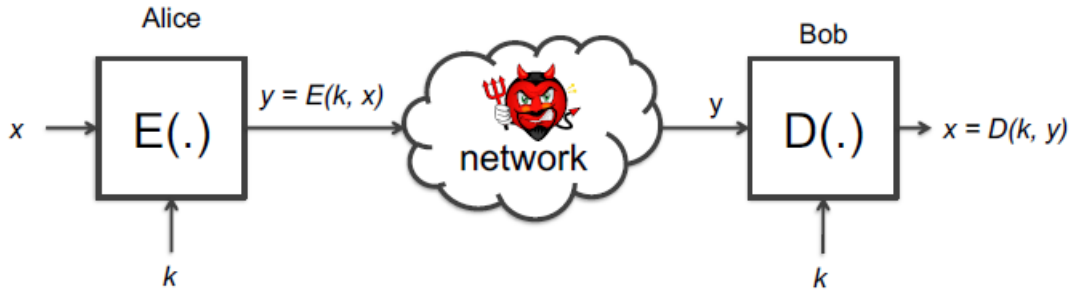
Contents

1 Crittografia Simmetrica

1.1 Introduzione

Se parliamo di crittografia Simmetrica, stiamo considerando uno schema in cui sono presenti almeno 2 agenti (Bob, Alice) che devono scambiare un messaggio in maniera sicura senza che un terzo (che potrebbe avere intenzioni malevole) possa capire il messaggio. Il modello quindi prevede una funzione di cifrazione (Enc) e una funzione di decifrazione (Dec), una chiave condivisa (k) e due tipi di messaggio, x detto messaggio in chiaro (plaintext) e y detto messaggio cifrato (ciphertext) ; la funzione di criptazione è determinata da un algoritmo che pubblicamente conosciuto, quello che invece rende sicuro l'utilizzo della cifratura simmetrica è la segretezza e lunghezza della chiave.



1.1.0.1 Definizione di Cifrario Un cifrario, o schema di cifratura, è defito su una tripletta composta da (K,P,C) usata nelle funzioni di (Gen,Enc,Dec) definite con questi domini:

$$Gen : Z^+ \rightarrow K funzionegeneratricedichiavi \quad (1)$$
$$Enc : P \times K \rightarrow C funzionedicifratura \quad (2)$$
$$Dec : C \times K \rightarrow P funzionedidecifratura \quad (3)$$
$$x \in P, y \in C, k \in K \quad (4)$$
$$y = Enc(k, x) \quad (5)$$
$$x = Dec(k, y) \quad (6)$$

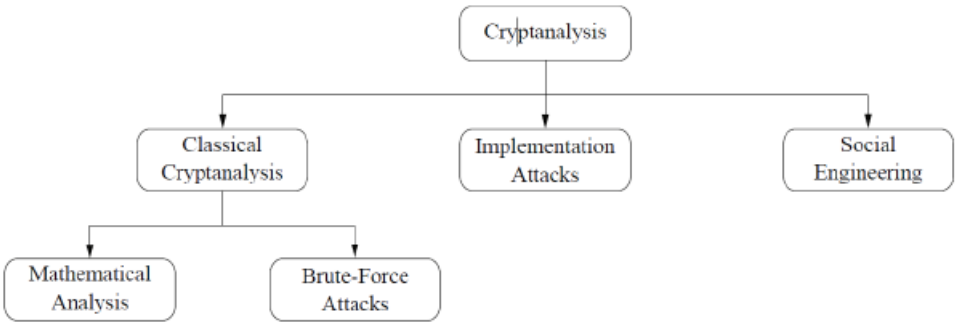
Proprieta di cifratura La cifratura deve anche rispettare le seguenti proprieta:

Correttezza : $\forall p \in P \wedge k \in K, \exists Dec(k, Enc(k, p)) = p$

Sicurezza : Un cifrario simmetrico è sicuro $\iff \forall (p, c), p \in P \wedge c \in C \Rightarrow$

- dato c ciphertext, è "difficile" determinare p plaintext senza conoscere la chiave k, e viceversa
- è "difficile" determinare k chiave, ammenoche non sia stata già usata una volta

Esempio : Sostituzione monoalfabetca Vediamo quindi un tipo di cifratura a sostituzione, dove la chiave è la permutazione dell' alfabeto. L' algoritmo prevede la sostituzione delle lettere della parola con le corrispondenti dell' alfabeto shiftato, e per decriptare si usa l' algoritmo al contrario. Le chiavi quindi possono essere circa 26! cioè circa $4 \cdot 10^26$, quindi tentare un attacco a forza bruta non è possibile, ma è possibile applicare tecniche di Crittoanalisi analizzando alcune proprietà che riguardano i linguaggi come : la frequenza delle lettere, la generalizzazione delle coppie e triple di lettere, la frequenza delle parole corte se sono identificati i separatori.



