Federated Identification Architecture

Arezoo Haghshenas
Department of Computer Tehran South Branch, Islamic Azad University
Tehran, Iran

Mir Ali Seyyedi
Department of Computer Tehran South Branch, Islamic Azad University
Tehran, Iran

ABSTRACT
Service Oriented Architectures are an abstract concept which exposes capabilities in distributed, domain-spanning environments as services. These modern systems have three characteristics: They are heterogeneous, distributed and loose Coupling. With increasing popularity of Service Oriented Architecture (SOA), this is no longer possible since interacting systems are generally not located within a single security domain anymore. Using SOA without extra attention to security issues leads to various problems. Federated Identification is one of the most important security issues in collaborative systems which are not in the same security domain. To deal with this security issue, several Federated Identity Architecture initiatives have appeared recently. Federated identity architecture lets users dynamically distribute identity information across security domains, increasing the portability of their digital identities. All Federated Identity Architectures keep user's distributed mapping and/or centralized mapping of user's identifiers for federated identity. Saving the mappings for each user and updating them when changes happen will bring a Heavy Overload for the architecture. In this paper Federated Identification Architecture is presented which provides a Centralized Identity Provider (CIP). The architecture presented is highly beneficial in SOA and distributed environments. All security domains can integrate with this architecture using very few adjustments. Advantage of using CIP model is that users can accesses a service by using any of the identifiers which they prefer. The chosen identifier is not always the identifier recognized by the requested service.

General Terms
Security

Keywords
Identification, SOA, Federated Identity

1. INTRODUCTION
Service Oriented Architectures are an abstract concept which exposes capabilities in distributed, domain-spanning environments as services [13]. These modern systems have three characteristics: They are heterogeneous, distributed and loose Coupling.

With increasing popularity of Service Oriented Architecture (SOA), this is no longer possible since interacting systems are generally not located within a single security domain anymore. Companies externalize their system into the "cloud" or to the third party provider offering that particular functionality as a service [12]. A system which is connected to other systems is open to security problems. Using SOA without extra attention to security issues leads to various problems. One of the security aspects is Identification. Federated Identification is one of the most important security issues in collaborative systems which are not in the same security domain. A security domain is a set of users and systems that are controlled by the same security policy. Federated Identification noted ever since Service Oriented Architecture became significant.

Identification is the use of an identifier that allows a system to recognize a particular subject and distinguish it from other users of the system [4].

To interpret the concept of Identification this paper will use the following definitions regarding to identity and identifiers:

Digital Identity – The electronic representation of an entity within a domain of application.

Entity – A person, a group of persons, an organization, a process or even a device, that is, any subject able to make transaction.

Domain of Application – The application scope where the digital identity has validity, for example: a company, a hospital, a club, a university or the internet. Not that an entity may have several identities within the same domain of application. For instance, a professor could have identities of both professor and student in case he takes continuous education classes.

Identifiers – A digital identity is composed of identifiers or attributes, which can be assigned, selected or they can be implicit to the user. Examples of attributes are: date of birth, address, employee ID, Social Security Number, among others [3].

Each Entity has several Identities which are located in various security domains. As previously mentioned, each digital identity has a unique attribute which distinguish user in a particular subject. This attribute is Identifier. The excessive number of identities applied to each user leads to some problems.

Under this context, users feel uncomfortable handling several digital identities, one for each service. From the point of view of the service, the identity management process represents a very high administrative load in financial and operative terms. To deal with this problem, several Federated Identity Architecture initiatives have appeared recently [6].

Traditionally, the maximum scope of a user identity has been only one organization. The identity has not been shared with other organizations. If the user has used services outside her home organization (for example, her employer or school), she has had separate usernames and passwords for each service. However, as the networking of organizations has become more common, it has become a subject of interest to share (*i.e.*, federate) user identities between organizations. In a federation, there is a specific middleware service that federates her attributes from the home organization (called Identity Provider) to the service she is using (called Service

Provider) [2]. A federation is an association of organizations that come together to exchange information, as appropriate, about their users and resources in order to enable collaborations and transactions [5].

Federated identity architecture creates globally interoperable identities. It is based on the business agreements, and policy agreements that allow organizations to interoperate based on shared identity management. The goal of identity federation is to enable users of one domain to access resources of another domain seamlessly [11]. Federated identity architecture lets users dynamically distribute identity information across security domains, increasing the portability of their digital identities [1]. Liberty Alliance, Shibboleth and Web-Federation are Federated identity architectures.

Liberty Alliance is a group of more than 200 companies from divers sectors. It was launched in 2001 with the objective to establish a technological, business and policy framework for implementing a federated Identity Architecture [7, 10].

