crittoanalisi del DES

Di fatto, il modo più efficiente di violare il DES è un attacco a forza bruta: provare ogni possibile chiave, una dopo l'altra.

Ci sono due attacchi "a livello teorico" applicabili a ogni cifrario a blocchi

crittoanalisi differenziale: "scoperta" da Biham e Shamir (ca 89). Il DES resiste molto bene alla CD: la tecnica era nota all'IBM e/o all'NSA al momento della progettazione, ed è stata tenuta segreta.
Per violare il DES con la CD, servono 2^{47} chosen plaintext.

crittoanalisi lineare: scoperta da Matsui nel 93, migliorata via via. Servono $\sim 2^{41}$ known plaintext. Non ci sono prove che il DES sia stato sviluppato per resistere a questo tipo di attacco.

crittoanalisi differenziale

- attacco complesso
- idea: studiare in comportamento di coppie di blocchi di testo attraverso la cifratura invece che di un singolo blocco
- si fissa una coppia di plaintext arbitraria - ma con una fissata differenza (XOR)
- usando le differenze nei risultanti testi cifrati, si assegna una probabilità alle possibili chiavi
- esaminando un numero sufficiente di coppie, una chiave emerge come più probabile
- il DES resiste molto bene alla CD – anche il numero di round è quello giusto

crittoanalisi lineare

- attacco più complesso
- si cerca di individuare delle approssimazioni lineari per la trasformazioni del DES
- una relazione del tipo: (lo XOR di alcuni bit per PT) XOR (lo XOR di alcuni bit del CT)=lo XOR di alcuni bit della chiave
- questa approssimazione lineare sarà valida con una certa probabilità p: se $p \neq \frac{1}{2}$, posso sfruttare questo bias

- il DES non è progettato in modo particolare per resistere a questo tipo di attacco
- in particolare la S_5

debolezze del DES

- ha una "proprietà di complementazione":
- $e_k(x) = y \iff e_{\overline{k}}(\overline{x}) = \overline{y}$.
- se si ha un testo in chiaro scelto posso ridurre un attacco a forza bruta di un fattore 2
- voglio scoprire la chiave k - posso generare (chosen plaintext) i due CT $y_1 = e_k(x), y_2 = e_k(\overline{x})$
- attaco a forza bruta: provo le varie chiavi k_a e calcolo $e_{k_a}(x)$
- se $e_{k_a}(x) = y_1$, la chiave cercata è $k = k_a$
- se $e_{k_a}(x) = \overline{y_2}$, allora la chiave cercata è $k = \overline{k_a}$
- con ogni tentativo elimino 2 chiavi

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

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

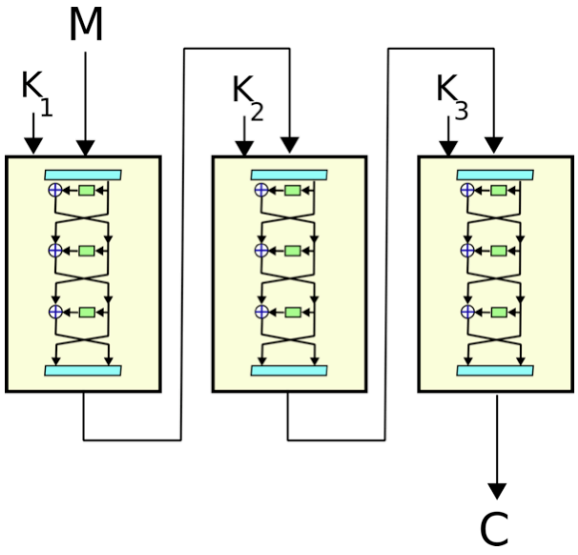
  non c'è nessun miglioramento: due (o più) iterazioni non servono a niente
- Già osservato che i cifrari a sostituzione hanno questa proprietà - la composizione di sue sostituzioni è una sostituzione (di più: formano gruppo)
- Le cifrature DES non sono chiuse rispetto alla composizione, né formano gruppo.
- 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.