

Cryptography and Network/Information Security

Block Ciphers and Data Encryption Standard

Outline

- Cifrari a blocco e a flusso
- Progetto e struttura dei cifrari Feistel
- DES
 - Dettagli
 - Resistenza
- Criptanalisi differenziale e lineare
- Principi di progetto dei cifrari a blocco

Cifrari a blocco moderni

- Fino ad ora trattati sistemi classici
- Oggi si usano cifrari a blocco moderni
- Algoritmi crittografici usati più largamente
- Usati per fornire servizi di confidenzialità e di autenticazione
- Data Encryption Standard (DES)
- Usato per illustrare i principi di progetto dei cifrari a blocco

Cifrari a blocco e a flusso

- Cifrari a blocco elaborano i messaggi dopo averli suddivisi in blocchi ognuno dei quali è criptato/decriptato
- Una sorta di operazione su un grande carattere

64 bit o più

- Cifrari a flusso elaborano i messaggi un bit o byte alla volta quando li criptano/decriptano
- Molti cifrari attuali sono cifrari a blocco
- Campo di applicazione più ampio di quelli a flusso

Principi dei cifrari a blocco

Maggior parte dei cifrari a blocco attuali basati sulla architettura del *Cifrario di Feistel*

Struttura necessaria per l'esigenza di decriptazione del ciphertext efficientemente

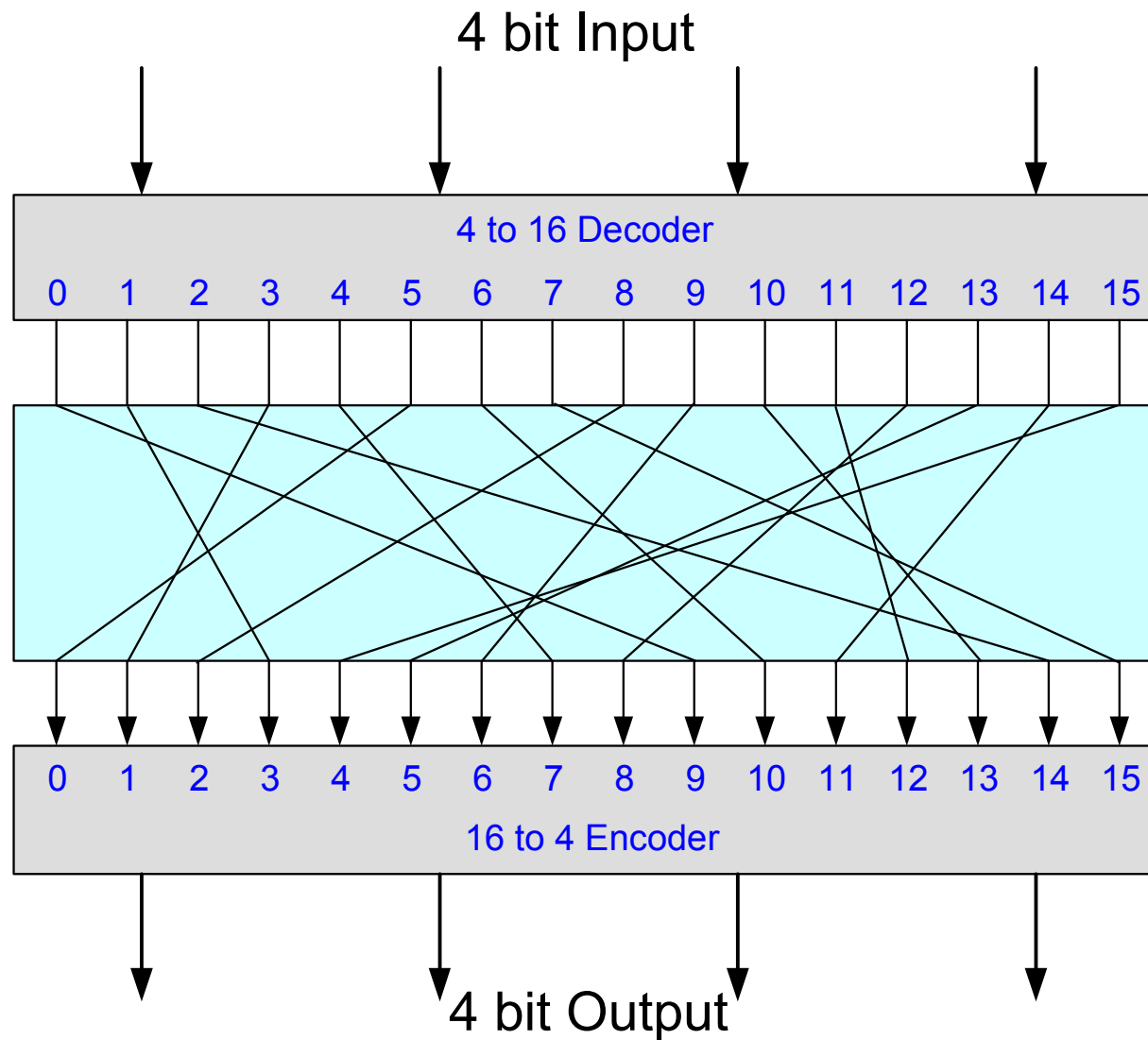
Cifrari a blocco si comportano come una sostituzione oltremodo grande

Per operare su blocchi da 64 bit servirebbe una tabella di sostituzione con 2^{64} entries

Questa tabella viene creata usando dei blocchi più piccoli

Si usa l'idea del cifrario a prodotto

Cifrario a blocco ideale



Shannon e i cifrari a Substitution-Permutation

- Nel 1949 C. Shannon introdusse l'idea delle reti a sostituzione-permutazione (S-P network)
- Base dei cifrari a blocco moderni
- S-P networks basate sulle due operazioni crittografiche primitive già viste
 - *Sostituzione (S-box)*
 - *Permutazione (P-box)*
- Si ha confusione e diffusione dei messaggi e delle chiavi

