Introduzione all'Informatica

Loriano Storchi

loriano@storchi.org

http://www.storchi.org/

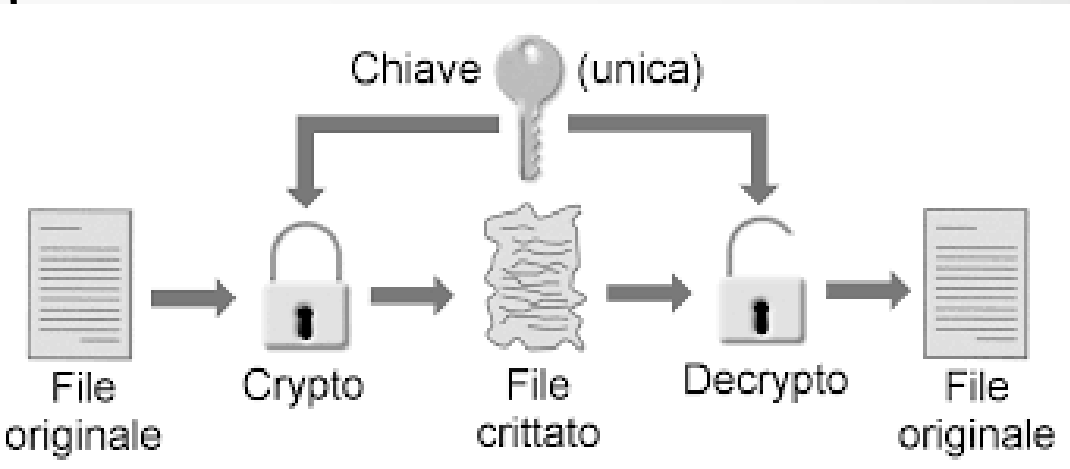
CRITTOGRAFIA CENNI DI BASE

Crittografia cenni di base

- Nei calcolatori le informazioni sono memorizzate come sequenze di bit
- **Le tecniche crittografiche modificano queste sequenze (stringhe) per ottenere sequenze diverse che poi possono essere trasmesse e ritrasformate dal riceventi nella sequenza originale**. Le funzioni matematiche usate nella trasformazione usano una o piou' chiavi segrete
  - **Cifratura Simmetrica:** usata addirittura a partire dagli Egizi e dagli antichi Romani
  - **Cifratura Asimmetrica**: risale agli anni 70

Cifratura simmetrica

- **La chiave usata per cifrare e decifrare, e quindi da mittente e destinatario e’ unica**. Ad esempio cifrarion di Cesare sostituire ogni carattere con altro sfasato di k posti (la chiave e’ il valore di k)
- Cifrario monoalfabetico



a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
m	n	b	v	c	x	z	a	s	d	f	g	h	j	k	l	p	o	i	u	y	t	r	e	w	q

