PROCEEDINGS OF SCIENCE

Toward Single Sign-on Establishment for Inter-Cloud Environment

Eisaku Sakane[*]
National Institute of Informatics
E-mail: sakane@nii.ac.jp

Takeshi Nishimura
National Institute of Informatics
E-mail: takeshi@nii.ac.jp

Kento Aida
National Institute of Informatics
E-mail: aida@nii.ac.jp

Motonori Nakamura
Kyoto University
E-mail: nakamura.motonori.2c@kyoto-u.ac.jp

This paper investigates a mechanism that establishes single sign-on for inter-cloud computing environment built as the optimized result of the needs of users. Arranging requirements and issues for the mechanism, a single sign-on system for an inter-cloud computing environment is presented. As concrete service in the inter-cloud environment, we deal with Amazon Web Service with SAML version 2.0 and implement a prototype system. We also evaluate the prototype implementation and consider applicability to the other services.

International Symposium on Grids & Clouds 2019, ISGC2019
31st March - 5th April, 2019
Academia Sinica, Taipei, Taiwan

[*] Speaker.

1. Introduction

As diversification of cloud services is making progress, a service-oriented approach is more important to conduct efficiently researches with distributed infrastructures. In such situation services are chosen from multiple cloud vendors according to the demands of users and an inter-cloud computing environment will be formed naturally.

Single sign-on mechanism is indispensable for inter-cloud computing environment that is composed of various services – each service is provided by different cloud vendors. In general a single sign-on mechanism should work within a cloud environment, provided that a cloud vendor offers many services on the cloud. However, a single sign-on mechanism that extends across multiple clouds will not be established at the beginning of use. For example, suppose that an academic researcher utilizes an inter-cloud computing environment. If the researcher can obtain an electronic credential from the home organization that the researcher belongs to, it is convenient for the researcher to access each service in the inter-cloud environment simply with the credential.

Since many of cloud vendors, of course, already support major authentication technologies such as SAML (Security Assertion Markup Language) [1] and OAuth [2], technically the credentials issued by the home organization will be usable for access to public clouds, for instance, Amazon Web Service. However, it is often hard for the identity provider (IdP) operated by the home organization to manage the attributes that the cloud vendor requires specifically, because the operating department of the IdP assumes the responsibility to assign only natural attributes that ensure a researcher is a constitute member of the organization. It will be quite a burden to manage various attributes for users individually unless the cloud service is provided for all members as common service. Based on the credential issued by the home organization, a mechanism for handling necessary attributes information is needed, which should not impose a burden on administrators of the IdP.

This paper designs a single sign-on mechanism for the inter-cloud environment without a burden on IdP management and provides the implementation of the mechanism. The single sign-on mechanism introduces a gateway service. The gateway service generates credential that contains the required information or attributes by the service based on user's credential and sends the service provider (SP) the credential instead of the IdP. The proposed system basically maintains the conventional IdP management by the home organization and enables users to efficiently access services with single sign-on according to individual service usage form.

The remainder of this paper is organized as follows. We describe the requirements and issues for a single sign-on system for an inter-cloud computing environment in Section 2. In Section 3 we present the design and implementation of the single sign-on mechanism. Section 4 makes discussion about the proposed system. Section 5 refers to related work. Finally, Section 6 concludes the paper.

2. Requirements and Issues

This section organizes the requirements and issues for a single sign-on system for an inter-cloud computing environment.

Let us consider an academic researcher that utilizes various services provided by different cloud vendors. The researcher uses each account identity according to the service because each cloud vendor, in general, has different account system. In academia, the organization that the researcher belongs to plays a role as IdP. If the IdP operated by the researcher's affiliation (home organization) is able to send sufficient information so that all of service providers can authorize a requestor to the service, a single sign-on system will be realized in this sense and the researcher will efficiently utilize the service. However, that is not always the case.