1.1.0.2 Crittoanalisi

Complessità dell'attacco Viene definito da:

Complessità dei Dati: Numero previsto di unità dei dati in ingresso richiesti.

Complessità di Storage: Numero previsto delle unità di storage richiesti.

Complessità di Elaborazione: Numero previsto di operazioni richiesti per processare i dati in input o/e per riempire la memoria con dati.

Tipi di attacco Si possono classificare in:

- Ciphertext-only attack
- Known-plaintext attack
- Chosen-plaintext attack (CPA)

Se un metodo è sicuro contro i CPA, allora è sicuro anche contro gli altri.

Principi di Kerchoff

Massima di Kerckhoffs Un sistema crittografico dovrebbe rimanere sicuro anche se tutto del sistema, tranne la chiave, è di pubblico dominio.

Massima di Shannon Il nemico conosce il sistema.

Massima di Bruce Schneier Meno e più semplici sono i segreti da custodire per garantire la sicurezza del sistema, più facile sarà mantenerne la sicurezza.

Consigli per mantenera la sicurezza facile

- Le chiavi sono piccoli segreti.
- Conservare piccoli segreti è più facile che conservare grandi segreti.
- Sostituire piccoli segreti, una volta eventualmente compromessi, è più facile (ed economico) che sostituire grandi segreti.

1.1.0.3 Esempi di cifrari

Cifrario di cesare Siano PT, CT e K elementi dell'anello Z_{26}.

- **Cifratura (Encryption):**
$$y = x + k \mod 26$$
- **Decifratura (Decryption):**
$$x = y - k \mod 26$$
- **Esempio:**
  - Testo in chiaro (Plaintext, x): "`ATTACK`" $\Rightarrow$
  $$x = (0, 19, 19, 0, 2, 10)$$
  - Chiave (k):
  $$k = 17$$
  - Testo cifrato (Ciphertext, y):
  $$y = (0+17, 19+17, 19+17, 0+17, 2+17, 10+17) \mod 26 = (17, 10, 10, 17, 19, 1)$$
  - Risultato: "`RKKRTB`"

Cifrario Affine Definizione
Siano $a, b, x, y \in Z_{26}$.

- **Cifratura (Encryption):**
$$y = a \cdot x + b \mod 26$$
- **Decifratura (Decryption):**
$$x = a^{-1} \cdot (y - b) \mod 26$$
- La chiave è $k = (a, b)$, con $\gcd(a, 26) = 1$
- **Esempio:**
  - Testo in chiaro (Plaintext): "`ATTACK`" $\Rightarrow (0, 19, 19, 0, 2, 10)$
  - Chiave $k = (9, 13)$
  - Cifratura:
  $$y = (9 \cdot x + 13) \mod 26$$
  $$y = (13, 2, 2, 13, 5, 25)$$
  - Testo cifrato (Ciphertext): "`NCCNFZ`"

Calcolo dello spazio delle chiavi
Lo spazio delle chiavi si calcola come:

$$\text{Spazio delle chiavi} = N_a \cdot N_b$$

Dove:

- N_a è il numero di valori possibili di a tali che $\gcd(a, 26) = 1$.
In questo caso: $N_a = 12$
- N_b è il numero dei possibili valori di shift b in Z_{26}.
Quindi: $N_b = 26$

Pertanto, lo spazio delle chiavi è:

$$12 \cdot 26 = 312$$

1.2 Cifrario Perfetto

1.2.1 Introduzione

Nel contesto della sicurezza crittografica, si considera un attaccante con la capacità di condurre un *ciphertext-only attack*, ovvero un attacco in cui l'unica informazione disponibile è il testo cifrato.

1.2.1.1 Requisiti di sicurezza

Un cifrario è considerato sicuro se soddisfa i seguenti requisiti:

- L'attaccante non è in grado di recuperare la chiave segreta.
- L'attaccante non è in grado di risalire al testo in chiaro (plaintext).

1.2.1.2 Intuizione di un cifrario perfettamente sicuro

L'idea alla base di un cifrario perfettamente sicuro è che, indipendentemente da qualsiasi informazione pregressa che l'attaccante possa avere sul testo in chiaro, il testo cifrato non deve fornire alcuna informazione aggiuntiva su di esso.

1.2.2 Approccio Probabilistico

Nel contesto della crittografia, il messaggio M viene modellato come una variabile aleatoria, secondo una distribuzione di probabilità detta *distribuzione del plaintext*. Ad esempio, si potrebbe avere:

- $\Pr[M = \text{“attack today”}] = 0.7$
- $\Pr[M = \text{“don't attack”}] = 0.3$

Queste probabilità rappresentano la *conoscenza a priori* che un attaccante potrebbe avere riguardo al contenuto del messaggio.

