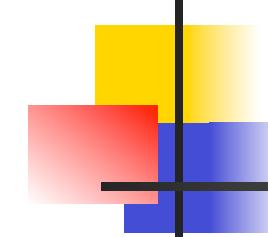


Corso di Crittografia

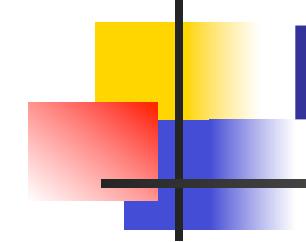
Prof. Dario Catalano

Cifrari Simmetrici (Prima Parte)



Introduzione

- Oggi (ri)parleremo di schemi di cifratura.
- Consistono in
 - Un algoritmo di cifratura ENC
 - Un algoritmo di decifratura DEC
 - Un algoritmo di generazione della chiave KeyGen

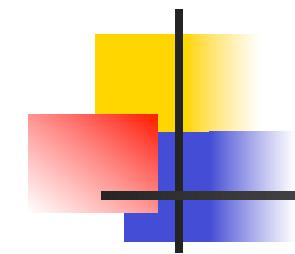


Ripasso

- L'algoritmo KeyGen (randomizzato) restituisce una stringa random (chiave) appartenente all'insieme Chiavi(k).
- L'algoritmo Enc (randomizzato o a stati) prende in input una chiave k e un messaggio m e restituisce un crittotesto C (o un simbolo speciale \perp)
- L'algoritmo Dec (deterministico) dall'input (k, C) produce m (o il simbolo \perp).

Cifrari a stati e Cifrari randomizzati

- Un cifrario simmetrico e' randomizzato se l'operazione di cifratura utilizza una certa quantita' di randomness.
- Un cifrario e' a stati se il risultato dipende da una certa quantita' chiamata stato (che viene opportunamente inizializzata prima di utilizzare il cifrario)



Esempi di cifrari simmetrici

- In questa lezione vedremo alcuni esempi (non tutti sicuri) di cifrari simmetrici.
- In particolare ci concentreremo sui cosiddetti modi d'operazione
- Tali meccanismi si basano su famiglie di permutazioni (ad es. Cifrari a blocchi) e stabiliscono come utilizzare la permutazione per generare il crittotesto.
- Supporremo (per semplicità) che la taglia del messaggio è un multiplo della taglia del blocco.

Modo ECB

(Electronic Codebook Mode)

- Sia $E : \mathcal{K} \times \{0, 1\}^n \rightarrow \{0, 1\}^n$ un cifrario a blocchi
- ECB produce un cifrario senza stati e deterministico.
- L'algoritmo di generazione della chiave si limita a restituire una stringa random.
- Guardiamo come cifrare e decifrare

Enc_{*k*}(*M*)

if ($|M| \bmod n \neq 0$) \vee ($|M| = 0$) return \perp
Sia $M = M[1] \dots M[m]$ ($|M[i]| = n$)
for $i = 1$ to m do
 $C[i] = E_k(M[i])$
 $C \leftarrow C[1] \dots C[m]$
return C

Dec_{*k*}(*C*)

if ($|C| \bmod n \neq 0$) \vee ($|C| = 0$) return \perp
Sia $C = C[1] \dots C[m]$
for $i = 1$ to m do
 $M[i] = E_k^{-1}(C[i])$
 $M \leftarrow M[1] \dots M[m]$
return M

Modo CBC\$

(Cipher block Chaining Mode)

- Sia $E : \mathcal{K} \times \{0, 1\}^n \rightarrow \{0, 1\}^n$ un cifrario a blocchi
- CBC\$, con vettore iniziale random IV produce un cifrario senza stati e randomizzato.
- L'algoritmo di generazione della chiave si limita a restituire una stringa random.
- Guardiamo come cifrare e decifrare

Enc _{k} (M)

if ($|M| \bmod n \neq 0$) \vee ($|M| = 0$) return \perp

Sia $M = M[1] \dots M[m]$ ($|M[i]| = n$)

$C[0] \leftarrow IV \leftarrow_R \{0, 1\}^n$

for $i = 1$ to m do $C[i] = E_k(C[i - 1] \oplus M[i])$

$C \leftarrow C[1] \dots C[m]$

return $\langle IV, C \rangle$

Dec _{k} (C, IV)

if ($|C| \bmod n \neq 0$) \vee ($|C| = 0$) return \perp

Sia $C = C[1] \dots C[m]$

$C[0] \leftarrow IV$

for $i = 1$ to m do $M[i] = E_k^{-1}(C[i] \oplus C[i - 1])$

$M \leftarrow M[1] \dots M[m]$

return M

Modo CTR\$

(modo counter randomizzato)

- Sia $F : \mathcal{K} \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$ una famiglia di funzioni (non necess. un cifrario a blocchi)
- CTR\$, produce un cifrario senza stati e randomizzato.
- L'algoritmo di generazione della chiave si limita a restituire una stringa random.
- Guardiamo come cifrare e decifrare.

