rafforzare il DES

- L'uso del DES è molto diffuso. Una volta che si è rivelato non più sicuro si può
  - passare a un algoritmo migliore, completamente diverso (AES) – può essere costoso
  - trovare il modo di rafforzare la sicurezza del DES
- tramite due iterazioni: si usano due chiavi (k_1, k_2), e la cifratura è

$$e_{k_2}(e_{k_1}(x))$$

- Questo per il DES porta il numero di possibili chiavi a 2^{112} (basta)

- ma attenzione - bisogna essere sicuri che il DES non sia idempotente!
- la cifratura non deve essere chiusa ripetto alla composizione
- se per ogni coppia di chiavi (k_1, k_2) esiste una chiave k_3 con

$$e_{k_2}(e_{k_1}(x)) = e_{k_3}(x),$$

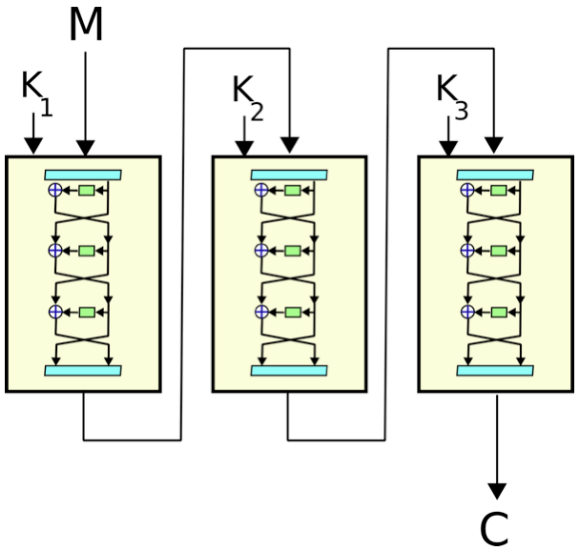
  non c'è nessun miglioramento: due (o più) iterazioni non servono a niente
- Già osservato che i cifrari a sostituzione hanno questa proprietà - la composizione di sue sostituzioni è una sostituzione (di più: formano gruppo)
- Le cifrature DES non sono chiuse rispetto alla composizione, né formano gruppo.
- Questo non è stato chiaro per molto tempo. Dimostrato solo nel 92 (Campbell e Wiener)

meet-in-the-middle

- si può mostrare che due iterazioni non bastano - il DES a due iterazioni è vulnerabile a un attacco meet-in-the-middle
- funziona per ogni cifrario iterato due volte – ha bisogno di spazio
- MIM attack: known plaintext: so che $e_{k_2}(e_{k_1}(x)) = y$
- calcolo $e_k(x)$ per ogni chiave k (e memorizzo i risultati)
- brute force $d_{k'}(y)$ per ogni chiave k'
- ogni corrispondenza può rivelare le due chiavi - per accertarsene, si testano altre coppie

triplo DES

Servono tre iterazioni: triplo DES



- il triplo DES può avere uno schema EDE
- $y = e_{k_3}(d_{k_2}(e_{k_1}(x)))$
- oppure uno schema EEE $y = e_{k_3}(e_{k_2}(e_{k_1}(x)))$

chiavi

- schema EDE
  - 2-DES: con due chiavi e tre iterazioni:

    $$y = e_{k_1}(d_{k_2}(e_{k_1}(x)))$$

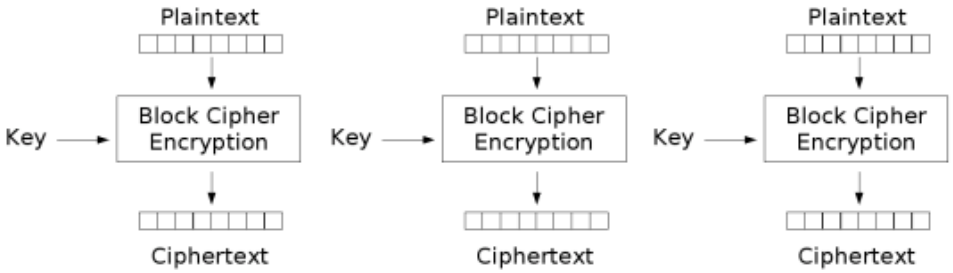
  - 3-DES: con tre chiavi e tre iterazioni:

    $$y = e_{k_3}(d_{k_2}(e_{k_1}(x)))$$

- Il vantaggio dell'EDE è la compatibilità col DES ordinario: con $k_1 = k_2 = k_3$, 3-DES=DES
- è ancora uno standard NIST, ma è lento!
- Sempre più spesso rimpiazzato dall'AES.

