e-privacy 2007 - Firenze

Anonymous Credentials Systems

Gianfranco Ciotti
g.ciotti@winstonsmith.info

Anonymous Credentials Systems: ossia come rendere anonimi i nostri dati personali durante transazioni telematiche e assicurarci in questo modo la protezione da furti di identità o altri tipi di abusi e violazione della nostra privacy.

Si è parlato in questi giorni di alcuni problemi legati alla profilazione e al tracciamento degli utenti e più volte è venuta fuori la definizione di “Persona Elettronica” o “**Identità Elettronica**” attraverso la possibilità di ricostruire l'identità di un individuo seguendo la traccia da esso lasciata durante vari transazioni sul web.

Quando inviamo i nostri dati personali per abbonarci ad una rivista o prenotare una vacanza, lasciamo dietro di noi una traccia di dati che rivela molteplici informazioni sulla nostra persona, sulla frequenza dei nostri acquisti o semplicemente sulle nostre preferenze, mediante le quali è potenzialmente possibile controllare alcune nostre azioni future.

Vedremo come gli Anonymous Credentials Systems (altrimenti detti Pseudonym Systems) ci vengono incontro eliminando questa traccia, attraverso l'utilizzo di identità artificiali, note come pseudonimi, e rendendo in questo modo le transazioni perfettamente anonime.

Un Pseudonym System ad esempio ci consentirebbe di acquistare libri o altro confermando solo il nostro limite di spesa senza dover rivelare il nostro numero di carta di credito e senza comunicare il nostro saldo bancario o fornire i nostri dati anagrafici. Darci, insomma, la possibilità di acquistare beni o accedere a servizi senza rivelare inutili informazioni personali.

A differenza di altri sistemi di gestione delle identità che trasmettono parti della vera identità di un utente, i modelli di Anonymous Credential System **tutelano la privacy comunicando solo gli pseudonimi** in modo tale che le reali informazioni di identità non possano mai essere intercettate o rivelate.

Soluzioni rivolte a minimizzare il rilascio di informazioni personali possono essere basate su tecniche atte ad anonimizzare il canale di trasporto dei dati tra utente e servizi. Queste tecniche possono rendere anonimo l'utente agli occhi dei servizi “esterni” (o di servizio: vedi Federation SSO) oppure, se desiderato, al service provider stesso.

Il provider dei servizi potrebbe richiedere l'autenticazione degli utenti (ad esempio per controllare l'accesso alle risorse): in questo caso gli utenti devono comprovare la loro identità oppure almeno il possesso di un qualche tipo di certificato.

Il certificato potrebbe contenere uno pseudonome dell'utente oppure solo gli attributi richiesti per accedere ad un determinato servizio.

Ad ogni modo, anche utilizzando certificati definiti dal X.509 o SPKI, l'utilizzo dei certificati da parte dell'utente rimarrebbero linkabili (problema del tracciamento) tra loro consentendo in certi casi di risalire ai dati dell'utente attraverso la combinazione incrociata di informazioni su più transazioni.

Questa **correlazione di dati** può essere vanificata, evitata, utilizzando un Anonymous Credential System. In questo tipo di sistema le organizzazioni (service provider e chi rilascia credenziali) conoscono gli utenti solo attraverso i loro pseudonomi. Una organizzazione, quindi, può rilasciare una credenziale ad un pseudonome e l'utente corrispondente può comprovare il possesso di questa credenziale ad una seconda organizzazione (che conoscerà l'utente mediante un altro pseudonome) senza **rivelare niente di più del possesso della credenziale stessa**.

Messa in questi termini “descrittivi” la cosa sembra più semplice di quello che in realtà è!

I Pseudonym System sono studiati da anni e sono stati formulati vari modelli purtroppo non sempre applicabili alla realtà. Quello che ci ha indotto a trattare questo argomento proprio quest'anno è stato l'annuncio (i primi di gennaio 2007) da parte di IBM del rilascio di un nuovo prodotto:

IDEMIX
(Identity Mixer)

Notizie

Notizie precedenti

Eventi

Press Room

Altri link

- Rivista OL3
- Pubblicazioni (US)
- Analisti IT (US)

Notizie

Idemix: un software per la tutela dell'identità dei consumatori sul Web

Dal centro di Ricerca IBM di Zurigo

IBM ha annunciato oggi un software che consente di nascondere o rendere anonimi i dati personali sul Web, assicurando così la protezione da furti di identità e altri abusi. Sviluppato dai ricercatori del laboratorio IBM di Zurigo, il software – nome in codice Identity Mixer, o "Idemix" abbreviato – darà ai consumatori la possibilità di acquistare beni e servizi su Internet senza rivelare informazioni personali.

