Università' degli studi di Roma "Tor Vergata"
Teoria della sicurezza e crittografia
Lezione 02

Authors: Angelo Silvestri, Marcello Politi, Samir Salman

Ottobre 2019

1 Go

Go(chiamato anche Golang) è un linguaggio di programmazione open source sviluppato da Google nel 2009. Caratteristiche principali: la sintassi è vicina al linguaggio C eccetto per la dichiarazione dei tipi e per la mancanza di parentesi tonde nei costrutti for e if, ha un sistema di garbage collection che si occupa autonomamente della gestione della memoria, sfrutta le goroutines(chiamate anche fiber),permette valori di ritorno multipli e molto altro. Esempio di codice in Go(Hello World):

```
package main
import "fmt"
func main() {
fmt.Printf("Hello, World\n")
}
```

2 Classi di crittografia

Possiamo dividere la crittografia in varie categorie: Hash, Simmetrica, Asimmetrica, Omomorfica

2.1 Hash

L'hash è una funzione non invertibile che mappa una stringa di lunghezza arbitraria in una stringa di lunghezza predefinita. La dimensione del dominio di una funzione hash è più grande della dimensione del codominio, quindi per un elemento y del codominio esiste almeno un elemento x del dominio che restituisce y. La funzione hash in crittografia è necessaria per garantire che un messaggio inviato da un ipotetico mittente A ad un ricevente B non venga alterato da un qualche esterno C. La qualità di una funzione hash si misura in base alla diversifcazione del risultato dell'hash rispetto ad una modifca della stringa di

partenza. Ad esempio, se utilizziamo la funzione hash su due stringhe leggermente diverse e la funzione ci restituisce due stringhe molto diverse tra di loro, allora la funzione hash è ritenuta buona.

2.2 Simmetrica

Con **cifratura simmetrica, o a chiave privata**, si intende una tecnica di cifratura che consiste nell' utilizzo dello stesso valore di chiave per cifrare e decifrare il messaggio, ragion per cui mittente e destinatario devono esserne in possesso e da qui il problema di come effettuare lo scambio di chiavi in modo sicuro. Il mittente cifra il messaggio M con la chiave k usando un algoritmo S di cifratura simmetrica.

$$(\mathrm{P};\, \mathrm{k}) = \mathrm{C}$$

Il destinatario riceve il messaggio C cifrato e può decifrarlo applicando l'algoritmo D di decifratura con la chiave k in suo possesso che ha lo stesso valore della chiave che ha usato il mittente per cifrare.

$$\mathrm{D}(\mathrm{C};\, \mathrm{k}) = \mathrm{P}$$

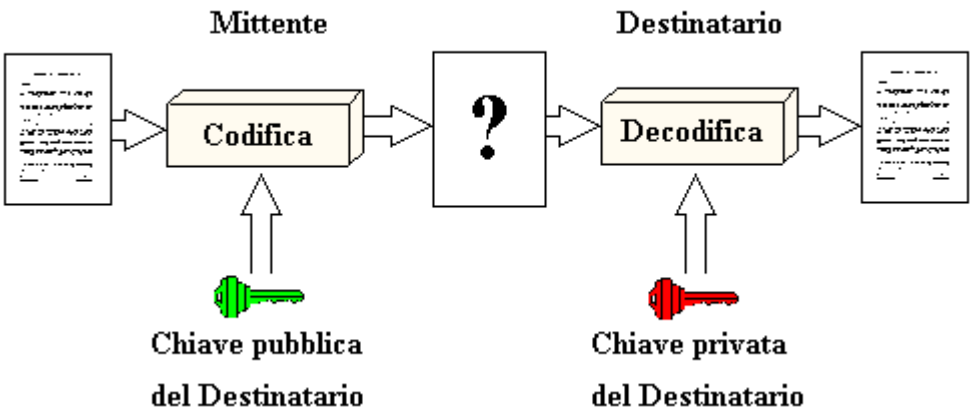
Il vantaggio di questo tipo di cifratura consiste nel fatto di essere molto veloce e semplice da implementare, non è invece adatta a scenari di comunicazione molti a molti dove avere una singola chiave privata per tutti risulterebbe avere poco senso.

2.3 Asimmetrica

La cifratura asimmetrica, conosciuta anche come a coppia di chiavi, a chiave pubblica/privata o anche solo a chiave pubblica, e un tipo di cifratura dove, come si deduce dal nome, ad ogni attore coinvolto nella comunicazione è associata una coppia di chiavi:

- La chiave pubblica, che deve essere distribuita.
- La chiave privata, appunto personale e segreta.

In questo modo si evita il problema connesso alla necessità di uno scambio in modo sicuro dell'unica chiave presente invece nella cifratura simmetrica. Il meccanismo si basa sul fatto che, se con una chiave cifra il messaggio,allora questo potrà essere decifrato solo con l'altra chiave. Il messaggio è cifrato dal mittente con la chiave pubblica del destinatario che la ha opportunamente distribuita a tutti i suoi probabili mittenti. Il messaggio viene decifrato con la chiave privata del destinatario.



Un vantaggio rispetto alla cifratura simmetrica è che in uno scenario di comunicazione molti a molti ognuno ha la sua coppia di chiavi; uno svantaggio è rappresentato dalla cifratura più lenta.