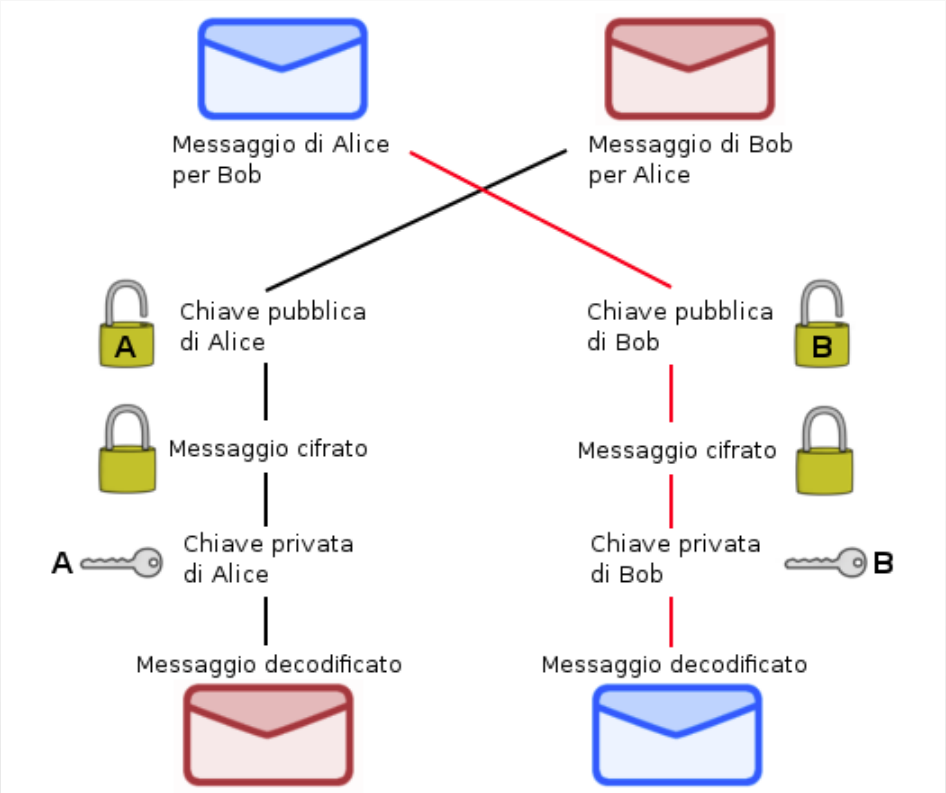
Cifratura simmetrica

- **Devo trovare un modo sicuro per scambiare la chiave** segreta che e' unica per mittente e destinatario
- Attacco a forza bruta (**brute-force**) nel caso ad esempio del cifrario di cesare provo tutti i valori di k e vedo quando ottengo frasi e parole "corrette"
- Esempi di algoritmi moderni : **Blowfish, Twofish, Standard DES o Triple DES, Standard AES**. Sono tutti basati su problemi matematici difficili da risolvere se non si conosce la chiave segreta

CIFRATURA ASIMMETRICA

Cifratura asimmetrica

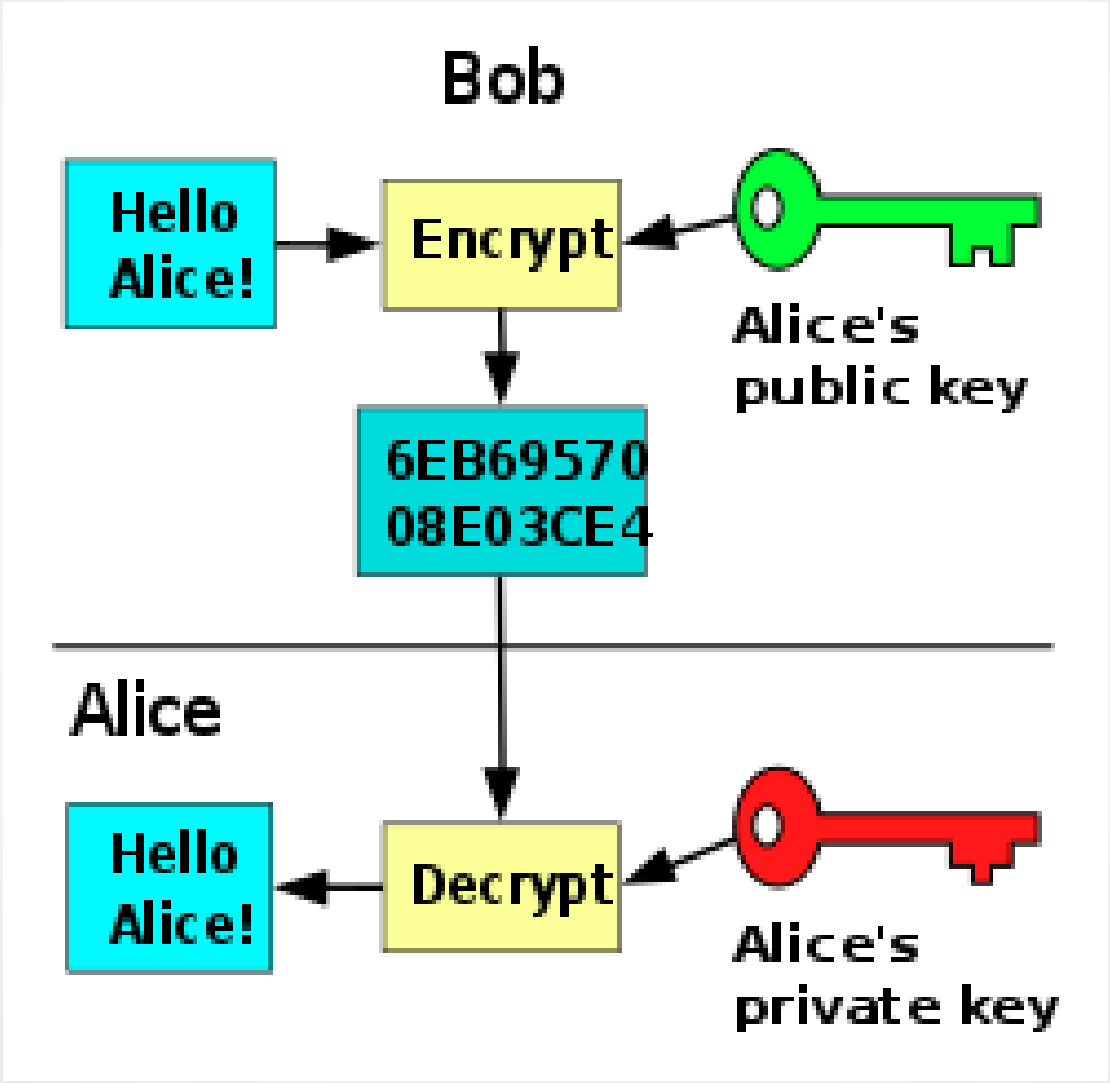
- **Le chiavi usate per cifrare e decifrare sono diverse**. Chiave privata che deve essere tenuta segreta serve a decifrare, quindi a riottenere il messaggio originale, quella pubblica a cifrare



