Federated Authorization
Implementing Grouper to federate user authorization

Andrea Biancini[1], Maarten Kremers[2], Maria Laura Mantovani[1] and Marco Malavolti[1]
[1]*Consortium GARR*
{*andrea.biancini, marialaura.mantovani, marco.malavolti*}*@garr.it*
[2]*SURFnet*
maarten.kremers@surfnet.nl

Keywords: Identity Federation, Authorization, Grouper, Shibboleth, SAML

Abstract: In this article the authors present the results and future activities of a task aimed at investigating solutions in the field of authorization within federated envirnoments. The general idea is to extend federated identities from the management of user authentication to include also authorization. To support these activities, Grouper has been embraced as a central solution for managing user groups and attributes. This article, in particular, will explore the general scenario for federated authentication and will also drive some conclusions about how a group and attribute solution could be introduced into existing federations. In the article, a very brief description of the use-cases will be presented. This use-cases taken into consideration will drive the research activities ongoing and will prove the effectiveness of the solusion described and implemented.

1 INTRODUCTION

All research and education communities have been pretty active in promoting Identity Federations (Gaedke et al., 2005) to simplify identity and access management within complex and heterogeneous environments. In fact, as web is in increasingly used as a platform for heterogeneous applications, we are faced with new requirements to authentication, authorization and identity management. Modern architectures have to control access not only to single, isolated systems, but to whole business-spanning federations of applications and services.

So far the Identity Federations, in particular the ones within the research and university community, have provided the higher benefits in implementing authentication processes. The main goals for an Identity Federation seem to be:

1. giving a delegated mechanism to manage user identification among different entities and within different subjects;
2. providing a set of attributes to an authenticated user to be used by the final application to have information about the user and his properties.

Into these tasks the existing Identity Federations, for instance the ones based on SAML (Cantor et al., 2005b; Cantor et al., 2005a) and implemented with Shibboleth (Morgan et al., 2004), are pretty effective and permit to reach high standard goals in distributed and delegated identity management. The same level of general agreement has not been achieved, so far, in the field of authorization. To distinguish authorization and authentication we can refer to the following definitions:

> **Authentication** is the act of confirming the truth of an attribute of a single piece of data or entity (the user of an application, for instance).
>
> **Authorization** is the function of specifying access rights to resources related to information security and computer security in general and to access control in particular. More formally, "to authorize" is to define an access policy.

The goal of this article is to describe an experimentation ongoing with the aim to extend the success of current identity federations to user authentication. The article will continue with the following structure: in the next section the problem of user authorization will be presented and described; then the following section will describe the solution implemented to tackle the problems identified; the last section will then propose the results achieved so far and the future work and activities.

2 USER AUTHORIZATION WITHIN IDENTITY FEDERATIONS

Together with authentication, authorization is a key aspect of identity management. From the point of view of an application, it is really important to know which operations can be performed by a specific user and which must be denied. Traditionally identity federations have solved the authorization problems with two approaches that are one the opposite of the next:

- On one hand, we may have SP managed authorization. In this case the authorization is managed completely by the SP that also implements a mechanism to explicitly express authorization rights for each user (or group).
- On the other hand, we may have IdP or Attribute Authority (AA) managed authorization. Where an Attribute Authority is defined as a trusted source of attributes for entities within the domain (ECMA, 2001). In this case the SP will not manage the authorization rights on a user by user basis, but it will leverages user attributes released by the IdP (or by an external AA) to associate users to specific authoritative groups.

These two approaches are both valid and prove to solve real problems in current applications. The main idea behind the design of an authorization process, is that of delegating the management of authorization information to the entity which has more interest in managing them. The fine grain management of such information is quite heavy and time consuming, so in order to have this process implemented effectively, it is really important that the subject more involved will be the one taking the responsibility to manage it.

3 THE SOLUTION PROPOSED

In this article we propose a solution that leverages an authorization process using an Attribute Authority (Novakov, 2013). Our solution involves the use of a central tool that could be used to manage authorization attributes for an inter/intra federation environment. This tool will centralize all groups and attributes definitions and will implement delegation mechanisms that could permit to have different administrators managing different user attributes.

