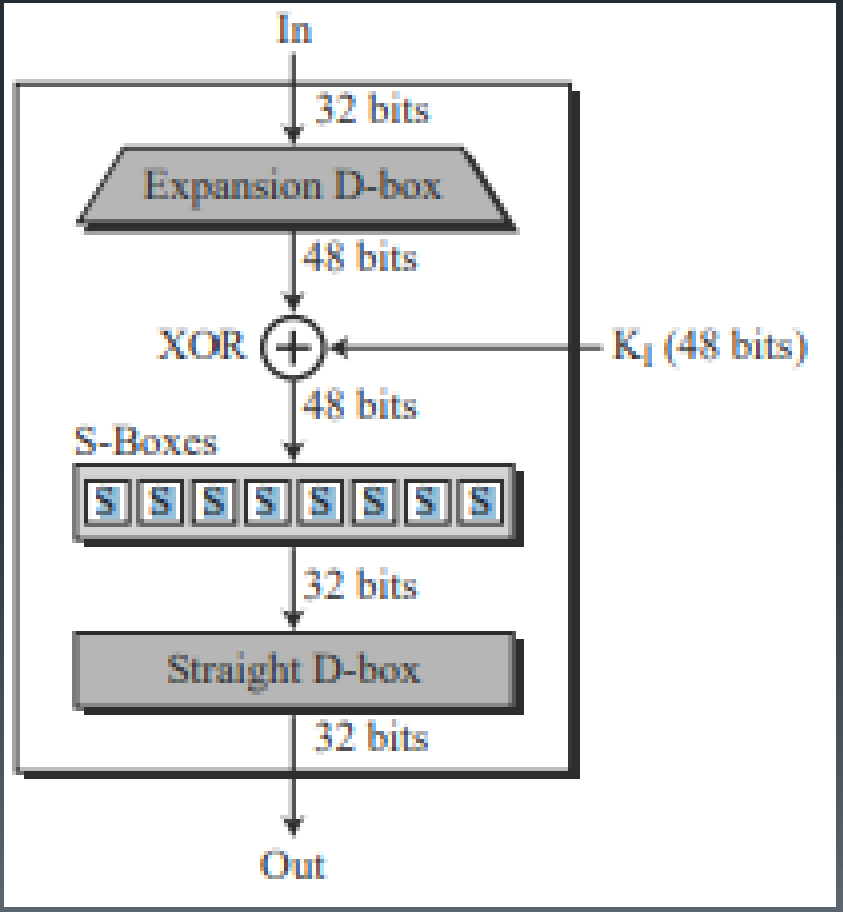
Data Encryption Standard



Mirco Bastari
Thomas De Alcubierre
Gabriele Marani
Nicola Sampaolesi
Federico Uncini

Che cos’è?

Il Data Encryption Standard (DES) è un sistema di cifratura a blocchi a chiave simmetrica, ciò significa che è in grado di operare solo su un gruppo di bit di lunghezza finita organizzati in un blocco.

Essendo un sistema a chiave simmetrica si ha una chiave per eseguire la crittazione del blocco e si usa la stessa per decifrarlo.

Storia

1973

Il National Bureau Standard (NBS) pubblica un bando in cui richiede un algoritmo di cifratura. Lo stesso anno, la IBM propone una prima versione del DES. In seguito la NSA lo certifica proponendo una variazione della lunghezza della chiave da 128 bit a 56 bit.

Storia

1976

Il DES fu scelto dal Federal Information Processing Standard (FIPS) per il governo degli Stati Uniti d'America e in seguito divenne di utilizzo internazionale.

1998

Il DES viene violato.

2000

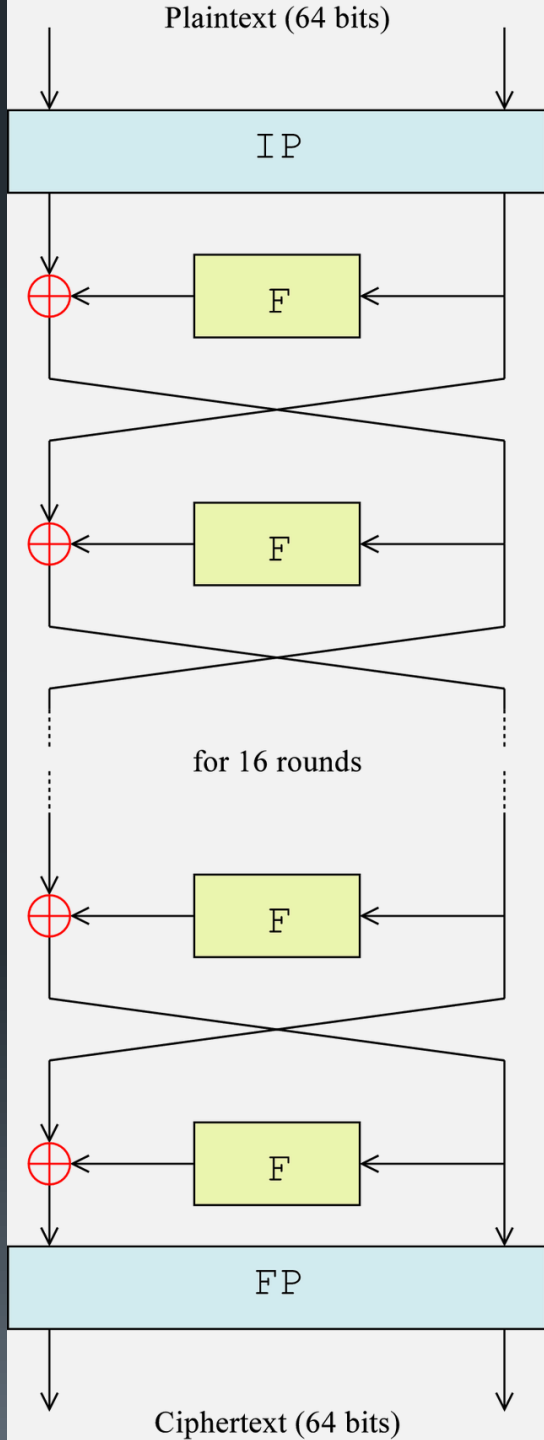
NBS sceglie il successore di DES:

