E-privacy 2010
Firenze, 28-29 maggio

Uso e abuso della PGP web of trust: potenzialità e rischi di un modello di fiducia semidecentralizzato

Tommaso Gagliardoni
gaglia[AT)anche(DOT]no

Licenza

Sommario

- Il problema dell'autenticazione
- Web of trust in PGP
- Web of trust nella pratica
- Problemi, attacchi, debolezze della WOT
- Possibili soluzioni, idee ed evoluzioni
- Conclusioni

Il problema dell'autenticazione

Scenario tipico

Alice  vuole spedire un messaggio segreto  a Bob 

Eve  è un nemico che vuole leggere il messaggio

Il canale a disposizione di Alice e Bob è **insicuro**

Alice propone a Bob di usare uno schema crittografico a chiave pubblica 

Bob spedisce ad Alice la sua chiave pubblica 

Alice usa la chiave pubblica di Bob per **cifrare** il messaggio

Bob riceve il messaggio e lo decifra con la sua chiave privata 

Il problema dell'autenticazione

Ecco come funziona

Il problema dell'autenticazione

PROBLEMA:

Chi assicura ad Alice che la chiave pubblica ricevuta sia davvero di Bob?

Il problema dell'autenticazione

Ecco cosa potrebbe succedere:

Il problema dell'autenticazione

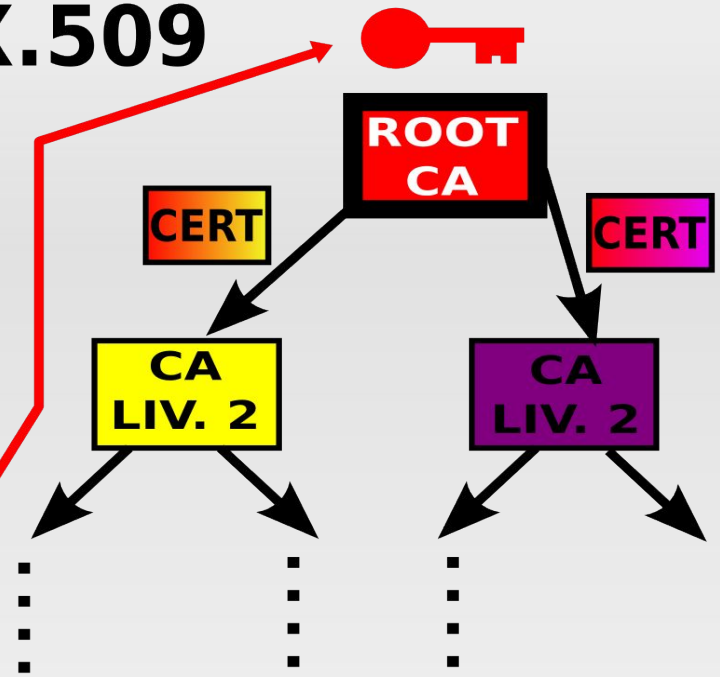
AUTHENTICATION

You're doing it wrong.

Il problema dell'autenticazione

Terze parti di fiducia: PKI X.509

1 Un'Autorità Certificante (ROOT CA) svolge il ruolo di garante principale ed emette certificati crittografici (firmati con la sua chiave privata) intestati ad altre autorità gerarchicamente inferiori (livello 2), le quali possono a loro volta emettere certificati a terze parti, a cascata.



4 Il certificato del sito web contiene tutte le informazioni per risalire al legame di fiducia con la ROOT CA, di cui l'utente possiede a priori la chiave pubblica, permettendo così di validare la sessione.

2 Il sito web che voglia fornire un'autenticazione richiede o compra un certificato da una di queste autorità

3 Il certificato viene scambiato all'inizio della sessione

Il problema dell'autenticazione

Problemi del modello centralizzato

Single point of failure: la compromissione della ROOT CA rompe *tutte* le catene di fiducia, la compromissione di una CA intermedia rompe tutto il ”sottoalbero” sotto di essa

Dipendenza: una volta acquisito un certificato da una CA si subordina la propria ”reputazione” a quella della CA (e di tutte quelle sopra di essa), la quale acquisisce ”potere di vita o di morte” sulla propria identità

