

CIFRARI A SICUREZZA INCONDIZIONATA

Nascondono l'informazione con certezza assoluta.

Anche nel caso $P = NP$

- COMPUTATIONAL

Porta da il crittoanalista

abbia a disposizione tutte ragionevoli computazionalmente
e funzionano fino a che $P \neq NP$

In un cifrario perfetto un crittoanalista non arricchisce il livello
di informazioni di un crittoanalista

Un cifrario è perfetto se la sua sicurezza è garantita qualunque sia
l'informazione catturata dal crittoanalista.

MSG : spazio dei messaggi

CRITTO : spazio dei crittogrammi

M : variabile aleatoria che descrive il comportamento
del mittente, assume valori in MSG

C : variabile aleatoria che descrive la comunicazione
sul canale, assume valori in CRITTO

$P(M=m)$ probabilità che A voglia inviare $m \in MSG$

$P(M=m | C=c)$ probabilità a posteriori che A abbia
invia $m \in MSG$ posto che sul canale
sfia transitando $c \in CRITTO$

Def. Un cifrario è perfetto se

$$\forall m \in MSG, \forall c \in CRITTO. P(M=m | C=c) = P(M=m)$$

Scenario: 30 cifrando/cifrante conosci:

- la distribuzione di probabilità con cui il mittente invia i messaggi;
- cifrario utilizzato;
- spazio delle chiavi.

e.g. 1) $\bar{m} \in MSG$

$$P(M=\bar{m}) = p > 0 \text{ con } p \in (0;1)$$

$$\exists \bar{c}. P(M=\bar{m} | C=\bar{c}) = 1$$

$$2) \exists \bar{c}. P(M=\bar{m} | C=\bar{c}) = 0$$

Sia in 1) sia in 2) E raffina la sua conoscenza,

in tutti i casi intermedi pure; l'unico caso in cui

non lo raffina è quello in cui il cifrario è perfetto.

TEOREMA di SHANNON

In un cifrario perfetto il numero delle chiavi deve essere maggiore
uguale il numero dei messaggi possibili.

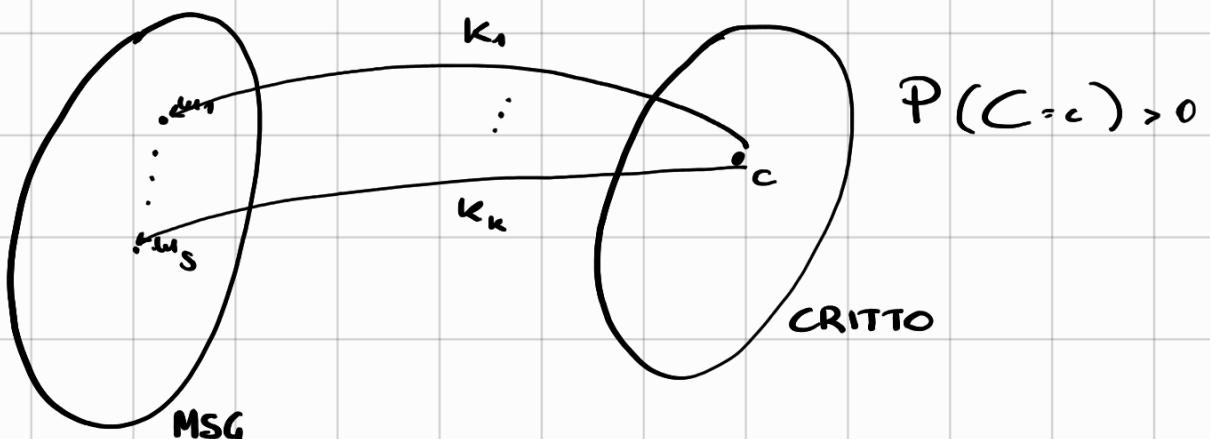
Definisco possibile un messaggio se t.c. $P(M=m) > 0$

Dim: Si procede per assurdo:

$$N_K = \#\{ \text{insieme delle chiavi} \}$$

$$N_M = \#\{ \text{dei messaggi possibili} \}$$

Si pone $N_M > N_K$



cavat: Chiavi diverse possono portare allo stesso

messaggio, $\rightarrow c$ corrispondono

$S \leq N_K$ messaggi possibili

ma $N_K < N_M$:

$S \leq N_K < N_M \Rightarrow \exists m' \in MSG \text{ possibili } t_c \in C \text{ non}$
 $\text{puo' corrispondere a } m'$

$$\Rightarrow P(M=m'|C=c) = 0 \neq P(M=m')$$

\Rightarrow il cifrario non è perfetto

$\Rightarrow Th.$

Q.E.D.

