
- Esistono numerosi altri schemi: - RC5, IDEA, two-fish, CAST, etc
- NIST ha sollecitato la progettazione di algoritmi da utilizzare per uno standard federale

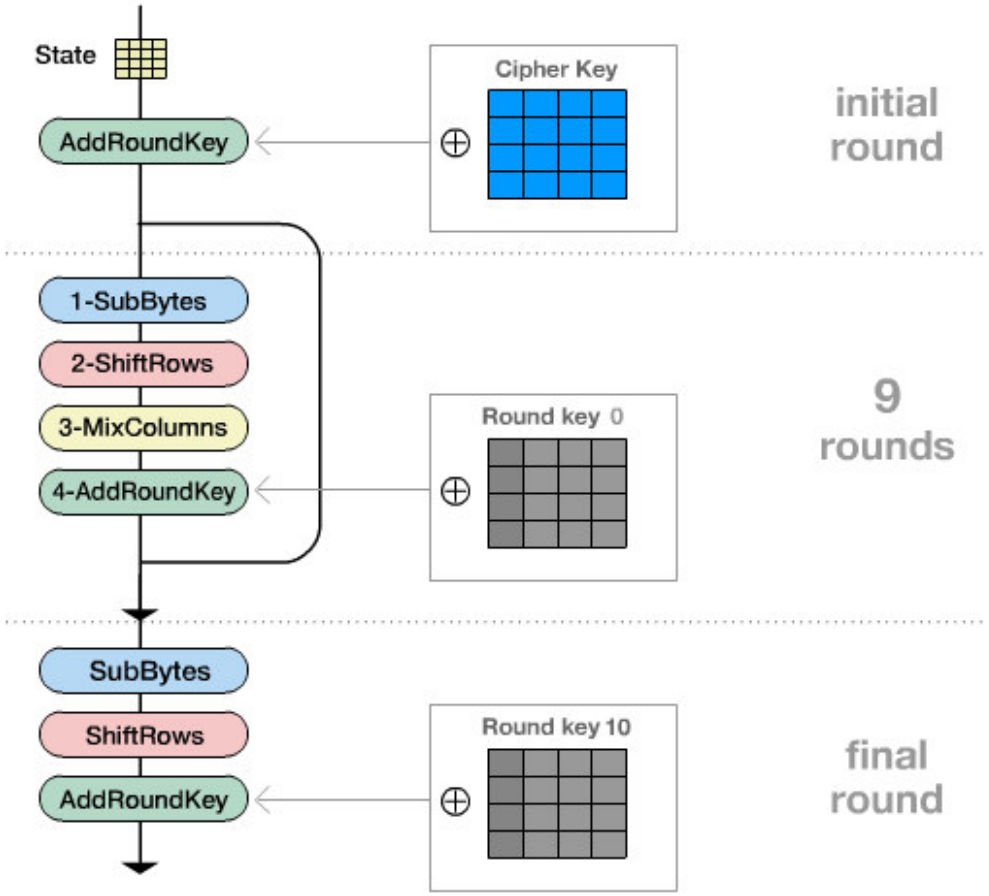
Advanced Encryption Standard

- NIST call for proposals nel 1997
- Nov, 2001 – **Rijndael** [rain´ dow]
- Cifrario a blocchi simmetrico (128 bits) e lunghezza delle chiavi 128, 192, 256
- Due crittografi fiamminghi: Joan Daeman e Vincent Rijmen

Overview di AES

4 Trasformazioni:

- Sostituzione di Bytes
- Shift Rows
- Mix Columns
- Add Round Key



Overview di AES

4 Trasformazioni:

- Sostituzione di Bytes

Un passo di sostituzione non lineare in cui ogni byte è sostituito da un altro secondo una lookup table.

- Shift Rows

Un passo di trasposizione in cui ogni riga dello stato è shiftata ciclicamente di un certo numero di passi.

- Mix Columns

operazione di mixing che lavora sulle colonne dello stato, combinando i quattro bytes di ogni colonna per mezzo di una trasformazione lineare

- Add Round Key

Ciascun byte dello stato è combinato con la round key; ciascuna round key è ottenuta a partire dalla chiave del cifrario usando un key schedule.

Overview di AES (implementazione Rijndael)

Animazione Rijndael !