Problema della **distribuzione delle chiavi** delle ROOT CA

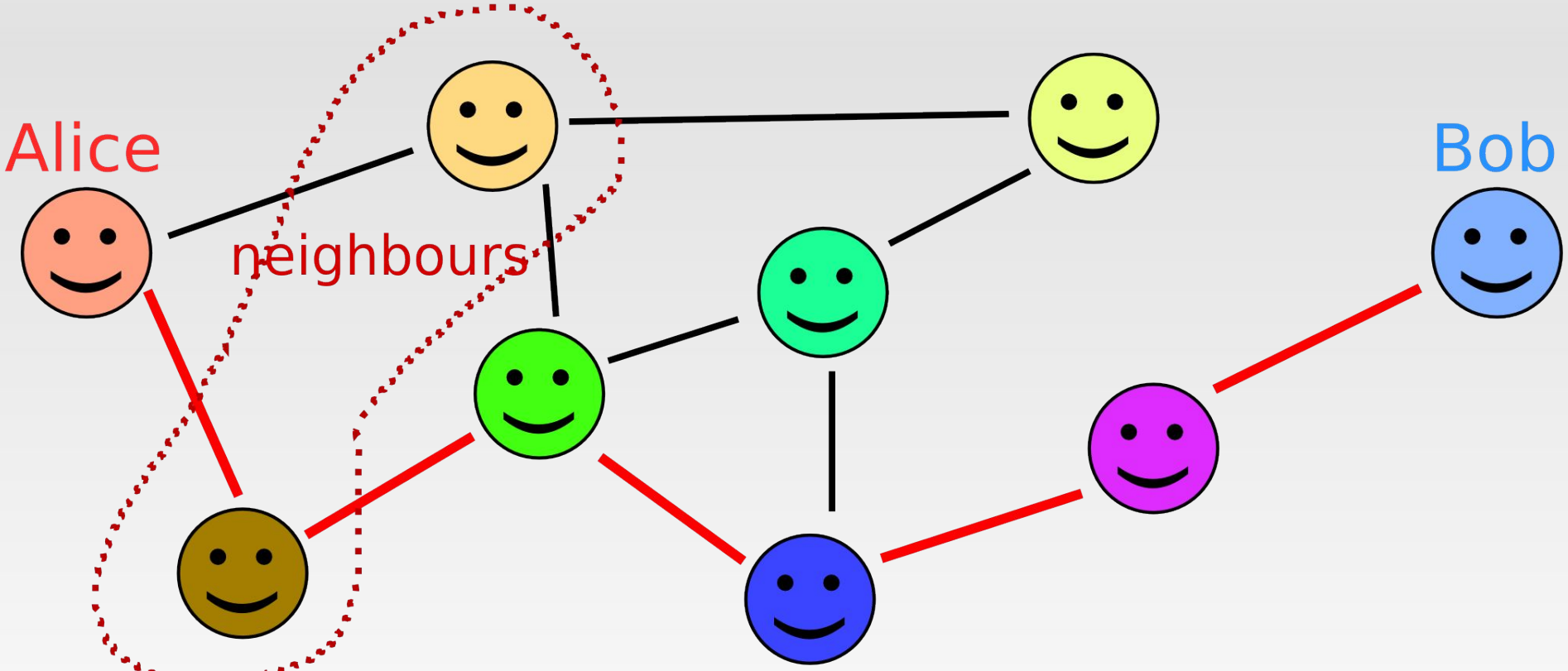
Vulnerabilità ad attacchi di tipo **Denial-Of-Service** (DOS)

Sommario

- Il problema dell'autenticazione
- Web of trust in PGP
- Web of trust nella pratica
- Problemi, attacchi, debolezze della WOT
- Possibili soluzioni, idee ed evoluzioni
- Conclusioni

Web of trust in PGP

Web Of Trust (WOT)



Ogni membro della rete ha relazioni di fiducia con un numero limitato di altri membri. È possibile stabilire una relazione di fiducia con uno sconosciuto "attraversando il grafo", ovvero fidandosi di intermediari dei quali si può stabilire la fiducia a partire da "vicini" (neighbours) fidatiti a priori.

Web of trust in PGP

Vantaggi della WOT

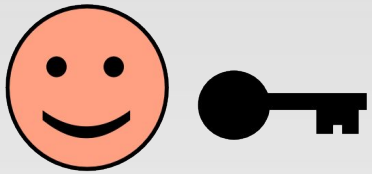
Scalabilità: si possono aggiungere o rimuovere facilmente ”nodi” o ”archi” alla rete (cioè aggiungere o rimuovere utenti e relazioni di fiducia)

Robustezza: se un nodo viene compromesso, la rete continua a funzionare quasi come se niente fosse successo

Gestione delle responsabilità decentralizzata: non c'è più un unica authority che deve accollarsi l'onere di gestire, direttamente o indirettamente, tutte le relazioni di fiducia

