An OpenID Based Authentication Mechanism in a Distributed System Environment

Yu-Lin Jeng

***Abstract*—There are various technical issues inside a distributed system and authentication issue is one of them. OpenID is a decentralized single sign-on authentication system on internet. With more and more applications are authenticated through OpenID mechanism, an OpenID based authentication architecture for a distributed system environment is designed in this paper. The proposed architecture emphasizes the advantage of OpenID deployed in a decentralized environment composed of system nodes. Besides, the proposed architecture supports the existing database authentication like AD or LDAP which are popular and widely used in enterprise. The detail illustration about the architecture that describes the network topology for authentication is given in the last paragraph. The proposed mechanism can solve the authentication issue in a distributed system.**

***Index Terms*—OpenID, distributed system, AD, LDAP.**

I. Introduction

The distributed system in this paper is defined as a system composed of several autonomous computational entities, each of which has its own computing resources. In this paper, computational entities are called system nodes. The distributed system is supposed to continuously coordinate the use of shared resources so that no conflicts or deadlocks occur in the system. The nodes are connected as a network or cluster either using an organizations internal network or the public internet. Every node has common user pool and data pool and provides services for user. Users can access the services through any node in the distributed environment. All data in each node can be synced by communication between nodes using synchronizers [1], [2]. Besides, there are also many challenges that are unique to distributed system. This paper focuses on the authentication issue of a distributed system and proposes an OpenID based authentication mechanism in a distributed system environment.

The rest of this paper is organized as following. A related survey of OpenID research is introduced in Section 2. Section 3 depicts a general architecture for the proposed distributed authentication system. Along with the architecture, three types of authentication process are proposed. Finally, conclusion remarks are shown in the last section.

Manuscript received May 1, 2012; revised June 12, 2012. This study is conducted under the "Cloud computing systems and software development projects" of the Institute for Information Industry which is subsidized by the Ministry of Economy Affairs of the Republic of China.

Yu-Lin Jeng is with the Cloud System Software Institute, Institute for Information Industry, Taipei, Taiwan (e-mail: jackjeng@iii.org.tw).

II. Related Survey

OpenID is a distributed open standard technology for user to be authenticated in a decentralized approach. The OpenID standard provides a framework for communication between identity provider and the OpenID acceptor. OpenID standard consists of user, IdP (Identity provider who provides OpenID service) and the acceptor (RP: Replying Party who uses OpenID information). Everyone can use OpenID system without extra cost as their authentication identifier. In this manner, user does not need to keep their ID and password for each site separately. The sites that provide services do not need to maintain visitor's ID and password accordingly. OpenID has been widely applied in several large web sites and has been emphasized in authentication mechanism research. [3] surveys OpenID providers and the data attached to it, as a means to learn the formats and semantics of information currently shared by OpenID users. Due to the features of OpenID, it is usually used for single sign on purpose. Research focuses on single sign on utilizes OpenID as an authentication service and applies OpenID in different applications [4], [5]. The security issue is also been emphasized in research of [6], [7]. Besides, a technique to strength user authentication is proposed to solve the phishing problem of relaying parties [8]. With the growth of mobile devices, OpenID has also been discussed in the mobile authentication application. [9] proposed an OpenID authentication mechanism using one-time-password to assure the security of communication among mobile phones, web service providers and OpenID providers. OpenID can be applied in various domains; this paper focuses mainly on the decentralized authentication mechanism in a distributed system and proposes the architecture in the following paragraph.

III. System Architecture

The proposed architecture can be divided into two sections. One is single node authentication, describes how a node does the authentication process using OpenID process. Besides, the proposed architecture supports the widely used authentication data schema including Active Directory (AD) and Lightweight Directory Access Protocol (LDAP). The other one is authentication topology, describes the three different types of authentication methods and its related process.

A. Single Node Authentication

A single node authentication process can be completed either by external OpenID provider or internal OpenID

provider. External OpenID providers perform robust authentication include AOL, Google, MySpace or Yahoo. Internal OpenID provider can operate authentication process through OpenID standard, and it can also support the existing databases or directories (AD or LDAP) for authentication.

