Metro Olografix Crypto Meeting 2006

Pescara, 25 marzo 2006

Condivisione anonima di informazioni:

remailer anonimi, server di pseudonimi, darknet

Marco A. Calamari - `marcoc@winstonsmith.info`

Progetto Winston Smith

Saluto

Buongiorno,

In questo intervento cercheremo di offrire un panorama completo dei mezzi disponibili in Rete per la comunicazione riservata.

Limiteremo il discorso alla trattazione dei sistemi ad alta latenza, quelli cioe' con tempi di propagazione non real time (come ad esempio Tor), ma piuttosto simili a quelli della posta elettronica.

Pur non avendo prerequisiti, una conoscenza concettuale dell'utilizzo della posta elettronica, in particolare crittografata, permettera' di trarre il massimo profitto da questo intervento.

Consigliamo anche una consultazione della bibliografia fornita.

Anonimato – chi ne ha bisogno ?

L'anonimato viene spesso visto come una necessita' di settori limitati della societa', spesso identificati come marginali se non addirittura criminali. In realta' molti altri settori della societa' possono trarre vantaggio dal comunicare in maniera anonima

Privati cittadini, che possono usare l'anonimato in casi particolari per tutelare la propria legittima privacy **(anonimato = privacy)**

- evitare di essere identificati dai corrispondenti
- evitare di essere profilati da ISP, pubblicitari e fornitori di servizi
- evitare conseguenze sociali negative di opinioni e comportamenti

Aziende ed enti governativi di vario tipo **(anonimato = sicurezza)**

- Analisi della concorrenza
- Tutela di relazioni strategiche riservate
- Riservatezza della locazione fisica di impiegati e risorse
- Resistenza all'analisi del traffico delle comunicazioni
- Indagini contro attivita' criminali sofisticate

Di cosa parleremo ?

- I sistemi anonimi ad alta latenza
- Una categorizzazione della problematica
- Remailer pseudoanonimi ed anonimi
- Server di pseudonimi
- Pubblicazione anonima di informazioni
- Conclusioni

I sistemi anonimi ad alta latenza

I sistemi ad alta latenza

Il problema di mantenere la riservatezza delle comunicazioni nei sistemi di comunicazione ad alta latenza richiede innanzitutto una definizione precisa di cio' che si vuole mantenere riservato.

Senza perdita di generalita', utilizzeremo la posta elettronica internet, ed i relativi RFC, come oggetto della nostra trattazione. Infatti, anche se la posta elettronica non e' l'unico strumento di comunicazione ad alta latenza, e' certamente di gran lunga il piu' diffuso.

Categorizziamo il problema

La comunicazione via posta internet, come la controparte cartacea, e' caratterizzata da quattro tipi di informazioni

- **Il contenuto del messaggio**
- **Il mittente del messaggio**
- **Il destinatario del messaggio**
- **L'esistenza di una comunicazione**

Diamo per acquisito che mantenere la riservatezza del contenuto sia un problema risolto e noto. Esistono infatti programmi di crittografia per tutti i sistemi operativi e per tutti i principali programmi di posta che, utilizzando metodi sicuri e verificati, permettono di rendere assolutamente riservato il contenuto di qualunque messaggio di posta elettronica. I capostipiti sono **Pgp** e **GnuPG**, ma ormai esistono molti altri programmi, sia proprietari che liberi

Remailer pseudoanonimi ed anonimi

Remailer anonimi

Il secondo problema e' quello di rendere non rintracciabile il mittente di un messaggio di posta elettronica.
Per ottenere questo si utilizzano appositi server di rete che sono chiamati **anonymous remailer** o **remailer anonimi**.
Nel loro sviluppo, il principio di funzionamento si e' evoluto, partendo da applicazioni elementari e non molto sicure fino a raggiungere l'attuale situazioni di applicazioni molto sicure e difficilmente attaccabili.

Il tipi di remailer di cui parleremo sono:

- **Remailer di Tipo 0 (Penet) detti pseudoanonimi**
- **Remailer di Tipo I (cypherpunk)**
- **Remailer di Tipo II (Mixmaster)**
- **Remailer di Tipo III (Mixminion)**

Remailer pseudoanonimi

Il piu' semplice tipo di remailer, ormai abbandonato, e' quello di **Tipo 0** o **pseudoanonimo**. Si tratta di server di posta che, ricevendo posta opportunamente formattata, estraggono dal messaggio l'indirizzo del destinatario e glielo inviano, sostituendo il proprio indirizzo nel campo del mittente.

