FEDERATED IDENTITY MANAGEMENT IN CROSS-CLOUD ENVIRONMENT

Monika K. Katwe[1], Mr. Manish M.Potey[2]
[1]M.E pursuing (Computer Engineering), [2]Associate Prof. Department of Computer Engineering
[1,2]K.J Somaiya College of Engineering, Mumbai.
Email:[1]monika.katwe@somaiya.edu, [2] manishpotey@somaiya.edu

Abstract- Identity management manages the identity of user across distributed network. This paper designed identity management in cross cloud environment to achieve single sign on for cloud users. The proposed system allows user to use unique set of credentials, even a user managed with different kinds of cloud environments. The given system offers a solution of open identification, authorization and federated identity management in which there is a trust party auditor which maintains all the credentials of users associated with same as well as different cloud and cloud provider can uniquely distinguish one user from other. This system performs two level authentications for cloud users and also maintain secure channel for user in order to ensure the confidentiality.

Index Terms - Single sign on, Authorization, Cloud computing, IID, Electronic identification, Federated identity management, Identity federation.

I. INTRODUCTION

Cloud computing is one of the fastest growing IT area which provides high scalability and elasticity and also offers pay for use options for cloud users so that, user can pay based on their usage. This makes cloud computing attractive in various areas like Government or Healthcare etc. The National Institute for Standards and Technology (NIST) classifies cloud computing into three different service delivery models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [1]. IaaS provides virtual computing resources such as virtual machines, networks and data storage. PaaS offers programming environments where customer can develop run and manage web application. In the third service model (SaaS), software applications are hosted by cloud service provider and provided as a service over the internet.

The today's need is to use Cloud computing in all areas where application can easily deploy on cloud and provides software as a sevice to users. The transfer of such applications to the cloud has a couple of advantages, e.g. less maintenance efforts or lower costs. But cloud service need to provide some level of security to such application. However, such security requirements are identification and authentication. Generally such application secured by username/password identification and authentication mechanisms for their PaaS and SaaS applications. Username /password schemes are still the dominant authentication approach used for protecting SaaS applications[1]. While username/password schemes may be sufficient for simple personalized services, they reach their limits in data sensitive areas such as e-Government or e-Health. Such areas require higher security requirements as usually sensitive data are processed. If such applications should be moved to the cloud, these high security requirements

must be fulfilled. One possibility to meet those requirements is the use of stronger authentication mechanisms for protecting SaaS applications, e.g. by the use of Intercloud ID(IID).

This approach demonstrated the use of IID solutions for secure cloud authentication. Therefore, system relies on the IID interoperability framework. It increases the usability and user comfort by additionally enabling single sign-on. By using this, users are seamlessly authenticated to several cross cloud services by their IID only once for authentication. The proposed system demonstrated single sign-on authentication between two public cloud service providers. The use of IIDs and SSO for cloud authentication paves the way for increasing future cloud adoption in sensitive areas such as e-Government or e-Health, as legal requirements can be easier fulfilled compared to username/password authentications. Proposed system uses the concept of IID for authenticating cross cloud application and for securely accessing local cloud application, implemented two level authentications.

For achieving single sign on in cross cloud environment, the concept of federated identity management where credential of all the users manages to their respective cloud which avoids remembering and entering different credential for different application. The proposed approach also provides federated Identity management [3] which uniquely identifies the user in cross cloud environment. It manages the identity of all the users associated with different cloud in third party auditor which provides centralized authentication, where IID's of the entire user getting stored and managed. Furthermore, the system also uses AES [8] encryption to securely upload and download of file so; a secure channel is maintained, to achieve privacy of the user and Integrity of the data.

The remainder of the paper is structured as follows. Section II gives some literature studied on single sign-on, some protocol used by existing system and need of cross cloud architecture. The Federated identity management in cross-cloud SSO architecture using IID is presented in Section III. Finally, draws conclusions in Section IV.

II. LITERATURE STUDIED

This section gives some preliminary study of proposed work. Here, the basic concept of SSO, existing identification and authentication approaches and some protocol relating to given work are briefly discussed.

A. Single Sign-On (SSO)

Generally cloud service provider offers one or more SaaS applications for the user comfort. The provider maintains user data in management database for further identification and authentication. If cloud service provider offers multiple SaaS application then separate user management need to be run for each application. Therefore, when user want to access n application from same or different cloud service provider then user must go through n times authentication. These continuous authentication processes may lower the user's reliability. To overcome this drawback, the concept of single sign-on (SSO) can be used. SSO allows the user to access one or more application by authenticating only once which avoids frequent reauthentication[2] so the user usability can increase. In SSO, user has to provide unique or single set of credential for accessing n application, so once the user is authenticated at one cloud service provider then they automatically identified in other cloud service provider. The SSO system provides various advantages, first user just need to remember single set of credential which avoids burden on user. Second authentication time and cost can be save. Third only single user management need to run at cloud service provider side which improves security and avoids maintenance of multiple databases. SSO system also have some drawbacks like If someone steals your credential then complete SSO sytem is available to the attacker and If the user management fails then user can not able to access single application. The whole system is disturbed.