IBM contribuirà con questo software al progetto Eclipse Higgins, un'iniziativa Open Source per lo lo sviluppo di software per una gestione delle identità "centrata sull'utente". Questo orientamento all'utente implica che ogni individuo potrà controllare, in modo attivo e sicuro, chi ha avuto accesso alle proprie informazioni personali online, quali conto corrente e numeri di carta di credito, o documentazione medica o lavorativa, anziché affidare la gestione di tali informazioni esclusivamente alle istituzioni, come avviene oggi.

Quando i consumatori inviano i propri dati personali per scaricare musica o abbonarsi a una newsletter online, lasciano dietro di sé una traccia di dati, che rivela informazioni sulle dimensioni, sulla frequenza e sulla fonte dei loro acquisti on line, mediante la quale è possibile risalire all'utente. Il software Idemix IBM elimina questa traccia, usando informazioni di identità artificiali note come pseudonimi per rendere le transazioni on line anonime. Ad esempio, il software consente di acquistare libri o abbigliamento senza rivelare il proprio numero di carta di credito. Può confermare il limite di spesa di una persona senza comunicare il suo saldo bancario, o fornire l'attestazione dell'età senza rivelare la data di nascita.

A differenza di altri sistemi di gestione delle identità, che trasmettono parti della vera identità di un utente, i sistemi creati con il software Idemix

Vedi anche

→ Visita il sito del Laboratorio di Zurigo

In realtà **IDEMIX** non è un vero e proprio prodotto, ma consiste “solo” in:

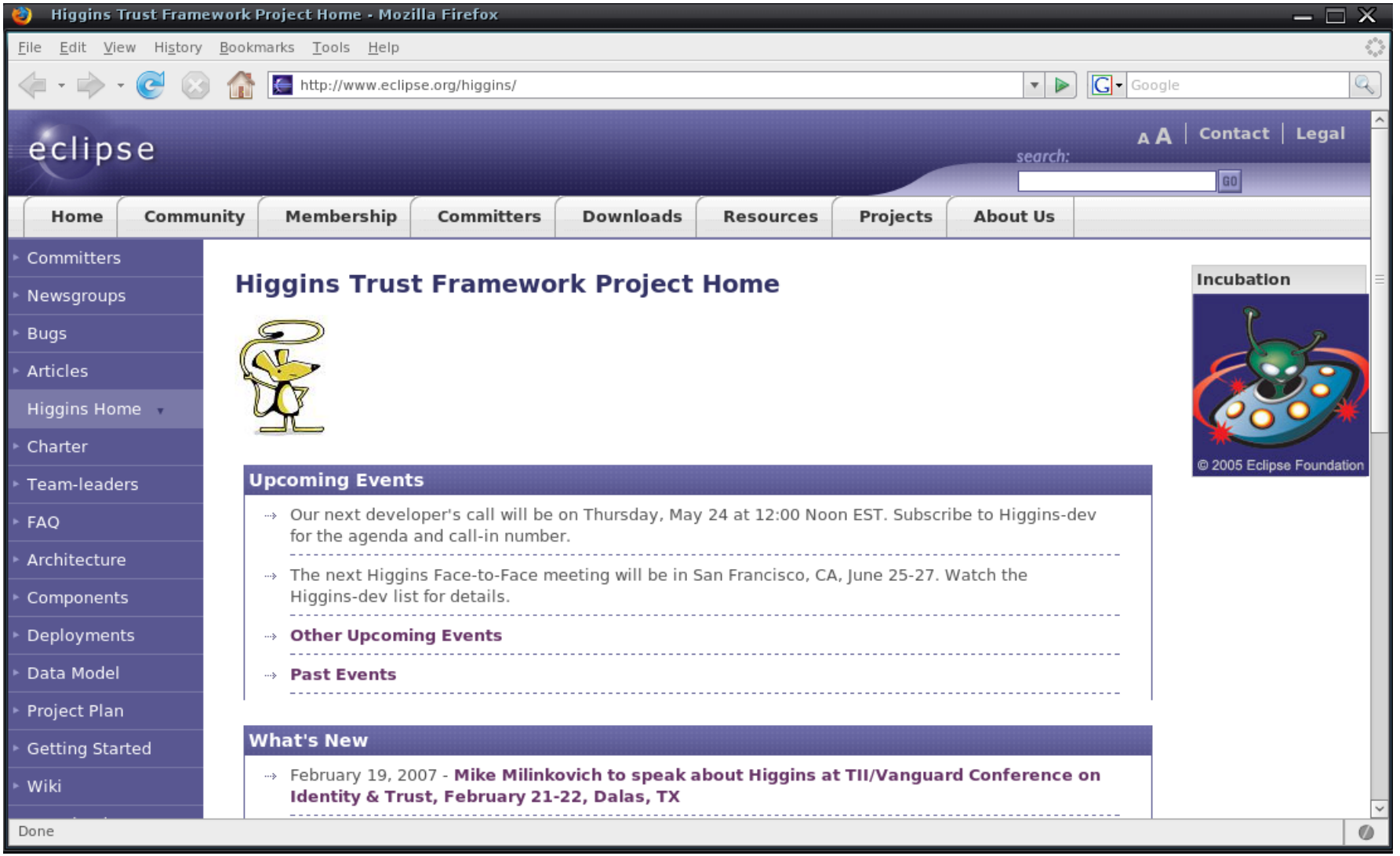
- **Risultati teorici** (algoritmi crittografici e proocolli per realizzare un sistema efficiente e che risolve alcune problematiche come vedremo dopo)