Shibboleth is an academic initiative of University members of internet2. Its objective is to facilitate the collaboration and access to protected resources among institutions without using external or temporary accounts. Some applications that could take advantage of this solution are access to library database information, distance learning courses, collaborative application for project development, etc. [8]. Web-Federation is an important component within the secure framework architecture for web services. As we know, Web service is a mechanism that supports communication between web applications located in different organization, and allowing the integration of application in heterogeneous environment. Web services bases its operation on the Service Oriented Architecture. Under this context, in 2002, IBM and Microsoft together with other companies defined a reference model to provide security to Web Services from a technological point of view as well as business activity policy [9].

All Federated Identity Architectures keep user's distributed mapping and/or centralized mapping of user's identifiers for federated Identity. Saving the mappings for each user and updating them when changes happen will bring a Heavy Overload for the architecture. In this paper federated identification architecture is presented which provides a *Centralized Identity Provider (CIP).* Each domain, which is a member of CIP, manages its local user's management system. There is an identity provider service available above each local user's management system which provides a standard media to access user's authentication information and other attributes. The main advantage of using CIP model is that users can accesses a service by using any of the identifiers which they prefer. The chosen identifier is not always the identifier recognized by the system.

2. FEDERATED IDENTIFICATION ARCHITECTURE

First, we describe Main components of Federated Identification Architecture:

- **Identity Provider (IdP):** User identity is saved and managed using this component. Users are identified in Identity Provider's security domain.
- **Service Provider (SP):** User's desired services are provided by this component. Security domain that contains Service Provider is not always identical to Identity provider's security domain.
- **Centralized Identity Provider (CIP):** A trusted third party which is trusted by all Identity Providers and Service Providers. They are both connected via CIP.

2.1 Identity Provider

An Identity Provider keeps credentials and attributes of the users (such as identifiers). The moment a request arrives; IdP sends user's identification asserts or attribute asserts to the requester. IdP components are presented in the figure 1.

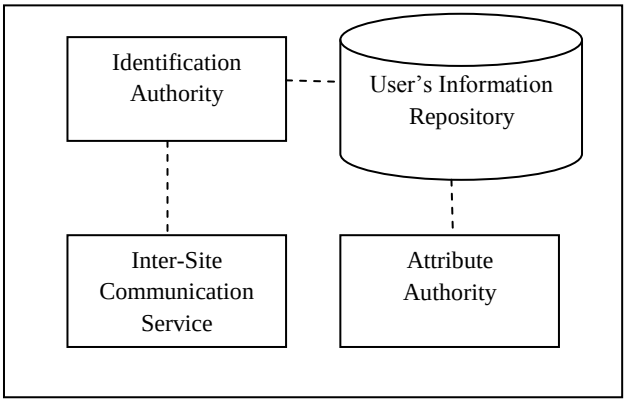

Fig 1: Identity Provider

- **Identification Authority:** Identification Authority distributes identification asserts to other components.
- **Inter-Site Communication Service:** This service cooperates with Identification Authority to generate identification asserts. This service is the first connection point to identity provider. This service starts identification process within the identity provider.
- **Attribute Authority:** Attribute Authority processes attribute requests. AA distributes attribute asserts. It Authenticates and Authorizes each request.

2.2 Service Provider

Service Provider manages secured resources. User's access to the services depends on the statements which service provider receives.