Advanced Encryption Standard (AES).

Meccanismo - Intro

Input Blocco di 64 bit

Se il testo da cifrare è più lungo di 64 bit, deve essere diviso in blocchi da 64.



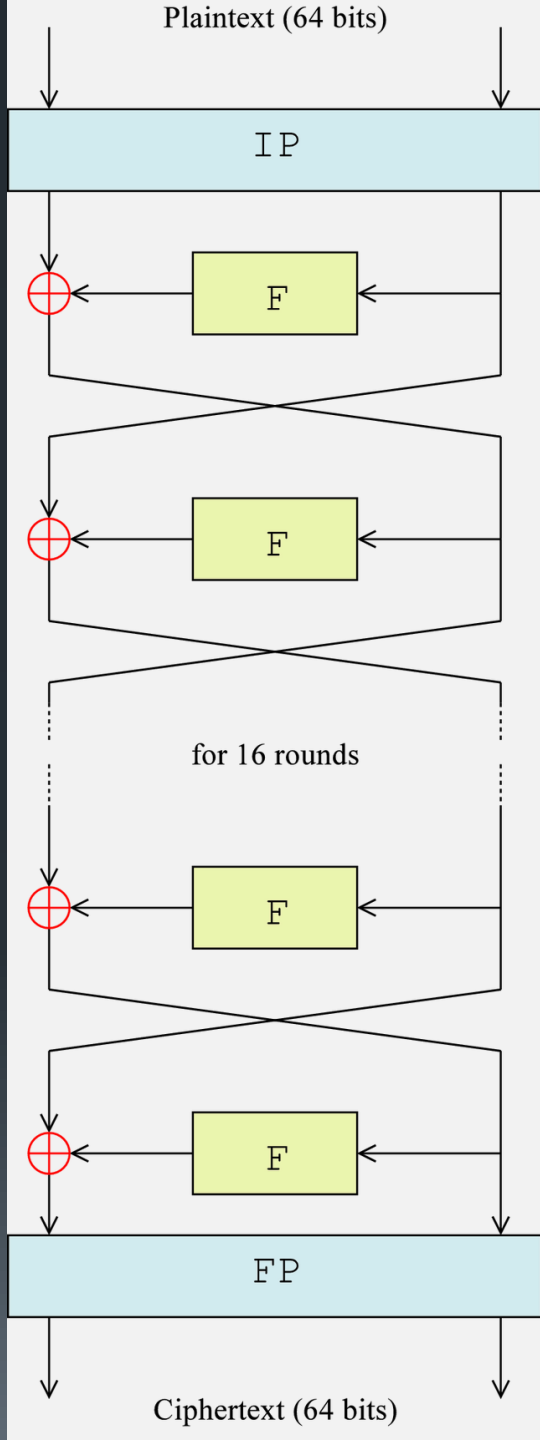
Meccanismo - Intro

IP Permutazione iniziale

FP Permutazione finale

I due blocchi di permutazione svolgono solamente la funzione di appesantire l'implementazione software dell'algoritmo rendendo più difficile forzare i messaggi criptati.

Questi blocchi svolgono funzione opposta.