- **Un prototipo** di implementazione del protocollo crittografico.

Il “solo” non è poi così poco e vedremo tra poco il perchè.

Un'altra cosa interessante di IDEMIX è il suo rilascio sotto **licenza GPL** e l'inserimento dei suoi risultati all'interno del Progetto

Eclipse Higgins:



Due parole al volo sul Progetto:

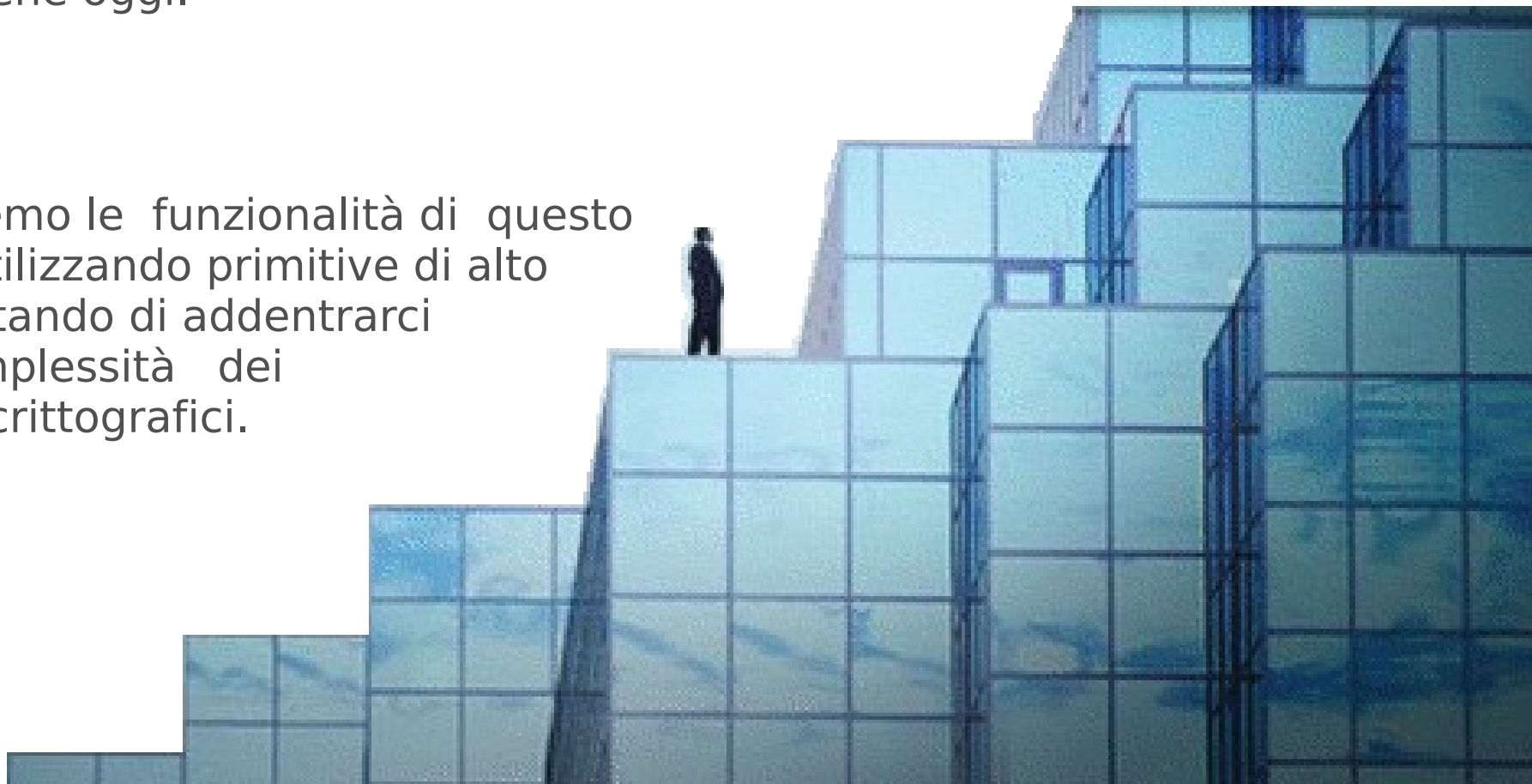
Eclipse è un progetto Open Source il cui scopo è la creazione di una piattaforma di sviluppo open comprensiva di framework estendibili, tools e tutto ciò che occorre per creare e manutenere software.

Il Progetto Eclipse Higgins è stato annunciato nel febbraio del 2006 dal Berkman Center for Internet and Society (Harvard Law School), IBM, Novell e Parity Communication. E' stata la prima iniziativa di **gestione delle identità centrata sull'utente** a seguire il modello Open Source, cui contribuiscono molti sviluppatori e grazie a questo viene perfezionato attraverso la collaborazione collaborativa.

la gestione delle identità centrata sull'utente:

questo **orientamento all'utente** implica che ogni individuo potrà controllare, in modo attivo e sicuro, chi ha avuto accesso alle proprie informazioni personali online, quali conto corrente e numeri di carta di credito, o documentazione medica o lavorativa, anziché affidare la gestione di tali informazioni esclusivamente alle istituzioni,
come avviene oggi.