Web of trust in PGP

Struttura di una chiave pubblica PGP



Master key (chiave pubblica principale)

```
myuser@localhost:~$ gpg --list-sig --fingerprint Alice
pub   4096R/AABBCCDD 2010-05-26
      Key fingerprint = 1111 2222 3333 4444  5555 6666 7777 8888
uid                  Alice <alice@wonderland.org>
sig 3       AABBCCDD 2010-05-26  Alice <alice@wonderland.org>
sig         EEFFEEFF 2010-05-27  Bob <bob@telespalla.it>
uid                  Sardina <sardina@riomare.it>
sig 3       AABBCCDD 2010-05-26  Alice <alice@wonderland.org>
sig         AA11AA11 2010-05-27  Charlie <charlie@manson.com>
sig         22552255 2010-05-28  Dave <dave@matthews.band>
sub   4096g/B0DD2B35 2010-05-26
sig 3       AABBCCDD 2010-05-26  Alice <alice@wonderland.org>
sub   2048R/C551B9F2 2010-05-26
sig 3       AABBCCDD 2010-05-26  Alice <alice@wonderland.org>
```

Identità principale (Alice)

Certificati di fiducia (firme di altre chiavi) assegnate all'identità principale

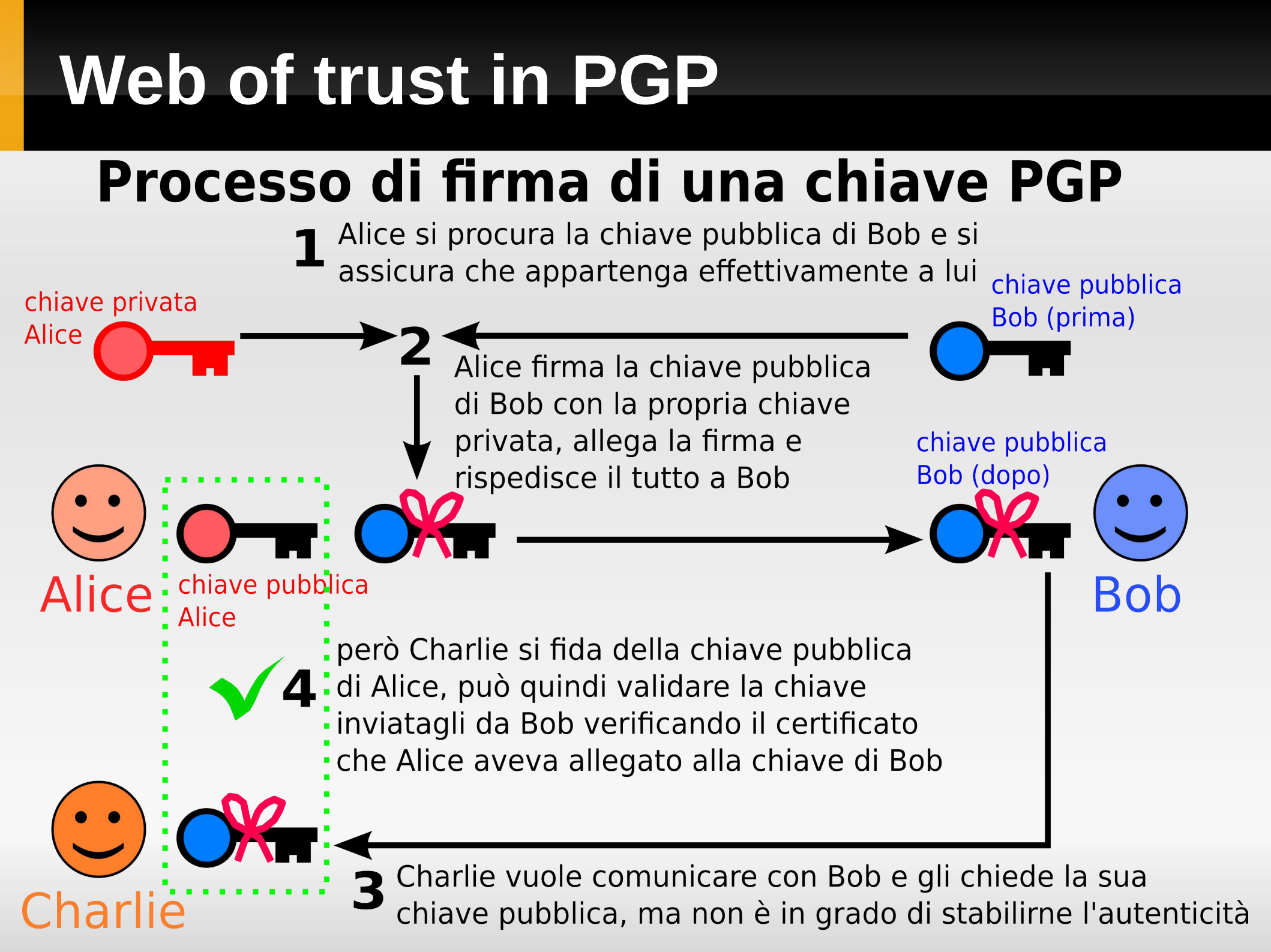
Identità addizionale

Certificati di fiducia (firme di altre chiavi) assegnate all'identità secondaria

Sottochiavi addizionali

Web of trust in PGP

Processo di firma di una chiave PGP

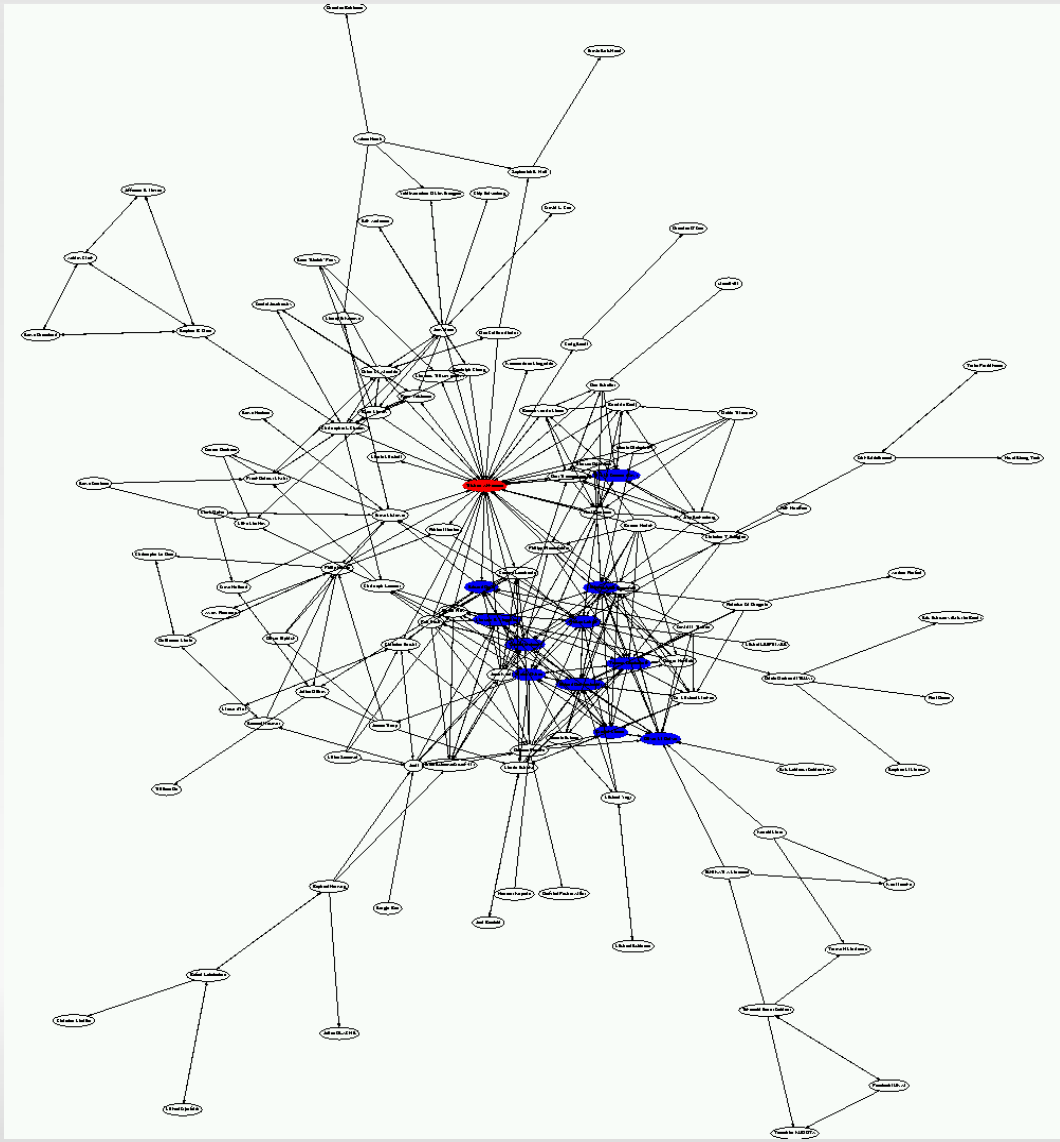


Web of trust in PGP

Grafi di fiducia PGP

Sono un modo per rappresentare la rete di relazioni di fiducia (WOT) tra gli utenti: ogni nodo rappresenta un utente (o meglio, una certa identità) mentre gli archi rappresentano le firme delle chiavi (possono essere mono o bidirezionali).

