

## Preferenze Privacy degli Utenti

### Parte IV

# Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
1.1	Rilascio diretto . . . . .	3
1.2	Controllo di accesso interattivo . . . . .	4
1.2.1	Interazione senza condizioni da parte del client . . . . .	4
1.2.2	Negoziazione <i>multi-step</i> . . . . .	4
1.2.3	Interazione a due step . . . . .	5
1.3	Preferenze degli utenti . . . . .	5
<b>2</b>	<b>Cost-sensitive Trust Negotiation</b>	<b>7</b>

# Capitolo 1

## Introduzione

### Privacy dell'identità degli utenti

Gli utenti preferiscono restare anonimi o comunque non condividere troppe informazioni quando operano nel cloud. Alcuni scenari:

- **Tecniche di comunicazione anonima**
- **Privacy in location-based services** (protezione della location quando sensibile)
- **Attribute-based control access:** è un problema lato server, non ci si basa più su chi un utente sia (l'identità) ma sugli attributi che ha (certificati che l'utente presenta)
- **Supporto alle preferenze privacy degli utenti:** problema lato utente; *se mi viene chiesto un documento d'identità, non è che do al server tutto il portafoglio*

Gli utenti potrebbero voler specificare le proprie scelte in termini di politiche del trattamento dei dati, quando:

- condividono delle proprie risorse con server esterni (ad esempio i social media)
- vengono rilasciate informazioni nelle interazioni digitali (ad esempio lascio la carta di credito per accedere a un servizio)

→ Due aspetti di **protezione:**

- **rilascio diretto:** regola quando, a chi e perchè un utente rilascia informazioni (es. sto comprando qualcosa)
- **uso secondario:** regola l'uso e la profilazione dei dati da terze parti; anche questo deve essere sotto il controllo dell'utente

## 1.1 Rilascio diretto

La community di ricerca ha sviluppato diverse tecniche per regolare le interazioni tra *parti sconosciute*, definendo dei meccanismi di **attribute-based access control**: consistono in una dipendenza dell'accesso rispetto alle proprietà che un utente ha. Quello che gli utenti possono fare dipende dagli attributi che possiedono, verificati attraverso i **certificati**.

L'*access control* non risponde più sì o no, ma risponde con i requisiti che il richiedente deve soddisfare per avere l'accesso. Non solo i server vanno protetti ma anche gli utenti, per questo vanno introdotte delle **forme di negoziazione**.

### Esempio

Se vogliamo cambiare filosofia, in un sistema aperto (non so chi è l'utente) se voglio chiedere *"tu soddisfi i requisiti per ottenere l'accesso?"*, nascono una serie di problematiche:

- come specificare l'autorizzazione
- *engine* per il controllo della politica
- anche la politica potrebbe essere confidenziale (*non voglio dirti che faccio certi controlli*)
- come chiedere le cose all'utente
- l'utente può avere delle controrichieste (*hai la certificazione per chiedermi la carta di credito? la cripti?*)

Questo dialogo deve terminare, deve essere **corretto** e **minimale** nei termini delle informazioni rilasciate; tipicamente vengono usati linguaggi basati sul paradigma logico.

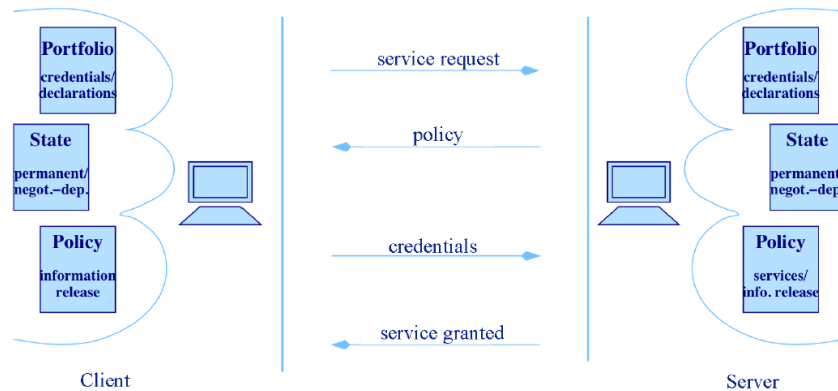
## 1.2 Controllo di accesso interattivo

Il client è colui che richiede il servizio (utente), ha con sé:

- **portfolio** (credenziali e proprietà)
- **stato** (stato di informazioni che vuole mantenere)
- **politica**

Lo stesso vale per il server, cioè colui che offre il servizio.