We observe two problems in such situation. One of problems is that the service is not *common* but *personal*. Here, we define *common* service as a service that is able to used with credential with fundamental attributes issued by the home organization, otherwise, *personal* service [3]. In this sense, it is actually hard for the IdP to manage necessary attributes that the *personal* service requires per individual researchers, because there are various researchers in university and various services that researchers demand. The other problem is, of course, that attributes the IdP should include in the information are different among each SP. SPs often require that the IdP should include SP-specific attributes in the information. For example, certain SP-specific Name attribute is obviously not usable to the other services. If provided services increase, it will be tough for IdP to manage attributes per service by itself.

Based on the problems described above, we arrange the requirements and issues that a single sign-on system should fulfill.

1. Issuance of credential with necessary attributes for *personal* services.
   The system should be able to add necessary attributes required by the service to authentication information issued by the IdP that user's home organization operates.

2. Handling attributes per service.
   The system should efficiently deal with attribute management per service.

In addition, the system should not impose more burden on IdP operated by home organization as possible.

3. Design and Implementation

In this section we describe the design of the single sign-on system for various services provided by different SPs. After that, we introduce GakuNin Cloud Gateway Service that an implementation of the design adopts, and illustrate the prototype implementation for Amazon Web Service supporting SAML version 2.0 as a concrete service.

3.1 Design

To resolve the issues mentioned in the previous section, the basic idea is for IdPs to delegate the responsibility for suitable attribute assignment to a trusted third party. The trusted third party assumes a role of the sender that sends necessary attributes for the service combined with fundamental authentication information. Concretely, we consider a gateway service that the trusted third party manages, and require that each of the following features should be provided by the gateway service (Figure 1):

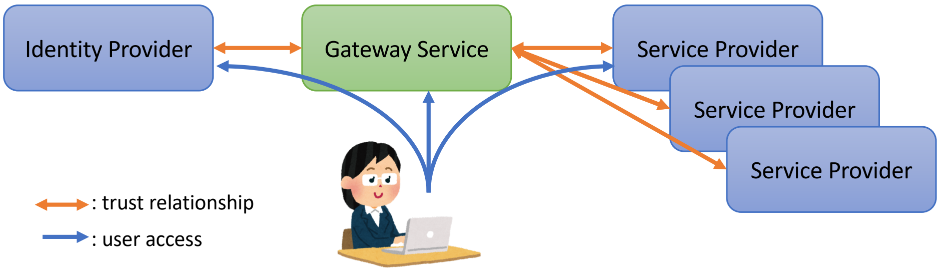


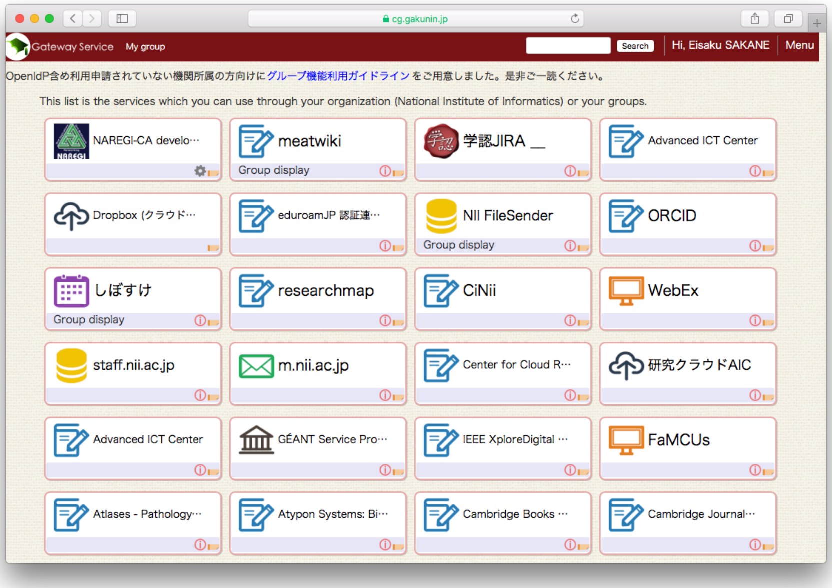
Figure 1: Gateway service approach.