cifrare messaggi lunghi

- Esistono diversi metodi per cifrare messaggi di lunghezza maggiore di un blocco
- Il più semplice è cifrare ogni blocco separatamente con la stessa chiave (ev padding)
- metodo ECB (Electronic CodeBook)

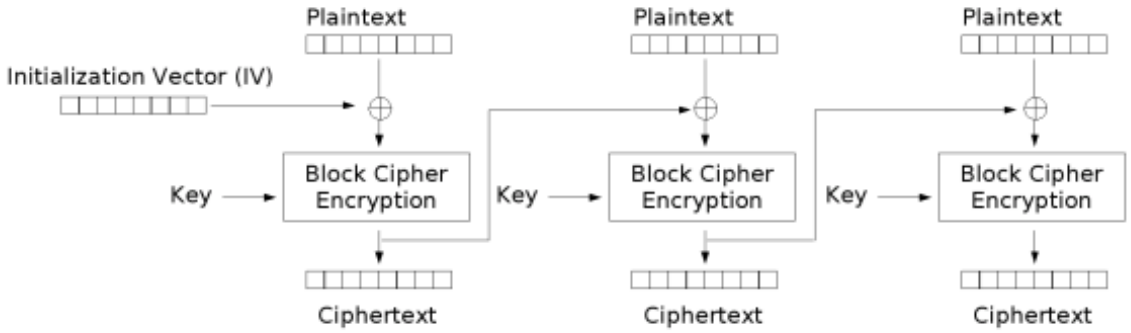


Electronic Codebook (ECB) mode encryption

- Se due plaintext sono uguali, anche i corrispondenti testi cifrati lo sono (possibili analisi di frequenza).
- Un attaccante può inserirsi e cambiare parte del messaggio senza essere scoperto (man-in-the-middle attack).
- si può procedere in parallelo
- in genere si usa solo nella trasmissione di testi molto brevi

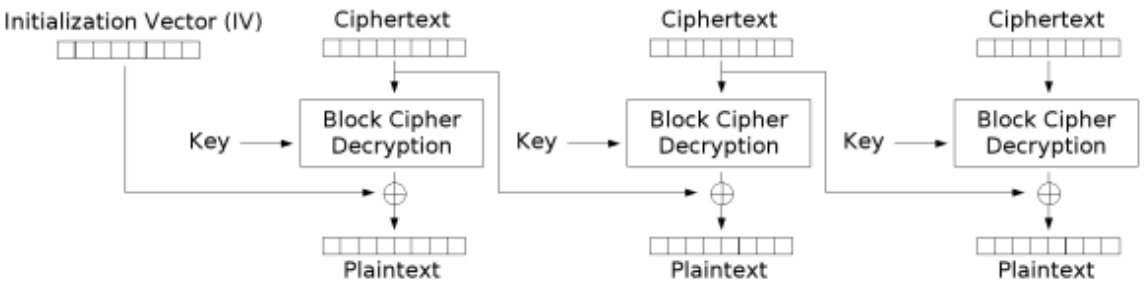
Cipher Block Chaining

Si introduce dipendenza: la cifratura di un blocco dipende dai blocchi precedenti.
Nel metodo CBC, il plaintext è messo in XOR con il testo cifrato precedente prima di essere cifrato.



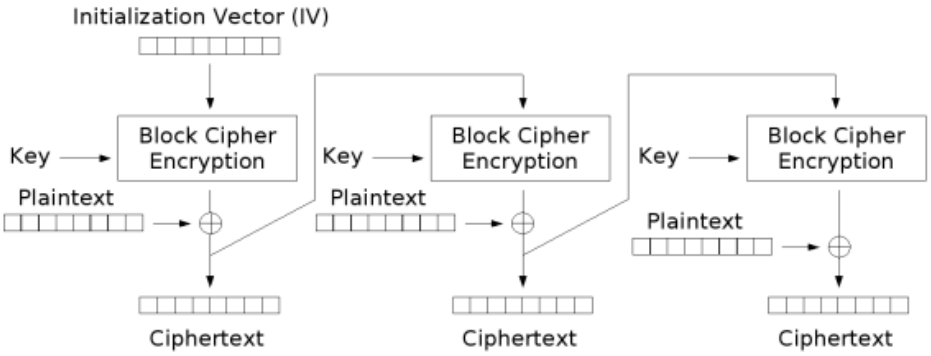
Cipher Block Chaining (CBC) mode encryption