Descriveremo le funzionalità di questo modello utilizzando primitive di alto livello, evitando di addentrarci nella complessità dei protocolli crittografici.



definizione descrittiva:

Un Anonymous Credential System consiste di utenti e organizzazioni.

Le organizzazioni conoscono gli utenti solo tramite pseudonomi.

Pseudonomi diferenti dello stesso utente non possono venire messi in relazione tra loro.

Una organizzazione può rilasciare una credenziale ad uno pseudonome e l'utente corrispondente potrà mostrare questa credenziale ad un'altra organizzazione senza rivelare altro se non il possesso della stessa.

La seconda organizzazione conoscerà l'utente con uno speudonome differente dal primo

definizione descrittiva:

Le credenziali possono essere mostrate o una sola volta (one-show credentials) oppure un numero illimitato di volte (multiple-show credentials).

Il possesso di una credenziale del secondo tipo può essere dimostrato tutte le volte che l'utente desidera, ma non potranno venire mai messe in correlazione tra loro.

Esiste poi un altro tipo di organizzazione (de-anonymizing organization) di cui parleremo più avanti.

proprietà desiderate:

l'impossibilità di poter creare una credenziale per un particolare utente attraverso l'abuso da parte di altri utenti ed organizzazioni esterne: ogni pseudonome e credenziale devono appartenere ad un ben definito utente [LRSW99].