1. Authenticate users with credentials issued by the home organization,
2. Provide a function that allows SP to control attributes (name and type) required by itself,
3. Provide a function that allows users or representatives to set the values of the required attributes,
4. Make a credential that includes the required attributes and the fundamental attributes managed by the home IdP, and send it to the SP, and
5. Provide users with a user-friendly interface for access to available services.

The gateway service mediates between an IdP and SPs, and forms a trust relationship with each provider. Thus, a trust relationship will be indirectly built between the IdP and SPs. Hereafter, we suppose that the IdPs, SPs and gateway service concerned join a trust federation, for the sake of simplicity. However, this is in a natural situation in view of the current state that there are several academic trust federations. Moreover, this paper does not refer to authorization process in detail and supposes that users are able to obtain suitable permissions for the services according to an "out-of-band" authorization process.

In the gateway service approach, access flow to a service provided by an SP (cloud vendor) is formed as follows. Firstly, the gateway service authenticates users, actually redirects the user to the IdP operated by the user's affiliation and obtains a valid credential from the IdP. Namely, the gateway service acts as an SP. Next, based on the credential the gateway service creates an assertion including the necessary attributes for the service that the user requested, where we have used the word "assertion" to distinguish with the credential issued by the IdP. Finally, the gateway service sends the assertion to the SP and enables the users to access the service. For all services that the gateway service supports, the required attributes per service should be managed correctly. Each of IdPs that users belong to will be able to entrust the gateway service with the attribute management, because the management of the required attributes does not depend on IdP. In other words, the gateway service is able to apply common attribute management system to any IdP. Just the attribute value will depend on IdP.

From an end-entity user's point of view, by accessing the gateway service as the starting point, users are able to utilize various services with single sign-on based on the credential issued by the home IdP.



OpenIdP含め利用申請されていない機関所属の方向けにグループ機能利用ガイドラインをご用意しました。是非ご一読ください。

This list is the services which you can use through your organization (National Institute of Informatics) or your groups.

Figure 2: GakuNin Cloud Gateway Service. Each icon represents an available service.

3.2 GakuNin Cloud Gateway Service

GakuNin is a full-scale development of the academic access management federation in Japan. GakuNin Cloud Gateway Service (CGS) [4, 5] is a web service portal, which has been developed and operated by National Institute of Informatics in Japan, and enables users to manage services available in academic research or education. Hereafter, the GakuNin CGS is called the CGS merely unless otherwise noted. In cooperation with IdPs and SPs that participate in the GakuNin (federation), the CGS checks service providers that the identity provider allows of sending SAML assertions to based on user's SAML assertion made by the identity provider, and displays users the list of available services (Figure 2). Each of services in the CGS is expressed as icon and can be controlled by not only organization such as university and institute but also group such as research project team. Therefore, users are able to easily make a virtual research organization on the CGS.

Typical access flow to services with the CGS is as follows. First of all, user accesses to the CGS. The internal SP, a component of the CGS, responds to the user and the embedded discovery service (DS), also a component of the CGS, shows the list of federated IdPs. After that, the embedded DS redirects the user to the home IdP selected by the user. After authentication by the IdP succeeded, the CGS displays the available services as icon. By clicking the icon of the service the CGS redirects the user to the SP that provides the selected service. The SP interprets the entityID of the IdP and redirects again to the IdP. The SP receives the assertion sent by the IdP with single sign-on. Finally, the user is able to utilize the service.

3.3 Prototype Implementation

The CGS thus far satisfies the requirements mentioned in Section 3.1 except the fourth requirement. Thus, we decide to add the required functions to the CGS. Furthermore, we focus on Amazon