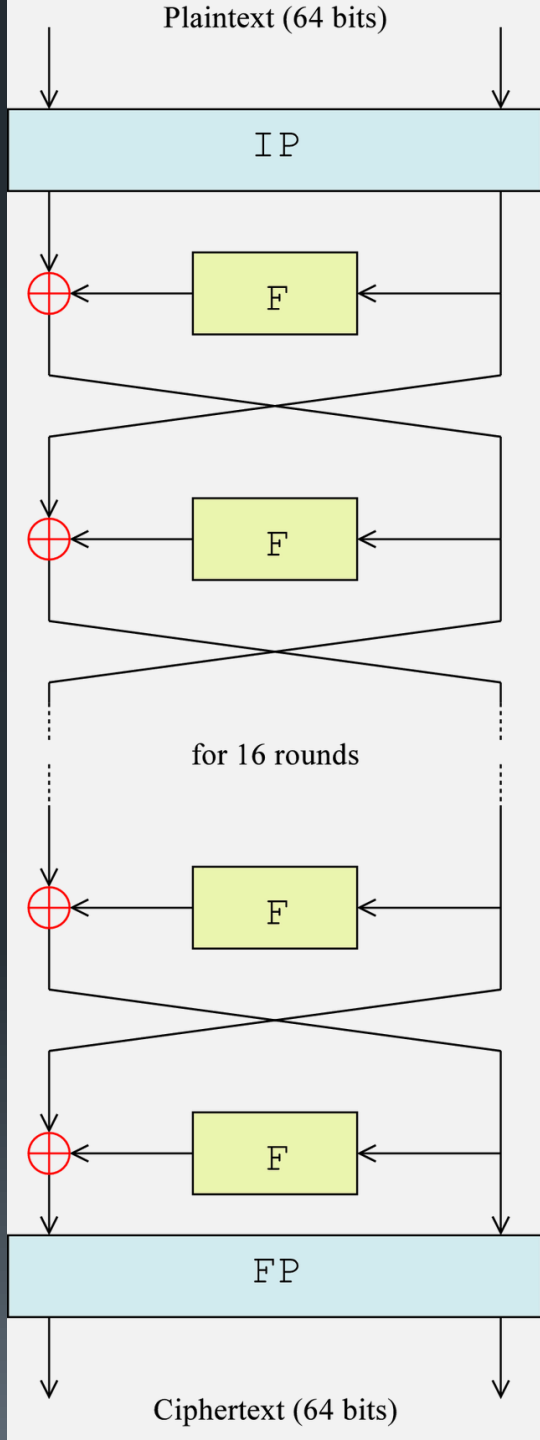


Meccanismo - Intro

F Feistel Cipher

I blocchi con scritto F svolgono la funzione di cifratura del messaggio.

La chiave deve essere di 64 bit (56 bit utili, 8 bit di parità). Nello schema di destra essa non è rappresentata.

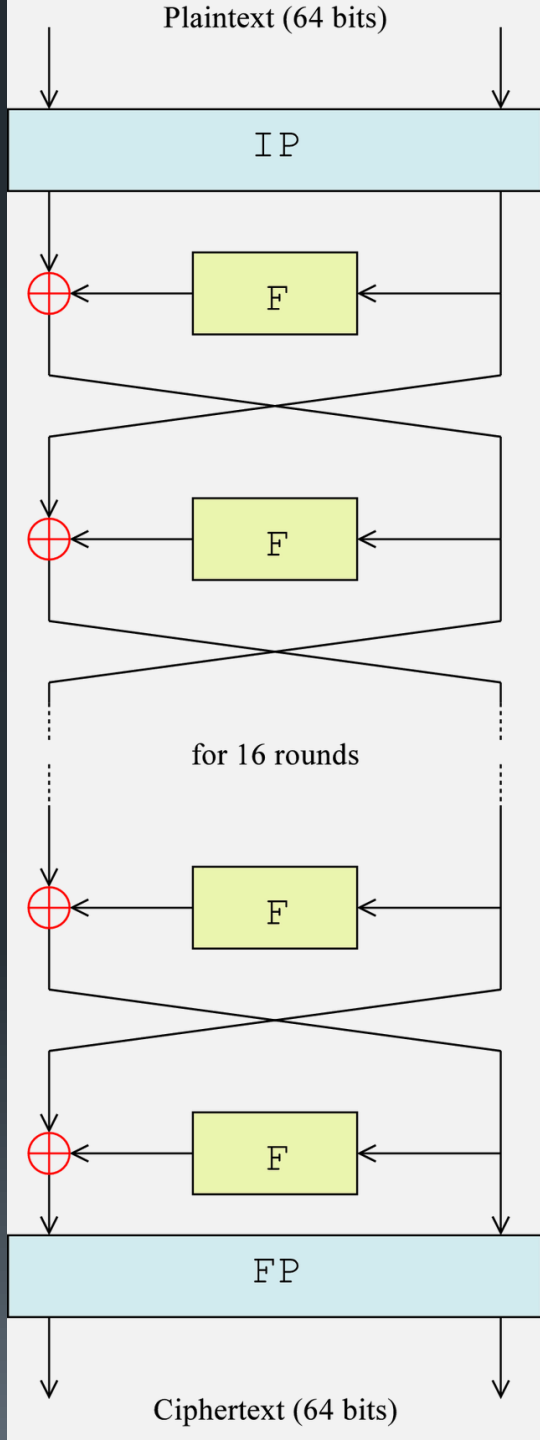


Meccanismo

- Il chiaro inizialmente di 64 bit viene diviso in due sotto-blocchi da 32 bit.
- Tali blocchi vengono poi alternativamente processati nella rete di Feistel.