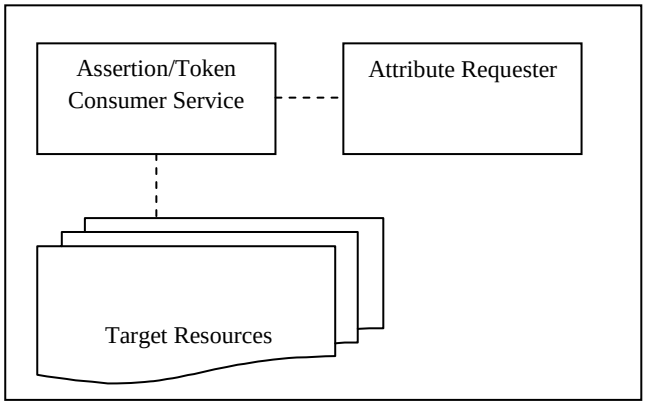

Fig 2: Service Provider

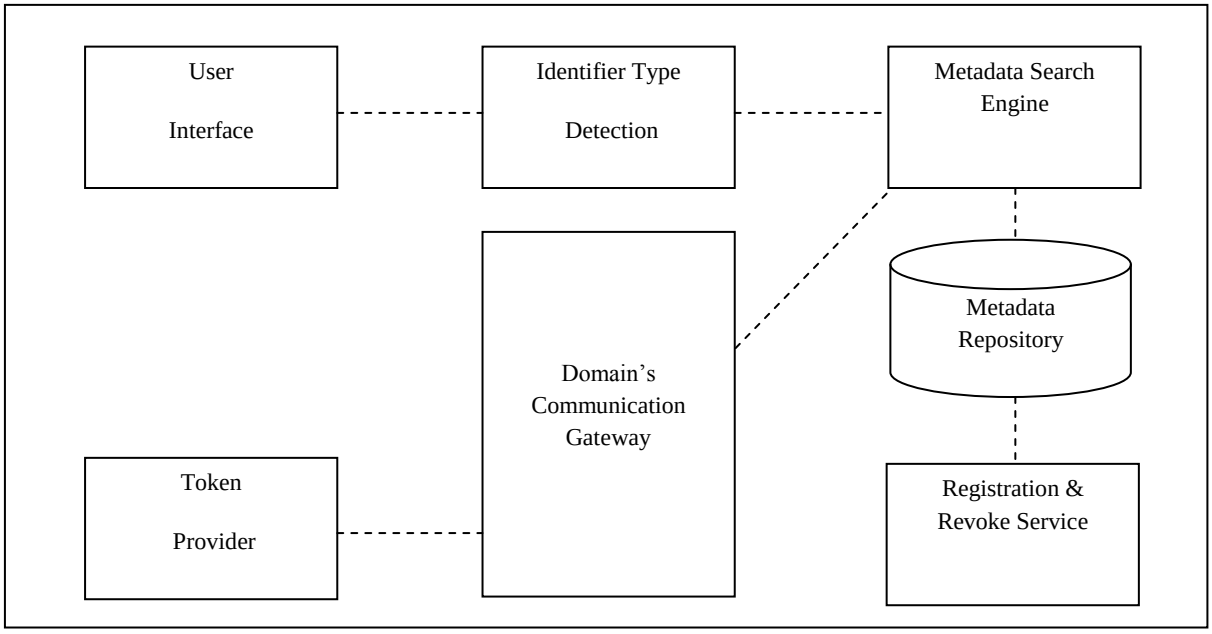

Fig 3: Centralized Identity Provider

- **Target Resources:** Resources which the users want to use them.
- **Assertion/Token Consumer Service:** This service processes identification asserts sent by Domain's Communication Gateway. Also it initiates authorized attribute requests. Also produces security content in service provider and guides users to the requested resource.
- **Attribute Requestor:** Attribute Requestor, in service provider and Attribute Authority in identity provider, might transfer attribute using back channels. This way Service Provider and Identity Provider work directly with each other.

2.3 Centralized Identity Provider

Centralized Identity Provider is a trusted party. This component saves security domain's metadata's element. All Identity Providers and Service Providers are connected via CIP.

- **User Interface:** This component is in charge of communicating with user. Using this component, users send their Identifiers and Credentials. When a user wants to access a secure service (Target Resource) in a Service Provider within the CIP, User Interface is the component which receives the data.
- **Identifier Type Detection:** Identifier Type Detection compares identifier type that the user enters, with its previously identified types. If the format of identifier used by the user doesn't match with the previously identified formats, an error message will be sent to user.
- **Metadata Search Engine:** This module is in charge of searching for the domain metadata that contains user's identity. Searching process will happen within the Metadata Repository. After recognizing the type of user's identifier, the domain metadata, which contains this identifier type, will be searched within the metadata repository.
- **Metadata Repository:** This repository is in charge of saving and updating metadata from various domains. This metadata can contain identifier types, existence or nonexistence of identifiers credibility, etc.
- **Domain's Communication Gateway:** This module is responsible for communicating with Identity Provider Domain and/or Service provider. Every request from IdP or SP and every response from them will take place through this module. This module is the only module in this component that can be communicate with different security domains.
- **Registration and Revoke Service:** This module is the module which registers new security domains within CIP. Deleting a security domain is this modules' responsibility as well.
- **Token Provider Module:** After a domain identified a user and sent its confirmation to the centralized identity provider, Token Provider will be in charge of creating a token for Service Provider. Token Provider is the module which creates the token.

3. IDENTIFICATION MODEL

For Non- identified access to secured resources Identification Model is shown in the figure 4:

1. First user sends a request for a service.
2. Then the user being redirected to CIP for request control.
3. After that, The CIP sends Login Form (User interface) to the user.
4. User enters his chosen Identifier.
5. Using entered Id; CIP identifies the IdP which user needs to transfer to for Identification.
6. CIP sends the identifier to the IdP recognized in the Last step for Identification.