Confusione e Diffusione

- Serve che i cifrari oscurino completamente le proprietà statistiche del plaintext
- Condizione soddisfatta da one-time pad ma in modo non praticabile
- C. Shannon suggerì di combinare le due operazioni crittografiche primitive di sostituzione (S-box) e permutazione (P-box) per avere:
 - ❖ **Diffusione** Diffondere la struttura statistica del plaintext sulla massa del ciphertext
 - ❖ **Confusione** Rendere la relazione tra plaintext e ciphertext quanto più complessa è possibile

Struttura del cipher Feistel

Horst Feistel ha concepito il cipher col suo nome

Basato sul concetto di cifrario invertibile a prodotto

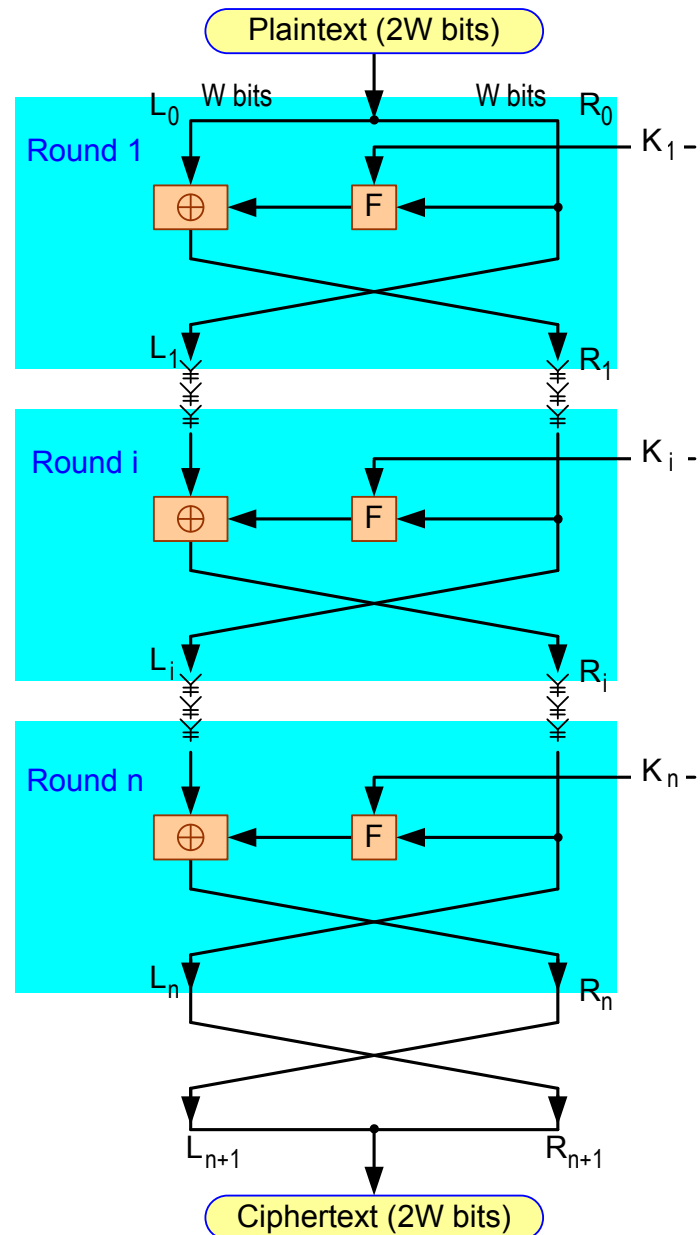
Il blocco d'ingresso diviso in parti due uguali, dette L e R elaborate in cicli n nel corso dell'algoritmo

Elaborazione tramite molte iterazioni che:

- Sviluppano una sostituzione sulla metà di sinistra (L)
- Usano funzione iterata sulla metà destra (R) e la subkey
- Quindi permutazione scambiando le due metà

Implementazione del concetto di Shannon (S-P net)