### 1.2.1 Interazione senza condizioni da parte del client



La policy del server sta ad indicare ciò che il client deve dimostrare, tramite i certificati, per poter accedere al servizio.

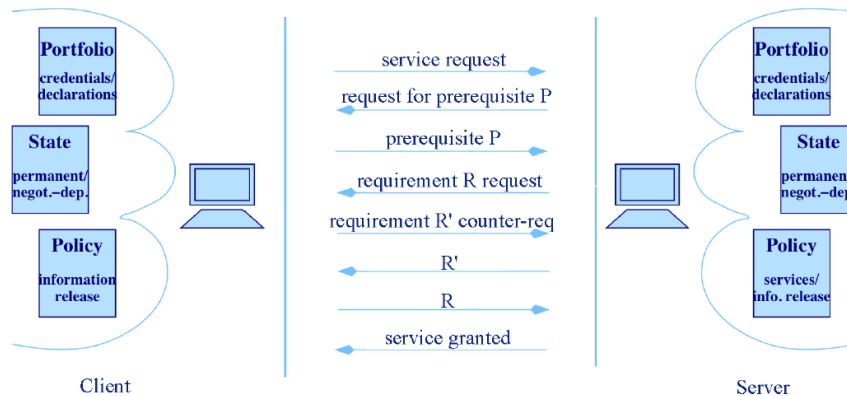
### 1.2.2 Negoziazione *multi-step*

In questo caso c'è una negoziazione tra client e server → bisogna stabilire fiducia tra le due parti.

Il server per essere *privacy-friendly* dovrebbe chiedere i dati tutti assieme.

### 1.2.3 Interazione a due step

Per essere *gentili* con l'utente viene fatta una distinzione tra i prerequisiti per l'accesso (necessari ma non sufficienti) e il requisito vero e proprio con eventuale controrichiesta da parte dell'utente.



### Esistenti/emegenti tecnologie di supporto a ABAC

- U-Prove/Idemix: fornisce avanzate tecnologie di gestione dei certificati (i certificati odierni ti permettono di estrapolare dal certificato solo l'informazione che voglio fornire all'altra parte, senza fornire tutto il certificato).
- XACML: standard di oggi per l'interoperabilità delle politiche di controllo degli accessi

## 1.3 Preferenze degli utenti

Le specifiche di controllo degli accessi non sempre si adattano bene con il problema lato utente:

- **+** sono espressive, potenti e permettono all'utente di specificare se determinate informazioni possono o non possono essere rilasciate
- **-** non permettono agli utenti di esprimere che preferirebbero rilasciare determinate informazioni piuttosto che altre, nel contesto in cui ne sia data la possibilità

→ È necessario fornire agli utenti strumenti per definire in modo efficace le preferenze sulla privacy riguardo al rilascio delle loro informazioni

### *Desiderata*

- **Context-based preferences:** sono disposto a rilasciare un'informazione solo se mi trovo in un certo contesto (*lascio la carta solo quando devo pagare*)
- **Forbidden disclosures:** certe cose insieme non le rilascio
- **Associazioni sensibili:** associazioni che sono sensibili, perché sono *quasi identifier* o perché non voglio che tu le veda
- **Limited disclosure:** *se mi chiedi di essere maggiorenne, te lo dimostro ma non voglio dirti la mia età*
- **Instance-based preferences:** *se la mia carta sta per scadere, preferisco lasciarti quella*
- **History-based preferences:** magari ho già rilasciato qualcosa in passato
- **Proof-based preferences:** *preferisco darti la prova che possiedo un passaporto invece che il passaporto stesso*
- **Non-linkability preferences:** *preferisco lasciarti informazioni che, linkate con terze parti, mi identificano di meno*

Esistono diversi approcci per regolare la preferenza sulla privacy per gli utenti, che andiamo a vedere nei prossimi capitoli.

## Capitolo 2

# Cost-sensitive Trust Negotiation