CODICI SEGRETI

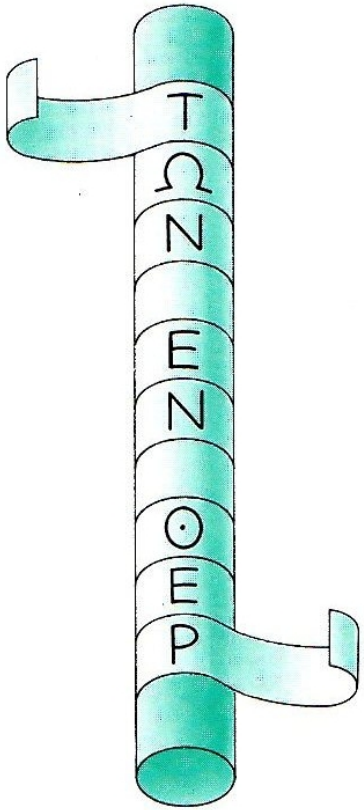
Agostino Dovier

Dip di Matematica e Informatica, Univ. di Udine

Ringrazio l'amico e maestro Andrea Sgarro per il materiale tratto dal suo meraviglioso quanto introvabile testo

Cifrari a trasposizione

Scitala spartana (≈ 400 AC)



Τ	Μ	Α	Κ	Α
Ω	Ο	Ν	Λ	
Ν	Π	Ο	Ε	Τ
	Υ	Ν	Η	Υ
Ε	Λ	Τ	Σ	Χ
Ν	Α	Ω		Α
	Ι	Ν	Μ	
Θ	Σ		Ε	
Ε		Ε	Ν	
Ρ	Θ	Υ		

Cifrari a trasposizione

Jules Verne (1828–1905): la griglia di Mattia Sandorf

ihnalz	zaemen	ruiopn
arnuro	trvree	mtqssl
odxhnp	estlev	eeuart
aeeeil	ennios	noupvg
spesdr	erssur	ouitse
eedgnc	toeedt	artuee

hazrxeirg nohaledec nadnepedn ilruopess
ssepourlindependancedelahongriexrzah
[ma]sse pour l'independance de la Hongrie. Xrzah

Cifrari a trasposizione

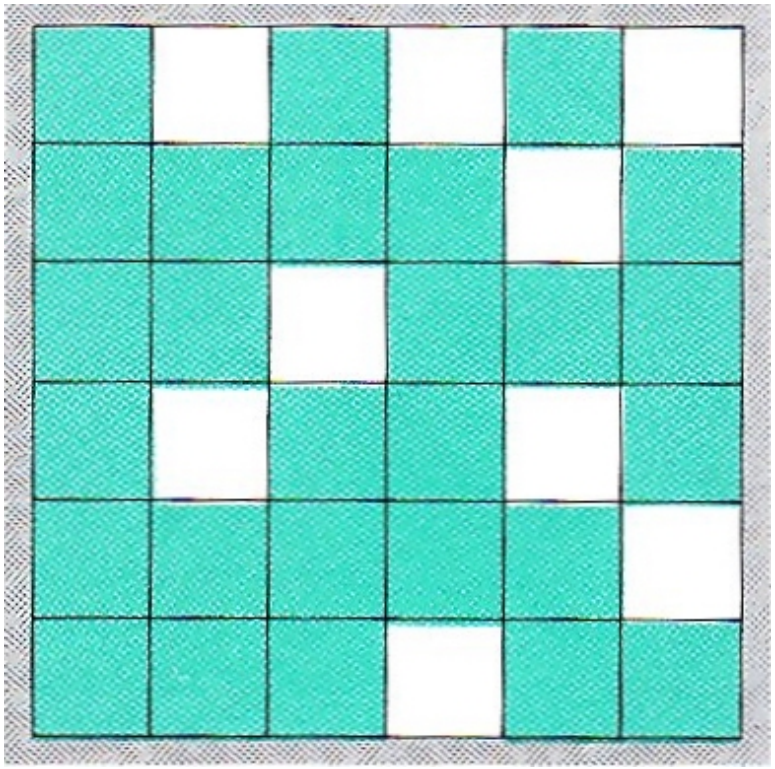
Jules Verne (1828–1905): la griglia di Mattia Sandorf

i	h	n	a	l	z
a	r	n	u	r	o
o	d	x	h	n	p
a	e	e	e	i	l
s	p	e	s	d	r
e	e	d	g	n	c

hazrxeirg nohaledec nadnepedn ilruopess
ssepourlindependancedelahongriexrzah
[ma]sse pour l'independance de la Hongrie. Xrzah