Chiaro: GABRIELE

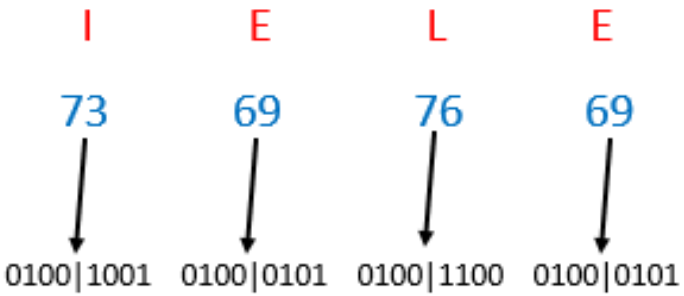
Suddivisione a 32bit: GABR | IELE

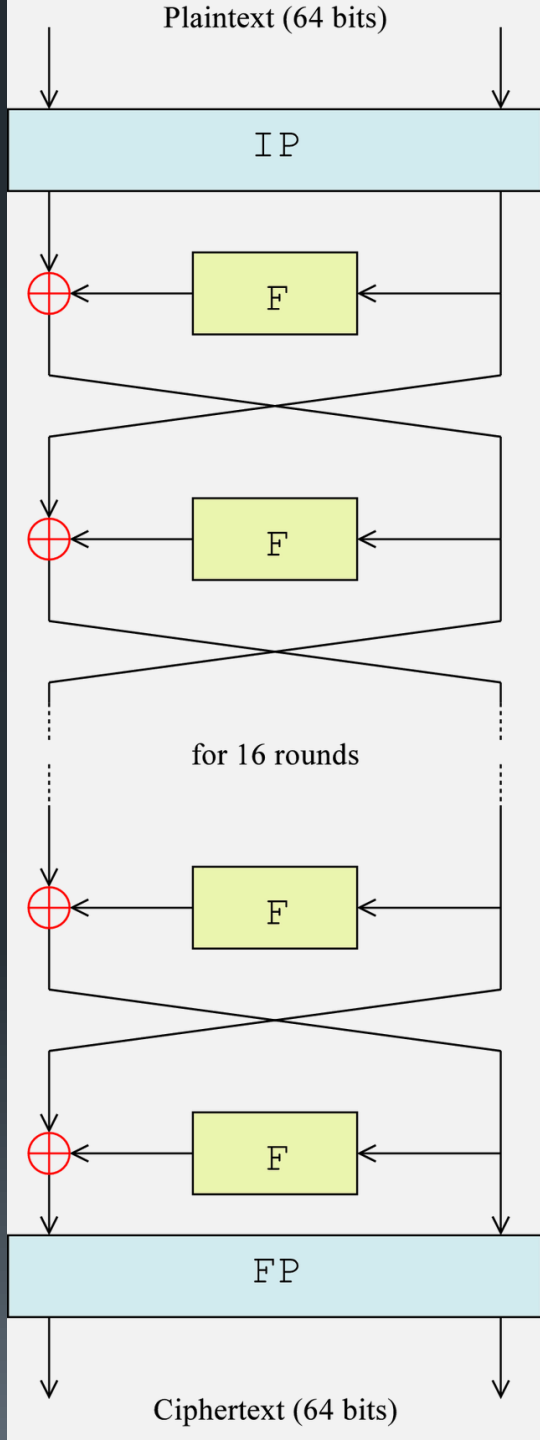


Meccanismo

- Si prende il secondo blocco di 32 bit e mediante la tabella ASCII si convertono i caratteri del chiaro in binario.
- Il risultato viene suddiviso in gruppi da 4 bit ciascuno.

DEC	HEX	CHARACTER
64	0x40	@
65	0x41	A
66	0x42	B
67	0x43	C
68	0x44	D
69	0x45	E
70	0x46	F
71	0x47	G
72	0x48	H
73	0x49	I
74	0x4A	J
75	0x4B	K
76	0x4C	L
77	0x4D	M
78	0x4E	N
79	0x4F	O
80	0x50	P
81	0x51	Q
82	0x52	R
83	0x53	S
84	0x54	T
85	0x55	U
86	0x56	V
87	0x57	W
88	0x58	X
89	0x59	Y
90	0x5A	Z

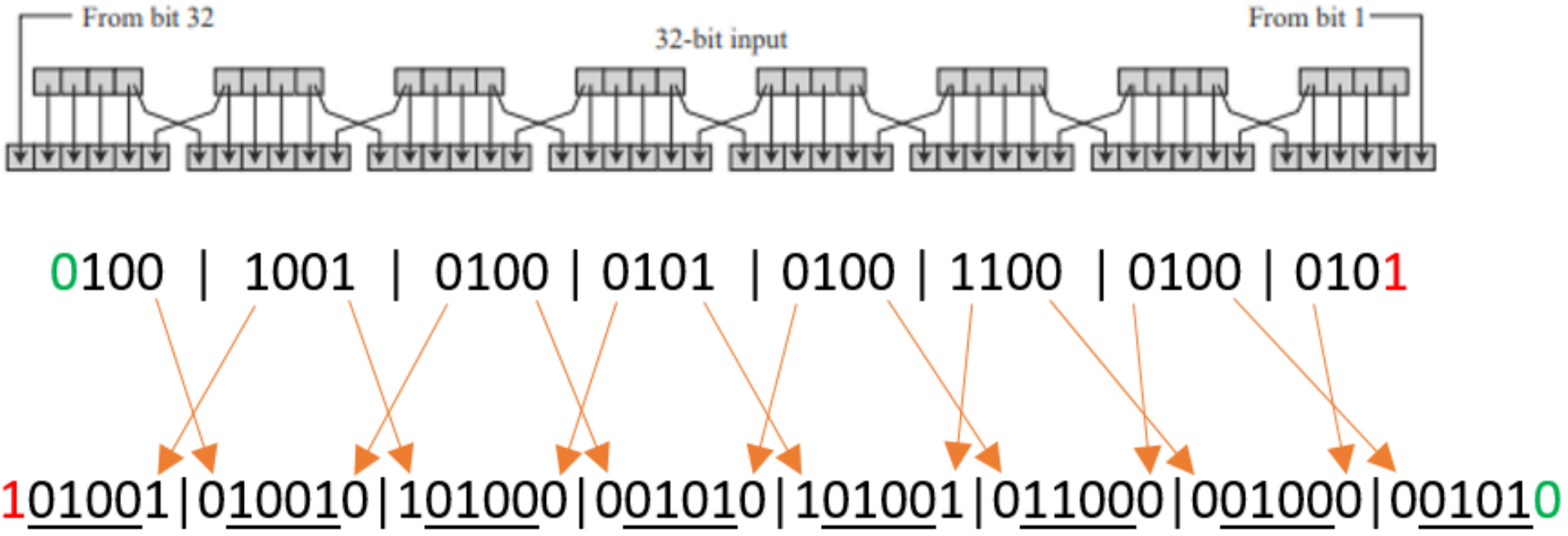




Meccanismo

- Inizio del blocco del Feistel Cipher.
- **Expansion D-box (Permutazione di espansione)**

Il blocco di 32 bit viene espanso a 48 bit.