ONE-TIME PAD (Blocco monouso)

KEY := Spazio delle chiavi

Si pone $MSG, CRITTO, KEY = \{0,1\}^n$

$$m, k \in \{0,1\}^n$$

$$c = m \oplus k \quad \text{con } \oplus \text{ xor bit a bit (somma mod 2)}$$

L'unica informazione a disposizione di E è se

l'ultimo bit di m, c sono uguali

$$\left. \begin{array}{l} \text{CIFRATURA} \\ \left\{ \begin{array}{l} m = m_1 \dots m_n \\ k = k_1 \dots k_n \\ c = c_1 \dots c_n, \text{ con } V_{sign.} c_i = m_i \oplus k_i \end{array} \right. \end{array} \right.$$

$$\text{DECIFRAZIONE } m = c \oplus k$$

$$\text{dim: } m = m \oplus k \oplus k = m \oplus (k \oplus k) = m$$

Si dimostra che one-time pad è perfetto:

Si pone. Una MSG. $|M| = n$. A parole: tutti i messaggi sono lunghi n

(si fa padding per messaggi più brevi)

o si cifra a blocchi lunghi n quelli più lunghi)

- $\forall m \in \{0,1\}^n$, $P(M=m) = p > 0$ A parole: tutte le sequenze di n bit formano messaggi possibili.

(si adegua probabilità bassa ma non nulla alle sequenze binarie prive di significato)

- Chiave scelta perfettamente a caso

per ogni messaggio.

Sotto queste condizioni è perfetto e impiega un numero minimo di chiavi

Dimo: 1) Minimalita':

segue immediatamente da $N_m = N_k = 2^n$.

2) Cifrario perfetto:

$$\forall m, c \quad P(M=m | C=c) = P(M=m)$$

$$P(M=m | C=c) = \frac{P(M=m \wedge C=c)}{P(C=c)}$$

Per proprietà di XOR, fissato m , chiavi

diverse producono cifogrammi diversi

$\Rightarrow \exists! k \mid m \xrightarrow{k} c$ con m, c fissati

$\Rightarrow P(C=c)$ è la probabilità di scegliere
a caso l'unica k che porta m in c
 $= (1/2)^n$

$$\frac{P(M=m \wedge C=c)}{P(C=c)} = \frac{P(M=m)}{P(C=c)} = P(M=m)$$

$\{M=m\}, \{C=c\}$ sono eventi indipendenti

oss. di rimuovere la seconda condizione e si considerano solo messaggi significativi

$$\#\text{MSG}_{\text{eng}} = 1,1^n \quad N_k, N_{\text{MSG}} = 1,1^n < 2^n$$

Si suppone di poter descrivere le chiavi con t bit con $2^{t, 1, 1}$

$$\text{cioè } t \geq \log_2 1,1 \approx 0,12 n$$

Per confondere E' e' però necessario
che molte coppie $(m, k) \in (\text{MSG}, \text{KEY})$

producono lo stesso crittogramma.

E' necessario pone $\#(MSG, KEY) \gg \# CRITTO$

Supponiamo chiavi lunghe t

α^n messaggi

$$\# CRITTO = 2^n;$$

Dove valere: $(2^t \cdot \alpha^n) \gg 2^n$

$$n \log_2 \alpha + t \gg n$$

$$t \gg n - n \log_2 \alpha$$

Nel caso di messaggi solo significativi: $\log_2 \alpha = 0.12$

$$t \gg n - 0.12n$$

$$t \gg 0.88n$$

Non permette di risparmiare

sulla lunghezza delle chiavi