Enc_{*k*}(*M*)

m $\leftarrow \lceil |M|/L \rceil$; *R* $\leftarrow_R \{0, 1\}^\ell$

Pad $\leftarrow F_k(R + 1) || F_k(R + 2) || \dots || F_k(R + m)$

Pad \leftarrow Primi $|M|$ bits di *Pad*

C' $\leftarrow M \oplus Pad$

C $\leftarrow R || C'$

return *C*

Dec_{*k*}(*C*)

if ($|C| < \ell$) return \perp

Sia $C = R || C'$ ($|R| = \ell$)

m $\leftarrow \lceil |C'|/L \rceil$

Pad $\leftarrow F_k(R + 1) || F_k(R + 2) || \dots || F_k(R + m)$

Pad \leftarrow Primi $|C'|$ bits di *Pad*

M $\leftarrow C' \oplus Pad$

return *M*

Modo CTRC (modo counter non-randomizz.)

- Sia $F : \mathcal{K} \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$ famiglia di funzioni (non necessariamente un cifrario a blocchi)
- CTRC, produce un cifrario con stati e non randomizzato.
- Viene mantenuto un contatore globale ctr (inizialmente uguale a 0)
- L'algoritmo di generazione della chiave si limita a restituire una stringa random.
- Guardiamo come cifrare e decifrare.

Enc_{*k*}(*M*)

m $\leftarrow \lceil |M|/L \rceil$;

If $ctr+m \geq 2^\ell$ return \perp

Pad $\leftarrow F_k(ctr+1) || F_k(ctr+2) || \dots || F_k(ctr+m)$

Pad \leftarrow Primi $|M|$ bits di *Pad*

C' $\leftarrow M \oplus Pad$

ctr $\leftarrow ctr+m$

return $\langle ctr-m, C' \rangle$

Dec_{*k*}($\langle i, C \rangle$)

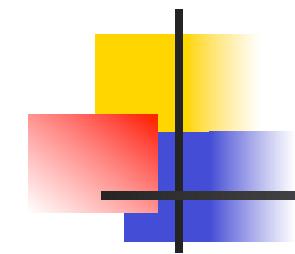
m $\leftarrow \lceil |C|/L \rceil$;

Pad $\leftarrow F_k(i+1) || F_k(i+2) || \dots || F_k(i+m)$

Pad \leftarrow Primi $|C|$ bits di *Pad*

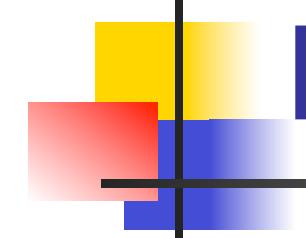
M $\leftarrow C \oplus Pad$

return *M*



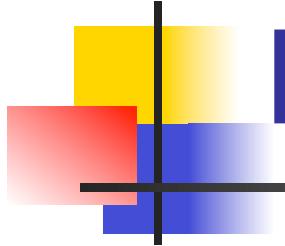
Privacy

- Prossimo obiettivo: arrivare ad una soddisfacente definizione di sicurezza (privacy).
- L'avversario **A** vede un certo numero di crittostesi e vuole cercare di “trarre” informazione da essi.
- Sembra piu’ facile parlare di insicurezza piuttosto che di sicurezza.



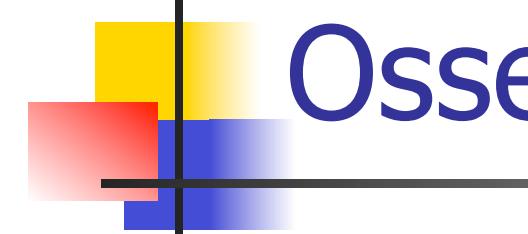
Privacy – II

- Se A e' capace di "trarre" M da C, allora lo schema non e' sicuro.
- Potremmo pensare di considerare sicuro il contrario...
- Nemmeno questo e' un buon approccio
 - Informazioni parziali potrebbero essere rivelate.
 - Sarebbe opportuno definire una nozione di sicurezza piu' forte.



Privacy – III

- Esiste una lista pressocche' infinita di proprieta' che rendono insicuro un sistema.
- Dobbiamo pensare alla sicurezza in maniera piu' diretta per poter sperare di riuscire a definirla.



Osservazioni

- Non possiamo sperare di definire sicuro uno schema nel quale C non rivela *alcuna informazione* su M
 - Alcune informazioni a priori sono inevitabili.
- La soluzione perfetta e' trasmettere M da Alice a Bob utilizzando un canale ideale.
 - Proteggiamo solo la comunicazione.
 - Deviamo dal modello ideale perche' la taglia del messaggio e' spesso nota.
- Riguardiamo il modo ECB, alla luce di un esempio.

$\text{Enc}_k(M)$

if $(|M| \bmod n \neq 0) \vee (|M| = 0)$ return \perp

Sia $M = M[1] \dots M[m]$ ($|M[i]| = n$)

for $i = 1$ to m do

$C[i] = E_k(M[i])$

$C \leftarrow C[1] \dots C[m]$

return C

- Il sistema e' del tutto deterministico
- Cio' non dipende dalla "sicurezza" di E

ATTACCO SU ECB

$$E: K \times \{0,1\}^m \rightarrow \{0,1\}^n$$

A(ECB) :

$$x, y \leftarrow_R \{0,1\}^m // x \neq y$$

$$M_0 \leftarrow x || x$$

$$M_1 \leftarrow x || y$$

$$C \leftarrow O_{ECB}(M_0, M_1) // C = C_1 || C_2$$

if ($C_1 \neq C_2$) return 1

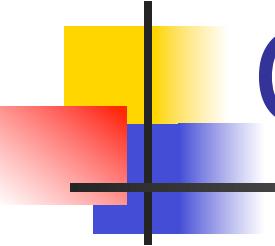
else return 0

OGNI SCHEMA DETERMINISTICO E SENZA STATI È INSICURO

$$\text{Adv}^{\text{ind-CPA}}(A) = 1$$

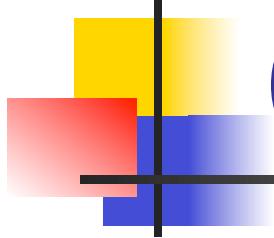
A(SE):