Cifrari a trasposizione

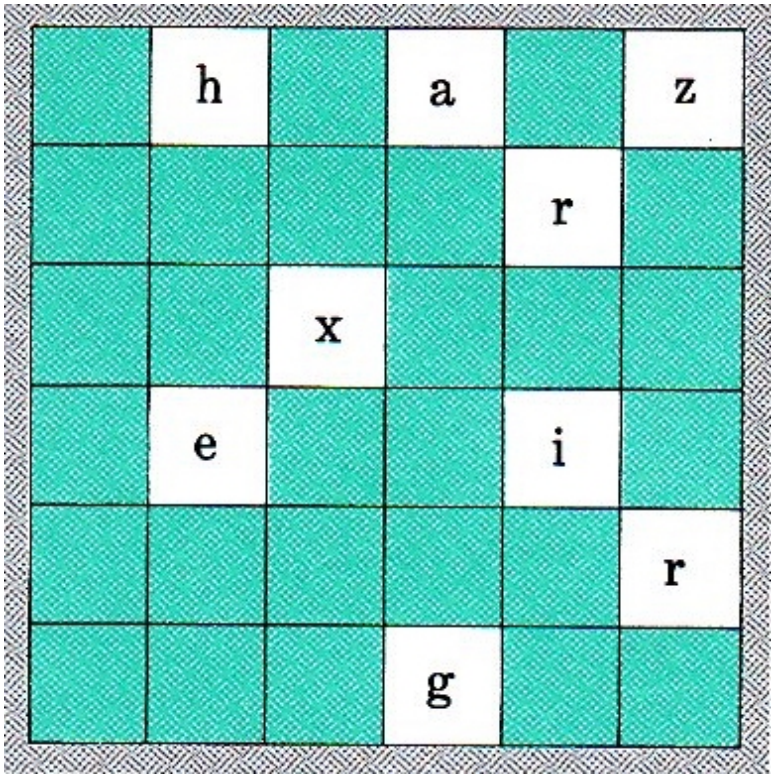
Jules Verne (1828–1905): la griglia di Mattia Sandorf



hazrxeirg nohaledec nadnepedn ilruopess
ssepourlindependancedelahongriexrzah
[ma]sse pour l'independance de la Hongrie. Xrzah

Cifrari a trasposizione

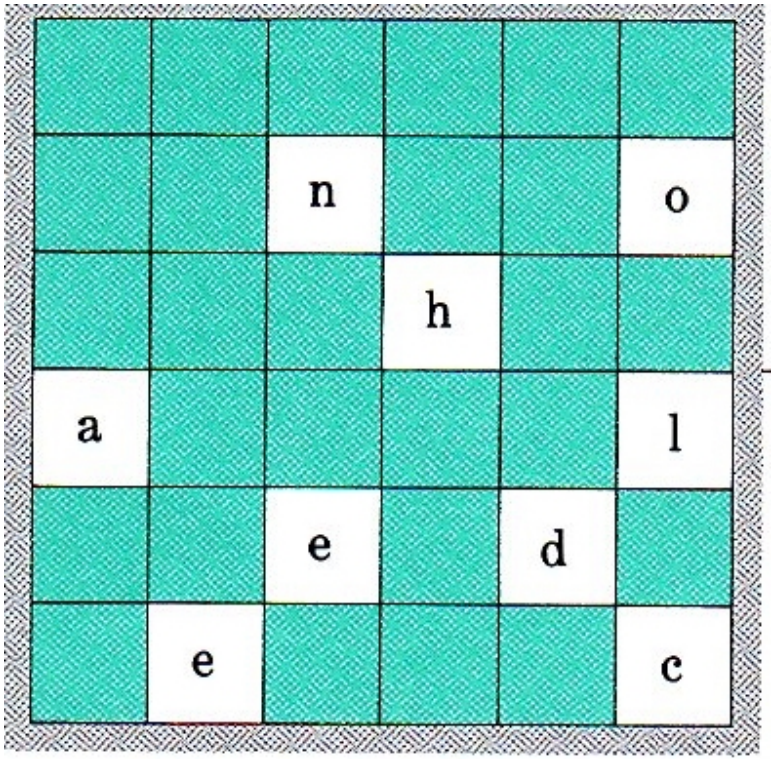
Jules Verne (1828–1905): la griglia di Mattia Sandorf



hazrxeirg nohaledec nadnepedn ilruopess ssepourlindependancedelahongriexrzah [ma]sse pour l'independance de la Hongrie. Xrzah

Cifrari a trasposizione

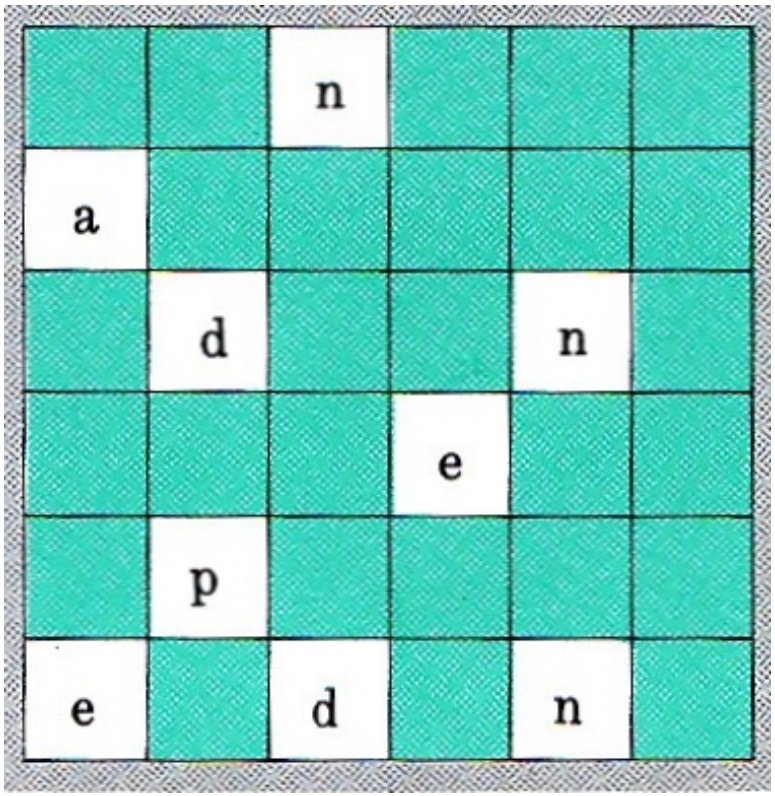
Jules Verne (1828–1905): la griglia di Mattia Sandorf