Questo a destra è il grafo di fiducia relativo al Debian Developer Team Keyring (al 30/8/2000)



Sommario

- Il problema dell'autenticazione
- Web of trust in PGP
- Web of trust nella pratica
- Problemi, attacchi, debolezze della WOT
- Possibili soluzioni, idee ed evoluzioni
- Conclusioni

Web of trust nella pratica

Keyservers

Un Keyserver (KS) è in pratica un repository (di solito pubblico) per *tutte* le chiavi PGP/GPG.

Chi vuole che la sua chiave pubblica sia liberamente raggiungibile da tutti non deve fare altro che caricarla su un qualsiasi KS (ad esempio: hkp://keys.gnupg.net) utilizzando l'apposito protocollo offerto da PGP/GPG

Per prelevare una chiave pubblica da un KS basta lanciare l'apposito comando da GPG/PGP (ad esempio: --recv-keys <ID>, ove ID può essere l'ID chiave o un nome di identità)

Tutti i KS pubblici sincronizzano automaticamente i loro database, così che la scelta di un particolare KS è indifferente

Web of trust nella pratica

Key-signing party

Resta il problema di come cominciare a stabilire relazioni di fiducia la prima volta che ci si vuole creare una WOT.

I Key-Signing Party servono a questo.

Questi party vengono di solito organizzati da gruppi di utenti per ampliare la loro WOT. Ci si reca al luogo dell'appuntamento e si scambiano con le altre persone bigliettini con nome, email e *fingerprint* della propria chiave pubblica.

Tornati a casa si prelevano da un KS le chiavi delle persone che abbiamo conosciuto, si firmano con la nostra chiave privata e si caricano di nuovo sul KS.

Per poter partecipare serve ovviamente che anche la propria chiave pubblica risieda su un KS pubblico.

Web of trust nella pratica

Keypath services

Esistono anche servizi chiamati Keypath servers (KPS) o PGP pathfinders, che di solito sono estensioni di un KS.

Questi servizi ricevono in input due chiavi pubbliche qualsiasi e cercano attraverso il grafo di fiducia del KS se esiste un percorso, ovvero una catena di relazioni di fiducia, che collega le due chiavi.

Ad esempio, se Alice riceve la chiave pubblica di Bob ma non ne può verificare l'autenticità, può interrogare un KPS ricevendo una risposta del tipo:

"Tu, Alice, conosci Charlie, il quale a sua volta conosce Dave, il quale a sua volta conosce Bob."

Spetterà poi ad Alice stabilire se questo percorso è affidabile o no (ad esempio Alice potrebbe non fidarsi di Dave)

Sommario

- Il problema dell'autenticazione
- Web of trust in PGP
- Web of trust nella pratica
- Problemi, attacchi, debolezze della WOT
- Possibili soluzioni, idee ed evoluzioni
- Conclusioni

Problemi, attacchi, debolezze della WOT

Attacchi a KS e KPS

Essendo nodi cruciali per il funzionamento pratico della WOT, KS e KPS si prestano ad essere bersagli ”privilegiati” per un nemico che volesse attaccare l'infrastruttura.

Un KS può essere manomesso, ad esempio, per sostituire una o più chiavi pubbliche con copie ”rogue” appartenenti al nemico, allo scopo di realizzare attacchi Man-In-The-Middle (MitM).

Oppure può essere ”poisonato” con numerose chiavi pubbliche del tutto simili ad una certa chiave target, rendendo molto difficile l'individuazione della chiave giusta di una certa persona.

I KPS possono essere manomessi per non mostrare alcuni percorsi di fiducia, o per mostrarne di falsi.

Infine entrambi sono suscettibili ad attacchi di tipo Denial-Of-Service (DOS).

Problemi, attacchi, debolezze della WOT

Data mining

Dal database globale delle chiavi PGP si può potenzialmente estrarre una grande mole di dati.

A cominciare dagli indirizzi email degli utenti, che sono liberamente accessibili dagli spammers.

Inoltre si possono individuare *clique*: gruppi di utenti che condividono scopi o interessi, anche se questi non rendono pubblica tale ”affiliazione”.

Infine il solo effettuare una query ad un KPS per interrogarlo sull'affidabilità di una certa chiave può far scattare molti campanelli d'allarme: qualcuno potrebbe venirvi a chiedere come mai lo volete sapere...

Problemi, attacchi, debolezze della WOT

Esempio: encrypted friend-of-a-friend scam

Supponete di ricevere una email diretta a voi, cifrata con la vostra chiave pubblica:

”Ciao, sono un amico di Alice e Bob, ho bisogno urgentissimo di aiuto per un problema che tu potresti essere in grado di aiutarmi a risolvere (mi hanno dato Alice e Bob il tuo contatto), ti prego di fare <azione pericolosa> immediatamente, è urgente!”

Alice e Bob sono vostri amici, li conoscete bene.