E' chiaro che questa forma di anonimizzazione e' estremamente debole; e' infatti sufficiente che il server che si utilizza venga compromesso per rendere osservabili tutti i mittenti dei messaggi.

Inoltre, anche senza compromettere il server, una osservazione dei messaggi entranti ed uscenti dal remailer rende facile dedurre il mittente reale di ogni messaggio che passa attraverso di esso.

Remailer pseudoanonimi

Il piu' famoso remailer di questo tipo, **anon.penet.fi**, fu appunto violato costringendo per vie legali il suo operatore (gli operatori dei remailer vengono chiamati tradizionalmente “**remop**”) a fornire l'indirizzo reale del mittente.

Il famoso caso legale, intentato dalla **Chiesa di Scientology**, da allora **nemica “storica” dei remailer anonimi**, vide soccombere **Julf Helsingius, il remop di anon.penet.fi**, che chiuse poi volontariamente il remailer.

Questo evento, apparentemente negativo, segno' invece un autentico boom nella ricerca di base sui remailer

Remailer di Tipo I

E' evidente che il principio di funzionamento dei remailer di Tipo 0 era viziato alla base; non deve infatti essere possibile violare un remailer semplicemente forzando l'operatore a “collaborare”, od osservando il traffico in entrata ed in uscita da esso, pena la sostaziiale inefficacia del remailer stesso.

Con questi principi in mente furono creati i remailer di **Tipo I** o **Cypherpunk**, dal nome della mail list in cui si svolse l'attivita' collaborativa di progettazione e programmazione.

I remailer Cypherpunk, che vengono usati “**in catena**” facendo attraversare almeno 3 remailer ad ogni singolo messaggio, superano il problema della forzatura dell'operatore impedendo anche a lui di conoscere l'indirizzo del mittente e/o del destinatario di un messaggio. In questo modo non sono possibili attacchi legali come quello di cui fu oggetto anon.penet.fi

Remailer di Tipo I

I remailer cypherpunk utilizzano massicciamente Pgp ed i metodi crittografici a chiave pubblica. Ogni remailer possiede una coppia di chiavi pgp.

Il **mittente** del messaggio sceglie una **catena di almeno tre remailer**, e crittografa in successione il messaggio per ciascuno dei tre remailer scelti, aggiungendo ad ogni passo l'indirizzo del successivo remailer a cui inoltrarlo.

Se paragoniamo il processo crittografico alla chiusura di un messaggio cartaceo in una busta, questo equivale ad inserire la prima busta che contiene il messaggio indirizzato al destinatario finale in una seconda, indirizzata all'ultimo remailer della catena, e questa in una terza, indirizzata al remailer intermedio, e questa in una quarta, indirizzata al primo remailer.

Remailer di Tipo I

Complicato ? Solo in apparenza. La “busta di buste” funziona esattamente come l'analogo cartaceo, con l'aggiunta del fatto che, per le proprieta' degli algoritmi crittografici, nessun remailer puo' “aprire” le buste non indirizzate a lui.

Quindi il primo remailer conosce l'indirizzo del mittente, e l'ultimo quello del destinatario, e **la compromissione di uno od anche due dei tre remailer non e' sufficiente per violare l'identita' del mittente** di un determinato messaggio ricevuto da un certo destinatario.

L'utilizzo dei remailer Cypherpunk e' teoricamente possibile usando direttamente Pgp ed un normale programma di posta; in realta' e' praticamente indispensabile un programma dedicato, come ad esempio **Jack B. Nymble**

Remailer di Tipo II

Malgrado l'apparente inattaccabilita', i remailer Cyperpunks sono attaccabili con metodi piu' sofisticati basati sull'**analisi del traffico in entrata ed in uscita** da ciascun remailer.

Anche se i messaggi sono illeggibili, la loro dimensione ed la loro sequenza temporale permettono ad un attaccante dotato di sufficienti risorse di violare le comunicazioni, in particolare in quei casi in cui vi siano invii regolari di messaggi tra due corrispondenti.
Per neutralizzare questi tipi di attacco furono realizzati i remailer di **Tipo II** o **Mixmaster**.