In particolare deve essere impossibile da parte di più utenti il poter dimostrare il possesso di alcune credenziali ad una organizzazione per far ottenere una particolare credenziale ad uno di questi che da solo non potrebbe ottenere. In questi casi si dice che il sistema possiede una **consistency of credentials**).

Poiché le organizzazioni potrebbero essere entità autonome, è desiderabile che siano autonome nel poter scegliere le proprie chiavi indipendentemente dalla altre entità. (comodità di gestione, sviluppo di proprie interfacce software)

Deve essere garantita la **privacy dell'utente**: un' organizzazione non deve poter conoscere altro di un utente a parte il fatto che l'utente possiede particolari credenziali, anche se collabora con altre organizzazioni. In particolare due **pseudonomi di uno stesso user non devono poter venire messi in correlazione** (no tracciamento e profilazione) [Cha85, CE87, Dam90, Che95, Bra99, LRSW99].

proprietà desiderate:

Il sistema deve essere efficente!

- protocolli efficenti

- interazioni tra minori entità possibili

- “quantità” di comunicazione minima.

Es. un utente che possiede una credenziale di tipo multiple-show, può dimostrare il suo possesso senza dover richiedere all'organizzazione di rilascio la rigenerazione della stessa ogni volta.

L'utente deve essere disincentivato nel voler **condividere le proprie credenziali** con altri utenti. Un metodo per fare ciò si basa sul PKI-assurred non-trasferability: la condivisione di una credenziale coincide con il condividere una particolare chiave segreta “esterna” al sistema (es: codice del proprio bancomat) [DLN96, GPR98, LRSW99]. Ad ogni modo questa chiave potrebbe non esistere! L'alternativa è la così detta all-or-nothing non-traferability: la condivisione di uno pseudonome o di una credenziale implica l'automatica condivisione di tutte le altre credenziali e pseudonomi dell'utente. I due metodi sono differenti e potrebbero anche venire utilizzati contemporaneamente.

proprietà desiderate:

Nei casi in cui Oi e Ov coincidano è possibile utilizzare l'approccio proposto da Stubblebine, Syverson e Goldschlag: ogni credenziale può essere utilizzata una sola volta e ogni volta che questo avviene ne viene generata una nuova: solo chi ha utilizzato l'ultima credenziale può usare quella nuova; questo rende scomodo e “tediante” lo share delle C.

Per finire potrebbe essere desiderabile avere la possibilità di **revoca dell'anonimato**, nel caso l'utente svolgesse operazioni (transazioni) illegali.

Se si rivela la vera identità dell'utente si parla di **global anonymity** revocation; se si rivela solo un particolare speudonome di una paritcolare organizzazione di rilascio (Oi) si parla di **local anonymity** revocation.

Entrambe i metodi devono essere **opzionali** (ritornando al discorso della centralità dell'utente: quindi è l'utente che decide se la revoca è possibile oppure no, così come decide quali credenziali mostrare).

Prima di IDEMIX:

lo scenario di più utenti che pur rimanendo anonimi all'interno di un sistema possono scambiare credenziali da una organizzazione ad un'altra è stato trattato per la prima volta da Chaum [**Cha85**].

Poco dopo, sempre Chaum in collaborazione con Evertse [**CE87**] proposero una soluzione basata su una semi-trusted terza parte coinvolta durante la transazione.

Ma... questa terza parte è indesiderabile per i motivi di efficienza ed autonomia da parte delle organizzazioni.

Prima di IDEMIX:

Altri schemi sono stati proposti:

Damgard [**Dam90**] – funzioni one-way e zero-knowledge proof (non applicabile nella pratica)

Chen [**Che95**] - efficiente ma senza risoluzione nel caso di utenti che abusano del sistema e l'utente deve richiedere più signature dall'organizzazione di rilascio per poter mostrare una credenziale non tracciabile più volte.