This scheme permits to separate clearly authentication and authorization processes, identifying specific components for both tasks. Authentication will continue to be managed by IdPs while authorization will leverage a new central component within the Federation. This central component must provide a delegation mechanism to permit a proper management of authorization directly to the subjects more heavily impacted by problems and benefits of the authentication process.

This central component in the architecture, will integrate with the existing SAML federations, for example based on Shibboleth, and will provide additional attributes to user after their login. An example of the additional attributes provided can be: (i) the groups the user participate to; (ii) or some additional attribute relevant to authorization (like for example the role a user has in the groups).

The central controller for authorization should also be able to offer additional functionalities to the entities participating to the identity federation. For instance, it must provide APIs to get a list of groups and a list of users participating to groups. For this purpose a very specific protocol is emerging to provide an answer exactly to this need. This protocol is called VOOT (Virtual Organization Orthogonal Technology), as described in (Kremers, 2012) and is a subset of OpenSocial used to manage group membership. The primary motivation for VOOT is that it is a simple tool for managing virtual organization in research and education federations and is gaining growing consensum.

3.1 The main tools used

The tool that will be used to implement the central system to manage authorization is Grouper, by Internet2. Grouper (Grouper, 2014) is an enterprise access management system designed for the highly distributed management environment and heterogeneous information technology environment common to Universities. Operating a central access management system that supports both central and distributed use cases, it permits to reduce risk.

Grouper is an enterprise access management system designed specifically for the distributed environment of the university and research communities. This tool operates a central access management system while permiting delegation and beingt easily interoperable at a federation level. The tool can in fact be used to achieve a delegation mechanism for access management rules and is intended to provide mechanisms and interfaces to permit group management.

The reason to use a tool like Grouper is that it permits a controlled collaboration in authorization attributes management. With Grouper it is possible to set up groups, roles, and permissions for many purposes, such as populating and administering standing

committees, ad hoc research teams, departments, or classes. After this definition, the existing application can use the group, role, and permission information stored into Grouper to make authorization decisions.

As already described in this work, Grouper is also able to operate as a single point of control for groups and authorization attributes. Once a person is added or removed from a group, the group-related privileges are automatically updated in all of the collaborative applications integrated in the federation. Grouper allows efficient management of the membership roster at a single point.

Grouper is open-source software licensed under the Apache 2.0 license and it is very widely used to manage authoritative tasks within distributed application. The project is very vital and is continuously updated and extended to support new use cases and provide new functionalities.

3.2 Use cases implemented

To prove the feasibility and effectivenes of the architecture described, with the central authorization system, we identified some use cases. In particular we identified three different kinds of applications from real needs of our users. The applications chosen are application that, on one hand, are particularly interesting from an authorization point of view. They, on the other hand, are also the more widely used applications in our communities and so are good candidates for experimenting centralized authorization on relevant scenarios.

3.2.1 MediaWiki

MediaWiki (MediaWiki, 2014) is a structured Wiki, typically used to run a project development space, a document management system, a knowledge base, or any other groupware tool.

To prove our use case to work effectively, we will try to access MediaWiki with federated identities and we will verify that the proper access grants will be released to the user depending on group definitions inside Grouper. This use case, in summary, will only require user groups and attributes to be retrieved during the login phase in order to give the user the right access rights depending on the groups he participates to (the AA process of integrating SAML attributes of the user).

After login, the user will be placed in the right groups and this will influence which pages and sections he has access to and with which right (read/modify/create), see Figure 1.

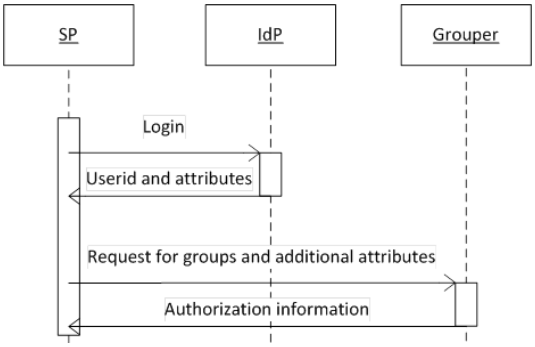

Figure 1: The processes implemented for integrating MediaWiki with Grouper.