B. Protocols

Following are standard protocol has been widely used for implementation of SSO system. These protocols also support cross-cloud environment.

1. **Security Assertion Markup Language (SAML):** The Security Assertion Markup Language (SAML) [6] is XML-based open

standard format. It is especially designed for the secure exchange of authentication, and authorization data between service provider and identity provider.

2. **WS-Federation:** WS-Federation [7], is XML-based specification, especially designed for enabling identity federation across different security realms. It is a part of the WS-Security framework. Microsoft's Windows Azure [8] cloud platform relies on WS Federation.

3. **OpenID:** OpenID [5] is a decentralized SSO approach especially used for web-based services. In OpenID Users typically authenticate by username/password authentication mechanisms and receive a URL-based OpenlD identifier. Authentication is managed by OpenID providers. OpenID allows user to use an existing account to sign in to multiple websites, without needing to create new passwords. Google for example is an OpenID provider.

4. **Shibboleth:** In this protocol the federation process between the identity providers is conducted on the base of the list containing the names of the providers and some predefined rules. This common rules act like agreement used between the providers. One of the missing features of this model is the complexity of the management of the provider's list.

The proposed system uses the concept of OpenID[5] for generation of IID, by using this a secure access cloud application in cross-cloud environment is possible.

C. Current Situation for Cloud Authentication

Every cloud service provider offer SaaS application for satisfying customer requirement and generally those applications are secured by some dominant authentication mechanism like username /password. For managing users credential, each cloud service provider hosts its own and separate user management. This type of mechanism has some drawback like username/password is very weak authentication scheme in some sensitive areas like Government sector and another drawback is, user have to remember different set of credential for accessing cloud application. The overwhelming amount of different username/passwords decreases the level of security, because people tend to re-use them or write them down close to their computers. That means, the user first has to register at each provider and second has to authenticate separately. Figure 1 illustrates this current sample situation for cloud authentication.

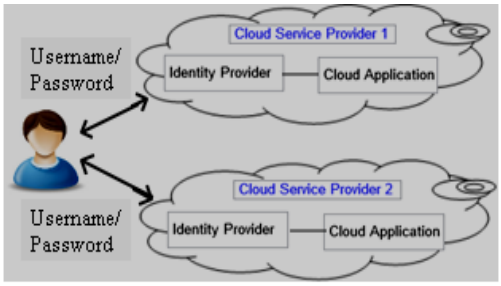

Fig.1 Current situation for cloud authentication [1].

Here, user wants to access two SaaS cloud applications of two different cloud service providers at the same time. The identity provider is hosted in provider's domain and manages user data of the individual domain only. It is assumed that the user is already registered at both identity providers of cloud service provider. Before getting access of application, the user has to identify and properly authenticate at the individual identity provider of each cloud service provider one after another. That means user has to provide separate set of registered credential to respective identity providers for authentication. In this situation it creates burden on user to remember separate set of credential and authenticate separately. To overcome this issue, the proposed system introduces a cross-cloud SSO authentication. By applying this architecture, users need to authenticate only once but still get access to applications of multiple cloud service providers.

D. Cross-Cloud Single Sign on Architecture

To implement secure SSO in cross cloud environment, users normally just need to remember one strong password and the risk of re-use or of writing it down become lower. Another option for increasing authentication security is the use of IID. IIDs allow for unique user identification and strong user

authentication. For instance, the proposed design supports concept of IIDs which can be renewed after some threshold for security reasons. The proposed architecture combines both, secure authentication using IIDs and single sign-on to take advantage of the benefits of both solutions for a cross-cloud SSO authentication scenario.

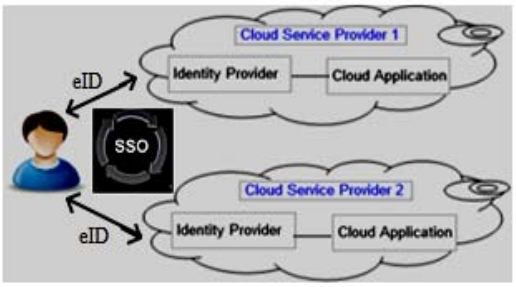

Fig.2 Architecture for cross-cloud SSO [1].

The cross-cloud architecture shown in Figure 2 supports strong IID authentication at different SaaS cloud service application, providing single sign-on between those applications at the same time. This means, by using system generated IID a user needs to authenticate once at one cloud service application. After that, the user is automatically and seamlessly authenticated for other cloud application.