La mail è cifrata con la vostra chiave pubblica, quindi è palesemente diretta a voi e solo a voi!

Inoltre sembra urgente e importante, dopotutto è cifrata!

Sommario

- Il problema dell'autenticazione
- Web of trust in PGP
- Web of trust nella pratica
- Problemi, attacchi, debolezze della WOT
- Possibili soluzioni, idee ed evoluzioni
- Conclusioni

Possibili soluzioni, idee ed evoluzioni

Analisi del problema

Tutti i problemi sopracitati hanno un'unica origine: l'accentramento delle informazioni e dei servizi. La WOT decentralizza il potere di fiducia ma la sua implementazione pratica attuale non riesce a decentralizzare le informazioni nè l'autorità.

Ovvero: colpire un KS è molto più efficiente che colpire un singolo utente perché un KS gode, nella pratica, di un'autorità che il singolo utente non ha.

Analogamente il data mining funziona perché esistono delle autorità (KS, KPS) che dispongono di tutte le informazioni sulla WOT, e sono quindi punti di analisi privilegiati.

Il rimedio dunque è: decentralizzazione spietata.

Possibili soluzioni, idee ed evoluzioni

Database globale locale

Una prima idea molto naif potrebbe essere quella di far sì che ogni singolo utente disponga di un database in locale completo di *tutte* le chiavi PGP e di tutte le relazioni di fiducia.

Innanzitutto ci sarebbe il problema della sincronizzazione: far sì che tutti gli utenti dispongano in locale in ogni momento di tutte le chiavi è decisamente utopistico.

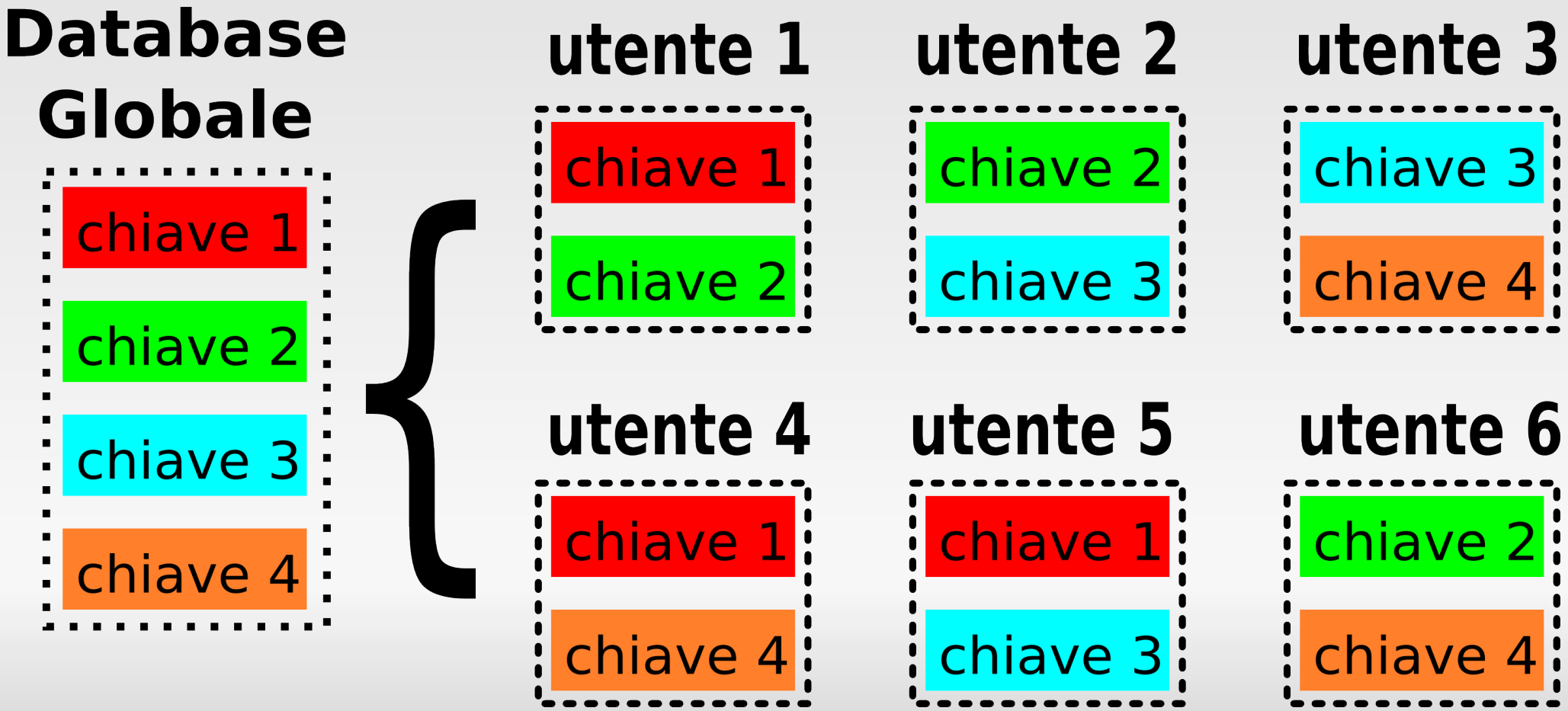
Poi ci sarebbero limiti di performances e di spazio.