3.2.2 Moodle

Moodle (Moodle, 2014) is a learning platform designed to provide educators, administrators and learners with a single robust, secure and integrated system to create personalised learning environments. The scope of the technical activities of the PoC will be to integrate the Moodle courses with groups defined inside Grouper. The general idea is that of describing in Grouper a structure of groups that could be reflected in Moodle as courses and classes. At this point the users can be assigned to group (with different entitlements) to represent the role each user will have for the specified course teacher or student, for instance).

To prove the use case to work effectively, we will try to access Moodle with federated identities and we will verify that the proper access grants will be released to the user depending on group definitions inside Grouper. Moreover, Moodle must be integrated with Grouper to obtain the list of courses (defined as groups in Grouper), the list of teachers and the list of students for every course. This means that this use case will extend the complexity of the previous. Apart from retrieving user groups and attributes during login with the AA SAML process, in fact, the Moodle system has the need to have lists of all groups (which are meant to be courses inside Moodle) and lists of all participants to the groups (to list teachers and students of every course). These interfaces could be implemented in VOOT with a specific connector for Grouper.

After the groups definition in Grouper, Moodle will use a specific module (called enrollment plugin) to retrieve via the VOOT interface: (i) the list of group that will be translated into courses inside Moodle; and (ii) the members the groups and their roles, this will permit to specify teachers and students of each course. After federated login to Moodle, the user will then be able to see course matherials for the courses he is a student of and to administer course matherials for the

courses he is a teacher of. The integration process described is represented in Figure 2.

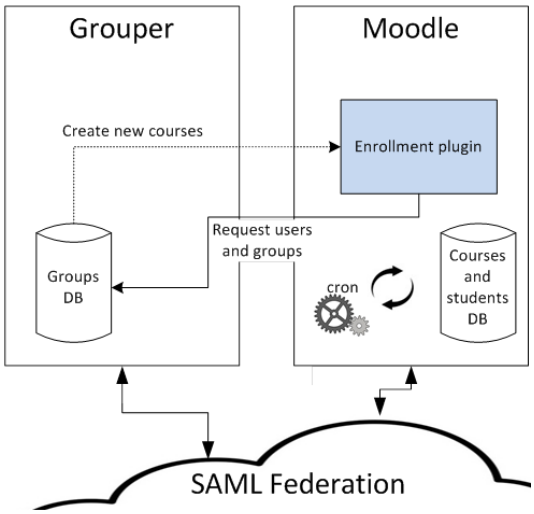

Figure 2: The key components in the integration between Moodle and Grouper.

3.2.3 GARRbox

GARRbox is a custom solution, developed by GARR, to provide a cloud personal storage for the Italian community of bio-medicine. We decided to try and integrate GARRbox into the Grouper authorization management process because of to two main reasons:

1. on one hand, GARRbox being a custom application, can help understand how emerging applications can be designed and modelled to be fully compliant with the delegated authorization process introduced into this task;
2. on the other hand, GARRbox poses some interesting aspect about user authorization which goes a little beyond the cases described so far. GARRbox, in addition to groups, needs to manage directly additional authorization attributes for the users (like for instance the size in GB of their virtual disk quota inside the application).

To prove the use case to work effectively, we externalized in Grouper the groups and the authorization attributes needed by the GARRbox application. GARRbox has been integrated with Grouper to obtain the list of groups, the users participating to every group and also to obtain a set of authorization attributes for every user accessing the application. In summary, this specific use case will be the more complex and general possible. Apart from the requirements of the previous use cases, in fact, GARRbox needs to permit a proper mechanism to delegate the administration of user authoritative attributes to the proper administrators (eventually belonging to different organizations).