hazrxeirg nohaledec nadnepedn ilruopess
ssepourlindependancedelahongriexrzah
[ma]sse pour l'independance de la Hongrie. Xrzah

Cifrari a trasposizione

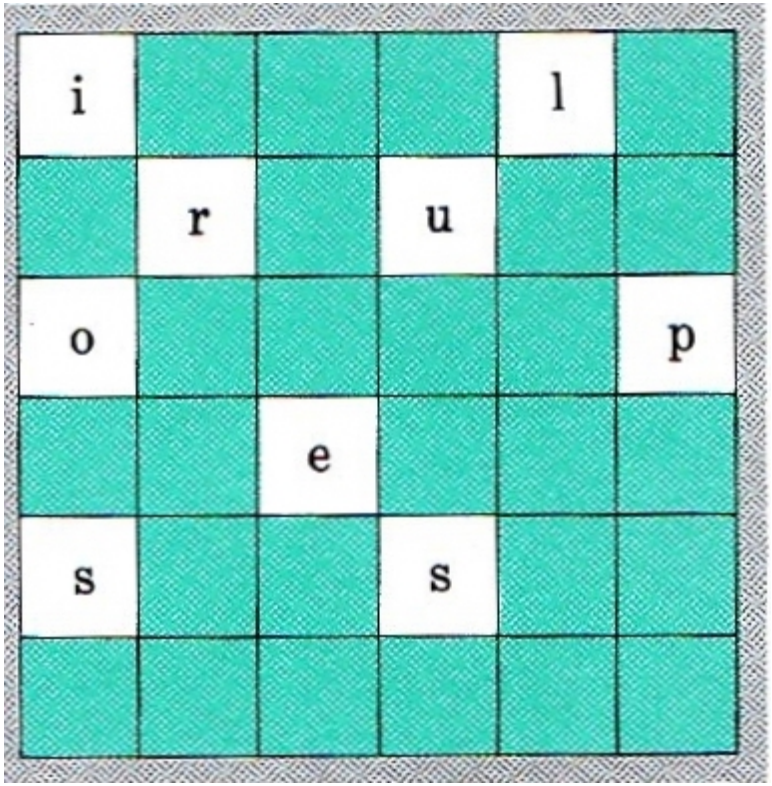
Jules Verne (1828–1905): la griglia di Mattia Sandorf



hazrxeirg nohaledec nadnepedn ilruopess
ssepourlindependancedelahongriexrzah
[ma]sse pour l'independance de la Hongrie. Xrzah