Cifratura asimmetrica

- **Al momento in cui l’utente genera la coppia di chiavi** deve custodire gelosamente la chiave **privata (segreta) KS** ed invece distribuire solo quella **pubblica KP**. **KS ad esempio in una smartcard** e’ sara’ in quel caso la smartcard stessa a operare la cifratura.
- Se l’utente Bob vuole scrivere un messaggio privato all;utente Alice, Bob usera’ la chiave pubblica KP di Alice e spedira’ il **crittogramma** ottenuto, Solo Alice in possesso della parte privata della chiave KS potra’ riottenere il messaggio originale (in chiaro)
- Cifratura asimmetrica usata anche nei processi di autenticazione

Cifratura asimmetrica



Cifratura asimmetrica: RSA

- L'algoritmo piu' noto ed usato e' l'**RSA** (nomi degli inventori Rivest, Shamir, Adleman)
- Anche per autenticazione o garantire integrita' di un documento (anche firma digitale)
- **Basato su numeri primi, cioe' quei numeri naturali che sono divisibili solo per 1 o per se stessi (2, 3, 5, ..., $19249 \cdot 2^{13018586} + 1$)**
- **In pratica KS e KP sono i fattori primi di un numero grande**
- Interesse nei numeri primi e negli algoritmo di fattorizzazione (**algoritmo di Shor quantum computer**)

PGP - OpenPGP

OpenPGP

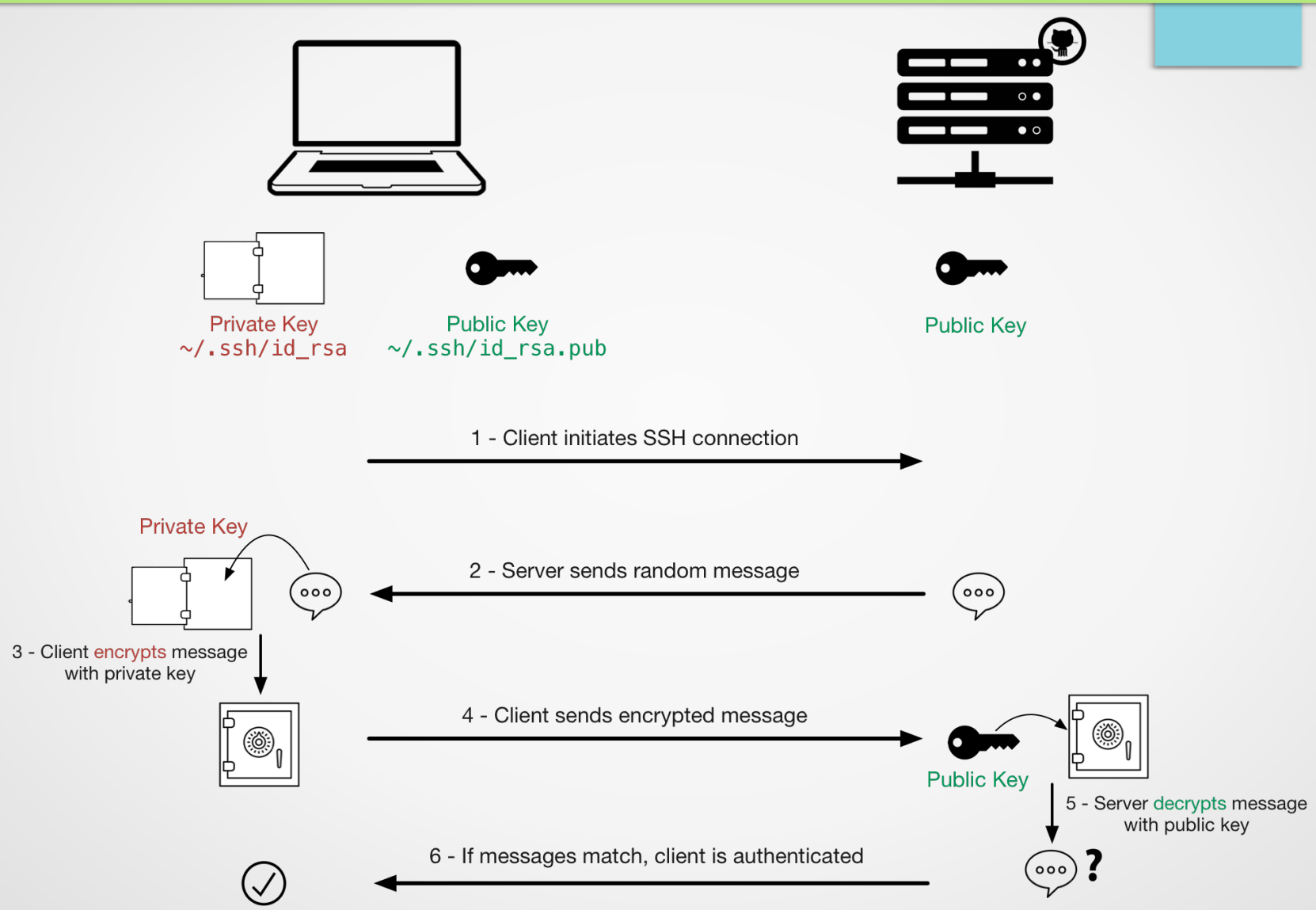
- RSA è un algoritmo (in realtà, due algoritmi: uno per la crittografia asimmetrica e uno per le firme digitali - con diverse varianti). **PGP** è un software, **ora un protocollo standard, generalmente noto come OpenPGP**.
- **OpenPGP definisce i formati per gli elementi di dati che supportano la messaggistica sicura**, con crittografia e firme e varie operazioni correlate come la distribuzione delle chiavi.
- Come protocollo, **OpenPGP si basa su una vasta gamma di algoritmi crittografici.** Tra gli algoritmi che OpenPGP può utilizzare è **RSA**.