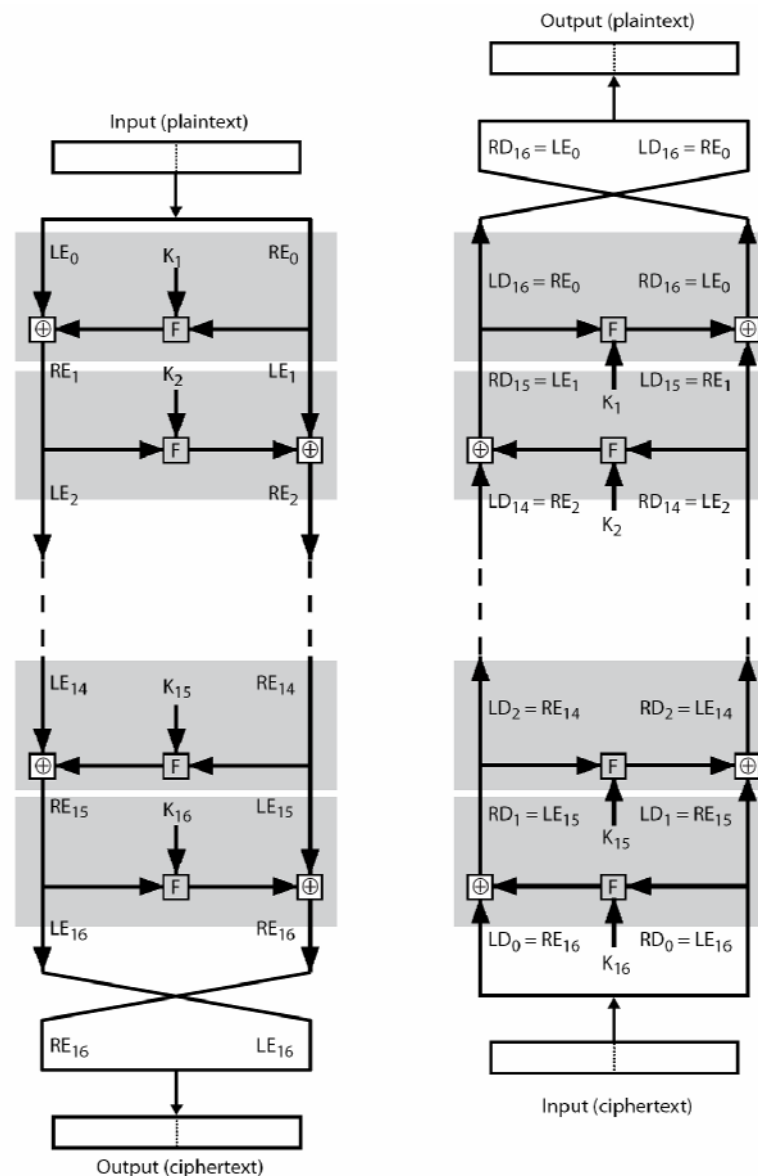
Struttura del cipher Feistel



Parametri del cipher Feistel

- ❖ Dimensione del blocco
- ❖ Dimensione della chiave
- ❖ Numero di cicli (rounds)
- ❖ Algoritmo di generazione delle subkey
- ❖ Funzione di ciclo
- ❖ Encryption/Decryption veloce via software
- ❖ Facilità di analisi

Decriptazione del cipher Feistel



Data Encryption Standard (DES)

- Cipher a blocco più ampiamente usato nel mondo
- Adottato dal 1977 dall'NBS (ora NIST)
- Reaffirmed 25 Oct 1999

Come FIPS (Federal Information Processing Standard) PB 46-3

- Cifra blocchi di dati lunghi 64 bit usando key da 64 bit
- Uso molto diffuso
- Notevoli controversie circa la sua sicurezza

Storia del DES

- ❖ Sviluppo a partire dal cipher Lucifer (IBM)
 - Lavoro svolto dal team guidato da H. Feistel nei tardi anni 60
 - Uso di blocchi di dati lunghi 64 bit con key da 128 bit
- ❖ Successivamente risviluppato come cipher commerciale sulla spinta della NSA e altri
- ❖ Nel 1973 l'NBS emette una richiesta di proposte per un cipher nazionale standard
- ❖ IBM presentò un verione rivista di Lucifer che venne accettata come il DES

Controversia sul progetto DES

- ❖ Lo standard DES è pubblico
- ❖ Si è avuta però una notevole polemica circa il progetto
 - A proposito della key da 56 bit (vs 128-bit di Lucifer)
 - Perché i criteri di progetto erano classificati
- ❖ Nel tempo si è visto che il progetto era appropriato
- ❖ L'uso del DES si è diffuso
 - Specialmente nelle applicazioni finanziarie
 - Ancora standardizzato per l'uso in applicazioni legacy

Definizione ufficiale

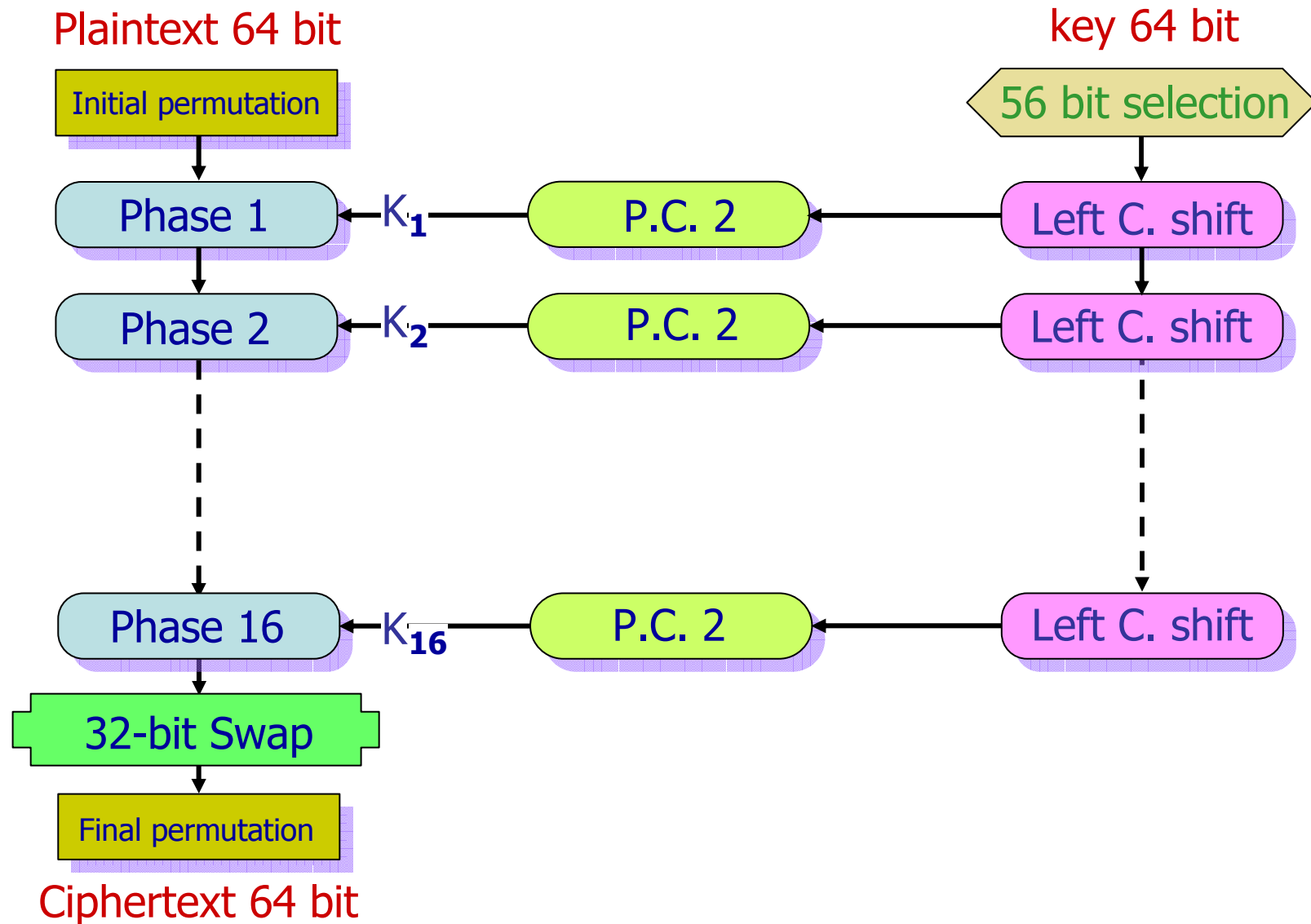
The algorithm is designed to encipher and decipher blocks of data consisting of 64 bits under control of a 64-bit key.