Inside Grouper, in this case, there will be a folder (that in Grouper is a "group of groups") for each client institution. Each folder will have an administrator inside the technology department of the client organization. This administrator will have the responsibility to map users inside the groups and to manage their authorization attributes. The user registration within groups will permit to specify also additional information (attributes in Grouper) describing the disk usage quota every user has in the GARRbox system. The delegation scheme, in this case, is designed so that every organization is assigned an overall quota (for instance 1 TB) that it can split among its users (for example 10 GB each). An administrator is then configured in the system to take the responsibility to manage user activation and personal quota assignment.

The integration process described is represented in Figure 3.

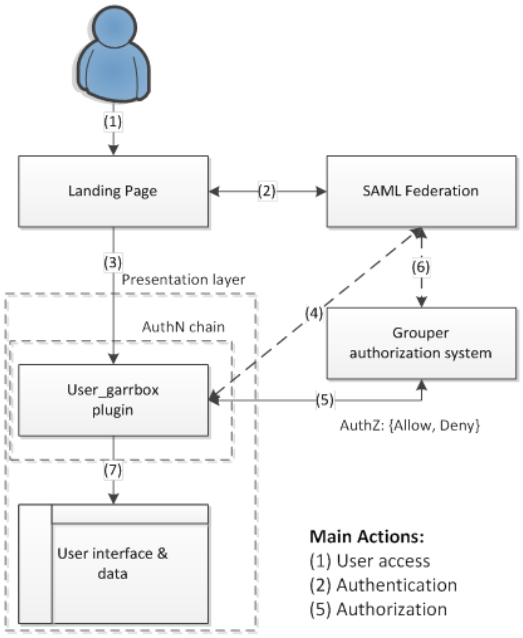

Figure 3: The key components in the integration between GARRbox and Grouper.

3.3 Advantages from the different perspectives

This paragraph will describe how a group management solution could be integrated into existing federations to bring an added value in the field of authorization. In this paragraph the main perspectives for

all the entities involved in a federation will be presented. The group management solution will be described to underline the benefits and interests to implement it from the SP, IdP and Federation point of view.

3.3.1 Service Provider point of view

Talking about authorization, the main interest from the SP point of view is to have a mechanism to delegate groups and attributes management in a controlled way. The SP, in the simplest situation, manages directly all fine grained aspects of authorization. In this case the SP has to know every user and has to maintain a user database with all the relevant information about users. The main problem, in this case, it that of keeping these information up to date during time. Since user management, in federations, is usually performed by the IdP, the SP may have the interest in delegating to IdPs the authentication attribute management.

In the scenario we propose in this article, the situation is yet slightly different. Instead of having the SP delegating authorization process to IdPs, in fact, we introduce a new component in the federation that is able to centralize group and attribute management and delegate authority for this aspects to the right entities and subjects. The SP, then, will delegate part of the authorization process to this external entity.

What we expect to realize is a delegation scheme in which:

- Authentication attribute representation is completely externalized out of the SP to the Grouper system. In this way Grouper will contain all relevant information about groups, authorization attributes and will permit to declare all authorization attributes for every user in a centralized place.
- The actual implementation of the authorization processes will happen inside the SP (or, better, within the application itself). The SP will then allow or deny specific grants to the accessing uses by retrieving relevant authorization information from Grouper.

Given this general scheme, the SP will have the need to retrieve all relevant authorization attributes by Grouper in order to implement the proper grant policies and to permit the user to perform only authorized operations.

3.3.2 Identity Provider point of view

The interest in authentication management from an Identity Provider point of view can be really different depending on the general scenario. Without the realization of a central system to manage authorization attributes, they could only be managed either by the SP or by the IdP. In the first case, the entity managing the IdP will not be completely independent in administering access rules for their users. In the other case, the entity managing the IdP will have full control of user definition and attributes, but will have to understand the internal logic of the SP to implement properly the authorization properties.

In the proposed scenario, using Grouper, it could be possible to implement a delegation mechanism that would permit to split the problem in half and thus simplifying the activities at the IdP side: the authorization attributes will be managed inside Grouper and thus in a single point; inside Grouper the proper delegation mechanism could be implemented to permit each entity administrator to deal only with the attributes of their interest. This will simplify management but at the same time will permit to maintain full control on user attributes.