I remailer Mixmaster aggiungono tre importanti meccanismi a quelli presenti nei remailer cypherpunk, che riassumiamo brevemente.

Remailer di Tipo II

Rendere uniformi le dimensioni dei messaggi: i messaggi vengono suddivisi in pezzi di dimensione uguale, e l'ultimo pezzo viene reso della lunghezza degli altri aggiungendovi informazioni casuali. Il primo remailer della catena spezzetta i messaggi e l'ultimo li riassembla, spedendo poi il risultato al destinatario.

Spedire i messaggi in maniera casuale: i messaggi in uscita da un remailer non vengono spediti appena pronti, ma posti in un **pool**, una coda, insieme ad altri. Ad intervalli prestabiliti il remailer spedisce una certa percentuale dei messaggi in coda presi a caso, impedendo quindi di correlare i messaggi in arrivo con quelli in uscita.

Generare traffico fittizio: per offuscare maggiormente il traffico vengono generati falsi messaggi, messi in coda con quelli veri; l'ultimo remailer riconosce i messaggi fittizi e li scarta.

Remailer di Tipo II

L'utilizzo dei remailer Mixmaster richiede l'**impiego di un client,** che e' fornito anche con il software del server. L'attuale implementazione di riferimento Mixmaster gestisce anche i messaggi di Tipo I.

Per questo motivo non esistono ormai piu' software di remailing di Tipo I, e la rete dei remailer esistenti e' formata da server Mixmaster che smistano ambedue i tipi di messaggi.

Il gia' citato client per windows **Jack B. Nymble** permette di inviare anche messaggi di tipo Mixmaster.

Remailer di Tipo III

Anche i remailer Mixmaster non sono esenti da problemi; sono infatti di gestione complessa e suscettibili di vari tipi di attacco, principalmente di **negazione del servizio**, visto che si poggiano sulla posta normale come veicolo di trasporto e che richiedono una infrastruttura separata per la **distribuzione delle chiavi crittografiche**.

Per questi motivi e' in fase di realizzazione la nuova generazione di remailer anonimi, detta di **Tipo III o Mixminion**.

I remailer Mixminion utilizzano principalmente un proprio **protocollo client/server** per spedire messaggi. Questo velocizza molto la consegna dei messaggi stessi e rende i remailer indipendenti da una preesistente (e ben monitorata!) infrastruttura di posta elettronica.

Remailer di Tipo III

I remailer Mixminion possono comunque **interfacciarsi anche con le reti Mixmaster e con la normale posta elettronica,** permettendo la massima flessibilita' di utilizzo.

Ma la piu' importante caratteristica dei remailer Mixminion e' che tutta l'infrastruttura di gestione e validazione delle chiavi crittografiche e' gestita automaticamente dai remailer stessi, tramite una funzionalita' di directory distribuita che rende molto difficile un attacco.

Il client (a linea comandi) fa parte del software del remailer; e' in fase di sviluppo **MMS**, una interfaccia grafica per windows equivalente a Jack B.Nymble, che ne facilita molto l'uso.

Benche' il protocollo Mixminion sia sperimentale, si trova in uno stadio avanzato di maturita' che lo rende gia' utilizzabile.

Remailer – Stato dell'arte e prospettive

Attualmente la rete Mixmaster dei remailer di Tipo I e II si trova in uno **stadio di maturita'**. Costituita da circa **35 remailer**, utilizza due implementazioni del protocollo; **Mixmaster** per *nix realizzato in linguaggio C, e **Reliable**, un software per windows realizzato in VB. Pur con una certa macchinosita' di manutenzione, la rete Mixmaster e' operativa ed utilizzabile senza problemi.

La rete dei remailer **Mixminion** di Tipo III (**40 remailer**) si trova ancora in una fase di sviluppo; mentre i protocolli di routing e crittografici sono ormai stabili, il servizio di directory e' in via di evoluzione. Mixminion e' quindi dichiarato ancora "non affidabile".
E' invece opinione comune che Mixminion, come robustezza dei protocolli, sia ormai pari o superiore a Mixmaster.
Per quanto invece attiene alle possibilita' di sovversione o DoS della rete, la situazione non e' ancora soddisfacente.