The single node authentication process starts from the login request of a user in Fig. 1. User can select either external OpenID provider or internal OpenID provider to run the authentication process. Once the user determines an OpenID provider, the login request invokes IdP and redirect the user to IdP webpage to do the authentication process. If the user select internal OpenID provider, the user would be redirected to do the authentication by private database (AD or LDAP). When the user is authenticated by external or internal Idp, the user can be redirected to the other service node.

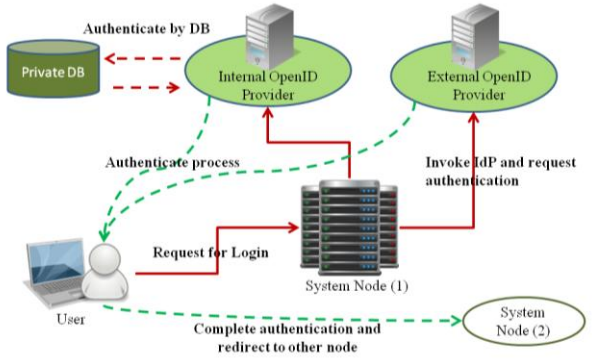

Fig. 1. Authentication process for a single node.

B. Authentication Topology

The distributed system uses a semi-central authentication model, that is, in a distributed system a user's home node will be its central authentication node. Since users are distributed in different home replicas, there is not a node that responsible for all authentication tasks and there is no single node failure in this model accordingly. Whenever a node is fail, a substitute node will be chosen from existing alive replicas. At that time, the substitute node will take more authentication loads (approximate double loads).

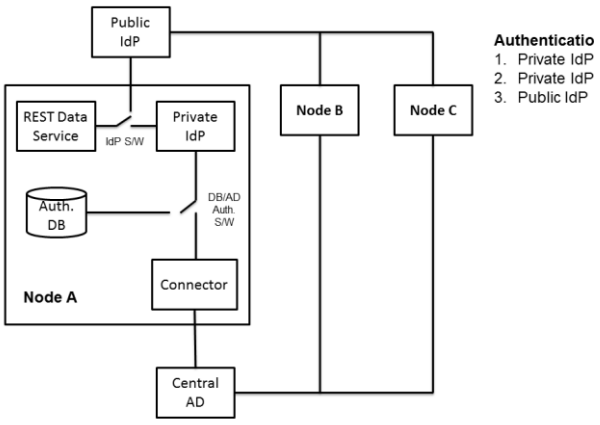

Fig. 2. Network topology for authentication.

Fig. 2 shows the network topology of the proposed distributed authentication mechanism. In the network topology, there are three types of authentication method described in follows.

C. General Flow and Interaction:

Scenario: Kevin's home node is Node A. Kevin asks for a resource on Node B.

Authentication Method: Private IdP with Auth. DB (LDAP):

1) Kevin enters his OpenID and password and tries to log on Node A.
2) Node A's Private IdP checks Kevin's login information on its authentication database (Auth. DB) and then responses a credential to Kevin.
3) Kevin uses the credential to access a resource on Node B.
4) Node B validates the credential via Kevin's home node – Node A.
5) Node B marks the credential as a valid credential in its local memory and responses the resource to Kevin.

Authentication Method: Private IdP with AD:

1) Kevin enters his OpenID and password and tries to log on Node A.
2) Node A's Private IdP checks Kevin's login information with AD via its local Connector and then responses a credential to Kevin.
3) Kevin uses the credential to access a resource on Node B.
4) Node B validates the credential via Kevin's home node – Node A.
5) Node B marks the credential as a valid credential in its local memory and responses the resource to Kevin.

In the above description, if Kevin does not interact with Node B anymore, the valid credential in Node B will be timeout after a fixed of time, which is managed by Node B. After timeout, if Kevin tries to use the same credential to interact with Node B, a credential verification between Node A and B will be re-invoked again.

If Kevin tries to interact with Node B and the credential is still alive on Node B, the Node B will not check the correctness of the credential with Node A again.

Assuming there is a connection problem between Node A and B, the Node A and B are separated into two different sub-distributed systems. Also, assumes the credential is timeout. It Kevin tries to use the same credential to interact with Node B, the Node B will try to validate credential with a new node (a Node A substitutes in sub-distributed system that Node B belongs to). The Node A substitutes can be calculated by consistent hash function.

Whenever a node tries to validate a credential with another node or the user credential is timeout. The user will be asked for re-logon to its home node again. If the mechanism needs to run in mobile device, it has to implement a special API for mobile device (APPs) to do authentication.