Infine questo approccio elimina la centralità dei KS e KPS, ma non distribuisce le informazioni della WOT: il data mining è ancora possibile, e con esso tutti i pericoli correlati.

Possibili soluzioni, idee ed evoluzioni

Filesystem distribuito

Tutte le informazioni, invece di risiedere in un singolo database, vengono distribuite con ridondanza tra gli utenti



Possibili soluzioni, idee ed evoluzioni

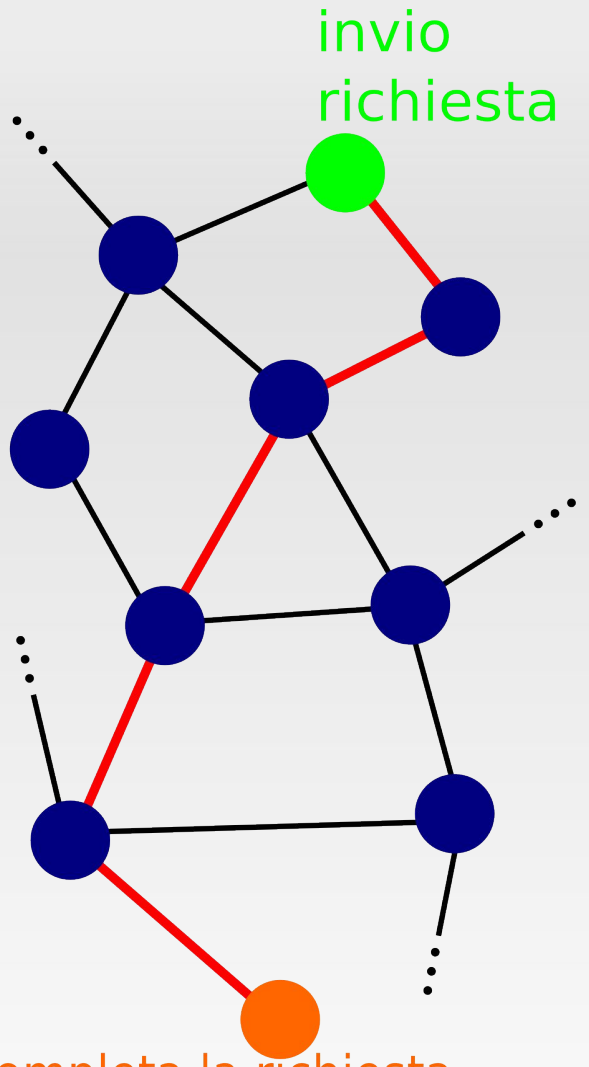
Routing

Le richieste di qualsiasi tipo di lettura e scrittura nel database vengono implementate usando tecniche di routing ”cieco” nel grafo.

Ogni nodo è connesso solo a peer fidati (modello ”darknet”) ed ogni connessione è cifrata ”a cipolla” con le chiavi di tutti i nodi che fanno parte del percorso.

Nessun nodo del percorso conosce fonte e destinazione della comunicazione, ma solo il nodo precedente e successivo.

Una richiesta qualsiasi viene ”broadcastata” verso i peer i quali a loro volta la instraderanno ai loro peer (con un certo Time-To-Live). Il nodo che riesce a soddisfare la richiesta risponde inviando un messaggio di successo a ritroso.



Possibili soluzioni, idee ed evoluzioni

Upload di una chiave

Il nodo emana un comando di ”salvataggio della chiave nel database locale” al primo dei suoi peer, con un TTL pari a ”n”.

Il primo peer salva la chiave nel proprio database locale, e ripete la stessa richiesta al primo dei suoi peer, con TTL pari a ”n-1”

Se il TTL diventa zero o se non ci sono più peer a disposizione, ogni nodo di questa sequenza restituisce un messaggio di ”termine” al nodo che aveva generato la richiesta superiore.

Il processo può essere parallelizzato per velocizzare.

La dimensione del database di ogni nodo è limitata e le chiavi meno usate vengono mano a mano cancellate.

Possibili soluzioni, idee ed evoluzioni

Fetching di una chiave

Il nodo emana un comando di ”ricerca chiave” ai suoi peer.

Se un peer trova la chiave nel suo database la restituisce, altrimenti forwarda la richiesta ai suoi peer.