$m_0, m_1 \in M // m_0 \neq m_1$
 $C_1 \leftarrow O_{\text{ENC}}(m_0, m_1)$
 $C_2 \leftarrow O_{\text{ENC}}(m_1, m_1)$
if ($C_1 == C_2$) return 1
else return 0



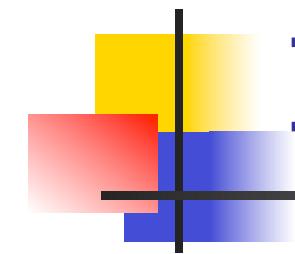
Conseguenze

- Un “buon” cifrario deve essere probabilistico o mantenere uno stato
- Questa e’ una cosa piuttosto sorprendente (pensando a 2000 anni di crittografia)



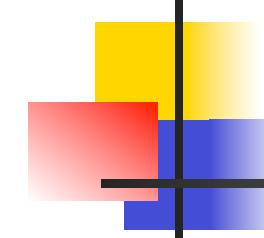
Indistinguibilità (relativamente ad attacchi cpa)

- **A** sceglie q coppie di messaggi M_0^i, M_1^i (della stessa lunghezza)
- **A** riceve q crittotesti C^i
- I crittotesti cifrano o solo i messaggi di destra o solo quelli di sinistra (secondo un bit segreto b)
- L'obiettivo di **A** e' indovinare b



Indistinguibilità (cont.)

- A ha la possibilità di accedere ad un “oracolo” $\text{Enc}_k(\text{LR}(\dots, b))$.
- A può rivolgere una domanda del tipo (M_0, M_1) all’oracolo.
- L’oracolo $\text{Enc}_k(\text{LR}(M_0, M_1, b))$ risponde con un crittostesso C che cifra M_b .
- Due mondi $\text{Enc}_k(\text{LR}(M_0, M_1, 0))$ e $\text{Enc}_k(\text{LR}(M_0, M_1, 1))$
- A deve capire in che mondo si trova.

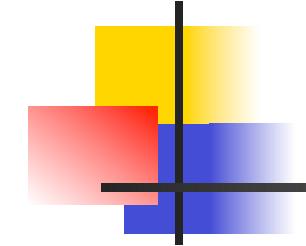


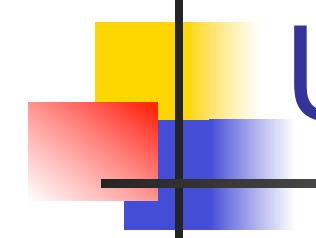
Definizione

- SE=(KeyGen, Enc, Dec) cifrario simmetrico

$Esp_{SE}^{ind\text{-}cpa\text{-}1}(A)$	$Esp_{SE}^{ind\text{-}cpa\text{-}0}(A)$
$K \leftarrow_R KeyGen$	$K \leftarrow_R KeyGen$
$b \leftarrow A^{Enc_K(LR(.,,1))}$	$b \leftarrow A^{Enc_K(LR(.,,0))}$
Return b	Return b

$$Adv^{ind\text{-}cpa}(A) = |\Pr[Esp_{SE}^{ind\text{-}cpa\text{-}1}(A) = 1] - \Pr[Esp_{SE}^{ind\text{-}cpa\text{-}0}(A) = 1]|$$

- 
- L'avversario puo' "vincere" con probabilita' $1/2$
 - Un tale avversario non puo' essere considerato una minaccia.
 - Il vantaggio, misura quindi la capacita' dell'avversario di far meglio che sparare a caso.



Un approccio alternativo

- $SE = (\text{KeyGen}, \text{Enc}, \text{Dec})$ cifrario simmetrico

$\text{Esp}_{SE}^{\text{ind-cpa-cg}}(A)$

$b \leftarrow_R \{0,1\}; K \leftarrow_R \text{KeyGen};$

$b' \leftarrow_R A^{\text{Enc}_K(\text{LR}(\dots, b))};$

If $b=b'$ return 1 else return 0

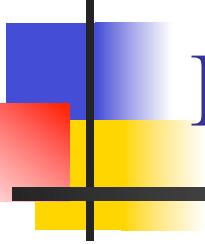
$$\text{Adv}^{\text{ind-cpa}}(A) = 2\Pr[\text{Esp}_{SE}^{\text{ind-cpa-cg}}(A) = 1] - 1$$