OpenPGP

- **Philip R. Zimmermann** è il creatore di **Pretty Good Privacy**, un pacchetto software di crittografia email. Originariamente concepito come uno strumento per i diritti umani, il **PGP** è stato **pubblicato gratuitamente su Internet nel 1991**.
- Ciò ha reso Zimmermann l'obiettivo di un'indagine criminale di tre anni, perché il governo riteneva che le restrizioni all'esportazione degli Stati Uniti per software crittografico fossero violate quando PGP si diffuse in tutto il mondo.
- **GNU Privacy Guard** (GnuPG o GPG) è un software libero progettato per sostituire la suite crittografica PGP.

AUTENTICAZIONE

Esempio autenticazione SSH



FIRMA DIGITALE

Firma Digitale

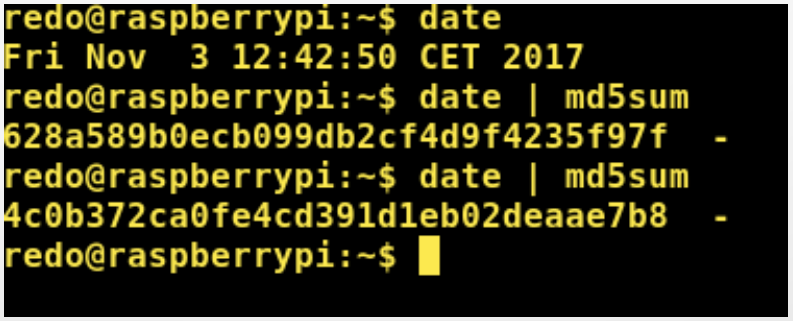
- Viene apposta a documenti digitali in modo da garantire
  - **Autenticità** : quindi garanzia della provenienza del messaggio
  - **Integrità** : il messaggio non e' stato modificato in alcun modo
  - **Non ripudiabilita** : La fonte del messaggio non puo' disconoscere di averlo firmato
- Solo il mittente puo' apporre quelle particolare firma
- Chiunque puo' verificare chi ha firmato il messaggio (documento digitale, testo, suono, immagine, filmato)
- **Ingredienti di base sistema di cifratura asiemmetrico e funzione di hash**

Firma Digitale: Modalita’ di firma

- **CASDES**: estensione **pdf.p7m** il file puo’ essere letto con i software di firma (File protector, DiKe ....)
- **PADES**: estensione **pdf** il file viene firmato con software di firma e letto con Acrobat Reader
- **XADES**: estensione **xml.p7m** , il file xml e’ letto in automatico da software che lo riceve