Server di Pseudonimi

Server di Pseudonimi

Come abbiamo visto all'inizio, il problema di rendere privati mittente e destinatario di messaggi di posta e' **praticamente risolto** dalle tecnologie dei remailer sviluppati ed in sviluppo. Infatti un uso esteso di questi programmi (per ora di la' da venire) permette di non rivelare le identita' del mittente di un messaggio, ed in parte anche quella del destinatario e l'esistenza stessa di un processo di comunicazione, grazie al traffico criptato ed a quello fittizio generabile dai remailer di tipo Mixmaster e Mixminion.

Esiste pero' il **problema della risposta**; chi riceve un messaggio attraverso un remailer non puo' rispondere al mittente se il mittente stesso non svela la sua identita' nel messaggio.
Non e' quindi possibile, utilizzando un remailer, rispondere ad un messaggio giunto tramite remailer oppure ricevere messaggi ad un indirizzo anonimo tramite la rete dei remailer.

Server di Pseudonimi

Esiste quindi la necessita' di creare degli **pseudonimi**, cioe' indirizzi di posta fittizi, **non riconducibili ad un indirizzo reale** ma tramite cui e' possibile far giungere messaggi ad un destinatario senza conoscerne l'identita'.

In effetti gia' i remailer di Tipo 0 permettevano di rispondere ad un messaggio giunto da essi; avendo infatti essi **una conoscenza completa della corrispondenza tra indirizzo fittizio ed indirizzo reale del mittente**, permettevano di rispondere con la massima semplicita' ad un messaggio giunto tramite essi.
Infatti un messaggio proveniente da tizio@provider.com veniva inoltrato come tizio@anon.penet.fi; bastava rispondere normalmente ed il messaggio giungeva al remailer che lo ritornava all'indirizzo originale. L'aggiunta delle caratteristiche necessarie nei remailer di tipo I e II impedisce questo processo.

Server di Pseudonimi

Sono stati quindi creati i server di tipo **Newnym**. Un server di questo tipo, usando gli stessi metodi di crittografia ampiamente usati nei remailer, permette ad un utente di creare un indirizzo di posta fittizio, uno **pseudonimo**, sul server stesso.

Tramite opportuni comandi un utente puo' crearsi uno pseudonimo sul server, e puo' trasmettere al server un messaggio che uscira' da esso con il suo pseudonimo come mittente. Per far cio' utilizzera' ovviamente una catena di remailer, per salvaguardare fin dall'inizio la sua identita'.

Puo' inoltre caricare sul server di pseudonimi uno o piu' **reply block**, che sono sequenze crittografate di istruzioni su come far pervenire un messaggio, arrivato allo pseudonimo, all'indirizzo reale del mittente tramite una serie di remailer anonimi che permettono di **conservare l'anonimato della risposta**.

Server di Pseudonimi

L'utilizzo dei server Newnym e' praticamente possibile solo tramite client appositi, come il piu' volte citato Jack B. Nymble.

Una buona notizia e' che lo sviluppo del protocollo Mixminion tende nuovamente a **riunificare remailer e pseudonym server** nello stesso software, come ai tempi dei remailer Tipo 0.

I remailer Mixminion, che ricordiamo implementano sia il server che un client testuale, permettono di creare ed utilizzare **SURB (Single Use Reply Block)** che consentono gia' oggi di realizzare una parte delle funzionalita' di uno pseudonimo direttamente tramite il protocollo Mixminion. Un utente puo' infatti generare uno o piu' reply block, simili a quelli dei server Newnym ma validi per un solo uso e con durata limitata nel tempo, che permettono di spedirgli un messaggio tramite la rete Mixminion non dovendo rivelare la propria identita'

Server di Pseudonimi

Un utente Mixminion puo' quindi scrivere ad un destinatario in forma anonima, ed **allegare al messaggio un SURB** che permette di rispondergli senza che sia costretto a rivelare la propria identita'.

Puo' anche **pubblicare**, ad esempio su una pagina web, una serie di SURB che consentano a **chiunque** di inviargli un messaggio.

Il metodo, come si vede e' macchinoso e certamente non user friendly, ma le funzionalita' di base ci sono tutte e sicuramente verranno presto realizzati frontend come MMS, in grado di automatizzare anche questo processo.