Cifrari a trasposizione

Jules Verne (1828–1905): la griglia di Mattia Sandorf



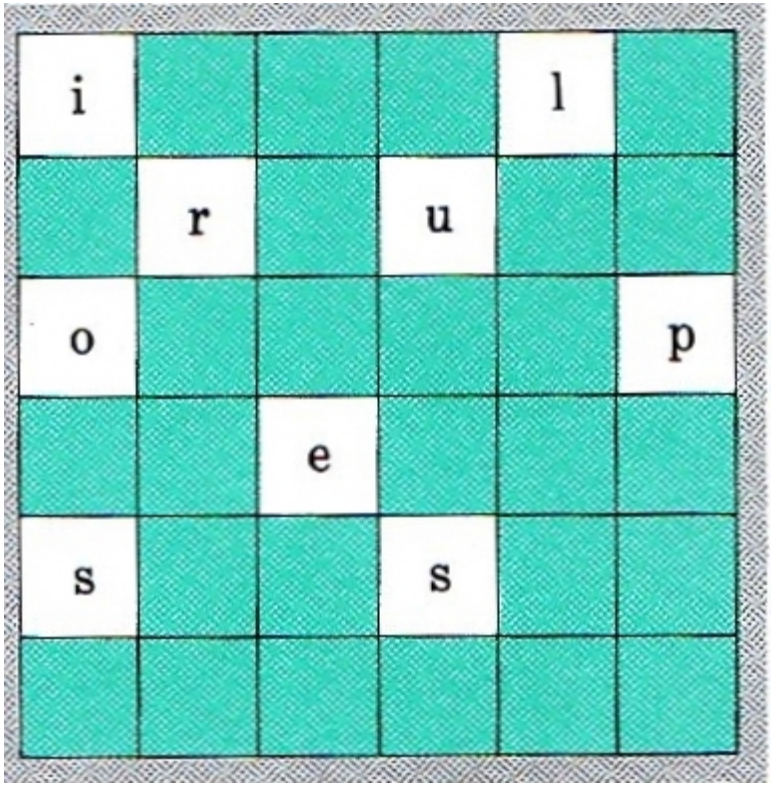
hazrxeirg nohaledec nadnepedn ilruopess

ssepourlindependancedelahongriexrzah

[ma]sse pour l'independance de la Hongrie. Xrzah

Cifrari a trasposizione

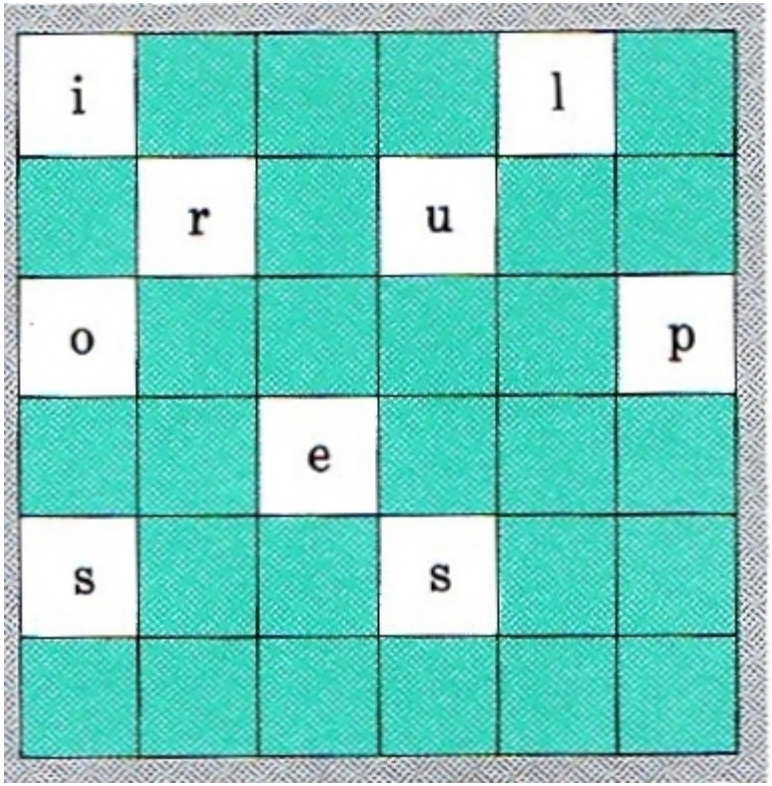
Jules Verne (1828–1905): la griglia di Mattia Sandorf



hazrxeirg nohaledec nadnepedn ilruopess
ssepourlindependancedelahongriexrzah
[ma]sse pour l'independance de la Hongrie. Xrzah

Cifrari a trasposizione

Jules Verne (1828–1905): la griglia di Mattia Sandorf



hazrxeirg nohaledec nadnepedn ilruopess
ssepourlindependancedelahongriexrzah
[ma]sse pour l'independance de la Hongrie. Xrzah

Data Encryption Standard

Breve storia

- 1973 L'NBS (National Bureau of Standards) ora NIST (National Institute of Standards and Technology) richiede un algoritmo standard di cifratura
- 1975 IBM (Horst Feistel et al) pubblica il DES nel Federal Register
- 1976/77 Il DES è approvato/pubblicato come standard
- Il funzionamento del DES è pubblico (soddisfa il principio di Kerckhoffs). Non sono necessariamente rese note le *ragioni* di alcune scelte.
- 1990 Biham e Shamir usando la crittanalisi differenziale, forzano un DES-like a 15 fasi. Ma il DES ne ha 16.

Data Encryption Standard

Breve storia

- Nel 1997 e 1998 il DES viene violato pubblicamente in risposta ad una competizione organizzata dalla RSA.
- 1998 Il DES cracker viola una chiave DES in 56 ore.
- 1999 DES cracker e distributed.net assieme violano una chiave DES in 22 ore e 15 minuti.
- 1999 Il DES raccomanda di usare la versione Triple DES (chiave di 168 bit)
- 2001 AES (Rijndael) viene pubblicato come standard per sostituire il DES.

Data Encryption Standard

Chiave

- Del DES tutto è noto tranne la chiave.
- La chiave del DES è costituita da 8 bytes.
- L'ultimo bit di ogni byte è un bit di disparità.
- Dunque la vera lunghezza della chiave è 56 bits (spazio di $2^{56} \approx 7.2 \times 10^{16}$).
- La codifica di un messaggio avviene dividendolo in blocchi di 64 bits e processandone uno alla volta.

Data Encryption Standard

Chiave

- Del DES tutto è noto tranne la chiave.
- La chiave del DES è costituita da 8 bytes.
- L'ultimo bit di ogni byte è un bit di disparità.
- Dunque la vera lunghezza della chiave è 56 bits (spazio di $2^{56} \approx 7.2 \times 10^{16}$).
- La codifica di un messaggio avviene dividendolo in blocchi di 64 bits e processandone uno alla volta.