III. FEDERATED IDENTITY MANAGEMENT IN CROSS-CLOUD SSO

A. Terminologies

1. **Identity provider (IDP):** IDP [3] [4] is used to manage the identities (Basic information) of user which were registered for accessing cloud application. It focuses on the authentication of the users as well as on the management of identity information, which can be shared with IDP of other cloud.
2. **Third party auditor:** It is used to gather as well as synchronized basic identities of user from other identity providers. All the identities of users associated with other cloud are centrally managed in third party auditor. Based on the gathered information third party auditor permits the user to redirect other SaaS cloud application.
3. **Intercloud ID (IID):** It is unique number generated by the system for each user, which uniquely identifies the user from other users in cross-cloud environment. IID is centrally managed in third party auditor as well as managed in respective IDP. It can be renewed after some threshold value for achieving security.

B. Proposed Design

The Proposed system adopted cross cloud framework to support authentication at different cloud services on the one side, and to support single sign-on between these different applications on the other side. This model consider three different clouds, where following applications are deployed on two clouds to achieve cross-cloud environment and third cloud is used as third party auditor i.e. cloud C which centrally manages the identities of all the users associated with different clouds. Based on stored identities, third party auditor gives the permission to users for accessing SaaS application of other cloud using IID only. That means in third party auditor all the identities of IDP1 and IDP2 is centrally managed and the users are registered with IDP1 or IDP2 to access SRP and ME and RU application. For example If user is registered with IDP1 in cloud A to access SRP and ME application and want to access the application of cloud B then third party auditor can give the permission to access other cloud application by validating their IID, which is managed in third party auditor. Users don't have to enter their credential again. Accordingly, it can achieve single sign on in cross cloud environment by managing the identities in third party auditor. The model presented in figure 3 shows proposed design.

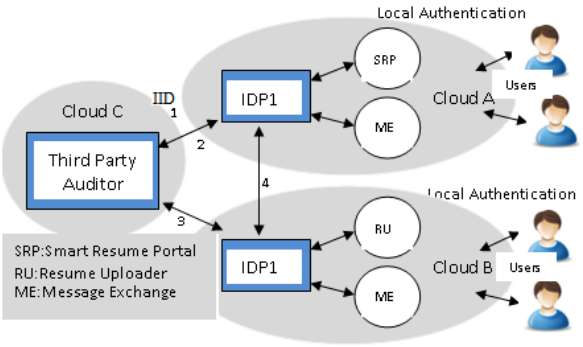

Fig.3 Proposed system model.

Following are the applications deployed on clouds to show SSO in cross-cloud environment.

1. **Smart Resume Portal:** SRP is used for resume filtering performed by sales person depending on their project requirement.

2. **Resume Uploader**: It is used to upload the resume by the engineer person. Depends of their skill and experience sales person can select their resume for further task.
3. **Message Exchange:** It is used to perform the communication between sales person and engineer person. Both of them registered with different cloud.

C. Authentication Mechanism

It is a trust based collaborative approach between cloud service provider and third party auditor. This system performs different levels of authentication for local SaaS application and application belongs to other cloud.

Algorithm for authenticating the user for accessing application running on same cloud

Provisioning and Security are key concerns for cloud Service providers. System attempt is to demonstrate a secure cloud authentication for accessing SaaS application which provides single sign on. The identities of cloud A and B user are managed by IDP1 and IDP2 respectively. Here two level of authentication is used, in first level user have to enter their login credential and in second level, Perform IP filtering or one time password method to securely authenticate the user. Following are the set of steps which is used for authenticating the users for accessing SaaS application of their registered cloud.

1. A user interface will be shown which has the option of creating and managing user account for cloud application. User have freedom to register, access update their credential on cloud portal.

2. When the user registers first time, his/her IP address is recorded by the system automatically.

3. User comes to portal login page and there is an option of sign in for particular application like SRP, RU etc.

4. When user enters their credential on login page, the credential as well as IP is verified and site will redirect to the user interface. Now user has a right to access the application. Here system uses username, password and pattern as credential criteria.

5. If user is accessing from other machine definitely IP is not matched, then one time password is automatically generated by the system and send it to his email-id.

6. User have to enter the one time password in portal login page then system will change previously recorded IP and set current machine's IP in the database and provide access to SaaS application.

All above steps are executed i.e local authentication, when user tries to access the application of cloud where user has registered. But when user tries to access the application of other cloud then user don't have to enter their credential again. User can get access by entering IID only.

Algorithm for authenticating the user for accessing application running on other cloud

By using federated identity in cross-cloud environment, user can able to access the application of other cloud from their registerd cloud without reauthentication. Following are the set of steps for accessing Saas application running on different cloud. It is assumed that cloud A user want access the application of cloud B.