Meccanismo

- Per semplicità si suppone di avere una chiave di 48 bit. (Passa al generatore delle chiavi)
- La chiave viene utilizzata per realizzare lo XOR con la precedente sequenza di bit.

Chiave: PAROLA

P	A	R	O	L	A
80	65	82	79	76	65
01010000	01000001	01010010	01001111	01001100	01000001

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

XOR

010100000100000101010010010011110100110001000001 ⊕

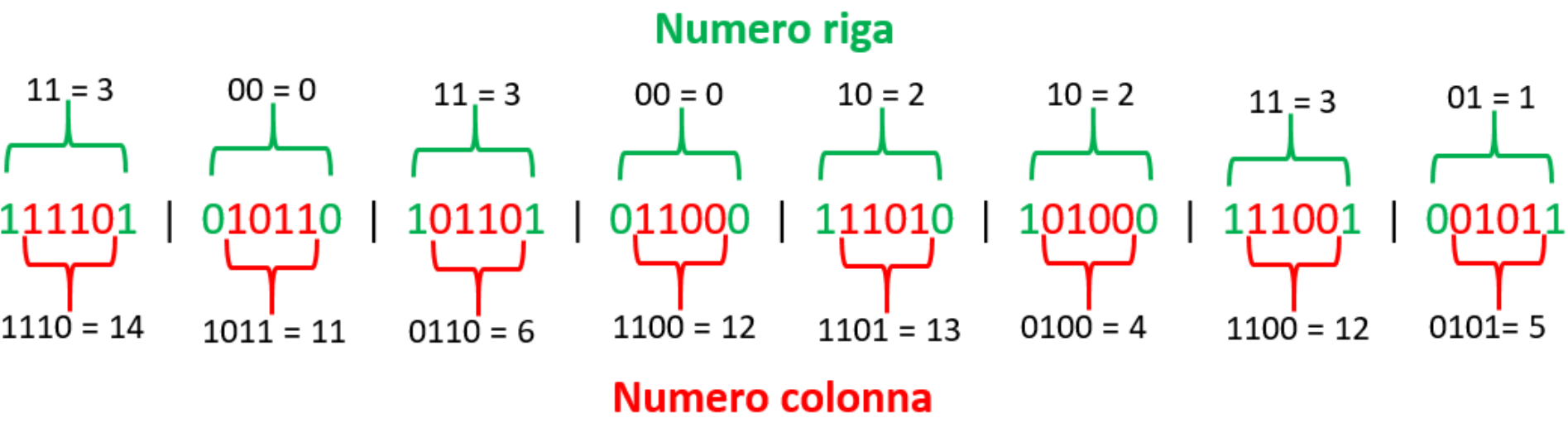
101001010010101000001010101001011000001000001010 =

111101010110101101011000111010101000111001001011

DEC	HEX	CHARACTER
64	0x40	@
65	0x41	A
66	0x42	B
67	0x43	C
68	0x44	D
69	0x45	E
70	0x46	F
71	0x47	G
72	0x48	H
73	0x49	I
74	0x4A	J
75	0x4B	K
76	0x4C	L
77	0x4D	M
78	0x4E	N
79	0x4F	O
80	0x50	P
81	0x51	Q
82	0x52	R
83	0x53	S
84	0x54	T
85	0x55	U
86	0x56	V
87	0x57	W
88	0x58	X
89	0x59	Y
90	0x5A	Z

Meccanismo

- Il risultato viene diviso in gruppi di 6 bit.
- I numeri iniziale e finale determinano la riga.
- I numeri centrali determinano la colonna.



Meccanismo

- I numeri trovati vengono utilizzati per identificare una cella da ogni tabella S-box (Sostitution).
- Il gruppo **N** di 6 bit è associato la S-box **N**.

Table 6.3 *S-box 1*

	0	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	10	03	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

Table 6.4 *S-box 2*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10
1	03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05
2	00	14	07	11	10	04	13	01	05	08	12	06	09	03	02	15
3	13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09

Table 6.5 *S-box 3*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	10	00	09	14	06	03	15	05	01	13	12	07	11	04	02	08
1	13	07	00	09	03	04	06	10	02	08	05	14	12	11	15	01
2	13	06	04	09	08	15	03	00	11	01	02	12	05	10	14	07
3	01	10	13	00	06	09	08	07	04	15	14	03	11	05	02	12

Table 6.6 *S-box 4*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	07	13	14	03	00	6	09	10	1	02	08	05	11	12	04	15
1	13	08	11	05	06	15	00	03	04	07	02	12	01	10	14	09
2	10	06	09	00	12	11	07	13	15	01	03	14	05	02	08	04
3	03	15	00	06	10	01	13	08	09	04	05	11	12	07	02	14

Table 6.7 *S-box 5*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	02	12	04	01	07	10	11	06	08	05	03	15	13	00	14	09
1	14	11	02	12	04	07	13	01	05	00	15	10	03	09	08	06
2	04	02	01	11	10	13	07	08	15	09	12	05	06	03	00	14
3	11	08	12	07	01	14	02	13	06	15	00	09	10	04	05	03