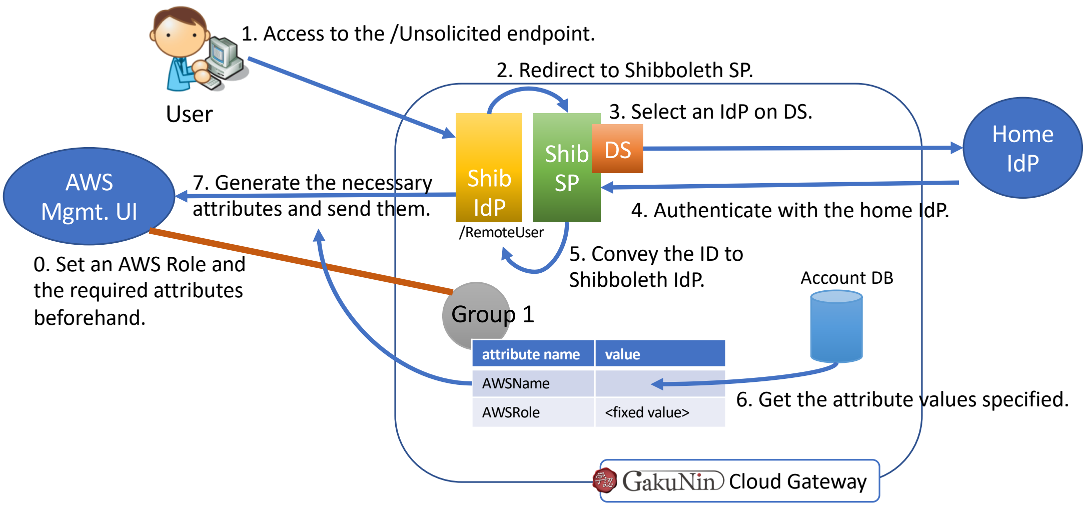

1. Access to the /Unsolicited endpoint.
User
2. Redirect to Shibboleth SP.
3. Select an IdP on DS.
Shib IdP
Shib SP
DS
Home IdP
AWS Mgmt. UI
7. Generate the necessary attributes and send them.
/RemoteUser
4. Authenticate with the home IdP.
5. Convey the ID to Shibboleth IdP.
0. Set an AWS Role and the required attributes beforehand.
Group 1
Account DB
attribute name
value
AWSName
AWSRole
<fixed value>
6. Get the attribute values specified.
GakuNin Cloud Gateway

Figure 3: Overview of the GakuNin Cloud Gateway Service supporting Amazon Web Service with SAML version 2.0 federation. Access flow to AWS management console is illustrated.

Web Service (AWS), for the sake of simplicity. Indeed, AWS can be considered to be a concrete example, because the AWS requires that SAML assertion contains specific attribute elements with the Name attribute set to, e.g., "https://aws.amazon.com/SAML/Attributes/Role". These attributes are obviously different with commonly used SAML attributes such as the eduPerson and eduOrg attributes [6]. Of course, the AWS enables users to treat the commonly used attributes in the GakuNin for authorization, however, they are optional.

Figure 3 illustrates the overview of the system that fulfills all the requirements. Main parts in Fig. 3 are the CGS, the home IdP that the user's affiliation operates, and the AWS. The CGS involves internally the Shibboleth IdP and SP, the DS, and an account database (DB). There are also characters in the proposed system, namely, end-user, group administrator, AWS account administrator, and service connector administrator. It is supposed that the user in Fig. 3 belongs to a group, called "Group 1", formed on the CGS. Group administrator manages a group formed on the CGS. AWS account administrator is a person that able to create AWS Role that allows SAML-federated access to her/his AWS resources. It will be the possible case that the group administrator and the AWS account administrator are the same person. In contrast to the internal SP, actual SP is able to require and administrate the attributes required by itself as service connector administrator in the CGS.

To establish single sign-on to the AWS, a trust relationship between the AWS and the group on the CGS must be built in advance (Step 0 in Fig. 3). Concretely, according to the AWS Identity and Access Management specification, AWS Roles and ID providers for SAML federation should be created. Building the trust relationship is conducted by importing each metadata with in the ordinary Shibboleth manner. On the other hand, the group administrator must set the indispensable attributes such as AWSRole element with the Name attribute, which is based on the information about the AWS Role already created on the AWS. This AWSRole is common for members of the Group 1 and can be set as fixed value.