Lysyanskaya, Rivest, Sahai, Wolf [**LRSW99**] – più efficiente ma con lostesso problema di [Che95]

Il concetto di revoca dell'anonimato lo troviamo negli schemi dei sistemi di pagamento [**BGK95**].

problemi comuni:

> poca efficienza
> impossibilità di revoca
> condivisione degli stessi gruppi logaritmici discreti.

idemix : pseudonymity for e-transactions

Risolve i problemi lasciati insoluti dagli altri modelli e per la prima volta introduce la possibilità di revoca dell'anonimato.

Ricordiamo ancora una volta che questa feature è assolutamente facoltativa: è l'utente che sceglie se essere assolutamente anonimo durante una transazione oppure se la sua anonimità può essere revocata.

descrizione operativa:

Un utente U può ottenere una credenziale C da un organizzazione (issuing) Oi e poi mostrarla ad una seconda organizzazione Ov (verifying).

La credenziale è sempre rilasciata ad uno pseudonome N sotto il quale l'utente U è registrato (oppure solo conosciuto) da Oi.

La credenziale può contenere alcuni attributi (attr).

Quando la credenziale viene mostrata l'utente può decidere quali attributi della stessa verranno rivelati (attr')

La registrazione di uno pseudonimo, il rilascio della credenziale e la sua verifica sono tutte attività regolate da particolari protocolli tra l'utente e le organizzazioni.

descrizione operativa:

Un utente U ha una (singola) master secret Su che è linkata a tutti gli pseudonomi e credenziali rilasciategli.

Le organizzazioni di rilascio e di verifica hanno tutte una coppia di chiavi pubblica/privata.

L'organizzazione che rilascia la credenziale utilizza la sua chiave privata per generare la credenziale stessa.

La credenziale può venire poi verificata utilizzando la chiave pubblica dell'organizzazione che l'ha rilasciata, sia dall'utente nel momento in cui la riceve per utilizzarla, sia dall'organizzazione alla quale l'utente la mostrerà successivamente.

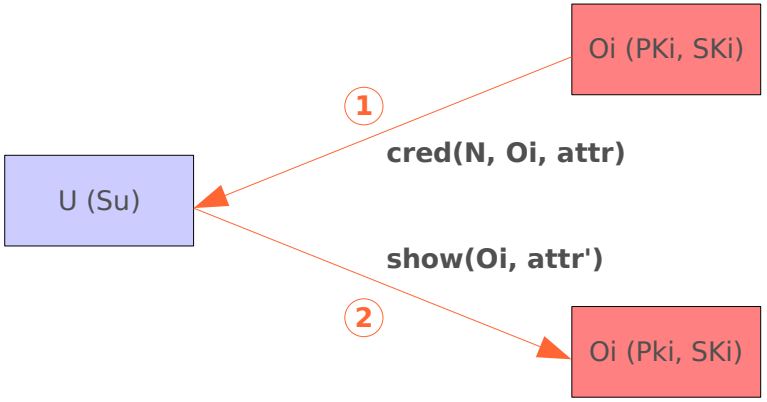
Quando l'utente mostra la sua credenziale, utilizza la chiave pubblica dell'organizzazione verificatrice.

Ottenere una credenziale da Oi e mostrarla successivamente a Ov è così schematizzabile:

U contatta Oi e stabilisce uno pseudonimo N con Oi.

Se N lo permette, Oi produce una credenziale C con particolari attributi (attr), firma una “dichiarazione” (statement) contenente attr, N e spedisce il tutto a U.

Ora U può mostrare queste credenziali a Ov.



Utilizzando la metodologia di zero-knowledge proof , U convince Ov di possedere una signature generata da Oi nello statement che continene N e attr.

E' da sottolineare il fatto che U non rivela nessuna altra informazione a Ov. In particolare U non manda a Ov le credenziali attuali.

Questo modo di mostrare una credenziale insieme alla proprietà di **zero-knowledge proof** assicura l'impossibilità di mettere in relazione tra loro (unlinkability) la stessa credenziale mostrata più volte e l'impossibilità di mettere in relazione tra loro credenziale e relativo pseudonome.

Questo significa che U può mostrare C a Ov (o ad altri verificatori) un numero illimitato di volte senza che le credenziali possano venire correlate tra loro o allo pseudonome di U (eccezione fatta per le credenziali one-show).