- I due approcci sono equivalenti

$$\begin{aligned}
& \Pr_r [E_{\text{SP}_{\text{SE}}}^{\text{ind-CPA-Cg}}(A) = 1] = \Pr_r [b = b'] = \Pr_r [b = b' \wedge b = 1] + \\
& + \Pr_r [b = b' \wedge b = 0] = \\
& = \Pr_r [b = b' \mid b = 1] \cdot \Pr_r [b = 1] + \Pr_r [b = b' \mid b = 0] \cdot \Pr_r [b = 0] = \\
& = \frac{1}{2} (\Pr_r [b' = 1 \mid b = 1] + \Pr_r [b' = 0 \mid b = 0]) = \\
& = \frac{1}{2} (\Pr_r [b' = 1 \mid b = 1] + 1 - \Pr_r [b' = 1 \mid b = 0]) = \\
& = \frac{1}{2} \Pr_r [b' = 1 \mid b = 1] + \frac{1}{2} - \frac{1}{2} \Pr_r [b' = 1 \mid b = 0] = \\
& = \frac{1}{2} + \frac{1}{2} (\Pr_r [b' = 1 \mid b = 1] - \Pr_r [b' = 1 \mid b = 0]) = \\
& = \frac{1}{2} + \frac{1}{2} \text{Adv}_{\text{SE}}^{\text{ind-CPA}}(A)
\end{aligned}$$

DA COI SEGUE:

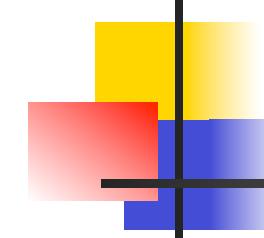
$$\text{Adv}_{\text{SE}}^{\text{ind-CPA}}(A) = 2 \Pr_r [E_{\text{SP}_{\text{SE}}}^{\text{ind-CPA-Cg}}(A) = 1] - 1$$



Corso di Crittografia

Prof. Dario Catalano

Cifrari Simmetrici (Seconda Parte)

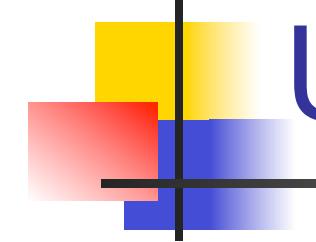


Definizione

- $\text{SE}=(\text{KeyGen}, \text{Enc}, \text{Dec})$ cifrario simmetrico

$\text{Esp}_{\text{SE}}^{\text{ind-cpa-1}}(A)$	$\text{Esp}_{\text{SE}}^{\text{ind-cpa-0}}(A)$
$K \leftarrow_R \text{KeyGen}$	$K \leftarrow_R \text{KeyGen}$
$b \leftarrow A^{\text{Enc}_K(\text{LR}(\dots, 1))}$	$b \leftarrow A^{\text{Enc}_K(\text{LR}(\dots, 0))}$
Return b	Return b

$$\text{Adv}^{\text{ind-cpa}}(A) = |\Pr[\text{Esp}_{\text{SE}}^{\text{ind-cpa-1}}(A) = 1] - \Pr[\text{Esp}_{\text{SE}}^{\text{ind-cpa-0}}(A) = 1]|$$



Un approccio alternativo

- $SE = (\text{KeyGen}, \text{Enc}, \text{Dec})$ cifrario simmetrico

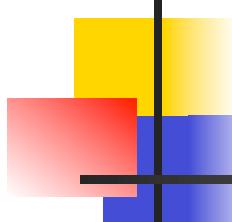
$\text{Esp}_{SE}^{\text{ind-cpa-cg}}(A)$
 $b \leftarrow_R \{0,1\}; K \leftarrow_R \text{KeyGen};$
 $b' \leftarrow_R A^{\text{Enc}_K(\text{LR}(\dots, b))};$
If $b' = b$ return 1 else return 0

$$\text{Adv}^{\text{ind-cpa}}(A) = 2\Pr[\text{Esp}_{SE}^{\text{ind-cpa-cg}}(A) = 1] - 1$$

- I due approcci sono equivalenti

Perche' e' una buona definizione?

- E' difficile stabilire se una definizione e' veramente quella buona.
 - Non basta trovare un controeSEMPIO
- Oggi cercheremo di convincerci che quella che abbiamo e' una buona definizione.
- Per adesso, pero', continuiamo ad "impratichirci" con essa, studiando qualche schema che non la soddisfa.



ECB non e' sicuro

$\text{Enc}_k(m)$

if $(|M| \bmod n \neq 0) \vee (|M| = 0)$ return \perp

Sia $M = M[1] \dots M[m]$ ($|M[i]| = n$)

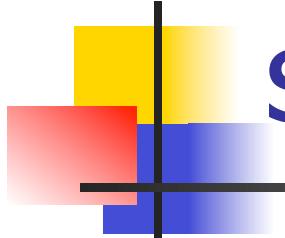
for $i = 1$ to m do

$C[i] = E_k(M[i])$

$C \leftarrow C[1] \dots C[m]$

return C

Costruiamo un avversario che rompe il sistema con un vantaggio considerevole.



Ogni schema deterministico e senza stati e' insicuro.

Teorema:

Sia $SE = (\text{KeyGen}, \text{Enc}, \text{Dec})$ un cifrario simmetrico (deterministico e senza stati)

Sia M lo spazio dei msg (contenente almeno due msg distinti della stessa lunghezza)

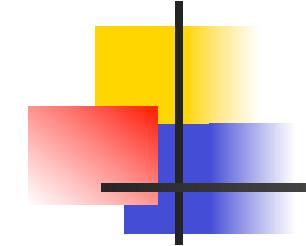
Esiste un avversario A , contro SE , tale che

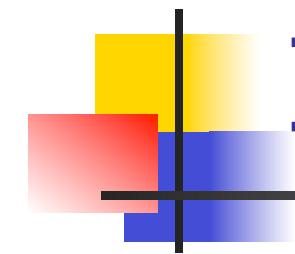
$$\text{Adv}^{\text{ind-cpa}}(A) = 1$$



Testiamo la definizione

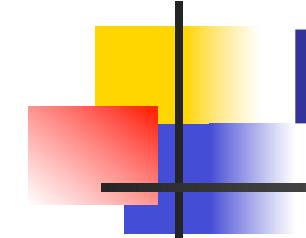
- Nelle precedenti lezioni abbiamo studiato un certo numero di proprietà che sono necessarie (ma non sufficienti!) per la privacy.
- Ad es. Resistenza ad Attacchi di tipo KR, o PR
- Un buon test per la nostra definizione è quindi verificare se essa implica queste proprietà.