Data Encryption Standard

Fase 0: trasposizione iniziale

Il blocco $I = x_1 x_2 \ldots x_{64}$ in ingresso viene trasposto mediante la trasposizione iniziale TI e diventa

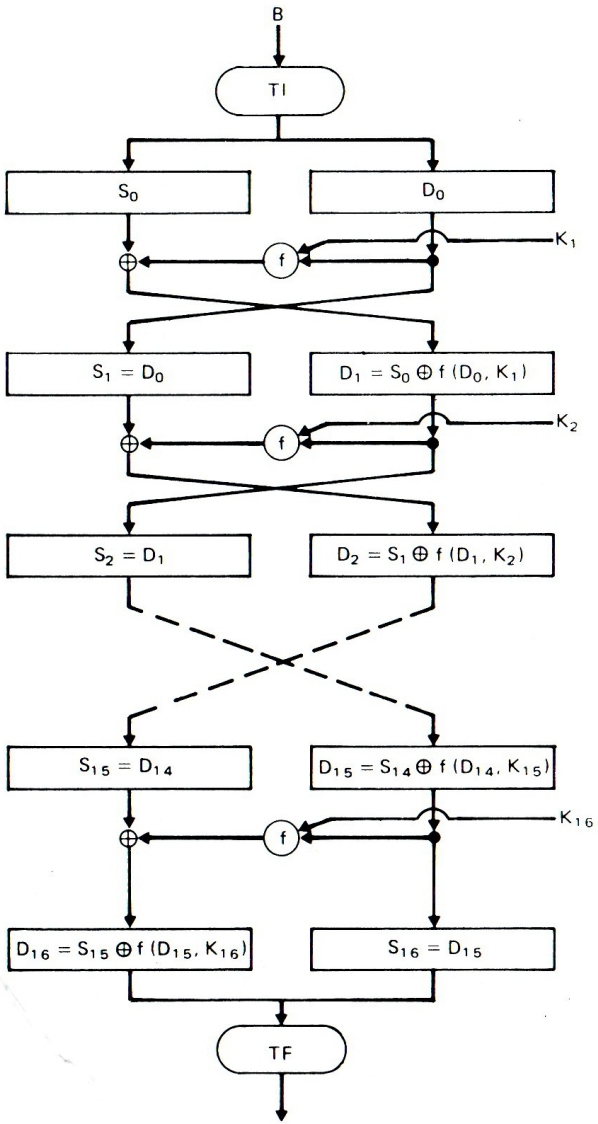
$$T_0 = \mathrm{TI}(I)$$

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Precisamente $T_0 = x_{58} x_{50} x_{42} \ldots x_{15} x_7$.

Data Encryption Standard

Fase 1–16: Algoritmo principale



- $T_0 = \mathrm{TI}(I)$
- Per $i > 0$, sia T_i il risultato (64 bit) della i-esima iterazione.
- Per $i \geq 0$, siano S_i e D_i t.c. $T_i = S_i D_i$, $|S_i| = |D_i| = 32$.
- Per $i = 1, \ldots, 16$, sia:

$$\begin{cases} S_i &= D_{i-1} \\ D_i &= S_{i-1} \oplus f(D_{i-1}, K_i) \end{cases}$$

- Ricordiamoci di $T_{16} = S_{16} D_{16}$
- Descriviamo ora f e K_i

Data Encryption Standard

La funzione f

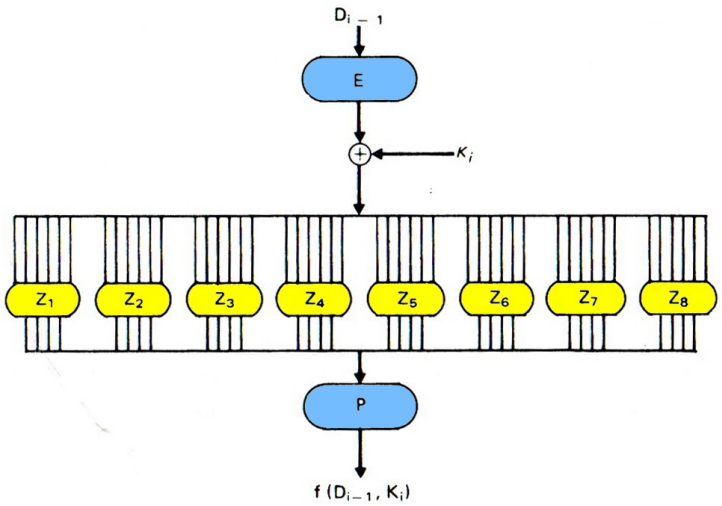
Dobbiamo calcolare $f(D_{i-1}, K_i)$.
Prima D_{i-1} (32 bit) viene espanso in un blocco di 48 bit $E(D_{i-1})$ usando la tabella

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Se $D_{i-1} = d_1 d_2 \cdots d_{32}$ allora $E(D_{i-1}) = d_{32} d_1 d_2 \cdots d_{32} d_1$.

Data Encryption Standard

La funzione f



- Si calcola la chiave K_i (48 bits).
- Si calcola $E(D_{i-1}) \oplus K_i$ e lo si legge come 8 blocchetti di 6 bits $\beta_1 \cdots \beta_8$.
- Ogni blocchetto di 6 bits viene trasformato in un blocchetto da 4 bits mediante la sostituzione Z_j con $j = 1, \ldots, 8$.