Deciphering must be accomplished by using the same key as for enciphering, but with the schedule of addressing the key bits altered so that the deciphering process is the reverse of the enciphering process.

A block to be enciphered is subjected to an initial permutation ***IP***, then to a complex key-dependent computation and finally to a permutation which is the inverse of the initial permutation ***IP*⁻¹**.

The key-dependent computation can be simply defined in terms of a function ***f***, called the cipher function, and a function ***KS***, called the key schedule.

Overview della Encryption DES



Initial Permutation (IP)

- Primo passo della elaborazione dei dati
- La IP rimescola i 64 bit in ingresso
- Bit pari nella metà LH, bit dispari nella metà RH
- Struttura molto regolare (facile in hardware)
- Esempio:

IP(675a6967 5e5a6b5a) = ffb2194d 004df6fb

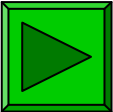
Initial Permutation (IP)

❖ Definita dalla tabella:

								→
↓	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

} Bit di posto pari

} Bit di posto dispari



L'output della Initial Permutation ha il bit 58 come suo primo bit, il bit 50 come secondo bit, e così via fino al bit 7 come ultimo bit

Struttura del ciclo DES

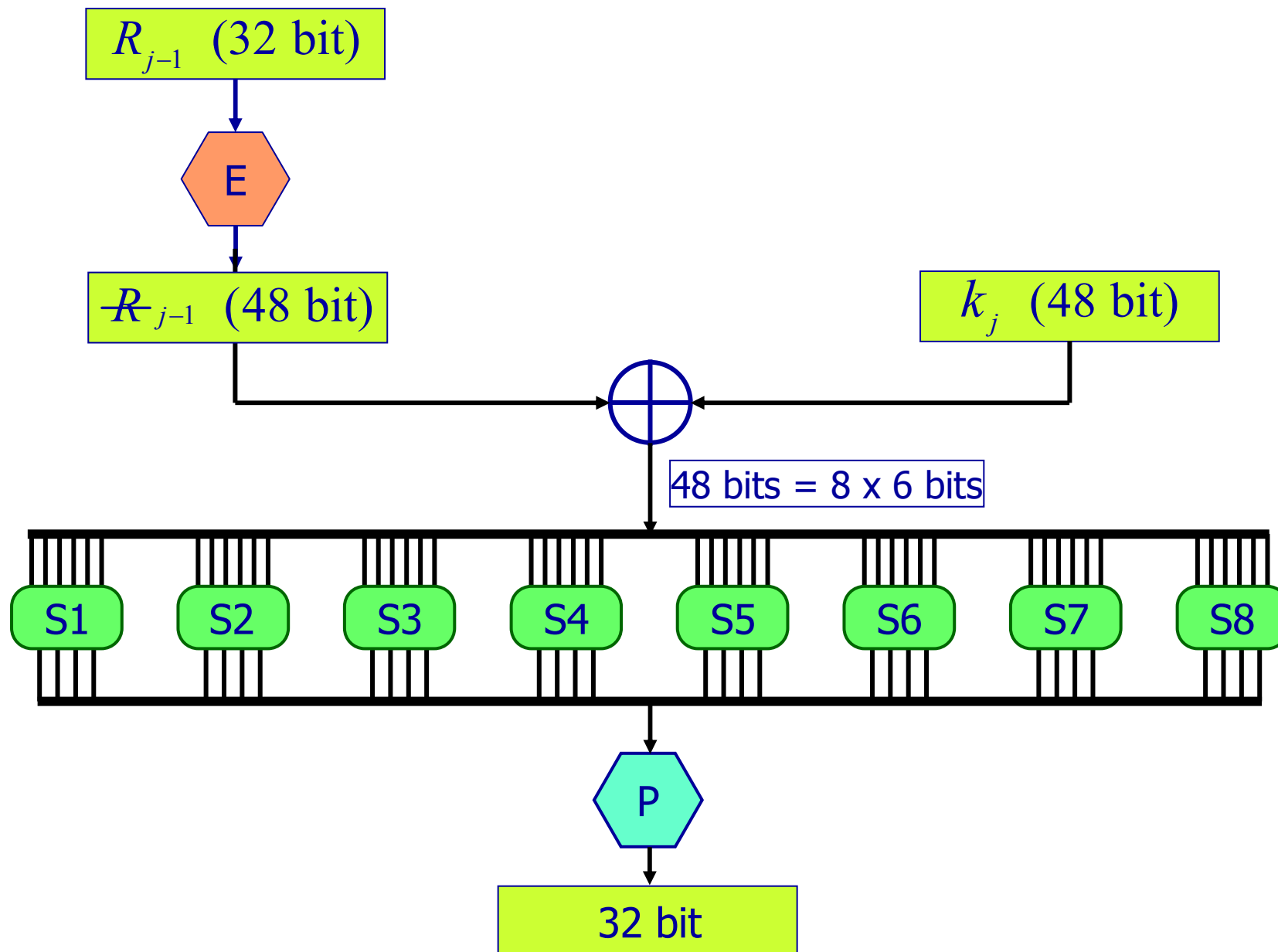
- Usa le due metà L e R di 32 bit
- Come per qualsiasi cipher Feistel può essere descritto come:

$$L_i = R_{i-1}$$

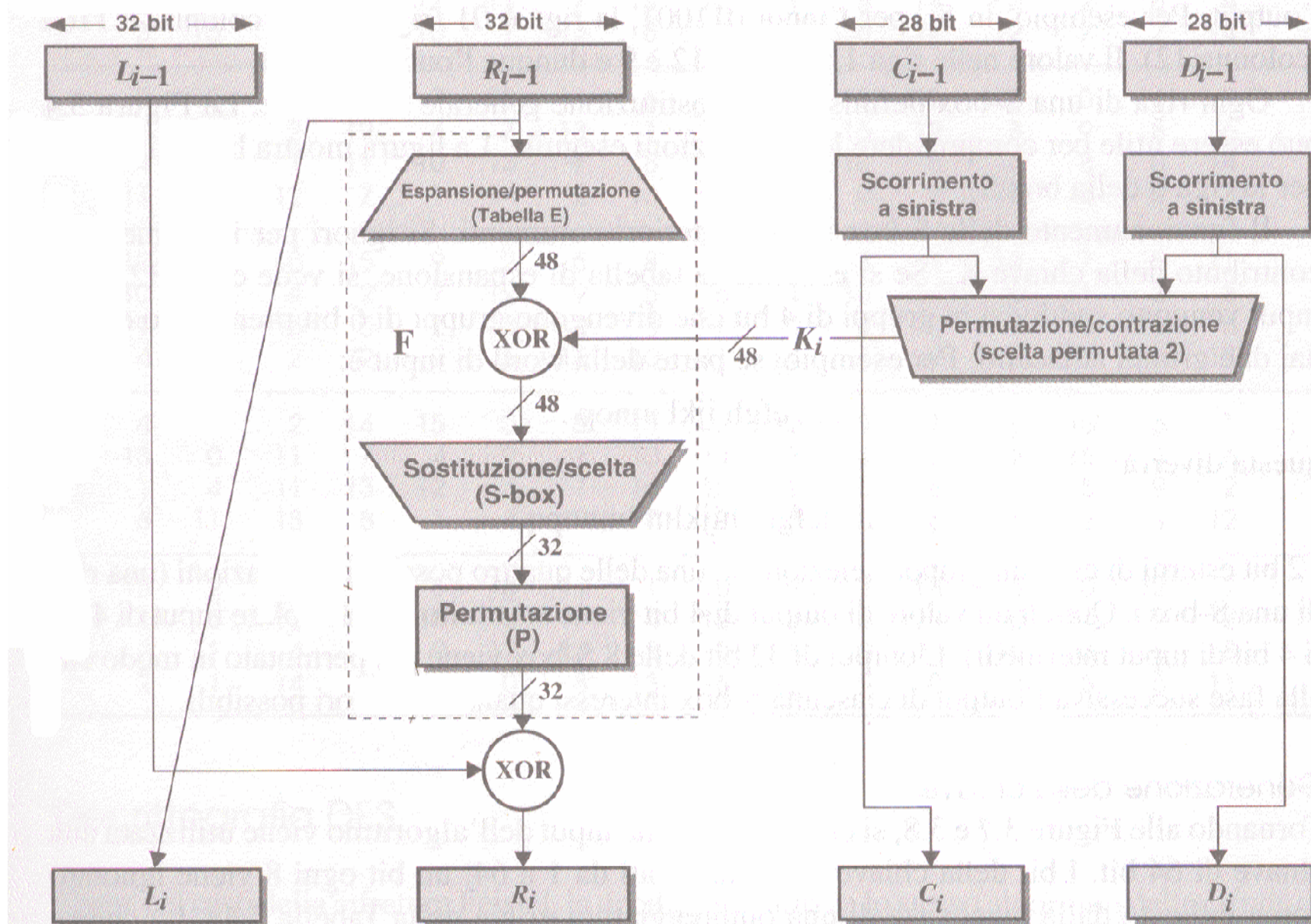
$$R_i = L_{i-1} \oplus F(R_{i-1}, k_i)$$

- F prende i 32 bit della metà destra R_{i-1} e i 48 bit della subkey e:
 - Espande R_{i-1} a 48 bit con l'espansione/permutazione E
 - Ne fa l'XOR con la subkey k_i
 - Passa attraverso 8 S-box ottenendo un risultato a 32 bit
 - Permuta i 32 bit con la permutazione P
- Output della permutazione P è XOR-ed con L_{i-1}

Struttura ciclo DES



Struttura ciclo DES



SWAP a 32 bit e permutazione finale

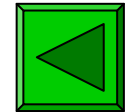
32-bit Swap

È un semplice swap tra le due metà (Left e Right)