Il generatore di chiavi Gen() definisce una distribuzione di probabilità sulla chiave K, ovvero:

$$\Pr[K = k] = \Pr[k \leftarrow \text{Gen}()]$$

Le variabili aleatorie M (messaggio) e K (chiave) si assumono indipendenti.

1.2.2.1 Processo di generazione del ciphertext

1. Si sceglie un messaggio m secondo la distribuzione di M.
2. Si genera una chiave k da Gen().
3. Si calcola il testo cifrato $c \leftarrow E_k(m)$.

Il testo cifrato C risultante è anch'esso una variabile aleatoria, e il processo di cifratura definisce una distribuzione di probabilità indotta su C.

1.2.2.2 Segretezza Perfetta

La *segretezza perfetta* è un concetto fondamentale introdotto da Claude Shannon nel 1949. In modo informale, si riferisce alla condizione in cui il testo cifrato non rivela alcuna informazione sul testo in chiaro. Formalizziamo l'idea di “informazione sul plaintext” in termini di distribuzione di probabilità.

Un sistema di cifratura gode di segretezza perfetta se, per ogni messaggio $m \in M$ e ogni testo cifrato $c \in C$ tale che $\Pr[C = c] > 0$, vale la seguente uguaglianza:

$$\Pr[M = m \mid C = c] = \Pr[M = m]$$

Questo implica che la probabilità a posteriori che un messaggio sia m, dato il testo cifrato c, è uguale alla probabilità a priori che il messaggio fosse m: osservare il testo cifrato non fornisce alcuna informazione aggiuntiva.

Una formulazione equivalente della segretezza perfetta è la seguente:

$$\forall m, m' \in M, \forall c \in C, \quad \Pr[E_k(m) = c] = \Pr[E_k(m') = c]$$

In altre parole, la distribuzione del testo cifrato è indipendente dal messaggio originale. La distribuzione a priori del plaintext (prima di osservare il ciphertext) e quella a posteriori (dopo aver osservato il ciphertext) devono coincidere.

1.2.2.3 Teorema di Shannon

Teorema di Shannon

- In un cifrario perfetto, $|K| \geq |M|$
- Ovvero, il numero di chiavi non può essere inferiore al numero di messaggi

Dimostrazione (per assurdo):

1. Supponiamo che $|K| < |M|$
2. Deve valere $|C| \geq |M|$, altrimenti il cifrario non è invertibile
3. Quindi, $|C| > |K|$
4. Sia $m \in M$ tale che $\Pr[M = m] \neq 0$; si calcoli $c_i \leftarrow E(k_i, m)$ per ogni $k_i \in K$
5. Per il punto (3), esiste almeno un c tale che $c \neq c_i$ per ogni $k_i \in K$
6. Allora $\Pr[M = m \mid C = c] = 0$, che è diverso da $\Pr[M = m]$, contraddicendo la definizione di cifrario perfetto

Fatto. Sia $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ uno schema di cifratura con $|M| = |K| = |C|$. Lo schema è perfettamente sicuro se e solo se:

1. $\forall k \in K$ è scelto con probabilità uniforme $1/|K|$ da Gen
2. $\forall m \in M \wedge c \in C, \exists! k \in K : E_k(m) = c$

Nota:

- La condizione 1 è facile da verificare
- La condizione 2 non richiede il calcolo di probabilità

1.2.2.4 Sicurezza incondizionata

La perfetta segretezza è equivalente alla sicurezza incondizionata. In questo modello, si assume che un avversario disponga di risorse computazionali illimitate. L'osservazione del testo cifrato non fornisce quindi alcuna informazione utile all'avversario riguardo al messaggio originale.

Condizioni necessarie

- I bit della chiave devono essere scelti in modo veramente casuale
- La lunghezza della chiave deve essere maggiore o uguale alla lunghezza del messaggio (secondo il teorema di Shannon)

1.2.2.5 Indistinguibilità perfetta

Sia $\Pi = (G, E, D)$ uno schema di cifratura definito su insiemi K, M, C.
Π ha *indistinguibilità perfetta* se e solo se:

$$\forall m_1, m_2 \in M,\ |m_1| = |m_2|,\ \forall c \in C,\ \text{con } k \leftarrow G \text{ uniforme}: \quad \Pr[E(k, m_1) = c] = \Pr[E(k, m_2) = c]$$

Fatto Π ha indistinguibilità perfetta $\iff$ Π è perfettamente sicuro.

1.2.3 One-Time Pad