Data Encryption Standard

La funzione f

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7	
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8	Z_1
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0	
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13	
0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10	
1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5	Z_2
2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15	
3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9	
0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8	
1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1	Z_3
2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7	
3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12	
0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15	
1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9	Z_4
2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4	
3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14	
0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9	
1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6	Z_5
2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14	
3	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3	
0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11	
1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8	Z_6
2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6	
3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13	
0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1	
1	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6	Z_7
2	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2	
3	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12	
0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7	
1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2	Z_8
2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8	
3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11	

- Consideriamo i bits $b_1 \ldots b_6$ del blocco j.
- I bits $b_1 b_2$ individuano la riga, i bits $b_2 b_3 b_4 b_5$ la colonna.
- L'intero trovato viene codificato in binario con 4 bits.

DATA ENCRYPTION STANDARD

LA FUNZIONE *f*

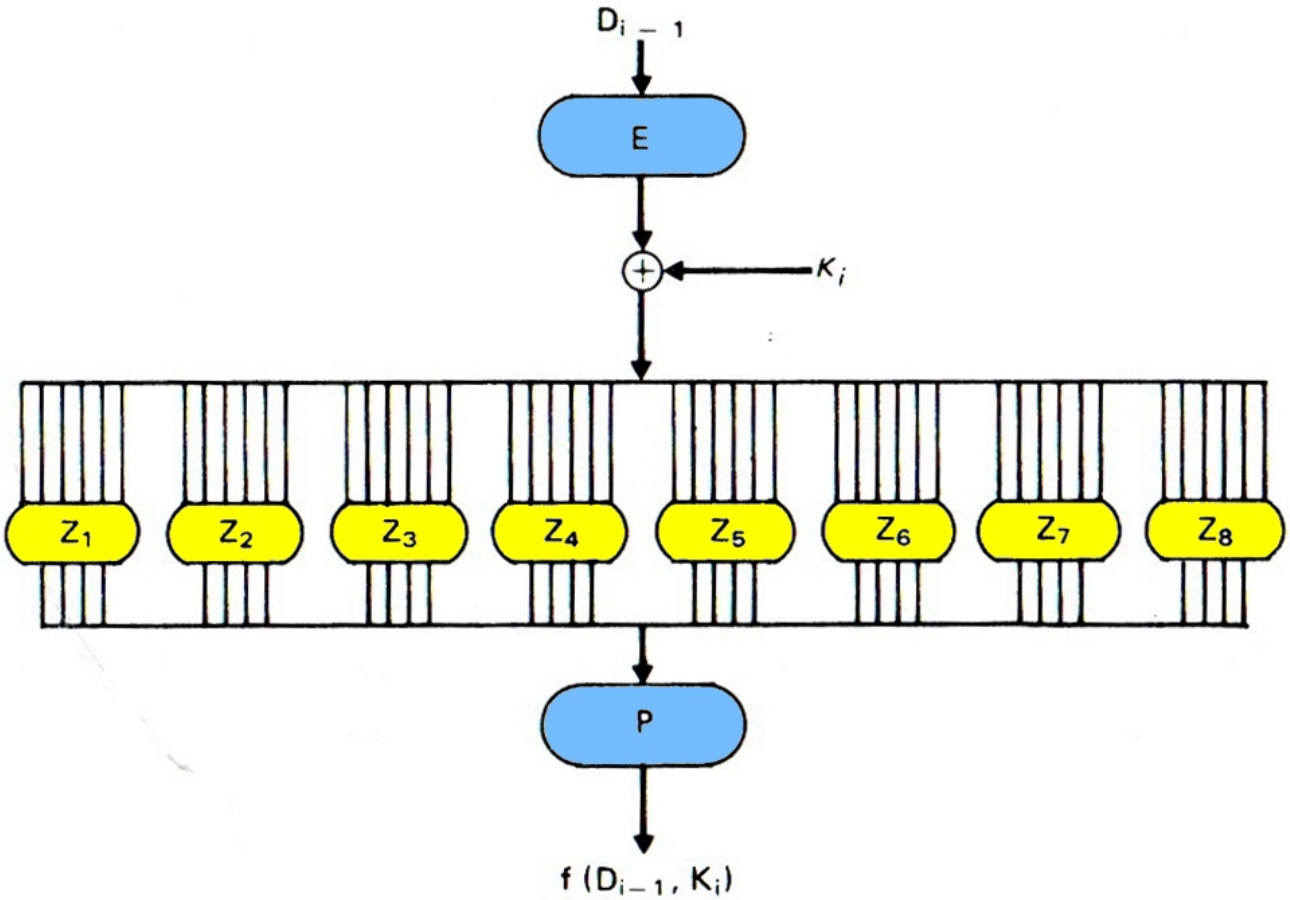
La concatenazione degli 8×4 bits (sia $b_1 \ldots b_{32}$) viene dunque trasposta usando la seguente tabella:

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

ottenendo dunque $b_{16} b_7 \cdots b_4 b_{25}$.