- 
- Ragionamento analogo a quello fatto per le PRF.
 - Uno schema sicuro in senso IND-CPA dovrebbe anche essere sicuro contro key recovery o plaintext recovery.
 - A titolo di esempio dimostriamo che IND-CPA implica PR-CPA.



IND-CPA → PR-CPA

- Dimostriamo che dato uno schema SE=(KeyGen,Enc,Dec) se esiste un avversario PR-CPA questo puo' essere "trasformato" in un avversario IND-CPA che utilizza piu' o meno le stesse risorse.
- Per semplicita' supponiamo che SE sia senza stati.
- Per far questo dobbiamo prima definire formalmente un avversario PR-CPA



Definizione

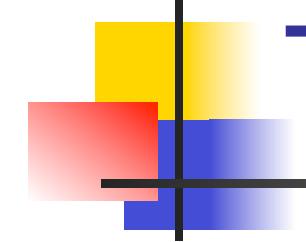
- SE: Cifrario simmetrico senza stati.

$\text{Esp}_F^{\text{pr-cpa}}(B)$

$$K \leftarrow_R \text{KeyGen}; M' \leftarrow_R \{0,1\}^m$$
$$C \leftarrow_R \text{Enc}_K(M')$$
$$M \leftarrow B^{\text{Enc}_K(\cdot)}(C)$$

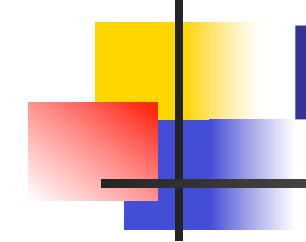
If $M=M'$ Return 1 else return 0

$$\text{Adv}(B) = \Pr[\text{Esp}_{SE}^{\text{pr-cpa}}(B) = 1]$$



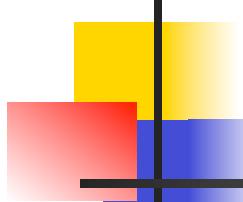
Teorema

- $SE = (\text{KeyGen}, \text{Enc}, \text{Dec})$ cifrario simmetrico (senza stati)
- $\{0,1\}^m$ spazio dei messaggi
- B avversario pr-cpa che fa q domande
- Allora esiste un avversario ind-cpa A (che fa $q+1$) domande tale che
$$\text{Adv}(B) \leq \text{Adv}(A) + (1/2)^m$$



Dimostrazione

- A ha a disposizione un oracolo $LR(.,.,b)$ e deve stabilire il valore di b.
- Per far questo A puo' sfruttare B.
- A dunque deve essere in grado di
 1. Rispondere alle domande di B
 2. Utizzare il messaggio restituito da B per determinare b.



Avversario A

$A^{\text{Enc}_k(LR(\cdot, \cdot, b))}$

$M_0, M_1 \leftarrow_R \{0, 1\}^m$

$C \leftarrow \text{Enc}(LR(M_0, M_1, b))$

Esegui B (con input C) come segue

If B chiede il messaggio X

$Y \leftarrow \text{Enc}_k(LR(X, X, b))$

Return Y a B.

Quando B si ferma e restituisce M

If $M = M_1$ return 1 else return 0.

$$\text{Adv}^{\text{ind-CPA}}(A) = |\Pr[\text{Exp}^{\text{ind-CPA-1}}(A) = 1] - \Pr[\text{Exp}^{\text{ind-CPA-0}}(A) = 1]|$$

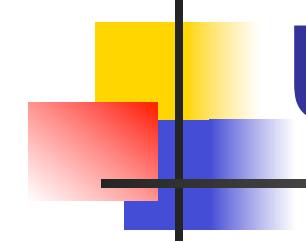
$$\Pr[\text{Exp}^{\text{ind-CPA-1}}(A) = 1] = \text{Adv}(B)$$

$$\Pr[\text{Exp}^{\text{ind-CPA-0}}(A) = 1] = \frac{1}{2^m}$$

$$\text{Adv}(B) = \text{Adv}(A) + \frac{1}{2^m}$$

$$\Downarrow$$
$$\text{Adv}(B) \leq \text{Adv}(A) + \frac{1}{2^m}$$

UNA PROPRIETÀ NECESSARIA È IMPLICATA DALL'INDISTIN-
GUITÀ



Un approccio piu' generale

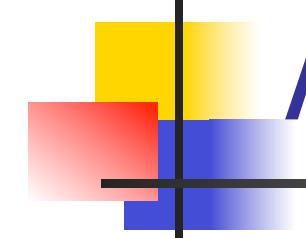
- Quanto detto suggerisce un metodo ancora piu' generale per provare che la definizione e' quella giusta.
- Consideriamo una qualunque proprietà di sicurezza e mostriamo che uno schema sicuro in senso IND-CPA la possiede.