- Bisogna generare e trasmettere l'IV (deve cambiare in ogni trasmissione)
- Gli errori in un blocco si propagano
- Nasconde eventuali pattern del plaintext
- la lunghezza del messaggio deve essere un multiplo della lunghezza del blocco (padding)
- la cifratura non può avvenire in parallelo – la decifratura sì



Cipher Block Chaining (CBC) mode decryption

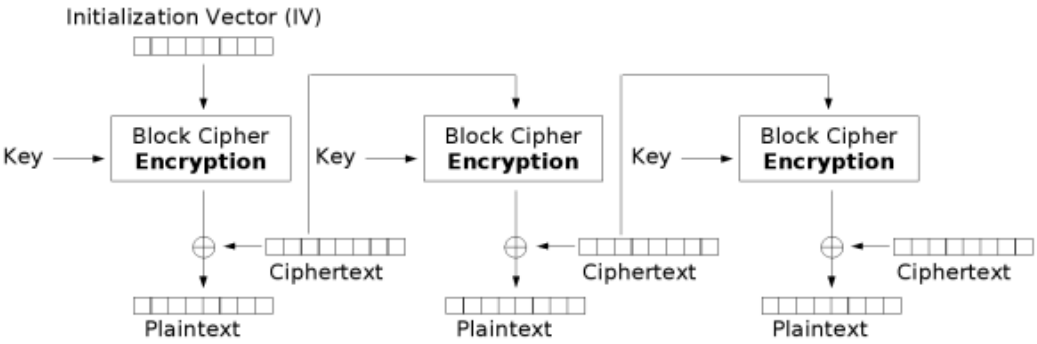
Cipher FeedBack



Cipher Feedback (CFB) mode encryption

- Il messaggio è cifrato con uno XOR (cifrario a flusso) (vantaggi - nel trasmettere un flusso di caratteri)
- Gli errori si propagano

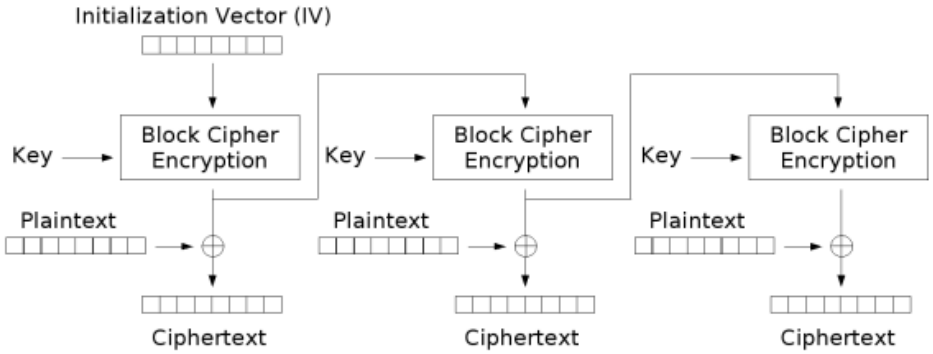
Cipher FeedBack - decifratura



Cipher Feedback (CFB) mode decryption

- si usa solo la cifratura, anche per decifrare
- l'IV deve cambiare in ogni trasmissione

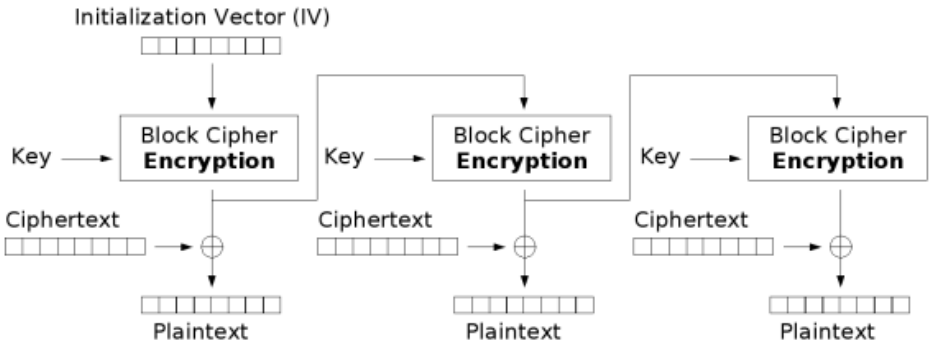
Output FeedBack



Output Feedback (OFB) mode encryption

- Si comporta come un generatore di numeri pseudocasuali.
- Il messaggio è cifrato con uno XOR (cifrario a flusso)
- Gli errori non si propagano
- Si può generare la keystream prima di conoscere il plaintext

Output FeedBack - decifratura



Output Feedback (OFB) mode decryption

- si usa solo la cifratura, anche per decifrare
- l'IV deve cambiare in ogni trasmissione