After the beforehand procedure as mentioned above, single sign-on to the AWS resource via the AWS management console is realized as follows. Firstly, the user accesses to the Shibboleth Unsolicited endpoint shown as an icon on the CGS (Step 1). Next, the internal IdP redirect the user to the internal SP and the DS shows the list of the federated IdP, after that, the user selects the her/his home IdP that will authenticate the user (Steps 2 and 3). If the authentication succeed, the internal SP will receive the authentication information from the home IdP and convey that to the internal IdP (Steps 4 and 5). Based on the authentication information, the values to be set to group member specific elements such as the AWSName with the Name attribute are obtained from the account DB (Step 6), and the internal IdP generates the assertion containing the necessary attributes and sends it to the AWS endpoint (Step 7). Finally, the user obtains a security credential from the AWS and will be able to access the AWS resource with the credential.

The new function that generates the required assertion based on the elementary assertion issued by the home IdP satisfies the fourth requirement described in Sec. 3.1. In this prototype implementation, we are able to manage three kinds of attributes: the service specific ones, the group specific, and the user specific. The service specific attributes can be used to distinguish the multiple services that the CGS supports, are common for all groups on the CGS, and will be fixed values. For the AWS, the value of the Audience element does not contain group or user specific information but does the AWS one only. The group specific attributes enables groups in the CGS to set each AWS Role. Indeed, the value of AWSRole is composed of the AWS Role name, the AWS account number, and the SAML ID provider name. Finally, the user specific values are, for example, the AWS Name, of which value that provides an identifier for the AWS temporary credentials for single sign-on and is used to display user information in the AWS management console [6].

4. Discussion

In this section we evaluate the prototype implementation of the single sign-on mechanism for inter-cloud environment.

Basically the prototype implementation is able to send the required attributes to the AWS without any changes of configuration of the home organization IdP. Although the burden of attribute management moves from the home IdP to the CGS, the burden of the management in the whole system that covers the federation will decrease because the CGS need not do anything in particular to apply the same function to the other IdP. In addition, even if the academic organization decide to provide AWS resources to all constitute members, namely, offer as common service, the IdP operated by the organization should be able to utilize the attribute management function of the CGS without modifying the IdP.

On attribute management by SP connector administrator, there is a specification limitation in the prototype implementation. Suppose concretely that a group, called Group 1, should set an attribute, e.g., eduPersonPrincipalName, and the other group, called Group 2, should set the other attribute, eduPersonOrgDN. In this case, the SP connector administrator should setup the required attributes on the CGS based on the union set of the attributes of Group 1 and 2. In other words, the SP connector administrator must always have been grasping all attributes that all groups require.

On security consideration, we discuss the possibility of unauthorized use via the CGS. In order to utilize SAML federation according to the User Guide [6], the following steps must be conducted: exchanging IdP's and AWS' metadata to establish a trust relationship between them, and specifying AWS Role and SAML ID provider. The AWS Role and SAML ID provider can be created for each AWS account. In our prototype implementation, the AWS will enter into the trust relationship just with the single internal IdP of the CGS for any groups on the CGS. The CGS puts group information into the eduPersonEntitlement attribute. Therefore, if the AWS Role correctly verifies the group information in the eduPersonEntitlement, suitable authorized use per each group can be realized in the trust relationship between the CGS and the AWS.

Although our current implementation supports only the AWS, we are able to consider the proposed system based on the CGS to be extended to the other services, because we adopt SAML technology as standard and the generation mechanism of the required assertion does not depend on the service. Of course, the proposed system should manage the set of attribute names and their value used per service. Furthermore, the performance of the extended CGS system should be evaluated according to the number of services supported. Currently we do not have the result of performance measurement. However, we do not believe that the performance will be degraded for the realistic number of services concerned.

5. Related Work

This section refers to AARC Blueprint Architecture by Authentication and Authorization for Research Collaboration (AARC) and proxy services: Check-In developed by EGI [8] and INDIGO IAM developed by INDIGO-DataCloud [9].

5.1 AARC Blueprint Architecture

The AARC Blueprint Architecture (BPA) [7] (AARC-BPA) is a set of software building blocks that can be utilized to implement federated access management solutions for international research collaborations.