Esempio

- $SE = (\text{KeyGen}, \text{Enc}, \text{Dec})$ cifrario simmetrico (senza stati)
- $\{0,1\}^m$ spazio dei messaggi
- B avversario che fa q domande e, dato un crittotesto $C = \text{Enc}_k(M)$ determina $P(M)$.
- Allora esiste un avversario ind-cpa A (che fa $q + 1$) domande tale

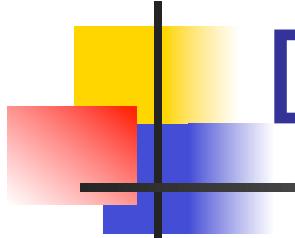
$$P : \mathbb{N} \rightarrow \{0, 1\}$$

$$\text{Adv}(B) = \text{Adv}(A)$$



Ancora una volta

- A ha a disposizione un oracolo $LR(.,.,b)$ e deve stabilire il valore di b.
- Per far questo A puo' sfruttare B.
- A dunque deve essere in grado di
 1. Rispondere alle domande di B
 2. Utilizzare il bit restituito da B per determinare b.



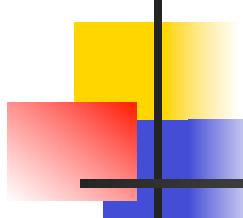
Definiamo l'esperimento di B

- SE: Cifrario simmetrico senza stati.

$\text{Esp}_{\text{SE}}^P(B)$

$$K \leftarrow_R \text{KeyGen}; M \leftarrow_R \{0,1\}^m$$
$$C \leftarrow_R \text{Enc}_K(M)$$
$$b \leftarrow B^{\text{Enc}_K(\cdot)}(C)$$
$$\text{If } b = P(M) \text{ Return 1 else return 0}$$

$$\text{Adv}(B) = 2\Pr[\text{Esp}_{\text{SE}}^P(B) = 1] - 1$$



Avversario A

$A^{\text{Enc}_k(LR(\cdot, \cdot, b))}$

$M_0, M_1 \leftarrow_R \{0, 1\}^m \ (lsb(M_d = d))$

$C \leftarrow \text{Enc}(LR(M_0, M_1, b))$

Esegui B (con input C) come segue

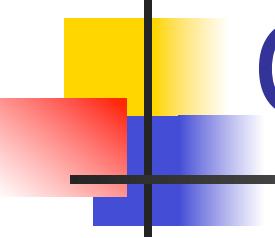
If B chiede il messaggio X

$Y \leftarrow \text{Enc}_k(LR(X, X, b))$

Return Y a B.

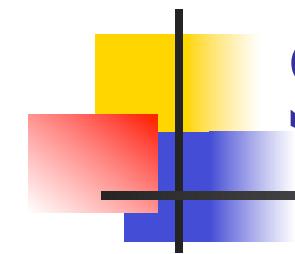
Quando B si ferma e restituisce b'

If $b' = 1$ return 1 else return 0.



Considerazioni

- Non possiamo provare tutte le possibili proprietà
- Eppure dovrebbe essere chiaro che la nostra definizione cattura proprio ciò che vogliamo
- Qualunque capacità abbia B, non dobbiamo far altro che scegliere due messaggi che siano diversi proprio in ciò che B riesce a calcolare.



Sicurezza dei modi CTR

- Entrambi (sotto certe condizioni) garantiscono sicurezza IND-CPA.
- Sorprendentemente, pero' la "qualita'" di tale sicurezza e' differente.
- CTRC permette perfetta sicurezza (ma solo utilizzando funzioni casuali)
- CTR\$ non puo' garantire perfetta sicurezza (anche usando funz. Casuali)
- Questo perche' non possiamo escludere la possibilita' di collisioni.

$\text{Enc}_k(M)$ Modo CTR\$

$m \leftarrow \lceil |M|/L \rceil; R \leftarrow_R \{0, 1\}^\ell$

$Pad \leftarrow F_k(R+1) || F_k(R+2) || \dots || F_k(R+m)$

$Pad \leftarrow$ Primi $|M|$ bits di Pad

$C' \leftarrow M \oplus Pad$

$C \leftarrow R || C'$

return C

$\text{Enc}_k(M)$ Modo CTRC

$m \leftarrow \lceil |M|/L \rceil;$

If $ctr+m \geq 2^\ell$ return \perp

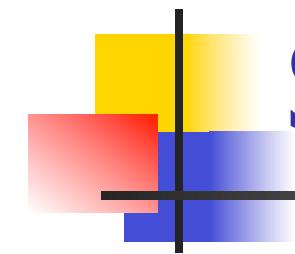
$Pad \leftarrow F_k(ctr+1) || F_k(ctr+2) || \dots || F_k(ctr+m)$

$Pad \leftarrow$ Primi $|M|$ bits di Pad

$C' \leftarrow M \oplus Pad$

$ctr \leftarrow ctr + m$

return $\langle ctr - m, C \rangle$



Sicurezza di CTRC

- Sia $F : \mathcal{K} \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$ fam. di funz.
- SE=(KeyGen,Enc,Dec) un cifrario CTRC
- A avversario IND-CPA contro SE, che fa al piu' q domande (per un totale di σ blocchi di L bit)
- Esiste B (avversario contro la sicurezza PRF di F), che fa σ domande e tale che

$$\text{Adv}_{\text{SE}}^{\text{ind-cpa}}(A) \leq 2\text{Adv}_F^{\text{prf}}(B)$$