Questa proprietà di non tracciabilità viene mantenuta anche se Ov e Oi dovessero coincidere (stessa organizzazione oppure stesso repository).

Da notare che questa proprietà di non tracciabilità fa si che l'utente risulti completamente anonimo nei confronti dell'organizzazione verificatore.

Ovviamente il vero anonimato presuppone che anche il canale di comunicazione utilizzato supporti un minimo di protezione!

Credential Options and Attributes:

Le credenziali possono avere **opzioni** (es. one-show, multi-show) ed **attributi**.

Esempi di attributi:

la data di scadenza della credenziale

l'età dell'utente

un sottotipo di credenziale...

Quando si mostra una credenziale l'utente può decidere **quali attributi comprovare e come**: un esempio se l'attributo è “età=40” si può comprovare solo “età > 18”.

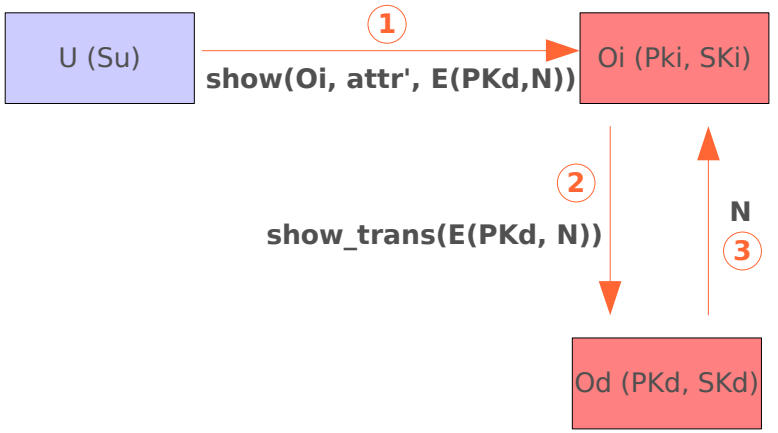
De-Anonymizible Show of a Credential:

In meccanismo di De-Anonymization permette di rivelare l'identità di un utente (global de-anonymization, anche detta **global anonymity revocation**) oppure di rivelare lo pseudonome rilasciata da una particolare organizzazione (local de-anonymization, aka **local anonymity revocation**).

Global può tornare utile nel momento in cui un utente cade in tentazione e fa qualcosa di illegale una transazione; la versione local è applicabile da una Oi nei casi in cui un utente abusi in un qualche modo nell'utilizzo della sua credenziale.

Entrambi questi metodi richiedono una cooperazione da parte dell'utente e l'esistenza di una particolare organizzazione Od (de-anonymizing organization).

De-Anonymizible Show of a Credential:



Utilizzando questa variante dello “show protocol”:

Od possiede una coppia di chiavi pubblica e privata (PKd e SKd).

U cripta N con la chiave pubblica di Od: E(PKd, N).

Questa encryption è verificabile, ossia Ov ha la prova che Od può decriptare e rivelare N (attraverso il protocollo di “show transcript”).

De-Anonymizible Show of a Credential:

Possono esservi più di una organizzazione Od in un sistema.

Includendo questa condizione di de-anonymizing U può decidere sotto quali condizioni consentire la perdita dell'anonimato.