Authentication Method: Public IdP:

1) Kevin chooses a Public IdP and enters his OpenID and password and tries to log on Node A.
2) The Public IdP responses a credential to Kevin and tells Node A the credential is valid.
3) Kevin uses the credential to access a resource on Node B.
4) Node B validates the credential via Public IdP.
5) Node B marks the credential as a valid credential in its local memory and responses the resource to Kevin.

In the above description, all replicas must have the ability to connect with the Public IdP. All replicas manage its local valid credentials (session tickets). Whenever a node meets an unknown credential, the node will try to validate the credential with Public IdP. The user will be asked for re-logon to its home node again.

IV. Conclusion

There are many technique challenges in distributed system architecture; distributed authentication is one of them. OpenID has the potential to make authentication a distributed service, and facilitating a distributed system architecture. This paper has focused on the authentication issue of a distributed system and proposes the distributed authentication architecture. The proposed architecture is based on the popular OpenID standard and also considers the use of existing database for authentication. It resolves the distributed authentication issue in a distributed system environment. Additionally, the original OpenID standard requires a publis IdP for authentication. However, most enterprises have their own authentication database like AD or LDAP. The proposed network topology for authentication process illustrates three types of authentication methods. The three types of authentication methods resolve the original public IdP issue for enterprise to support a private IdP with internal existing database (AD or LDAP). This study gave a direction of distributed authentication services in a distributed system, and wishes it could inspire follow-up researches in this area.

References

[1] Lynch, and A. Nancy. *Distributed Algorithms*. Morgan Kaufmann, ISBN 1-55860-348-4. 1996.

[2] Peleg and David. *Distributed Computing: A Locality-Sensitive Approach*. SIAM, ISBN 0-89871-464-8. 2000.

[3] Tapiador and A. Mendo, "A survey on OpenID identifiers," in *Proc. of the 7th International Conf. on Next Generation Web Services Practices (NWeSP)*, Salamanca, Spain, 2011, pp. 357-362.

[4] J. Bellamy-McIntyre, C. Luterroth, and G. Weber, "OpenID and the Enterprise: A Model-based Analysis of Single Sign-On Authentication," in *Proc. of the 15th IEEE International Enterprise Distributed Object Computing Conf. (EDOC)*, Helsinki, Finland, 2011, pp. 129-138.

[5] R. H. Khan, J. Ylitalo, and A. S. Ahmed, "OpenID Authentication As A Service in OpenStack," in *Proc. of the 7th International Conf. on Information Assurance and Security (IAS)*, Malacca, Malaysia, 2011, pp. 372-377.

[6] J. Wei, M. Zhang, X. Ding, and Y. Wang, "Research on Multi-Level Security Framework for OpenID," in *Proc. of the Third International Symposium on Electronic Commerce and Security*, Guangzhou, China, 2010, pp. 393-397.

[7] H. Kyung and J. S. Hun, "The Security Limitations of SSO in OpenID," in *Proc. of the 10th International Conf. on Advanced Communication Technology (ICACT)*, Gangwon-Do, Republic of Korea, 2008, pp. 1608-1611.

[8] Y. jae-Hwe and J. Moon-Seog, "A Mechanism to prevent RP Phishing In OpenID System," in *Proc. of the 9th IEEE/ACIS International Conf. on Computer and Information Science*, Saint Louis, MO, 2010, pp. 876-880.

[9] H. Wang, C. Fan, S. Yang, J. Zou, and X. Zhang, "A New Secure OpenID Authentication Mechanism Using One-Time Password (OTP)," in *Proc. of the 7th International Conf. on Wireless Communications, Networking and Mobile Computing (WiCOM)*, Wuhan, China, 2011, pp. 1-4.

Yu-Lin Jeng was born in Taiwan, Republic of China, on December 13, 1980. He received the B.S. degree in information management department from Hsing Kuo University of Management, Tainan, Taiwan, in 2004. He received his Ph.D. degree in the Department of Engineering Science from the National Cheng Kung University in 2009. His major research interest includes e-learning, artificial intelligence, data mining, and mobile engineering. He was a senior engineer of Innovative DigiTech-Enabled Application and Services Institute, Institute for Information Industry. Presently, he is a section manager of Cloud System Software Institute, Institute for Information Industry in Taiwan.