B PROVA A ROMPERE LA PRF USANDO L'AVVERSARIO A
B USA $g : \{0, 1\}^\ell \rightarrow \{0, 1\}^L$ TACE CHE $g \in F \circ g \in \text{FUNC}(\ell, L)$

B⁸:

$$b \leftarrow \{0, 1\}$$

RUN ADVERSARY A, REPLYING AS FOLLOWS:

WHEN A MAKES A QUERY (M_0, M_1) do:

$$C \leftarrow \text{Enc}(M_b) // \text{CTR} C \xrightarrow{\gamma}$$

RETURN C TO A

UNTIL A OUTPUTS b'

if($b' == b$) return 1

else return 0

FUNZIONA BENE SE
 $\gamma \in \text{Func}(l, l)$

DIMOSTRAZIONE

$$\Pr_r [E_{SP_F^{Pnf-1}}(A) = 1] = \Pr_r [E_{SP_F^{Pnf-0}}(A) = 1] = \\ = \Pr_r [E_{SP_{SE}^{\text{ind-CPA-}\gamma}}(A) = 1] = \frac{1}{2} + \frac{1}{2} \cdot \text{Adv}_{SE}^{\text{ind-CPA}}(A)$$

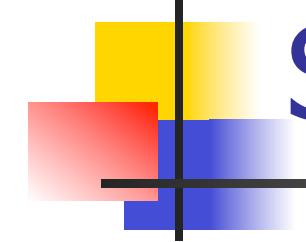
LEMMA

A AVVERSAARIO IND-CPA CON $\text{SE}[\text{Func}(l, l)]$ ALLORA
 $\text{Adv}_{SE}^{\text{ind-CPA}}(A) = 0$

QUINDI:

$$\text{Ad}v_F^{\text{Prf}}(B) = \cancel{\frac{1}{2}} + \frac{1}{2} \cdot \text{Ad}v_{SE}^{\text{ind} \cdot \text{CPa}}(A) - \cancel{\frac{1}{2}}$$

$$\text{Ad}v_{SE}^{\text{ind} \cdot \text{CPa}}(A) = 2 \text{Ad}v_F^{\text{Prf}}(B)$$



Sicurezza di CTR\$

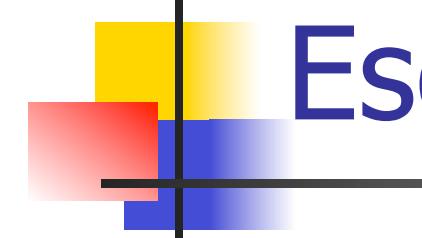
- Sia $F : \mathcal{K} \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$ fam. di funz.
- SE=(KeyGen,Enc,Dec) un cifrario CTR\$
- A avversario IND-CPA contro SE, che fa al piu' q domande
- Esiste B (avversario contro la sicurezza PRF di F) tale che

$$\text{Adv}_{SE}^{\text{ind-cpa}}(A) \leq \text{Adv}_F^{\text{prf}}(B) + \frac{0.5\sigma^2}{2^\ell}$$



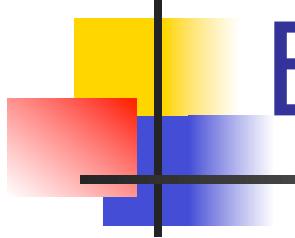
Interpretazione pratica

- Dimostriamo la sicurezza di un cifrario assumendo che determinate componenti siano sicure.
- I teoremi ci dicono che CTR è un cifrario ben congegnato.
- Esso è sicuro, purché si utilizzi un buon cifrario a blocchi come primitiva di base.
- È importante capire che anche l'efficienza della riduzione ha un peso.



Esempio

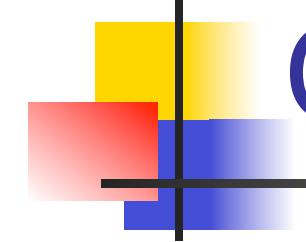
- $F = \text{AES}$ ($I = L = 128$), $q = 2^{30}$
- Ogni msg 2^{13} bit $\rightarrow \sigma = 2^{36}$ blocchi
- Voglio cifrare q msg (σ blocchi) con CTR\$.
Posso farlo in modo sicuro?
- A avv contro CTR\$, usando il teorema
costruisco B.
- Vogliamo valutare quanto e' grande il
vantaggio (come prf su AES) di B



Esempio (cont.)

- B fa $\sigma=2^{36}$ domande.
- Assumendo che AES e' sicuro, possiamo ipotizzare che il vantaggio di B non puo' essere superiore a $\sigma^2/2^{128}$.
- Il teorema ci dice

$$\text{Adv}_{SE}^{\text{ind-cpa}}(A) \leq \frac{\sigma^2}{2^{128}} + \frac{0.5 \cdot \sigma^2}{2^{128}} = \frac{1.5 \cdot 2^{72}}{2^{128}} \leq \frac{1}{2^{55}}$$



Cosa ci dice questo esempio?

- Se cifriamo piu' di $\sigma=2^{l/2}$ blocchi con CTR\$ non possiamo piu' dimostrare alcuna sicurezza, indipendentemente da quanto e' buono il cifrario a blocchi che utilizziamo.
- D'altro canto CTRC rimane sicuro fino a σ blocchi.
- Generalmente tale distinzione puo' non avere grande importanza.
- Ma e' lo stesso importante capirla!