7. This system uses the concept of OpenID[5] for generation of Intercloud ID(IID). It is automatically generated by the system when the user register first time, and it can be renewed after some decided threshold to avoid some attacks and increased security. IID can be used to access the application of other cloud.

8. When the user wants to redirect some other cloud, he/she has to enter recent IID for uniquely identifying the user.

9. The IID is currently verify with local database where user is registered and then it will encrypt using AES algorithm. Encrypted IID will send to third party auditor shown in step1 in proposed model.

10. In third party auditor decryption is performed and verifies IID with user identities managed in database.

11. If IID matches then one token is generated over IID using SHA-256 algorithm and encrypt the token using AES algorithm and send it to Cloud A and B both. Step 2 and 3 indicates sending of token to cloud A and B

Received hash token gets decrypted at both clouds. Cloud B stores received token into its database along with user identities (like who is going to access the application) and starts the timer for some threshold value. Here provides only 10 min thresholds to verify whether the third party auditor gives the permission to cloud A user or not.

12. On the other side one autofill token page will appear on cloud A user's screen. User have to submit the token within specified threshold i.e 10 min for getting access if the user is busy somewhere and not able to submit then he has to generate another token, If user has successfully submitted the token within specified threshold then encrypted token will send to cloud B mentioned in step 4.

13. On cloud B side decryption of token is performed. Now the cloud B will have two token, one is received from third party auditor and other is from cloud A if both token matches then cloud A user successfully redirect to cloud B to access the SaaS application. It indicates that third party auditor gives a permission to cloud A user to access the application of cloud B. Vice versa is also possible. To achieve the security, some threshold is applied to submit the token.

14. The given system maintains the database for managing the identities of user in each cloud for audit of the user (user operations) so that the accountability also can be shown.

15. When the user has successfully redirected, the secure channel is maintained between user and cloud application as well as between two different cloud applications associated with two different clouds. It uses AES 128 bit encryption for maintaining secure channel. So further communication can performed through secure channel only, accordingly integrity can be achieved.

IV. CONCLUSION

This model incorporates the two level authentications where in first level user has to remember his own login credential and in second level, uses the concept of IP filtering and one time password to achieve security. To achieve single sign on in cross cloud environment, it is used IID. With the help of IID user can easily authenticate to access the application of other cloud for that system manages the identities of all the users to their registered cloud as well as third party auditor who give the permission to access the application of other cloud. Here by remembering single set of user credential and IID, user can get access of any SaaS application running on different cloud. This model can be extended in future by adding some security features to overcome some attacks so the system can be more secure.

REFERENCES

[1] "Secure Cross-Cloud Single Sign-On (SSO) using IIDs" by Bernd wattendorfer, Arne Tauber E-Government Innovation Center (EGlZ) Graz University of Technology Graz, Austria.

[2] "Single Sign On For Cloud" by Pratap Murukutla National Institute of Technology,Karnataka, K.C. Shet National Institute of Technology,Karnataka.

[3] "Identity management based security architecture of cloud computing on multi-agent systems" by R.M. Lguliev Institute of Information Technology ANAS Baku, Azerbaijan, F.C. Abdullayeva Institute of Information Technology ANAS Baku, Azerbaijan.

[4] Balasubramaniam, S.; Lewis, G.A.; Morris, E.; Simanta, S.; Smith, D.B., "Identity management and its impact on federation in a system-of-systems context," Systems Conference, 2009 3rd Annual IEEE , vol., no., pp.179,182, 23-26 March 2009 doi: 10.1109/SYSTEMS.2009.4815794

[5] Khan, R.H.; Ylitalo, J.; Ahmed, A.S., "OpenID authentication as a service in OpenStack," Information Assurance and Security (IAS), 2011 7th International Conference on ,vol., no., pp.372, 377

,5-8Dec. 2011 doi: 10.1109/ISIAS.2011.6122782.

[6] Fatemi Moghaddam, F.; Karimi, O.; Hajivali, M., "Applying a single sign-on algorithm based on cloud computing concepts for SaaS applications," Communications (MICC), 2013 IEEE Malaysia International Conference on , vol., no., pp.335,339, 26-28 Nov. 2013 doi: 10.1109/MICC.2013.6805850

[7] Ghazizadeh, E.; Zamani, M.; Ab Manan, J.-L.; Khaleghparast, R.; Taherian, A., "A trust based model for federated identity architecture to mitigate identity theft," Internet Technology And Secured Transactions, 2012 International Conference for , vol., no., pp.376,381, 10-12 Dec. 2012

[8] Dreo, G.; Golling, M.; Hommel, W.; Tietze, F., "ICEMAN: An architecture for secure federated inter-cloud identity management," Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on , vol., no., pp.1207,1210, 27-31 May 2013.