Firma Digitale: HASH

- Algoritmo di **hash**: **MD5, SHA-1, RIPEMD, SHA-256:**
  - Una funzione che dato un flusso di bit di dimensione variabile restituisce una stringa di lettere o numeri di dimensione fissa (una sorta di bollino unico)
  - La stringa e' un identificativo univoco, la modifica di 1 solo bit del flusso sorgente produce un HASH diverso
  - Non e' invertibile quindi a partire dalla stringa restituita non e' possibile determinare il flusso originale



Firma Digitale: Come funziona

- Alice per firmare un dato oggetto O (un documento ad esempio):
  - **Calcola l'HASH di O** (detto anche **digest**)
  - **Cifra l'HASH ottenuto cona la propria chiave pubblica**
  - **Appende l'HASH cifrato (la firma) assieme alla sua chiave pubblica** all'oggetto O, chiamiamola F
- Bob per verificare la firma di Alice:
  - **Calcola l'HASH di O**
  - **Decifra l'HASH** cifrato (quindi la firma F di Alice che trova assieme al documento) **usando la chiave pubblica di Alice**
  - **Verifica** a questo punto che i valori siano uguali

CA E CERTIFICATI

CA e Certificati

- Come si puo’ essere sicuri che la firma usata sia effettivamente del firmatario ? Parafrasando come posso essere certo dell’associazione utente chiave pubblica ?
- **I Certificati Digitali servono a questo scopo. Essi contengono una seri di informazioni, come ade esempio la chiave pubblica ed i dati dell’utente**
- Cosi’ come i certificati cartacei permettono di avere informazioni a proposito dell’utente
- **CA, Certification Authorithy** , garantisce l’associazione fra firma digitale ed identita’ del titolare

CA e Certificati

- Certificati digitali: e’ costituito da (X.509):
  - **Chiave pubblica** del firmatario
  - Dalla **sua identita’** (nome cognome data di nascita etc etc)
  - **Data di scadenza** della chiave pubblica
  - **Nome della CA che lo ha rilasciato**
  - **Firma digitale della CA che ha messo il certificato**
- **Se ci fidiamo della CA (Autorita’ di Certificazione) possiamo verificare la sua firma e dunque l’idenrtita’ del firmatario**

CA e Certificati

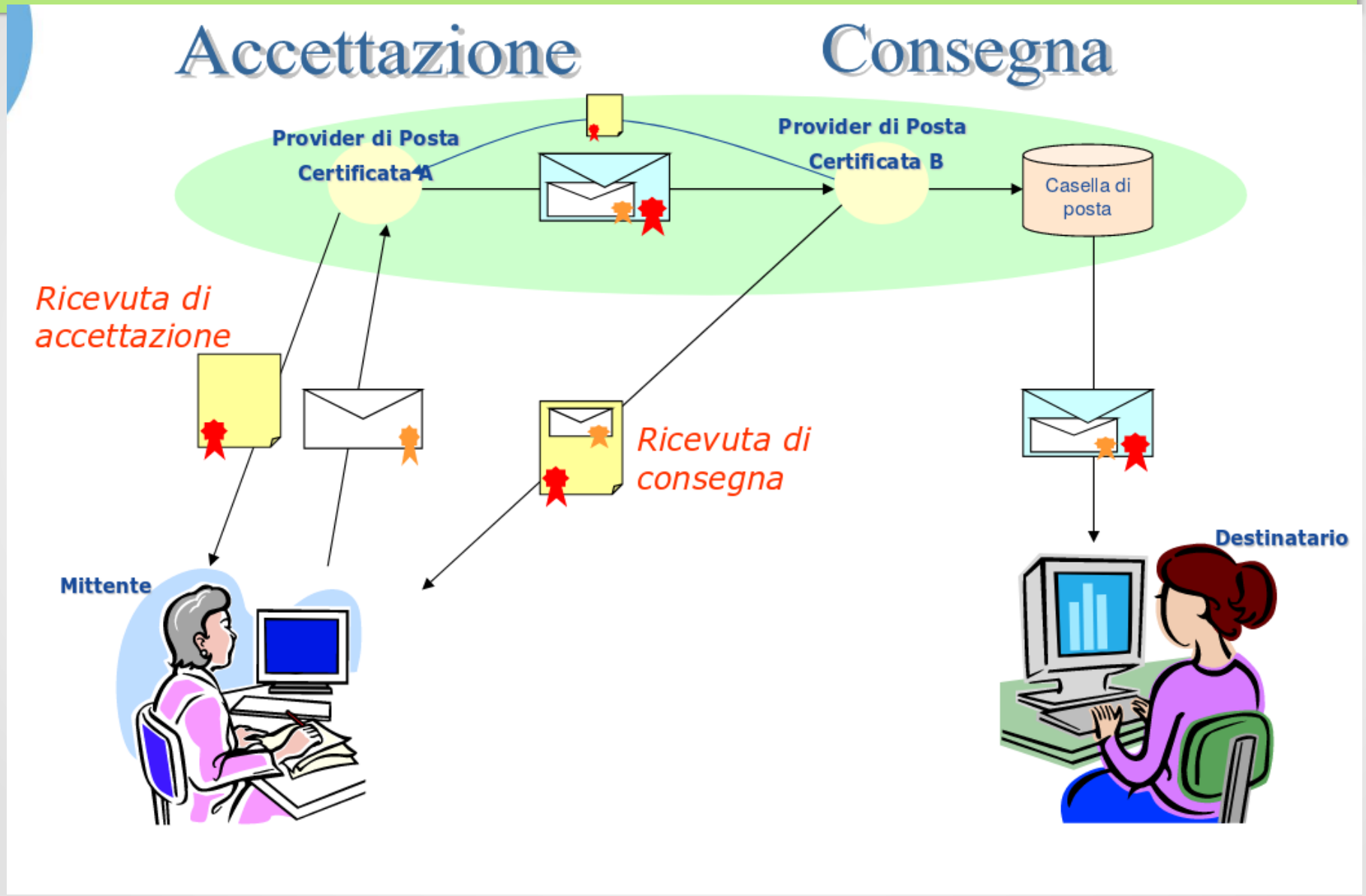
- **CA, Certification Authorithy** , garantisce l’associazione fra firma digitale ed identita’ del titolare
- Il certificato digitale viene firmato da un ente detto CA che ne attesta la bontà. L'operazione di firma avviene,da parte della CA,allegando al certificato i propri riferimenti e cifrando il tutto con la propria chiave privata
- Una data CA che firma il certifica potrebbe non essere ritenuta affidabile a quel punto ripeteremo il certificato dalla CA sospetta possiamo rivolgerci alla CA di livello superiore e cosi via fino a risalire a una CA ritenuta affidabile, e che quindi validi tutti la catena