Permutazione finale

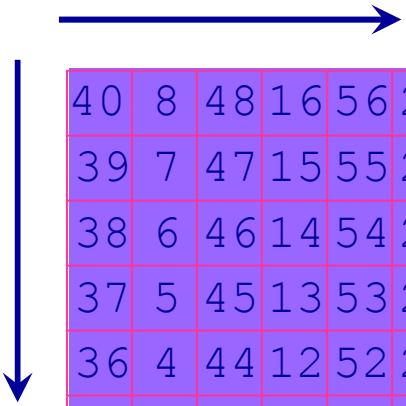
È $(IP)^{-1}$

❖ Definita dalla tabella:



L'output della Final Permutation ha il bit 40 del preoutput block come suo primo bit, il bit 8 come suo secondo bit, e così via fino al bit 25 come ultimo bit

Notare che quelli che erano rispettivamente il bit 1 e il bit 2 del blocco dopo la Initial Permutation si vengono a trovare nelle posizioni 40 e 8; quindi la Final Permutation li riporta nella loro posizione



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

Substitution Box S

- 8 substitution box che mappano 6 bit in 4 bit
- Ciascun S-box è formato da 4 piccoli box a 4 bit
 - I bit esterni 1 e 6 (bit di riga) selezionano una riga tra 4
 - I bit interni 2-5 (bit di colonna) sono sostituiti
 - Il risultato è formato da 8 gruppi di 4 bit o 32 bit
- La selezione di riga dipende da dati e da key
 - Caratteristica nota come autoclaving (autokey)

Esempio:

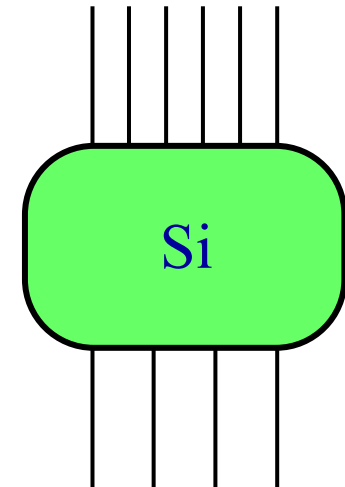
$S(18 \ 09 \ 12 \ 3d \ 11 \ 17 \ 38 \ 39) = 5fd25e03$

Substitution Box S

Output degli S-box definito da tabelle

Una tabella diversa per ognuno degli 8 S-box

S1	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13



Primo e ultimo bit usati per scegliere la riga e i quattro bit interni usati per selezionare il valore di output

e.g. con input **011001** la coppia **01** seleziona la seconda riga e il gruppo interno **1100** (12 D) seleziona il 12-mo elemento, 9

Permutazione P

—1→

↓2

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

Generazione subkey DES

Processo di generazione delle sub-keys a 48 bit ha alcuni passi:

1. Con la cosiddetta Permuted Choice 1 (PC1), si estrae una key a 56 bit dalla chiave a 64 bit, eliminando ogni 8-vo bit e la formazione di 2 blocchi C_0 e D_0 di 28 bit

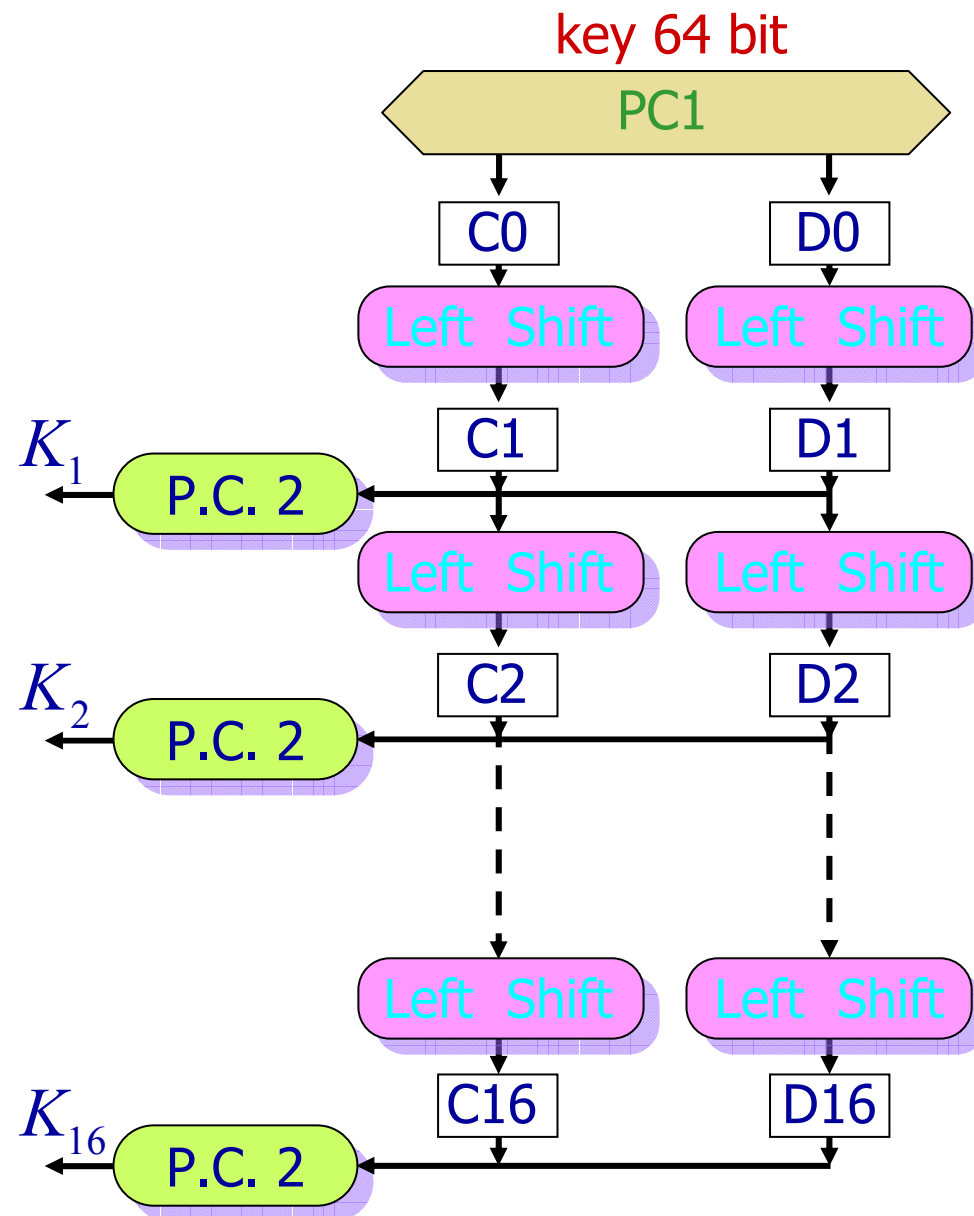
C_0	57	49	41	33	25	17	9
	1	58	50	42	34	26	18
	10	2	59	51	43	35	27
	19	11	3	60	52	44	36

