Access Control for XML Messaging Systems using Web Services

[1]Vinaya V, [2]Praveen M. S, [3]Pradeep G. S, [4]Jeevan Vijay, [5]Alok Ranjan
[1,2,3,4]Dept. of CSE, Canara Engineering College, Manglore, Karnataka, India
[5]Dept. of CSE, BANTWAL, Karnataka, India

Abstract
Web services over the Internet are widely used nowadays. We describe the design of an Access Control System using Web Services for information and content management. XML is quickly emerging as the standard data format for Internet technology. It is also very popular for the broader range of component technologies. XML is a useful way to introduce structure into the body of HTTP requests. Here we described the solution for control mechanism of the information systems which have unique message id, requesting and giving permissions to different channels, and accessibility of the user to the specific channels by giving permissions to the user to use XML Messaging Systems using Web Services.

Keywords
Authorization, Web Services, SOAP

I. Introduction

Information and content management systems require access controls to information content. In publish/subscribe style systems, for example, there will be various restrictions on access to both topics and particular privileges associated with that topic, or channel. We are particularly interested in access control systems that are associated with XML metadata systems. XML metadata is important for future Web service applications [1]. Open Grid Service Architecture (OGSA) [2] and Semantic Web [3] both depend on metadata [4]. We have examined the low-level requirements for managing the XML nuggets of such systems, which include the following: composing tools for creating valid, correct XML metadata nuggets; an architecture and implementation for delivering metadata in the form of messages to listeners who has access to use specific message channel; metadata browsers that can sort and display the nuggets by category; and user role/ access control system to define user levels and privileges. The implementation of these ideas is discussed in Ref. [5]. Specific applications of this system include a news group system and a book reference manager.

As a next phase of research, we want to, (a). decouple our prototype systems into Web Services with well-defined interface, (b) formalize the various roles and privileges that exist in the system. Web services invoke remote methods by using (typically) XML-based protocol and interface definitions. The message protocol, usually SOAP [6], is bound to a lower level transport protocol such as HTTP. The method interface (expressed in WSDL [7]) describes agreed-up set of methods, parameters and return type for the services. We want to take advantage of these technologies in our systems. Particularly, we want to use them to build a reusable Access Control System that can be used within other systems besides our metadata system.

Access control systems are important to any system with multiple users, and highlighting here other work in this area. For example, UNIX [8] provides a very simple but powerful form of access control that partially inspires some of our system. Akenti [9] is an access control system for distributed resources based upon Public Key Infrastructure. An important recent development for Web Services is WS-Policy [10], which provides secure environment for Web Services. Also for Web Services, SAML [11] can be used to convey both authentication and access control assertions in SOAP headers.

II. Literature Survey

Most web applications that offer user accounts do so in part to restrict certain visitors from accessing certain pages within the site. In most online message board sites, for example, all users - anonymous and authenticated - are able to view the message board's posts, but only authenticated users can visit the web page to create a new post. And there may be administrative pages that are only accessible to a particular user (or a particular set of users). Moreover, page-level functionality can differ on a user-by-user basis. When viewing a list of posts, authenticated users are shown an interface for rating each post, whereas this interface is not available to anonymous visitors.

We are particularly interested in access control systems that are associated with XML metadata systems. we described the solution for control mechanism of the information systems which have unique message id, requesting and giving permissions to different channels, and accessibility of the user to the specific channels by giving permissions to the user to use XML Messaging Systems using Web Services applications [1]. We have used the low level protocol called SOAP [6]. The XML metadata may be organized into various categories or channels, we assign all the channels a unique, hierarchical URI [12], which provides a method of inheriting permissions, similar to the UNIX file structure and then using the services of the Web service data model we will assign the channel for the end user to use it.

Akenti [9] is an access control system for distributed resources based upon Public Key Infrastructure. An important recent development for Web Services is WS-Policy [10], which provides secure environment for Web Services. Also for Web Services, SAML [11] can be used to convey both authentication and access control assertions in SOAP headers.

III. XML Messaging System

XML messaging represents a rapidly growing, dynamic area of IT, a situation that makes it exciting and tiresome at the same time. As B2B exchanges and other forms of inter-business electronic communication grow, XML messaging will be more widely deployed than ever. To start our exploration, we need to understand the basic premise of XML messaging and what the term messaging implies. For purposes of this article, I define message as follows:

A collection of data fields sent or received together between software applications. A message contains a header (which stores control information about the message) and a payload (the actual content of message).