HTTPS e POSTA CERTIFICATA

HTTPS

- Protocollo fondamentale ad esempio nell'e-commerce e home banking
- Quando mi collego ad un sito via https:
  - **Il server dichiara la sua identita' inviando il proprio certificato** di chiave pubblica **garantito da una CA**
  - **Il mio browser (client) verifica che l'URL corrisponda all'identita' contenuta nel certificato**
  - **Il client (browser) usando la chiave pubblica del server invia una chiave simmetrica temporanea per cifrare tutti il traffico dati**

Posta certificata

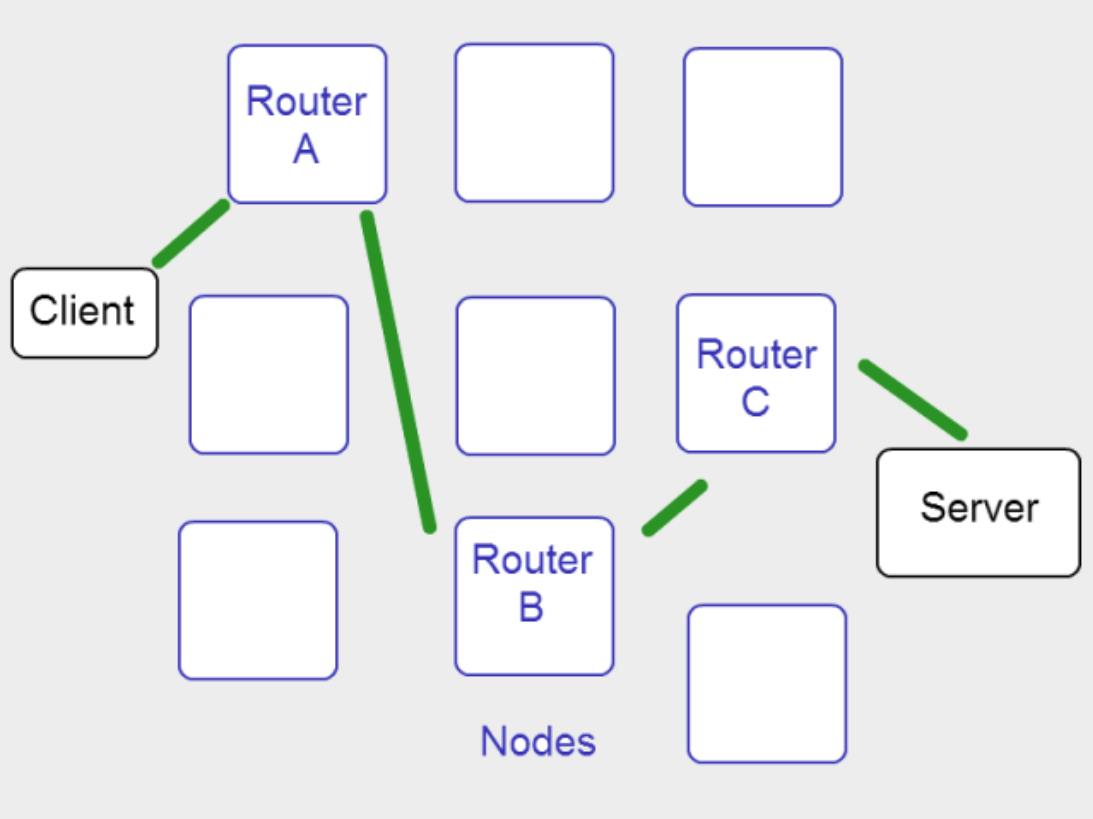


INTERNET, ALTRI ASPETTI

Deep web, Tor, bitcoin

- Onion routing un cenno. La rete tor e composta da volontari che usano i propri computer come nodi:

Tor crea un percorso random fra i vari nodi cosi da raggiungere il server partendo dal client

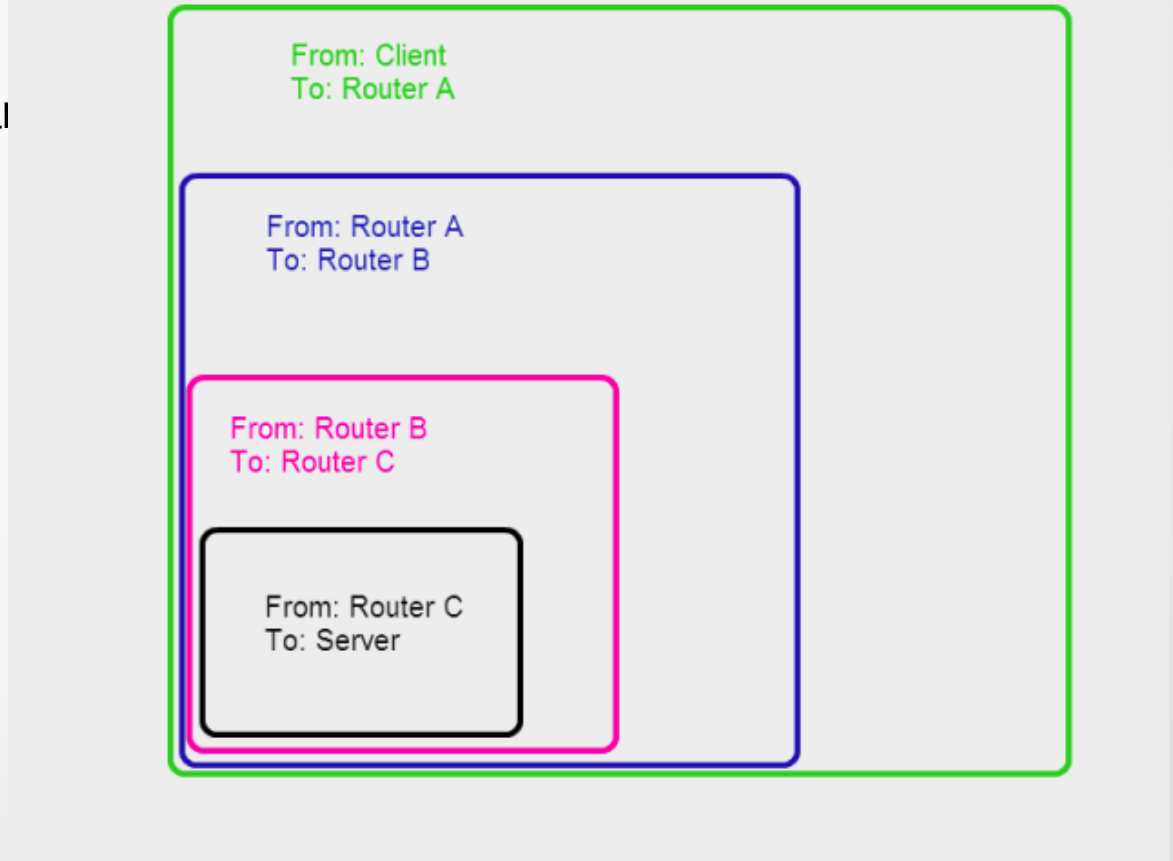


Deep web, Tor, bitcoin

- Posso raggiungere servizi “nascosti” **hidden service**, ma anche normali server via **exit nodes**