Nym – Stato dell'arte e prospettive

Parallelamente allo sviluppo di **Mixminion** e' iniziato quello di **Nymbaron**, un server di pseudonimi separato da Mixminion ma che ne usa la rete come trasporto ed i SURBS come componenti.

Nymbaron, come Mixminion, e' sviluppato in linguaggio Python, e si trova per ora in una release alpha in cui solo una parte delle funzionalita' sono implementate.

Come Mixminion e' una applicazione a linea comandi, e necessitera' di una interfaccia grafica per essere utilizzabile da non “addetti ai lavori”.

I sistemi per la pubblicazione anonima di informazioni

Pubblicazione anonima

Un sistema di pubblicazione anonima di informazioni, analogamente ai remailer anonimi, deve possedere alcune funzionalita' di base.

- **Privacy per chi pubblica informazioni**
- **Privacy per chi le legge**
- **Protezione per chi le conserva**
- **Resistenza alla soppressione di informazioni (censura)**

Queste funzionalita' non sono possedute dai piu' comuni e ben noti sistemi peer-to-peer.

Infatti quelli piu' utiizzati (Napster, WinMX, Gnutella, eMule) non possiedono i requisiti di privacy e protezione, e solo molto limitatamente quelle anticensura.

Pubblicazione anonima

Infatti rendono rintracciabile con mezzi semplici sia il detentore delle informazioni che il richiedente. Inoltre possedendo nodi master, o cumunque privilegiati, sono suscettibili ad azioni di diniego del servizio e/o di censura.

Infine non posseggono, di per se', nessuno storage permanente, essendo questo fornito ad hoc dai client che si collegano.

Il prototipo dei sistemi di pubblicazione resistenti alla censura e' stato l'ormai dismesso Publius.

Publius

Publius offriva la possibilita' di pubblicare contenuti statici, come una pagina html, su **una rete di server specializzati**.

Il contenuto veniva suddiviso in un numero **n** di pacchetti cifrati e ridondanti, in modo tale che, recuperando solo una parte di essi (tipicamente 10 su 15) fosse possibile ricostruire l'intero documento. Distribuendo poi i server su scala globale, rendeva molto difficile una azione censoria, che avrebbe dovuto riguardare molti server posti in molte giurisdizioni per rendere indisponibile un documento.

Solo **l'autore poteva cancellare un documento**, e quindi rappresentare il **single point of failure del sistema**, ove ad esempio fosse stato vittima di un attacco di tipo legale che lo costringesse a rimuovere il suo documento

Eternity

Eternity e' stato un altro interessante tentativo di realizzare un vero sistema di pubblicazione anonima resistente alla censura.
Si basava su di un meccanismo crittografico e di split concettualmente simile a Publius, ed **utilizzava i newsgroup USENET come datastore** per immagazzinare i documenti, sfruttando cosi' la enorme quantita' di ben dimensionati server di news esistenti.

Pur essendo resistentissimo ad attacchi di tipo censorio, non offriva metodi sicuri per pubblicare e recuperare informazioni, dovendosi poggiare su quelli tipici delle news

Freenet

Freenet e' stato, ed e' tuttora, **l'unico sistema di pubblicazione anonima reso disponibile alla generalita' del pubblico**. E' stato progettato fin dall'inizio per garantire tutte le funzionalita' proprie di un sistema di pubblicazione anonima, ed e' implementato come protocollo di comunicazione.

I nodi Freenet incorporano tutti un **datastore** per la memorizzazione dei documenti ed un **proxy locale che permette di navigare in Freenet con un normale browser**.

La pubblicazione ed il recupero delle informazioni avvengono da un normale nodo, ed il protocollo, **parzialmente stocastico**, che effettua queste operazioni garantisce la non rintracciabilita' sia di chi pubblica che di chi recupera.

Freenet

I contenuti sono suddivisi in atomi detti “chiavi” che sono crittografati e distribuiti casualmente tra molti server; essi poi passano dinamicamente da uno all'altro quando vengono richiesti.
Essendo le chiavi crittografate, **il gestore di un nodo che possiede una chiave non puo' conoscerne il contenuto**; questo assicura, nella maggior parte dei sistemi giuridici, una buona protezione di tipo legale.
Per la protezione totale contro la censura, nessuna informazione puo' essere cancellata da Freenet, nemmeno da chi l'ha pubblicata.
Dovendo ovviamente evitare la saturazione immediata di tutti i datastore, l'informazione, se necessario per fare posto nel datastore, **viene cancellata in automatico quando non viene richiesta**. Con questo metodo quindi le informazioni popolari sopravvivono a lungo, mentre quelle non consultate spariscono.