Table 6.8 *S-box 6*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	12	01	10	15	09	02	06	08	00	13	03	04	14	07	05	11
1	10	15	04	02	07	12	09	05	06	01	13	14	00	11	03	08
2	09	14	15	05	02	08	12	03	07	00	04	10	01	13	11	06
3	04	03	02	12	09	05	15	10	11	14	01	07	10	00	08	13

Table 6.9 *S-box 7*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4	11	2	14	15	00	08	13	03	12	09	07	05	10	06	01
1	13	00	11	07	04	09	01	10	14	03	05	12	02	15	08	06
2	01	04	11	13	12	03	07	14	10	15	06	08	00	05	09	02
3	06	11	13	08	01	04	10	07	09	05	00	15	14	02	03	12

Table 6.10 *S-box 8*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13	02	08	04	06	15	11	01	10	09	03	14	05	00	12	07
1	01	15	13	08	10	03	07	04	12	05	06	11	10	14	09	02
2	07	11	04	01	09	12	14	02	00	06	10	10	15	03	05	08
3	02	01	14	07	04	10	8	13	15	12	09	09	03	05	06	11

Meccanismo

- I numeri trovati vengono convertiti in binario.
- Si riprende la prima parte del chiaro e la si trasforma in binario.
- Si applica lo **XOR** tra i due risultati.

06 13 08 11 03 02 14 03

0110 1101 1000 1011 0011 0010 1110 0011

01101101100010110011001011100011 ⊕

01000111010000010100001001010010 =

00101010110010100111000010100001

DEC	HEX	CHARACTER
64	0x40	@
65	0x41	A
66	0x42	B
67	0x43	C
68	0x44	D
69	0x45	E
70	0x46	F
71	0x47	G
72	0x48	H
73	0x49	I
74	0x4A	J
75	0x4B	K
76	0x4C	L
77	0x4D	M
78	0x4E	N
79	0x4F	O
80	0x50	P
81	0x51	Q
82	0x52	R
83	0x53	S
84	0x54	T
85	0x55	U
86	0x56	V
87	0x57	W
88	0x58	X
89	0x59	Y
90	0x5A	Z