D_0	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

Generazione subkey DES (cont1)

2. Per ogni stadio j i
due blocchi C_{j-1} e
 D_{j-1} sono left shifted
di un numero fisso di
volte ottenendo la
nuova coppia di blocchi
 C_j e D_j

3. Per ogni stadio j i
due blocchi C_j e
 D_j sono applicati alla
Permutation Choice 2
ottenendo la key K_j



Generazione subkey DES (cont2)

Il numero dei Left Shifts,
secondo lo standard
dipende dalla iterazione J
ed è definito da una tabella

Number Iteration	Number of Left Shifts
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	9
10	1
11	2
12	2
13	2
14	2
15	2
16	1

Generazione subkey DES (cont3)

Tutte le subkey sono ottenute applicando alla coppia di blocchi

C_j e D_j la cosiddetta Permuted Choice 2, che è definita dalla tabella

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

Nella tabella non sono presenti gli 8 numeri 9, 18, 22, 25, 35, 38, 43 e 54, sicché la PC2 con un input di 56 bit da luogo ad un output di 48 bit

Il primo bit di K_j è il 14-mo bit di $C_j D_j$, il secondo bit il 17-mo, e così via fino al 48-mo bit di K_j che è il 32-mo di $C_j D_j$

Decriptazione del DES

- La decriptazione svolge gli stessi passi
- Con schema Feistel, si fanno di nuovo i passi di encriptazione usando le subkey in ordine inverso
 - La IP rifà all'indietro il passo di criptazione FP
 - Il primo ciclo con la SK16 rifà all'indietro il 16-mo passo di encriptazione
 -
 - Il 16-mo ciclo con SK1 rifà all'indietro il 1-mo passo di encriptazione
 - Infine la FP rifà all'indietro la Initial Permutation IP
 - Alla fine si rianno i dati originali

Effetto valanga

- Desiderabile che la chiave abbia un effetto valanga
- Un cambiamento di un input o di un bit della key dia luogo a cambiare circa metà dei bit di output
- In questo modo i tentativi di decriptazione per tentativi diventano impossibili
- DES presenta un forte effetto valanga

Forza del DES – Dimensione della chiave

- Chiavi a 56-bit hanno $2^{56} \cong 7.2 \times 10^{16}$ valori
- Attacchi a forza bruta sembrano difficili
- Progressi recenti hanno mostrato la possibilità
 - Nel 1997 su Internet in qualche mese
 - Nel 1998 con hardware specializzato in qualche giorno
 - Nel 1999 combinando le due cose in 22 h
- Bisogna comunque essere in grado di riconoscere il testo
- È però stata mostrata la necessità di metodi alternativi e più forti

Forza del DES – Attacchi analitici

- Oggi esistono diversi attacchi analitici al DES
- Utilizzano la struttura profonda del DES
 - Acquisendo informazioni circa le encryptions
 - Possono eventualmente ricavare alcuni/tutti i bit delle subkey
 - Se serve dopo possono fare una ricerca esaustiva del resto
- Generalmente questi sono attacchi statistici
- Includono:
 - Criptoanalisi differenziale
 - Criptoanalisi lineare
 - Attacchi relativi alla chiave

Forza del DES – Attacchi di timing

- Attacchi verso la reale cifratura
- La struttura della chiave (peso di Hamming) ha effetto sul tempo di cifratura/decifratura
- Possibile solo nel caso in cui l'opponent possa controllare questi tempi
- Particolarmente problematica per le smartcard
- In ogni modo, rappresenta solo un primo passo del possibile attacco

Criptanalisi differenziale

Uno dei più significativi avanzamenti recenti (pubblici) nella criptanalisi

Nota alla NSA e alla IBM al progetto del DES

Pubblicata negli anni 90 da Murphy et al.

Metodo potente per l'attacco ai block cipher

Usato per attaccare diversi cifrari a blocco con successo variabile

DES (come Lucifer) abbastanza resistente

Criptanalisi differenziale

- Attacco statistico contro i cifrari Feistel
- Usa una struttura di cifra non usata prima
- Il progetto delle reti S-P dà luogo a un'uscita della funzione f influenzata da ingresso e chiave
- Quindi è impossibile tracciare i valori all'indietro attraverso il cipher senza conoscere la chiave
- Criptanalisi differenziale confronta due coppie di encryptions in relazione tra loro

Criptanalisi differenziale

Confronta coppie di encryption

- Con una differenza nota nell'input
- Cercando una differenza nota nell'output
- Quando sono usate le stesse subkey

$$\begin{aligned}\Delta m_{i+1} &= m_{i+1} \oplus m'_{i+1} = \\ &= [m_i \oplus f(m_i, k_i)] \oplus [m'_i \oplus f(m'_i, k_i)] = \\ &= \Delta m_i \oplus [f(m_i, k_i) \oplus f(m'_i, k_i)]\end{aligned}$$