Freenet

Gli svantaggi dell'attuale versione di Freenet (0.5), che comunque possiede tutte le funzionalita' predette, sono:

- essere lenta e pesante (il server e' realizzato in Java ed e' probabilmente l'applicativo java piu' grande al mondo).

- essere un sistema in **evoluzione caotica**, visto che i programmatori testano le nuove versioni sulla Freenet reale, e documentano poco e male (d'altra parte con le poche risorse a disposizione sarebbe difficile fare altrimenti)

- essere suscettibile ad attacchi DoS diretti contro i nodi, i quali sono facilmente identificabili **ricavando le informazioni da Freenet stessa**. Proprio per risolvere questa classe di problemi e' intenzione del gruppo di sviluppo trasformare Freenet in una rete **non aperta ma "ad invito"** che verra' chiamata' **Dark**

Freenet – Stato dell'arte e prospettive

L'attuale rete Freenet 0.5 si trova ormai in una fase di stabilita' e maturita' che non aveva mai conosciuto nella sua, abbastanza travagliata, storia.
Stabilita' dei contenuti, dimensione dello storage e tempi di latenza sono senz'altro soddisfacenti. Non vi sono attivita' di sviluppo se non il fixing di bug di sicurezza, peraltro rari nella versione corrente.

Freenet 0.7 e' ad oggi limitata ad un client a linea comandi che implementa solo il protocollo FCP di base; il meccanismo Darknet di invito e' completamente manuale, realizzato via IRC.
E' prevedibile che, in tempi dell'ordine di un anno, il deployement di una rete di test 0.7 sottrarra' risorse alla rete 0.5, che prevedibilmente restera' stabile e popolata fino ad allora, analogamente a quanto avvenuto anni fa durante la transizione da Freenet 0.3 alla 0.5.

Conclusioni

Conclusioni

Possiamo concludere che **sistemi anonimi di comunicazione e pubblicazione di informazioni esistono e sono pienamente utilizzabili**, anche se spesso richiedono competenza tecnica e pazienza non alla portata della generalita' del pubblico.

Quello che manca e' **una diffusione della loro conoscenza e del loro utilizzo come strumento di comunicazione in Rete**.

E' opinione largamente condivisa dagli “operatori” di questo settore di nicchia che questo, e non problemi tecnologici, sia il vero ostacolo da superare per realizzare una vita in Rete realimente sicura e privata.

Nella bibliografia, oltre ai riferimenti sui temi trattati, sono elencati alcuni gruppi di discussione a cui e' possibile partecipare.

Grazie a tutti per l'attenzione

ci sono domande ?

Contattatemi pure all'indirizzo **marcoc@winstonsmith.info**

Il Progetto Winston Smith

http://www.winstonsmith.info/pws

freenet:SSK@Dgg5lJQu-WO905TrlZ0LjQHXDdlPAgM/pws/15//

Bibliografia

Bibliografia

- **CoS vs. Julf Helingius**
  http://www.xs4all.nl/~kspaink/cos/rnewman/anon/penet.html
- **Jack B. Nymble**
  http://e-privacy.firenze.linux.it/pws/software.html
- **Il remailer Mixmaster** http://mixmaster.sourceforge.net/
- **Il remailer Mixminion** http://www.mixminion.net/
  http://www.mixminion.it/
- **MMS** http://peculiar.wcw.net/mixminion-message-sender/
- **Newnym pseudonym server** http://sourceforge.net/projects/nymserv

Bibliografia

- **Publius** http://cs1.cs.nyu.edu/~waldman/publius/
- **Eternity service** http://www.cypherspace.org/adam/eternity/
- **Freenet** http://www.freenetproject.org/
  https://lists.firenze.linux.it/mailman/listinfo/freenet-list
- **Il Progetto Winston Smith** http://www.winstonsmith.info/pws/index.html
  https://lists.firenze.linux.it/mailman/listinfo/e-privacy
- **il Convegno e-privacy** http://e-privacy.firenze.linux.it/