Contribution ID: **56** Type: **Sessions contributions**

A solution for Access Delegation based on SAML

Wednesday, 21 May 2014 14:00 (20 minutes)

During this presentation we would like to show a solution to Access Delegation we have implemented based on 'SAML Condition for Delegation'. In particular, a message flow protocol has been designed and RESTful web services, based on OpenSAML library, have been implemented to provide Service Oriented Delegation.
Access Delegation consists in enabling a user or process to act on behalf of another user or process: in other words, a user or process obtains a subset of the privileges of another user or process preserving his/her/its identity. Several solutions are currently applied to implement it: e.g. Proxy Certificates, OAuth 2.0, and KeyStone bearer Token. According to the analysis performed the first solution, used in Grid Environment, has poor performances, the second one is not totally secure, even if it is very popular in the mass market. The third one does not preserve the identity of the delegate, so it is not a full delegation technique, but an 'impersonation' process which is much less secure.
SAML is a very popular solution for Identity Federation: OASIS specification 'SAML 2.0 Condition for Delegation' extends SAML Tokens to obtain delegation support. This solution has been chosen after some comparisons with the other mentioned technologies. It has been considered the best one because it is based on a consolidated standard, it implements full delegation, it is very secure because messages are signed and does not introduce a strong performance degradation.

Wider impact and conclusions

SAML 2.0 Condition for Delegation is a complete and secure specification providing a full delegation technique. SAML technology is also used for Identity Federation: the use of a single technology to obtain Delegation and Identity Federation provides a great flexibility to address different use cases. For example, inside a Federation a user belonging to a certain domain can delegate a user (or process) of another domain to perform some actions: this concept is useful for Cloud Storage federations where a user of a certain Cloud member delegates a Process in the Cloud to manage objects on another Cloud of the federation. A SAML based Identity Federation Module has also been implemented: it is integrated with Shibboleth and provides Identity Federation for stateless services. This feature can be used together with the Delegation Module to meet use cases similar to what has been described above and providing a very complete framework for Federated Identity and Access Management.

Description of work

The base architecture of SAML federation includes an Identity Provider (IdP) and a Service Provider (SP): on it we defined a message flow involving the actors of a typical delegation process, i.e.:
*the delegator, who is the actual owner of identity and privileges*the delegate, who is provided by the delegator with a subset of his/her privileges for a limited amount of time.
The message flow is the following: 1.the authenticated delegator asks the IdP for a delegation Assertion defining who is the delegate, which privileges have to be provided and the delegation lifetime 2.if the request is successful the IdP produces a signed 'delegation assertion' containing the requested information 3.the delegate is provided with the assertion, that is used together with his/her credentials to access the target 4.the target SP authenticates the delegate and assigns to him/her the privileges in the assertion.
This message flow is implemented by a set of RESTful web services. Since services for Authentication, Authorization and SAML Federation are provided, everything can be integrated to meet several use cases.
This consideration can be better explained by the following use case: let's consider a Federated Cloud Storage on which objects owned by some users are stored. The Federation enables users to have a consistent and homogeneous vision of his/her objects regardless of the actual location among the member clouds. Every member manages the privileges on its objects by its Identity Management System. If a user is logged in a certain cloud and has to get an object on a remote cloud without directly accessing it, he/she should 'delegate' the local cloud to access the remote one, by providing it with his/her read privilege on the object. In order to complete this use case, we may add SAML Federation to Delegation: in this case we would be sure that the

user has a single identity in the federation and identity mapping among different identities of the same user is not required

Primary author: FORMISANO, Ciro (Engineering Ingegneria Informatica)

Co-authors: TRAVAGLINO, Ermanno (Engineering Ingegneria Informatica S.p.A.); MATRANGA, isabel (engineering ingegneria informatica spa)

Presenter: FORMISANO, Ciro (Engineering Ingegneria Informatica)

Session Classification: Authentication & Authorisation

Track Classification: Integrated AAI services (Track Leaders: P. Solagna, A. Bonvin, J. Kewley)