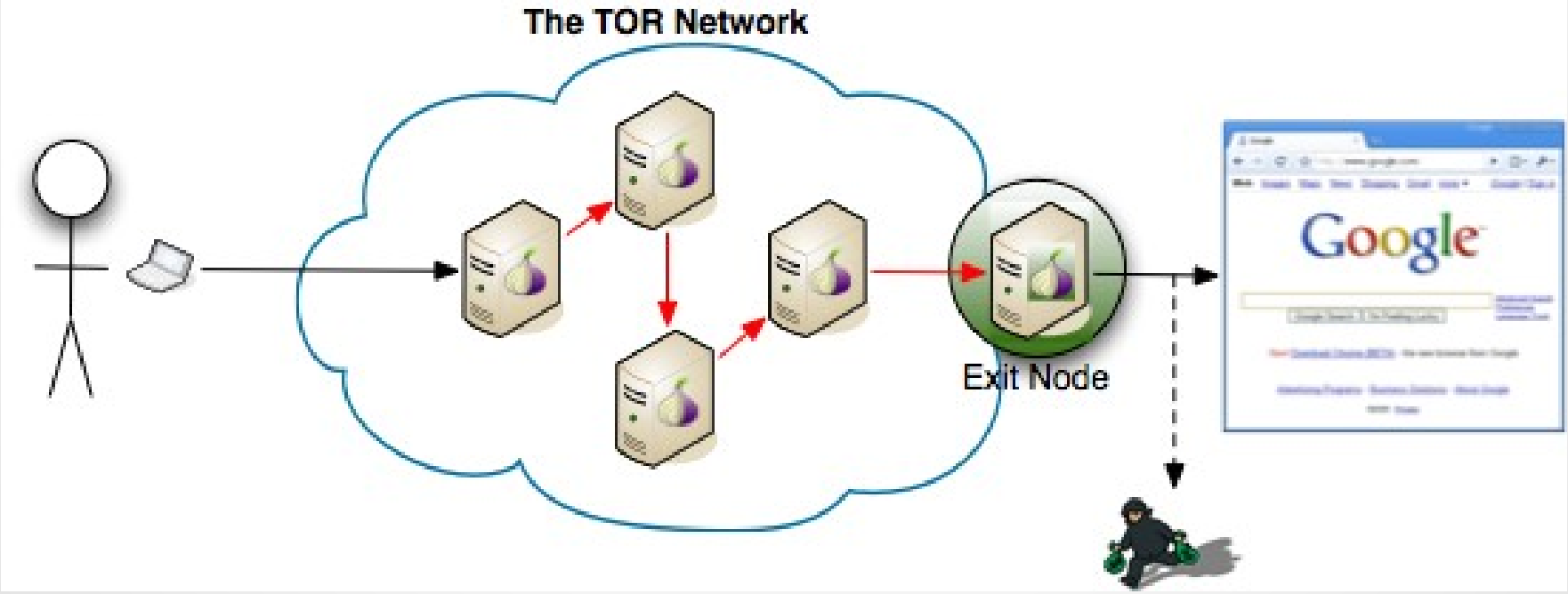
I pacchetti che transitano fra i var nodi sono incapsulati in strati successivi crittati, cosi’ come gli strati di una “cipolla”

Ad ogni passaggio il nodo conoscenze solo il punto di provenienza e quello di arrivo



Deep web, Tor, bitcoin

- Posso raggiungere servizi “nascosti” **hidden service**, ma anche normali server via **exit nodes**



Deep web, Tor, bitcoin

- **Bitcoin** (cito wikipedia) A differenza della maggior parte delle valute tradizionali, Bitcoin non fa uso di un ente centrale: esso utilizza un database distribuito tra i nodi della rete che tengono traccia delle transazioni, ma sfrutta la crittografia per gestire gli aspetti funzionali, come la generazione di nuova moneta e l'attribuzione della proprietà dei bitcoin.
- Basata su una **rete P2P (blockchain** database distribuito strutturato a blocchi**)**
- Basata su crittografia ed algoritmi di hashing (SHA-256), BitCoin utilizza l'algoritmo hash SHA-256 per generare numeri "random" verificabili in un modo tale da richiede una quantità prevedibile di tempo di calcolo. Generare un hash SHA-256 con un valore inferiore al target attuale risolve un blocco.