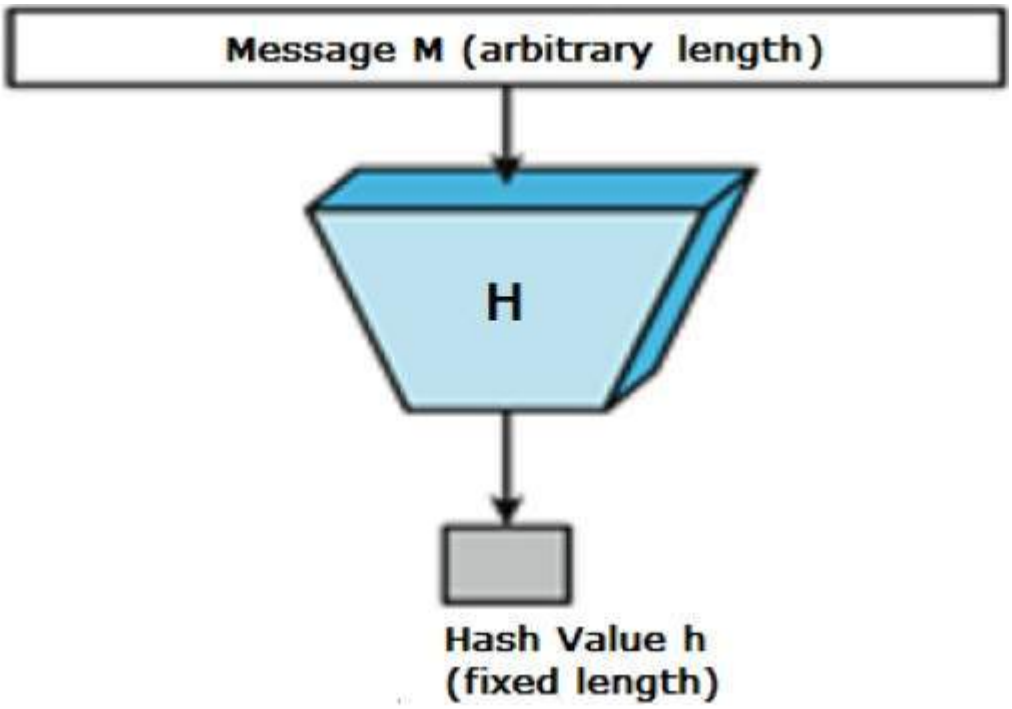
2.4 Funzioni Hash

Si definisce funzione di hash qualunque funzione che trasformi i dati in input in un output di lunghezza costante. Una funzione di hash gode delle seguenti proprietà:

- Il messaggio da cifrare M può essere di qualsiasi dimensione;
- Il messaggio cifrato h è sempre della stessa dimensione
- La funzione Hash H(M) è facile da calcolare
- Ogni valore di h ha la stessa probabilità di essere restituito

Al fine di essere utilizzata in crittografia, una funzione di hash deve godere anche delle seguenti proprietà:

- Unidirezionalità: noto h deve essere computazionalmente impossibile trovare M tale che H(M) = h.
- Resistenza debole alle collisioni: conoscendo M deve essere computazionalmente impossibile trovare M' tale che H(M) = H(M').
- Effetto valanga: una piccola modifica di M deve alterare tutto h.



2.4.1 Il paradosso del compleanno

2.4.1 Paradosso del compleanno La probabilità che in un gruppo di persone almeno due siano nate nello stesso giorno è molto più alta di quanto si possa immaginare. Con sole 23 persone si ottiene una probabilità del 50(

Questo paradosso ha importanti ricadute nella crittografia e nel dimensionamento del blocco da cifrare, in particolare per indicare che le funzioni hash crittografiche abbiano la proprietà di "resistenza forte alle collisioni".

Ad esempio una funzione di hash che produce un risultato su n bit sarà reputata insicura quando verranno generati $2^{n/2}$ risultati in quanto si ha la probabilità di oltre il 50 (Nello specifico per calcolare la probabilità P(g) che in un gruppo g ci siano 2 persone che compiono gli anni nello stesso giorno ci calcoliamo prima la probabilità che non accada ossia P1(g) . Il porcesso è il seguente: data una qualunque persona del gruppo (indipendentemente dalla data del suo compleanno), vi sono 364 casi su 365 in cui il compleanno di una seconda persona avvenga in un giorno diverso; se si considera una terza persona, ci sono 363 casi su 365 in cui compie gli anni in un giorno diverso dalle prime due persone e via dicendo. Esprimendo in formule matematiche quanto sopra, la probabilità che tutti i g compleanni cadano in date diverse è:

$$P1(g) = \frac{364}{365}\frac{363}{365}...\frac{365-g+1}{365} = \frac{364!}{365^{g-1}(365-g)!}$$

e dunque la probabilità del suo evento complementare, cioè che esistano almeno due compleanni uguali è

$$P(g) = 1 - P1(g) = 1 - \frac{364}{365}\frac{363}{365}...\frac{365-g+1}{365} = 1 - \frac{364!}{365^{g-1}(365-g)!}$$