Data Encryption Standard

La funzione *f* (riassunto)



Data Encryption Standard

Calcolo delle chiavi parziali

La chiave K ha 64 bits (inclusi quelli di disparità) $k_1 \ldots k_{64}$.
Usiamo la seguente tabella TK per trasporla:

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Otteniamo 56 bits $k_{57}k_{49}\cdots k_{12}k_4$.
Li spezziamo nei semiblocchi s_0 e d_0 da 28 bits.

Data Encryption Standard

Calcolo delle chiavi parziali

I blocchi s_0 e d_0 vengono shiftati ciclicamente a sinistra di uno o due posizioni usando la tabella a destra.
Si ottengono:

$$\langle s_1, d_1 \rangle$$
$$\langle s_2, d_2 \rangle$$
$$\vdots$$
$$\langle s_{16}, d_{16} \rangle$$

TABELLA 4.8 *Le rotazioni a sinistra RS*

Iterazione	Ampiezza della rotazione a sinistra
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	1
10	2
11	2
12	2
13	2
14	2
15	2
16	1

Data Encryption Standard

Calcolo delle chiavi parziali

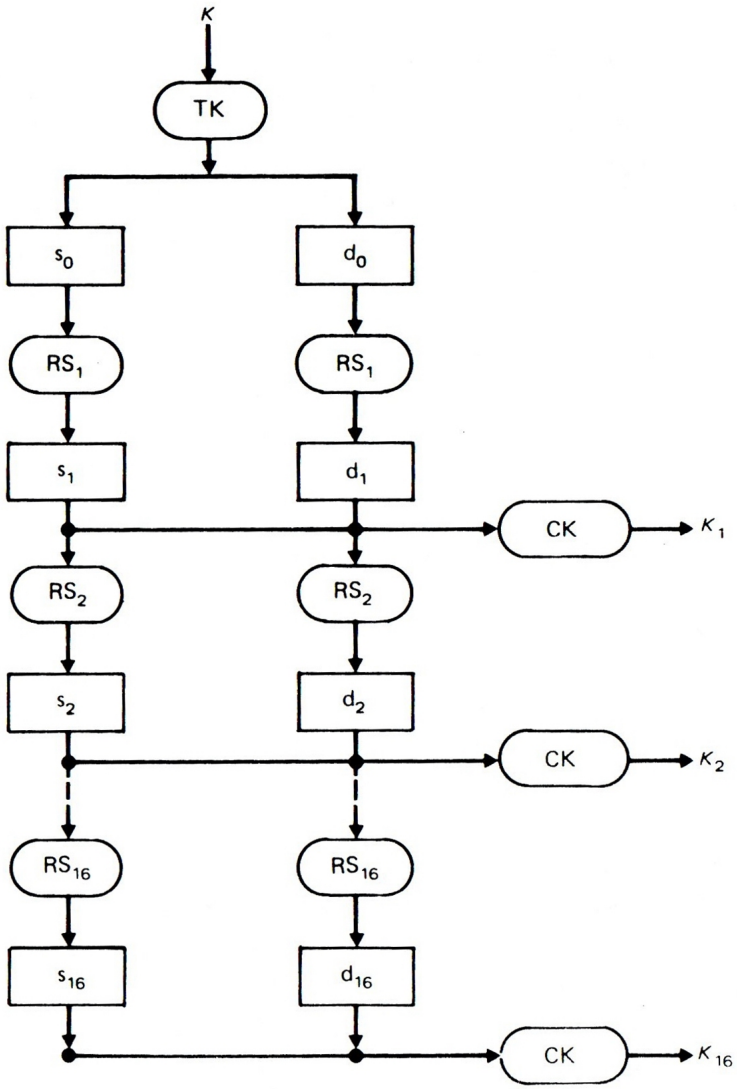
Le varie coppie $\langle s_i, d_i \rangle$ ottenute, sono filtrate e trasposte usando la matrice CK a fianco.
Di 56 bits ne rimangono solo 48.

TABELLA 4.7 *La compressione CK*

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Data Encryption Standard

Calcolo delle chiavi parziali: schema



Data Encryption Standard

Calcolo delle chiavi parziali: schema

Alla fine T_{16} viene data in pasto alla trasposizione finale TF:

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Che è inversa della trasposizione iniziale TI

Data Encryption Standard

Decifrazione

- L'algoritmo è lo stesso.
- Unica differenza: le chiavi parziali si usano in ordine invertito (prima $\langle s_{16}, d_{16} \rangle$, ultima $\langle s_1, d_1 \rangle$).

Data Encryption Standard

Decrittazione: crittanalisi differenziale

- Idea di Biham–Shamir 1990.
- Si basa sull'assunzione di aver accesso alla "macchina" con chiave fissata (ma ignota).
- Si generano molti messaggi in chiaro ottenendo i corrispondenti crittogrammi.
- Si lavora sulla differenza tra messaggi in ingresso e messaggi in uscita.
- Nel libro Wade–Trappe vedete come funziona con un DES-like (Feistel system) con tre o quattro iterazioni.
- In realtà si mostra che il metodo ha dei buoni risultati fino a 15 iterazioni.
- Ma il DES ne ha 16. Dunque i progettisti del DES sapevano quel che facevano.