G A B R

71 65 66 82

0100|0111 0100|0001 0100|0010 0101|0010

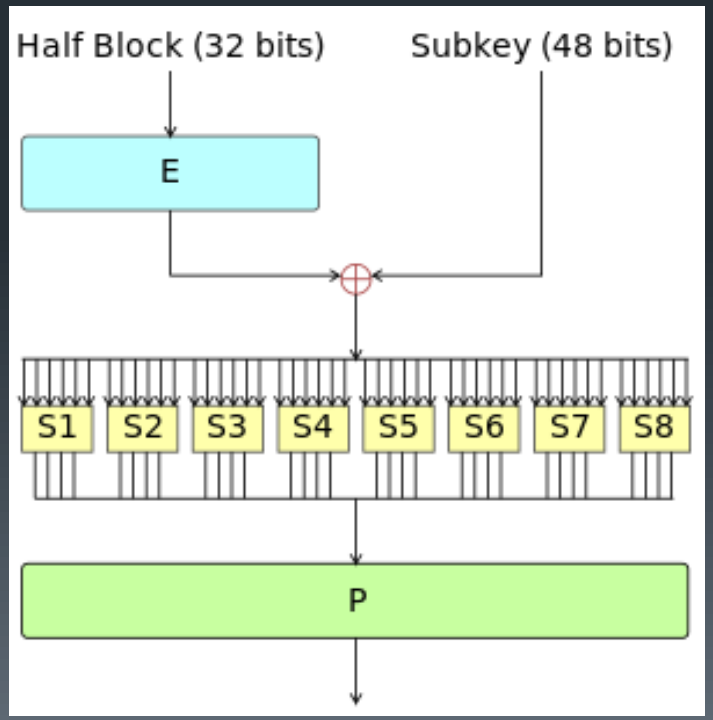
01000111010000010100001001010010

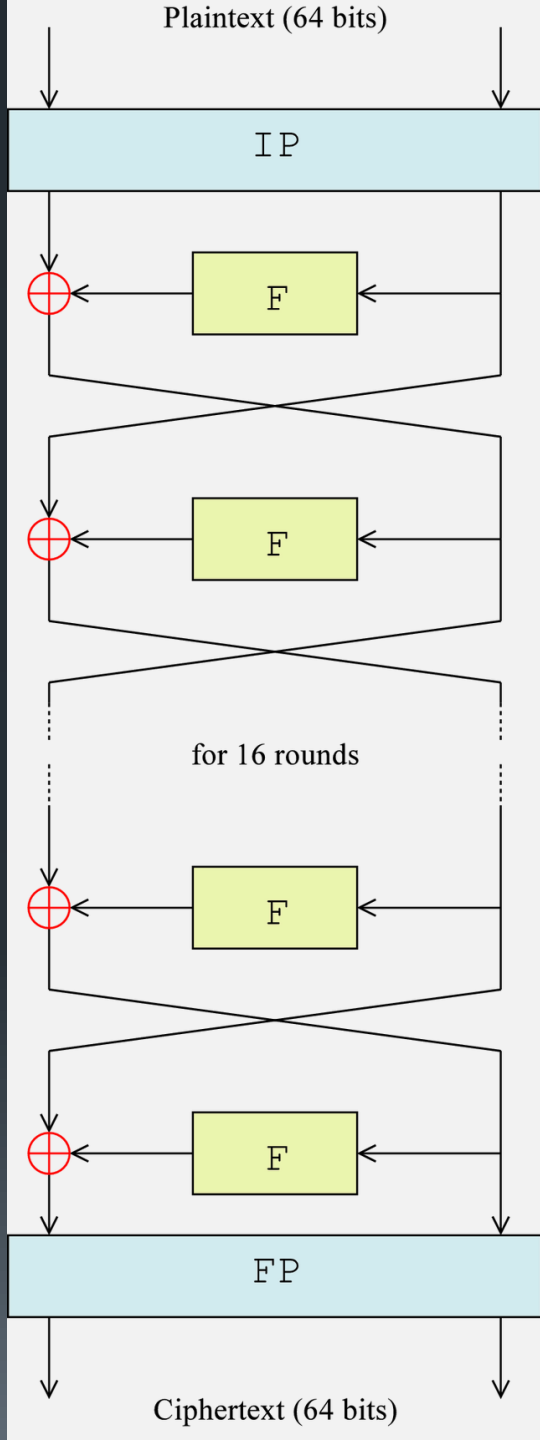
Meccanismo

- Una volta trovato il risultato:

00101010110010100111000010100001

Si ripete per 16 volte, ripartendo dal **Expansion D-box** (vedi slide 10).

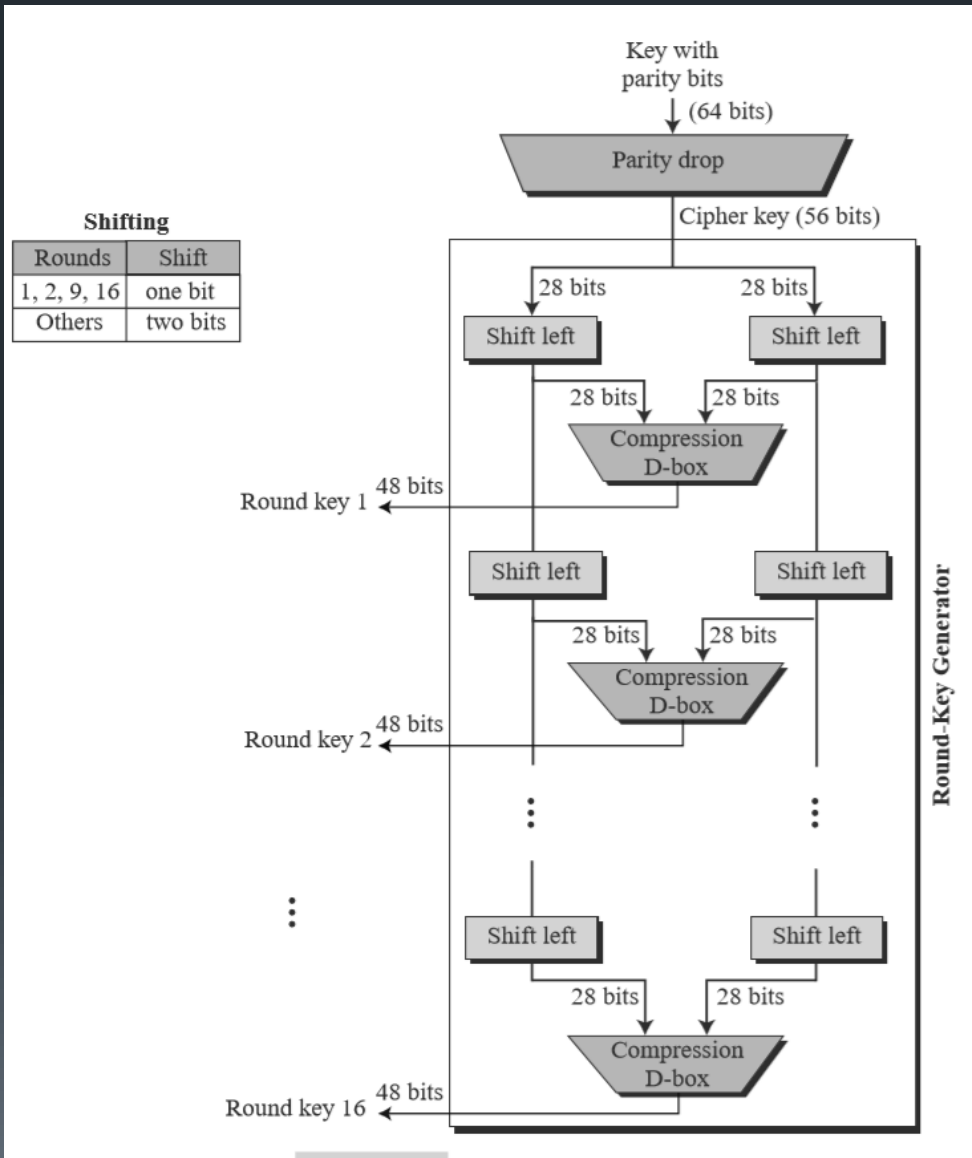




Meccanismo – Round-Key Generator

- Ad ogni round viene generata una nuova round-key (chiave temporanea) da utilizzare per lo XOR della slide 11.

- La chiave iniziale è di 64 bit.
- Essa perde gli 8 bit di parità.
- Viene divisa in due blocchi di 28 bit.
- Ad ogni round i due blocchi subiscono uno shift a sinistra.
- I due risultati vengono compressi mediante il D-box.