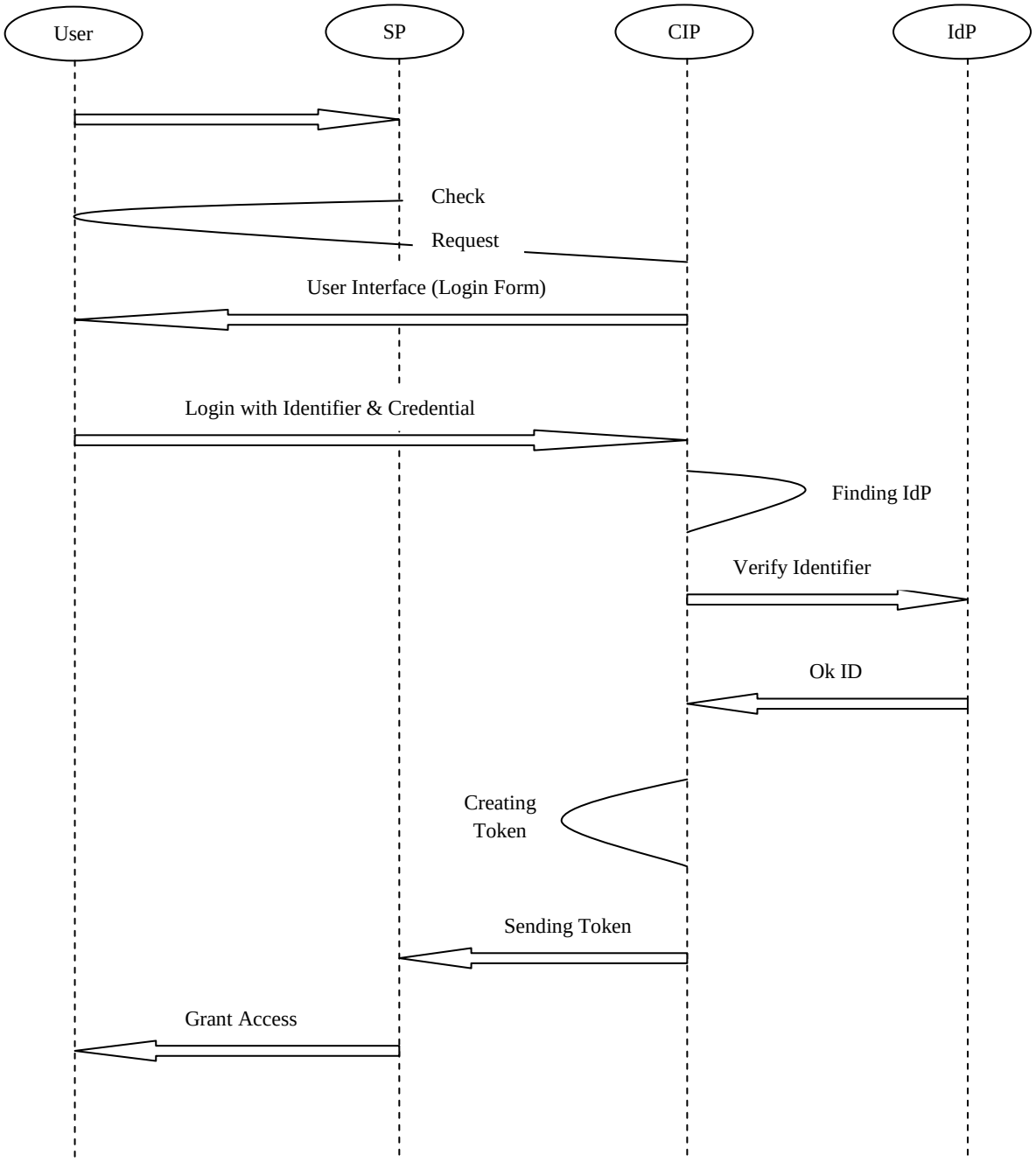

Fig 4: Identification Model

7. Depend on the Identification received from the last step, IdP sends OK or Error message.
8. If the message sent in the previous step was 'OK' then CIP generates a token for the requested service.
9. Generated token in step 8 is sent to Service Provider.
10. Finally, Requested Service will be sent to the user.