The GakuNin Cloud Gateway Service adopted to implement the prototype should be expressed in terms of layers defined by the AARC-BPA because the AARC-BPA is a general-purpose architecture and is useful to review the GakuNin CGS for function extension and so on. The detail architecture of the GakuNin CGS and the correspondence with the AARC-BPA will be published in the near future.

5.2 Proxy Services

EGI Check-in is a proxy service that operates as a central hub to connect federated identity providers (IdPs) with EGI service providers. Check-in allows users to select their preferred IdP so that they can access and use EGI services in a uniform and easy way [10].

INDIGO IAM is an identity access management system that can manage identities, enrollment, group membership, attributes and policies to access distributed resources and services [11].

The goal of the proxy services referred above is almost the same as the proposed system in this paper. Main difference between those proxy services and the GakuNin CGS is as follows: in those proxy services the other authentication technologies than SAML is taken into consideration in the current stage, while the GakuNin only SAML. Although it is sufficient for the academia in Japan and we currently do not have any plan to be usable the other authentication technologies, coexistence between multiple authentication technologies should be considered in the future.

6. Summary

In this paper, we presented a gateway service approach toward single sign-on establishment for inter-cloud computing environment.

Our contribution in this paper is as follows:

- We designed a system that realizes single sign-on for inter-cloud computing environment.
- We made a prototype implementation of the system by extending the GakuNin Cloud Gateway Service developed by National Institute of Informatics in Japan, in which we focused on single sign-on access to Amazon Web Service with SAML assertion as the first step.

In the current stage of research, we support only the Amazon Web Service. However, the proposed single sign-on mechanism will be applicable to the other services as long as SAML assertion is used to establish single sign-on.

We will support more services offered by various cloud vendors in SAML federation, e.g., Dropbox [12] as the next step. Also we plan to start operation officially in the GakuNin.

Acknowledgments

This work is partially supported by the Ministry of Education, Sports, Science and Technology in Japan.

References

[1] S. Cantor, J. Kemp, R. Philpott, E. Maler, *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*, (2005) https://www.oasis-open.org/standards#samlv2.0

[2] D. Hardt, Ed., *The OAuth 2.0 Authorization Framework*, RFC 6749 (2012) https://www.rfc-editor.org/info/rfc6749

[3] E. Sakane, T. Nishimura, K. Aida and M. Nakamura, *A Study of Credential Integration Model in Academic Research Federation Supporting a Wide Variety of Services*, in proceedings of *International Symposium on Grids and Clouds (ISGC) 2018 in conjunction with Frontiers in Computational Drug Discovery*, PoS(ISGC 2018 & FCDD)016, (2018).

[4] T. Nishimura, M. Nakamura, K. Yamaji, H. Sato and Y. Okabe, *Privacy Preserving Attribute Aggregation Method without Shared Identifier Binding*, *Journal of Information Processing* **22** (2014) 472.

[5] T. Nishimura, E. Sakane, K. Yamaji, M. Nakamura, K. Aida and N. Klingenstein, *Virtual Organization Platform Interoperability Provides the Long Tail an eScience Environment*, *Journal of Information Processing* **24** (2016) 609.

[6] Amazon Web Services, Inc., *AWS Identity and Access Management: User Guide*, (2019) https://docs.aws.amazon.com/IAM/latest/UserGuide/index.html

[7] The AARC project, *AARC Blueprint Architecture*, (2017) https://aarc-project.eu/wp-content/uploads/2017/04/AARC-BPA-2017.pdf

[8] EGI: advanced computing for research, https://www.egi.eu

[9] INDIGO DataCloud project, https://www.indigo-datacloud.eu/

[10] EGI Check-in, https://www.egi.eu/services/check-in/

[11] INDIGO-IAM, https://www.indigo-datacloud.eu/identity-and-access-management

[12] Dropbox, Inc., *eduGAIN, InCommon,and configuring Dropbox SSO*, https://help.dropbox.com/en-us/business/incommon-sso