Data Encryption Standard

Decrittazione: forza bruta (e HW dedicato)

- Lo spazio di 2^{56} sembrò “piccolo” fin dal principio.
- Già nel 1975 Diffie e Hellman stimarono che con circa 20 milioni di dollari si potesse costruire un calcolatore in grado di forzarlo in circa un giorno.
- Nel 1977 fu approvato come “standard”.
- Ogni 5 anni nasceva discussione circa il rinnovo di tale licenza
- Nel 1987, in barba a Kerckoffs, l’NSA propose un nuovo sistema (SW) di cui solo loro sapevano l’architettura (e garantivano forte protezione verso reverse engineering). Non andò in porto.
- Gli attacchi differenziali si dimostrarono inefficaci (e basati su un’ipotesi comunque molto forte).
- Il DES fu ricertificato anche nel 1992.

Data Encryption Standard

Decrittazione: forza bruta (e HW dedicato)

- Nel 1996 maturarono due (*) linee di attacco al DES:
  - Parallelismo massivo ditribuito (forza bruta a basso costo)
  - Architettura ad hoc (Michael Wiener, Bell, 1993)
- Nel 1997 i “rivali” della RSA (di cui parleremo a lungo) misero in palio 10000 $ per chi sarebbe stato in grado di trovare la chiave di un messaggio creato e cifrato da loro.
- Rocke Verser preparò un programma in grado di distribuire il lavoro su PC vari collegati in Internet. La partecipazione era invogliata economicamente (si sarebbe ricevuto il 40% della vincita nel caso il proprio PC fosse quello che trovava la chiave).

Data Encryption Standard

Decrittazione: forza bruta (e HW dedicato)

- Il messaggio era: Strong cryptography makes the world a safer place.
- Furono necessari 5 mesi (ovviamente con carico piuttosto irregolare). Fu esplorato circa il 25% dello spazio.
- L'anno successivo (1998) l'RSA ripetè il concorso, con il messaggio Many hands make light work.
- In questo caso la chiave fu trovata dopo 39 giorni, visitando l'85% dello spazio di ricerca.

Data Encryption Standard

Decrittazione: forza bruta (e HW dedicato)

- Nel 1998 la EFF (Electronic Frontier Foundation) mise a disposizione un budget di 200.000$ ad un team per lo sviluppo del DES cracker.
- Ad un PC tradizionale furono associati circa 1500 chips dedicati (a 40MHz) che emulavano il DES.
- Ogni chip aveva 24 unità di ricerca ed era delegato ad una porzione delle chiavi possibili.
- Un testo di 16 Bytes era diviso in due blocchi da 8 Bytes da analizzarsi in cascata.

Data Encryption Standard

Doppio e Triplo DES

- La chiave era promettente se sui primi 8 Bytes generava un testo in cui i simboli erano numero, lettera, o simbolo di punteggiatura.
- Se la chiave era promettente, si ripeteva sulla seconda metà.
- Numeri: $\frac{69}{256} \simeq \frac{1}{4}$ sono gli ASCII buoni. Dunque la chance che sia promettente è $\sim \left(\frac{1}{4}\right)^8 = 2^{-16}$. Combinando con la seconda metà si arriva a 2^{-32}. Rimangono per la CPU principale $2^{(56-32)} = 2^{24} \approx 16$ milioni di tentativi.
- A 100 tentativi al secondo sono 40 ore nel caso peggiore.

Data Encryption Standard

Doppio e Triplo DES

- Urge allora abbandonare il DES. Fu suggerito di usare il doppio DES o il triplo DES.
- Il doppio DES usa due codifiche consecutive $c = DES_{k_1}(DES_{k_2}(m))$
- La chiave ora è lunga 112 bits. Inoltre la combinazione di due codifiche non dà una singola codifica diversa (cosa che succede con altri codici).
- Tuttavia, avendo una copia m, c, la decrittazione è poco più complicata del singolo DES ($\sim 2^{57}$) (meet-in-the-middle attack).
- Il triplo DES si può ottenere in diversi modi, anche usando una chiave come *inversa*. Anche con il meet-in-the-middle attack lo spazio rimane (per oggi) considerevole ($\sim 2^{113}$).

Spazio per un laboratorio avanzato, magari usando GPU/CUDA invece dei chip dedicati.

Data Encryption Standard

Doppio e Triplo DES

- Urge allora abbandonare il DES. Fu suggerito di usare il doppio DES o il triplo DES.
- Il doppio DES usa due codifiche consecutive $c = DES_{k_1}(DES_{k_2}(m))$
- La chiave ora è lunga 112 bits. Inoltre la combinazione di due codifiche non dà una singola codifica diversa (cosa che succede con altri codici).
- Tuttavia, avendo una copia m, c, la decrittazione è poco più complicata del singolo DES ($\sim 2^{57}$) (meet-in-the-middle attack).
- Il triplo DES si può ottenere in diversi modi, anche usando una chiave come *inversa*. Anche con il meet-in-the-middle attack lo spazio rimane (per oggi) considerevole ($\sim 2^{113}$).

Spazio per un laboratorio avanzato, magari usando GPU/CUDA invece dei chip dedicati