4. IMPLIMENTING THE ARCHITECTURE

This Architecture implemented using ASP.NET and C# programming language. Traffic Department, Registration administration and Insurance Departments chose as security domains. Every security domain saved in Metadata Repository in XML format. One of the most important elements of these Metadata is the type of Identifier used for Identification. For instance, Identity provider in Registration Administration uses National Number Indicator as an Identifier. In this model a form designed to save new domains' XML files and/or to remove them. This form is equivalent to Registration and Revoke Service. Also, in this model the User Interface module implemented in the format of a form. After a service request arrives this form will be redirecting to the user. Any type of identifier user enters will be process in a routine which identifies the entered identifiers' type. The Routine responsible for this process is Identifiers Type Detection. Afterwards, another routine which is known as the Metadata Search Engine finds the XML that covers this

type of identifier; the search engine sends a function to the domain which covers the XML for identifying the user. In this step Domain's Communication Gateway communicates with Inter-Site Communication Service. Inter-Site Communication Service is a domain service which its location is in the domains' metadata. Domain's Communication Gateway evocates this service. If Inter-Site Communication Service sends 'valid' message, a token will be generated for the requested service. This token will be sent to the service.

5. CONCLUSION

The Architecture presented is highly beneficial in distributed environments. All security domains can integrate with this architecture using very few adjustments. One advantage of this architecture is that users can accesses a service by using any of the identifiers which they prefer. The chosen identifier is not always the identifier recognized by the requested service.

Another Advantage of this architecture is moderating data overflow and redundancy information. The reason to that is because; Users' data won't be needed to save in different security domains for the purpose of identity federation. Reducing data overflow and redundancy information will make data management easier and more convenient. Another advantage of this model is its flexibility. When a new security domain registers in CIP all user identities available in this domain will be integrating with every other domain available, simultaneously. Table 1 shows Liberty Alliance, WS-Federation and Centralized Identity Provider Model comparisons.

Table 1. Liberty Alliance, WS-Federation and CIP Model comparisons

Feature / Functionality	Liberty Alliance	WS-Federation	CIP Model
Account Federation	Account federation via Identity Mapping enabled by opaque identifiers (a key privacy feature)	Account federation via Identity Mapping enabled by the Pseudonym Service	Instead of saving the mappings for each account, CIP just save metadata of security domains
Data Redundancy and Memory Consumption	High because of saving identity federation information in different security domains	High because of saving identity federation information in different security domains	Low because of just saving domains metadata in CIP
Change Management Cost	High because Account Updates should apply in all domains	High because Account Updates should apply in all domains	Low because Account Updates just apply in domain of account

6. REFERENCES

[1] E. Maler, D. ond. "Options and Issues in Federated Identity Management". IEEE Security & Privacy. 2008

[2] M. Linden. "Organising Federated Identity in Finnish Higher Education". Computational Methods in Science and Technology, Volume 11, Issue 2, 2005. 109–118

[3] A. Jøsang**,** S. Pope**. "**User Centric Identity Management". AusCERT Conference**.** 2005

[4] S. Kamburugamuwa, K. Indrasiri, P. Perera, M. Pathirage. "Federated Identity Framework for Web Services". Department of computer science & engineering, university of Moratuwa,SRI LANKA.final year project.

[5] InCommon Federation. InCommon glossary http://ww.incommonfederation.org/glossary.cfm Referenced 2.5.2005.

[6] U. Fragoso-Rodriguez, M. Laurent-Maknavicius, J. Incera-Dieguez. "Federated Identity Architectures". 2005

[7] T. Wason. "Introduction to Liberty Alliance Identity Architecture", URL: http://www.projectliberty.org. Revision 1.0. Liberty Alliance Project, 2003

[8] T. Scavo, S. Cantor."Shibboleth architecture, Technical Overview".URL:http://shibboleth.internet2.edu/shibboleth-documents.html. Working Draft 02.June. 2005

[9] Security Roadmap. "Security in a Web Service World: A Proposed Architecture And RoadMap ". URL://www.128.ibm.com/developerworks/webservices/library/specification/ws-secmap/. IBM and Microsoft white paper. April 7,2002

[10] T. Jonathan, K. Yuzo. "Liberty ID-WSF Web Services Framework Overview". URL: http://www.projectliberty.org. Version 1.0.Liberty Alliance project. 2004

[11] M. H.kang, A. Khashnobish. "A Peer-to-Peer Federated Authentication System". Sixth International Conference on Information Technology: New Generations, IEEE Computer Society, 2009

[12] M. Wolf, I. Thomas, M. Menzel,C. Meinel. "A Message Meta Model for Federated Authentication in Service Oriented Architectures". IEEE. 2009

[13] M. MacKenzie, K. Laskey, F. McCabe, P. Brown, R. Metz. "Reference Model for Service Oriented Architecture 1.0." OASIS Committee Specification, February 2006