Quando un nodo trova la chiave e la rispedisce a ritroso nella catena, ogni nodo della catena la salva nel suo database per uso futuro (eventualmente cancellando chiavi vecchie).

Possibili soluzioni, idee ed evoluzioni

Trust path querying

Il nodo emana una richiesta di lettura per una certa chiave (di cui vuole conoscere affidabilità e percorso di fiducia), come nel caso del fetching, a tutti i suoi peer.

Stavolta però, se un peer non trova la chiave nel suo database, prima di inoltrare la richiesta ad altri peer controlla anche se quella chiave ha firmato qualcuna delle chiavi in suo possesso. Se sì, allora restituisce quella chiave a ritroso nella catena.

Dato che la relazione di peer nella rete è *più stretta* di quella di peer fidati nella WOT infatti, ogni nodo può certificare un percorso di fiducia per tutte le chiavi nel proprio database (poiché ognuna di queste chiavi sarà stata inserita da un altro peer fidato).

Ogni peer della catena a questo punto inoltrerà a ritroso le risposte ricevute, allegando il proprio ID.

Il nodo origine riceve così infine il percorso di fiducia, ma nessuno saprà con certezza che è stato lui a effettuare la query (dopotutto potrebbe essere semplicemente un anello di una catena più lunga).

Possibili soluzioni, idee ed evoluzioni

Precauzioni per l'uso

- **Captchas:** il nodo finale di una catena, prima di restituire un certo risultato, emana un captcha a ritroso nella catena, che giungerà al nodo-origine. Questo evita che bot automatizzati facciano crawling della WOT ricostruendo il database globale.
- **Relazioni di fiducia pesate:** possono essere utili per stabilire che ”il nodo X è più fidato del nodo Y” e che ”oltre una certa lunghezza, un trust path è da considerarsi insicuro”.
- **Niente pesi negativi:** possibilità di formazione di ”mafie” (teoria dei giochi).
- **Alta latenza:** rende difficoltosa l'analisi dei tempi di risposta della rete per capire chi ha originato una certa richiesta.
- **Traffico fake:** per eludere l'analisi del traffico, i nodi dovrebbero generare costantemente rumore di fondo.
- **TTL randomizzati:** evitano di trovare banalmente l'origine di una query.

Possibili soluzioni, idee ed evoluzioni

Vantaggi e svantaggi

Il vantaggio principale è che nessuna entità dispone del database globale della WOT. Quindi niente data mining selvaggio.

Ricostruirlo interrogando molte volte i nodi è comunque difficile a causa dei captchas (e semplici meccanismi di protezione anti-flood).

Anonimato delle query e robustezza della rete da DOS, manomissione o blackout sono anch'essi garantiti.

Lo svantaggio è che l'implmentazione di uno strumento del genere richiede che i nodi stiano connessi in rete per la maggior parte del tempo.

Possibili soluzioni, idee ed evoluzioni

Follia? Delirio? Fuffa?

Esiste già uno strumento rodato e scalabile che offre tutte le suddette proprietà di distribuzione e anonimato:

http://freenetproject.org/

La struttura modulare di Freenet permetterebbe probabilmente di realizzare un plugin apposito che si occupi della WOT nel modo sopra descritto, senza dover implementare tutto da zero.

Sommario

- Il problema dell'autenticazione
- Web of trust in PGP
- Web of trust nella pratica
- Problemi, attacchi, debolezze della WOT
- Possibili soluzioni, idee ed evoluzioni
- Conclusioni

Conclusioni

Il problema dell'autenticazione è cruciale, non solo per la cifratura delle e-mail, ma in una miriade di altri ambiti (sociologia, economia, semantic web, etc).

Il modello ”classico” di terza parte fidata che garantisce l'autenticità è facile da implementare ma fragile e obsoleto.

Il metodo WOT (qui analizzato nel caso di PGP) è più robusto sotto molti aspetti: la parola chiave è ”decentralizzazione”.

Ciononostante, nell'implementazione pratica la WOT di PGP ha bisogno ancora di affidarsi ad autorità terze, e questo introduce dei punti deboli nel sistema che si prestano ad ipotetici attacchi.

Passare ad un sistema totalmente distribuito significa adottare un metodo peer-to-peer per lo storage delle chiavi e per il routing delle richieste nella rete WOT.

Tra i vantaggi di questo metodo c'è quello di poter interrogare il sistema sulla fiducia di una particolare entità senza che nessuno possa nemmeno sapere chi ha fatto una tale interrogazione.

Un metodo plausibile per implementare questo schema potrebbe essere quello di appoggiarsi a Freenet, che presenta già nativamente molte delle suddette caratteristiche.

Fine

Domande?