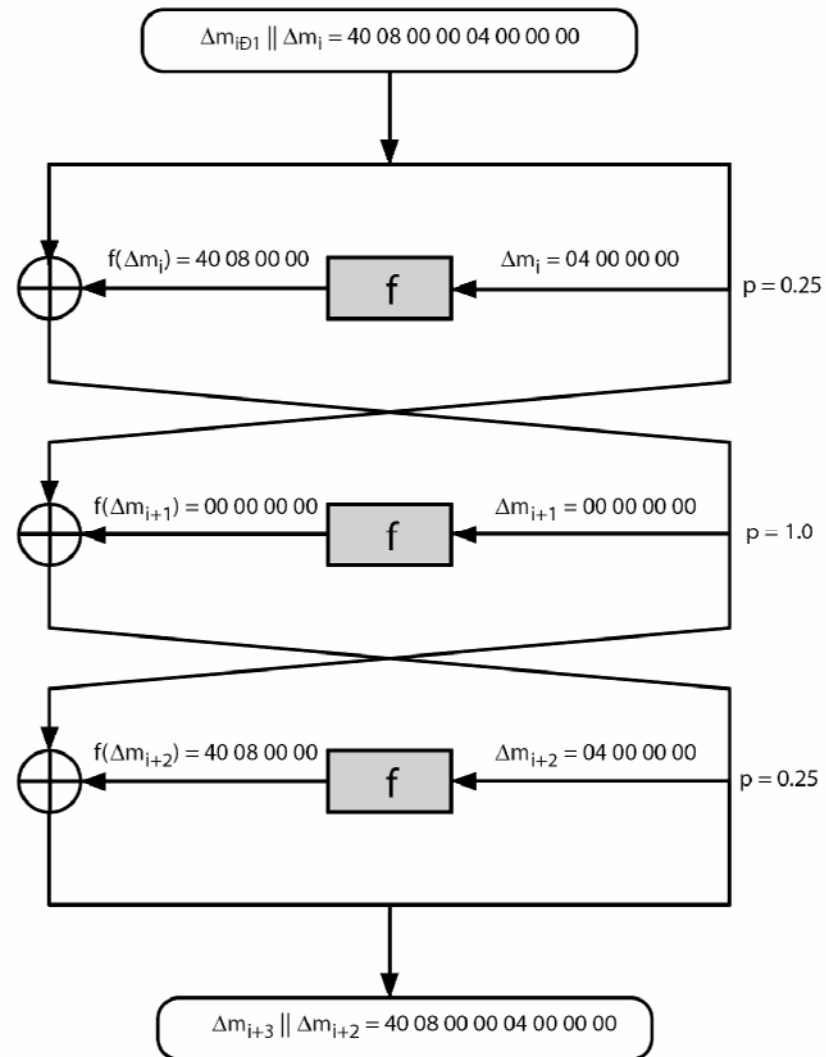
Criptanalisi differenziale

Alcune differenze di input danno luogo con probabilità p ad alcune differenze di output

Se si trovano con probabilità più elevata istanze di alcune coppie di differenze input/output allora:

- Possibile inferire la subkey usata in un round
- Dopo di ciò bisogna iterare il processo su molti round (con probabilità decrescenti)

Criptoanalisi differenziale



Criptanalisi differenziale

Sviluppa un attacco criptando ripetutamente delle coppie di plaintext con XOR di input noto fino ad ottenere lo XOR in uscita desiderato

Quando trovato:

- Se i round intermedi si adattano allo XOR richiesto si una coppia buona
- Se no, sia una coppia sbagliata, il rapporto relativo è l'S/N per l'attacco

Possibile dedurre allora i valori delle chiavi per i round

- Le coppie buone suggeriscono gli stessi bit di chiave
- Le coppie sbagliate danno valori casuali

Per numero di round grande, probabilmente è così basso che è richiesto che esistano più coppie con input di 64 bit

Biham e Shamir hanno mostrato come una caratteristica
Di 13 round iterata può rompere il DES completo a 16 round

Criptanalisi lineare

Trattasi di un altro sviluppo recente

Anch'esso un metodo statistico

Deve essere iterato sopra i round,
con probabilità decrescenti

Sviluppato da Matsui e altri nei primi anni 90

Basato sulla determinazione di approssimazioni lineari

Possibile attaccare il DES con 2^{43} plaintext noti

Più facile, ma praticamente non realizzabile

Criptanalisi lineare

Trova delle approssimazioni lineari con probabilità $\neq 1/2$

$$P[i_1, i_2, \dots, i_a] \oplus C[j_1, j_2, \dots, j_b] = K[k_1, k_2, \dots, k_c]$$

dove i_a, j_b, k_c sono le posizioni dei bit in P, C, K

Fornisce delle equazioni lineari per i bit della chiave

Ottiene un bit della chiave usando un algoritmo di massima verosimiglianza

Usa un gran numero di encryptions di tentativo

Efficacia data da $|p^{-1/2}|$

Criteri di progetto del DES

Presentati da Coppersmith nel 1994

- ❑ 7 criteri per gli S-box forniscono:
 - Non linearità
 - Resistenza alla criptanalisi differenziale
 - Buona confusione
- ❑ 3 criteri per la permutazione P forniscono:
 - Aumento della diffusione

Progetto dei cifrari a blocco

Principi fondamentali ancora come Feistel nel 1970

- ❑ Numero dei round
- ❑ Cifrari migliori con valore maggiore
- ❑ Funzione f
 - Se non lineare fornisce confusione, valanga
 - Esistono problemi circa la selezione degli S-box
- ❑ Schedulazione della chiave
 - Creazione complessa delle subkey, valanga