

# **Identity Management for SOA --- Architectural Pattern**

**Elena Ferrari, Barbara Carminati, Chi Hung Chi**

**Enhancing IDM with Automated Trust  
Negotiation Functionalities**

# Trust Negotiation Component

## Intent:

- Provide a user centric management of user attributes

## Motivation:

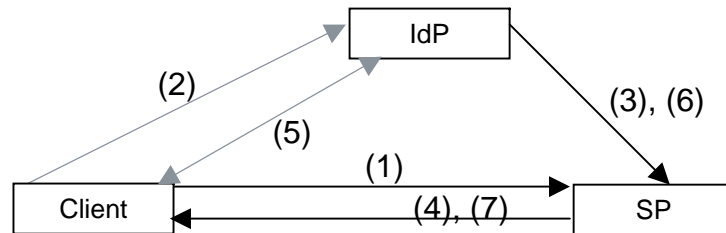
- Users do not have to submit their attributes more than once in the federation but previously released user attributes can be shared by different SPs upon a negotiation.
- Identity management is more user-centric than in traditional identity management systems in that the user can negotiate with an SP/IdP the requirements he/she would like to be satisfied (e.g., privacy preferences) for the release of his/her attributes.

## Applicability:

- All the environments where release of a service/resource are governed by access control policies specified not only in terms of user IDs but in terms of requirements and conditions against user attributes.

# Trust negotiation component

## Structure



## Participants

- Client
- Service Provider (SP):
  - Provide services regulated by policies on user attributes
- Identity Provider (IdP)

# Trust negotiation component

## Collaborations

- Client requires services (1) to SP; Client authenticates to IdP (2); IdP asserts identity to SP (3); SP requires some attributes from Client (4) to deliver the service. If Client trusts the SP, it negotiates with it the requirements for the release of its attributes, otherwise the negotiation and attribute disclosure is carried out with IdP (5). In this latter case, IdP sends SP an assertion stating that client's attributes satisfy SP's policies, without revealing the attributes themselves (6); SP delivers the service to the Client (7). When Client requests services to another SP, the SP can negotiate the release of user attributes with other SPs possibly storing them, if any.

## Consequences

- Enable the Client to have full control over the disclosure of personal attributes
- Speed up subsequent negotiations

## Implementation

TBD by taking into account the already proposed architectural patterns

Related Patterns: Private Identity Selector