Sicurezza del modo CBC\$

- Sia $E : \mathcal{K} \times \{0, 1\}^n \rightarrow \{0, 1\}^n$ un cifrario a blocchi e $\text{SE} = (\text{KeyGen}, \text{Enc}, \text{Dec})$ il cifrario CBC\$.
- A avv. IND-CPA contro SE, che fa al piu' q domande (per un totale di σ n-bit blocchi)
- Esiste B (avversario contro la sicurezza PRF di E) tale che

$$\text{Adv}_{SE}^{\text{ind-cpa}}(A) \leq \text{Adv}_E^{\text{prf}}(B) + \frac{\sigma^2}{2^{n+1}}$$

Attacchi a crittotesto scelto (CCA)

- Fino ad ora abbiamo considerato solo attacchi CPA.
- Un modello piu' forte e' quello CCA
- Tale modello potrebbe sembrare innaturale.

Definizione (ind-cca)

- SE=(KeyGen, Enc, Dec) cifrario simmetrico

$Esp_{SE}^{ind\text{-}cca\text{-}1}(A)$

$K \leftarrow_R KeyGen$

$b \leftarrow A^{Enc_k(LR(.,.,1)), Dec_k(.)}$

If A imbroglia Return 0
else return b

$Esp_{SE}^{ind\text{-}cca\text{-}0}(A)$

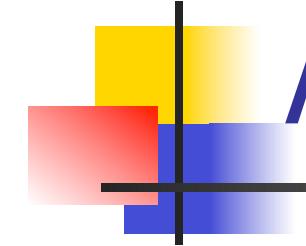
$K \leftarrow_R KeyGen$

$b \leftarrow A^{Enc_k(LR(.,.,0)), Dec_k(.)}$

If A imbroglia Return 0
else return b

$$Adv^{ind\text{-}cca}(A) = |\Pr[Esp_{SE}^{ind\text{-}cca\text{-}1}(A) = 1] - \Pr[Esp_{SE}^{ind\text{-}cca\text{-}0}(A) = 1]|$$

A imbroglia se interroga $D_k(.)$ su un crittotesto gia' restituito da $Enc_k(LR(.,.,1))$



Attacco CCA ai modi CTR

- Guardiamo il caso CTR\$
- Sia (r, C) e' un crittotesto ottenuto da M .
- Cambiamo un bit (i -esimo) di C .
Otterremo C' che cifra M' .
- M e' diverso da M' solo nell' i -esimo bit

$\text{Enc}_k(M)$ Modo CTR\$

$m \leftarrow \lceil |M|/L \rceil; R \leftarrow_R \{0, 1\}^\ell$

$Pad \leftarrow F_k(R + 1) || F_k(R + 2) || \dots || F_k(R + m)$

$Pad \leftarrow$ Primi $|M|$ bits di Pad

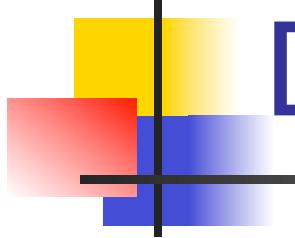
$C' \leftarrow M \oplus Pad$

$C \leftarrow R || C'$

return C

- Sia $F : \mathcal{K} \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$ fam di funzioni e $\text{SE}=(\text{KeyGen}, \text{Enc}, \text{Dec})$ un cifrario CTR\$.
- Esiste A avversario che fa una sola domanda (a ciascun oracolo) e lavora in tempo t (lineare in ℓ e L)

$$\text{Adv}_{\text{SE}}^{\text{ind-cca}}(A) = 1$$



Descrizione dell'avversario

$A^{\text{Enc}_k(LR(\cdot, \cdot, b)), \text{Dec}_k(\cdot)}$

$M_0 \leftarrow_R 0^\ell, M_1 \leftarrow_R 1^\ell$

$(r, C) \leftarrow \text{Enc}_k(LR(M_0, M_1, b))$

$C' \leftarrow C \oplus 1^\ell$

$M \leftarrow \text{Dec}_k((r, C'))$

If $M = M_0$ return 1 else return 0.

$$M_0 \leftarrow 0^l$$

$$C' = C \oplus 1^l = M \oplus PAD \oplus 1^l = 0^l \oplus PAD \oplus 1^l = PAD \oplus 1^l$$

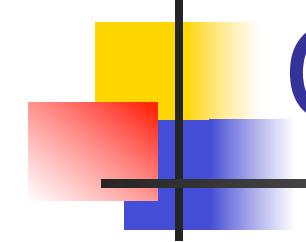
$$M = PAD \oplus PAD \oplus 1^l = 1^l$$

~~~~~

$$M_1 \leftarrow 1^l$$

$$C' = C \oplus 1^l = M \oplus PAD \oplus 1^l = 1^l \oplus PAD \oplus 1^l = PAD \oplus 0^l$$

$$M = PAD \oplus PAD \oplus 0^l = 0^l$$



# Considerazioni

---

- L'attacco e' indipendente dalla qualita' della funzione F.
- Se anche F fosse una funzione casuale, l'avversario avrebbe lo stesso vantaggio.
- Questo dimostra che gli attacchi CCA sono potenzialmente molto piu' devastanti degli attacchi CPA.