Shifting

Rounds	Shift
1, 2, 9, 16	one bit
Others	two bits

Meccanismo – Round-Key Generator

La compressione rispetta la seguente tabella.

Ogni numero identifica un bit dell'input.

L'output è dato dalla concatenazione in ordine dei bit identificati da ogni cella.

14	17	11	24	01	05	03	28
15	06	21	10	23	19	12	04
26	08	16	07	27	20	13	02
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

Approfondimento

Crittografia Simmetrica & Crittografia Asimmetrica

Crittografia simmetrica

La chiave per crittografare il messaggio è la stessa utilizzata per decrittarlo.

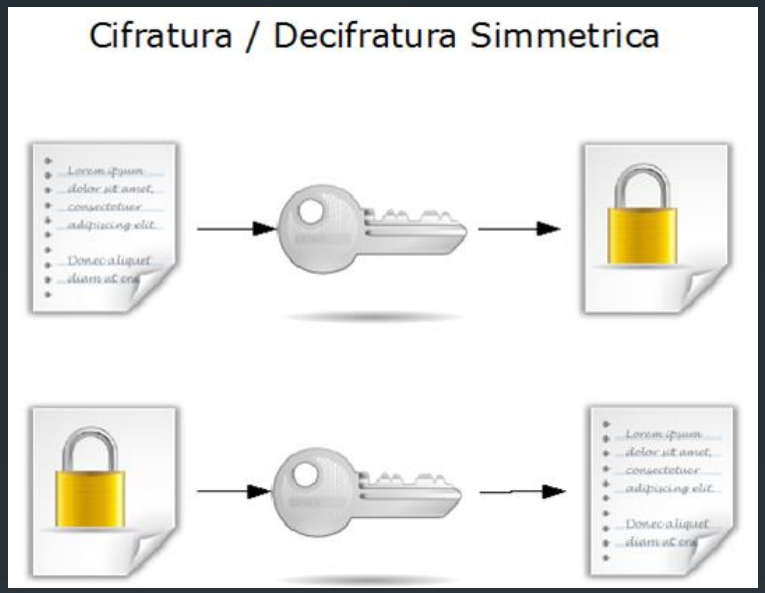
Tale meccanismo richiede che le parti si conoscano e abbiano un metodo sicuro di chiavi condivise.

La crittografia simmetrica non è perciò applicabile nel caso in cui le entità non si siano mai incontrate.

Funzionamento

Per inviare un messaggio da un mittente a una destinatario bisogna:

- Il mittente utilizzi la chiave privata con la quale codificherà il messaggio;
- Il destinatario decodificherà il messaggio codificato tramite la stessa chiave privata.



Crittografia asimmetrica

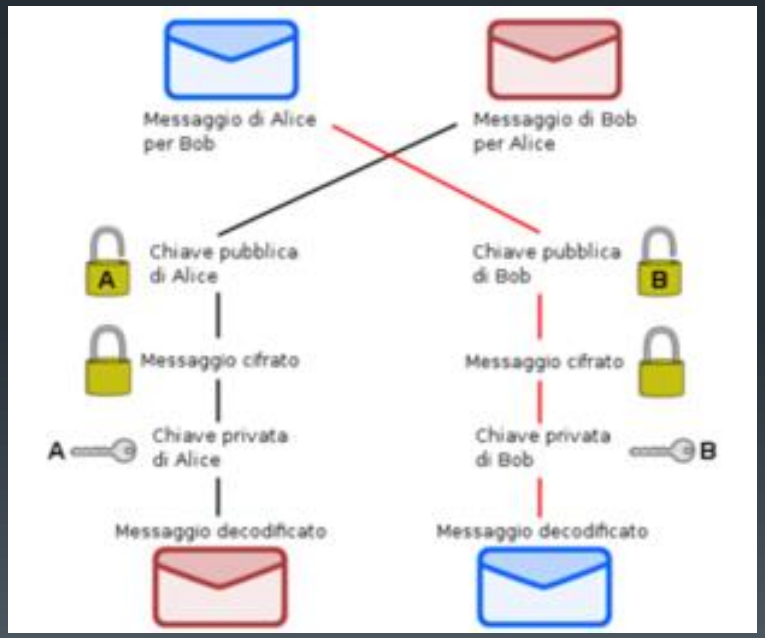
La crittografia asimmetrica o a coppia di chiavi utilizza due chiavi distinte:

- Chiave pubblica, distribuita pubblicamente;
- Chiave privata, personale e segreta;

Funzionamento

Il meccanismo a crittografia asimmetrica prevede l'utilizzo della chiave pubblica per cifrare il messaggio e quella privata per decrittarlo.

Funzionamento



- Il mittente ottiene la chiave pubblica del destinatario con la quale poi codifica il messaggio;
- Il destinatario riceve il messaggio e lo decritta tramite la propria chiave privata.

Vantaggi & Svantaggi

- La cifratura simmetrica è più veloce della crittografia asimmetrica, ma non è molto utile per l'e-commerce (mittente e destinatario non si conoscono);
- La cifratura simmetrica è più sicura a parità di lunghezza delle chiavi;
- La crittografia asimmetrica deve utilizzare chiavi più lunghe per raggiungere un livello di sicurezza adeguato e per questo è soggetta a rallentare il processo.

Fonti

- WikiPedia
- Università Rome Tre
- Cleveland State University
- Tutorialspoint
- Crittologia.EU