3.3.3 Federation point of view

Reasoning at a federation level, the implementation of a centralized authorization management system means creating the prerequisites to permit a trusted delegation mechanism among different entities and institutions. The Grouper system described, in fact, permit an effective delegation of responsibilities in the field of authorization attributes management. For this to happen effectively the Federation needs to ensure that:

1. the delegation will happen in a clear and secure way so that responsibilities are very clearly defined and attributed;
2. the different subjects interacting in the authorization definition process must rely on a reciprocal trust, which is usually built at a federation level;
3. the technical representation and exploitation of authorization attributes must be coherent with the already defined authentication process (to simplify technical adoption of such a solution by all the participants to the federation).

The federation is particularly interested in guaranteeing that the authorization process is implemented in a solid and fully shared way. In particular a strong attention will be provided to guarantee that the technical implementation of authentication processes leverages the existing standards and protocols to ease its introduction in real scenarios (SAML and VOOT in our case).

4 CONCLUSIONS AND FUTURE WORKS

The approach described in this article, permitted to create a central process to manage user authorization, extending the functionalities of traditional Identity Federations. The general idea presented in the article is that of having autorization managed outside the SPs but in a component separated from IdPs and placed at the Federation level. In this way this component is able to implement the proper delegation mechanism that permits the right subjects to be involved in authorization management.

To prove the feasibility and the advantages of this approach we implemented three major use cases which cover the main applications of interest to our communities. The use cases studied permitted to study problems of different complexity in implementing a central manager for user authorization. In particular, for our experience, we considered to be very important the following points:

1. A central solution for authorization but integrate seamlessly into existing Identity Federations in order to provide user groups and additional attributes during the login process.
2. The release of attributes and groups during the login phase is not enought. Services in the Federation may have the need to interrogate the central repository to obtain informations (like participation to groups, for instance). So this central system must provide a common interface providing trusted information to the services requesting. To perform this task we prefer standard solutions implementing commonly shared protocols, like for example VOOT.

The approach described proved to be effective and paves the way to have it implemented as a real functionality into exisiting Identity Federations. This preliminar PoC permitted to identify the key problems and main aspects of realizing a central system for authorization. Future developments could be to move from the laboratory to real production environments in order to test on the field the robustness of the choices made.

ACKNOWLEDGEMENTS

This research has been supported by the European Commission, within the FP7 programme the GN3+ project supported these activities with the specific Joint Research Activity 3, Task 1.

REFERENCES

Cantor, S., Hirsch, F., Kemp, J., Philpott, R., and Maler, E. (2005a). Bindings for the oasis security assertion markup language (saml) v2.0. Technical report.

Cantor, S., Kemp, J., Philpott, R., and Maler, E. (2005b). Assertions and protocols for the oasis security assertion markup language (saml) v2.0. Technical report.

ECMA (2001). *ECMA-138: Security in Open Systems Data Elements and Service Definitions*. ECMA (European Association for Standardizing Information and Communication Systems), Geneva, Switzerland.

Gaedke, M., Meinecke, J., and Nussbaumer, M. (2005). A modeling approach to federated identity and access management. In *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web*, WWW '05, pages 1156–1157, New York, NY, USA. ACM.

Grouper (2014). http://www.internet2.edu/products-services/trust-identity-middleware/grouper/.

Kremers, M. (2012). Supporting virtual organisations using voot. In *Internet2 Member Meeting, Fall 2012*, Philadelphia, PA, USA. Internet2.

MediaWiki (2014). http://www.mediawiki.org/.

Moodle (2014). https://moodle.org/.

Morgan, R. L., Cantor, S., Carmody, S., Hoehn, W., and Klingenstein, K. (2004). Federated security: The shibboleth approach. *EDUCAUSE Quarterly*, 27(4):12–17.

Novakov, I. (2013). Standalone saml attribute authority with shibboleth. In *CESNET Technical Report 5/2013*, Prague, Czech Republic. CESNET.