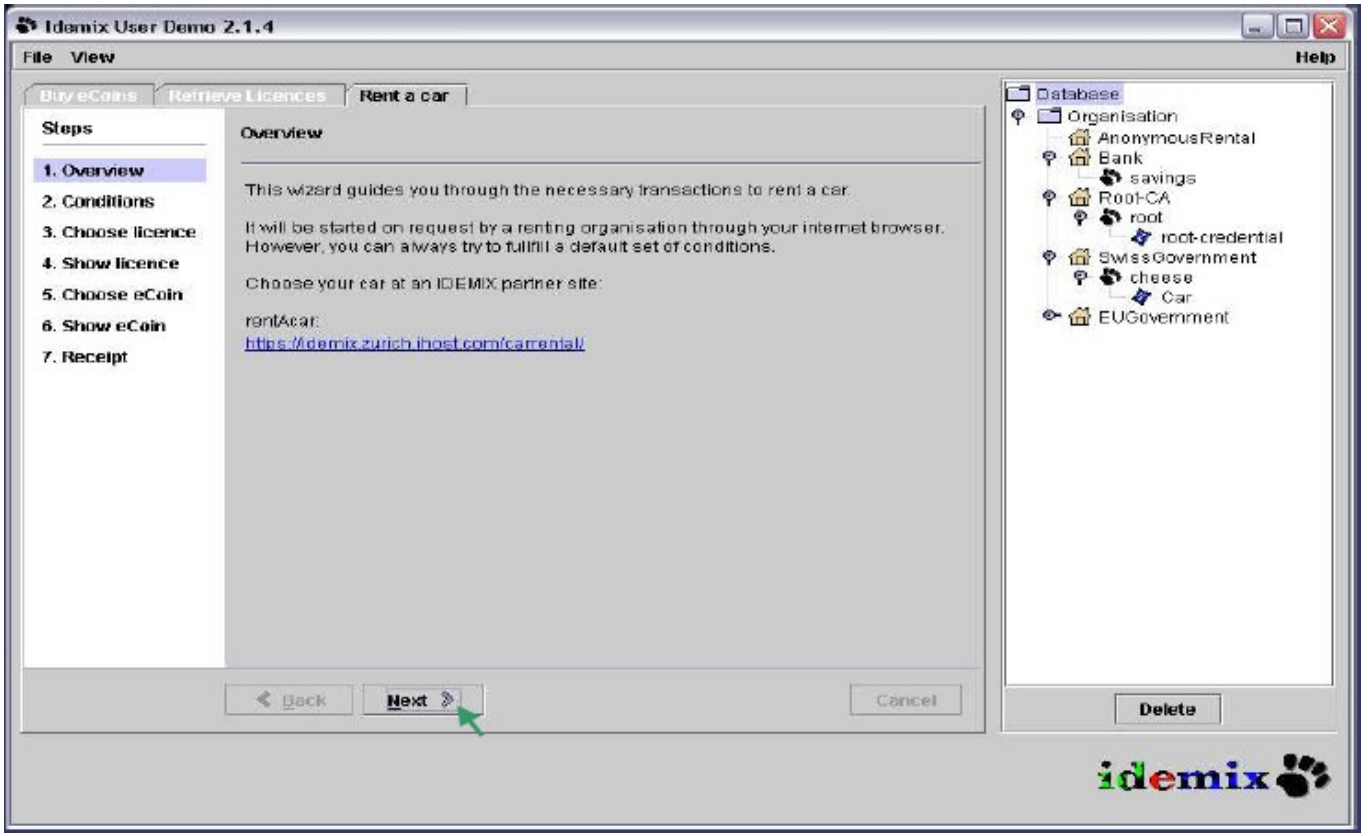
Quando necessario Ov può mandare una richiesta di transcript a Od e Od può decidere se le condizioni sono soddisfatte per de-nononimizzare N (metodo “local”).

Il metodo Global utilizza la stessa tecnica e richiede l'esistenza di una “Root Pseudonym Authority” che è in grado di mappare uno speudonome alla vera identità dello user.

Applicazioni

IDEMIX non è un vero prodotto ma è diventato parte del framework Eclipse.

Esiste comunque un prototipo sviluppato in Java:



Applicativi

Il core di questo prototipo è il **package NymSystem** che implementa i componenti UserNymSystem, OrgNymSystem e DeAnOrgNymSystem, ognuno dei quali offre le funzionalità per le specifiche operazioni crittografiche eseguite dalle varie entità.

Ad oggi, comunque, non esiste nessuna soluzione applicativa completa.

Alcune parti dello studio degli algoritmi e protocolli di comunicazione di IDEMIX sono stati incorporati all'interno di Tivoli per garantire l'anonimato in ambienti Federation e SSO.

Link:

Presentazione prodotto Idemix:

http://www.ibm.com/news/it/it/2007/01/260.html

Laboratori di ricerca di Zurigo:

http://www.zurich.ibm.com/security/idemix/

Progetto Eclipse Higgings:

http://www.eclipse.org/higgins/

e-mail:

g.ciotti@winstonsmith.info