Messaging uses messages to communicate with different systems to perform some kind of function. We refer to the communication as being message-oriented because we would send and receive messages to perform the operation, in contrast to an RPC (Remote

Procedure Call)-oriented communication. A simple analogy may help: think of messaging as email for applications. Indeed, messaging possesses many of the attributes of individuals sending email messages to one another.

XML METADATA XML metadata is important for future Web service applications. The term metadata is an ambiguous term which is used for two fundamentally different concepts. Although the expression "data about data" is often used, Basically Metadata is structured data which describes the characteristics of a resource, where Open Grid Service Architecture (OGSA) [2] and Semantic Web [3] both depend on metadata [4].

XML [13] messages can be used to provide a computer platform-, programming language-, and application endpoint-independent way for both synchronous and asynchronous communication to take place. Instead of concentrating on endpoint implementations (some of which may be produced by independent developers), we can develop application message formats and use common wire protocols (such as HTTP [14] and SMTP [15]) to transport these messages. These messages then become events in a general purpose message-oriented middleware system.

There are numerous potential applications for such a system, including a) a newsgroup System that allows users to post messages, which can then be delivered by email, through a Generated web page, or both; b) a training registration system that allows students to register for classes and instructors to post classes and course. Fig. 1 shows the Xml messaging system

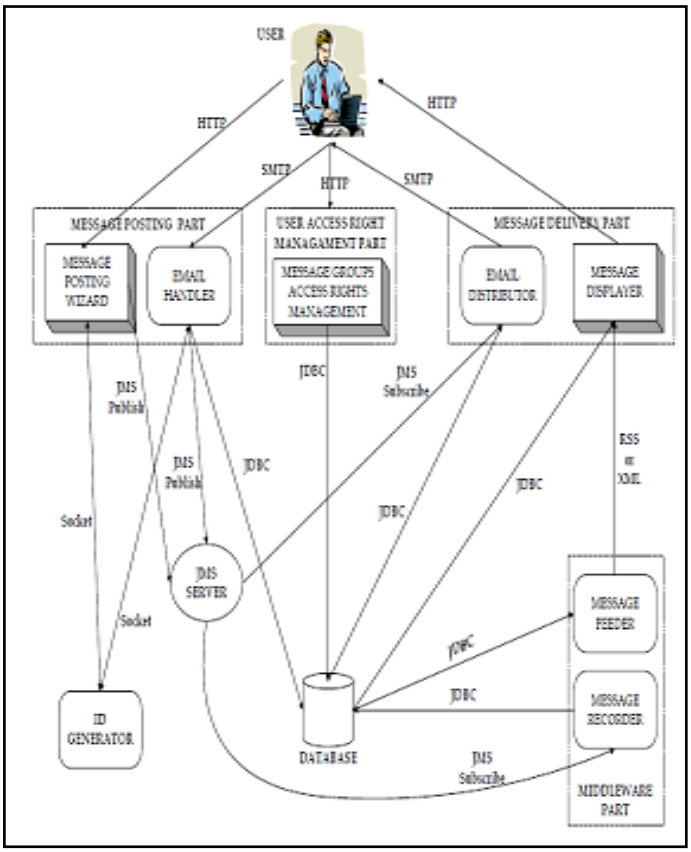

Fig. 1: Our XML Messaging System Architecture Supports Both Intermediate Delivery and Archiving of Messages

IV. Access Control Systems Requirement and Structure Data Models

XML metadata may be organized into various categories, or channels. In our newsgroup system, for example, we have many news topics. Each topic includes users with different types of roles, with multiple access privileges. Individuals may also possess different roles in different topics. The default "user" role has privileges to read and optionally write to one or more message channels. Users have additional options with regard to the choice of message delivery mechanism. That is, a user may request message notification by email, through a web interface, or both.

Each channel users has been assigned to the role and group. Each role has several privileges (properties). For example, a user may additionally request that attachments to topic postings be sent to him through email. User can make requests to change these properties, which are granted or denied by channel administrators.

Other possible roles include "administrator", and "super-administrator". Message Channel Administrators have the authority to assign users to a specific message channel. A channel user may have administration privileges over more than one message channel, and a specific channel has one or more administrators. They also modify the access rights of a user, denying a user the privilege of writing to a particular channel, for example.

Super administrators manage the entire messaging system. In addition to possessing administrator authorities for all channels, this role has the authority to create new messages channels and assign administrators to them. The channel defines the main object for information systems which defines as unique URI.

Each channel has assigned roles, and each role has rights to be confirmed by administrator in order to use the channels. These channels can be information messaging channel or file systems or any type of systems needed access control systems. The relation is shown in fig. 2. Each channel unique name, and defined roles. Role object have name and rights.

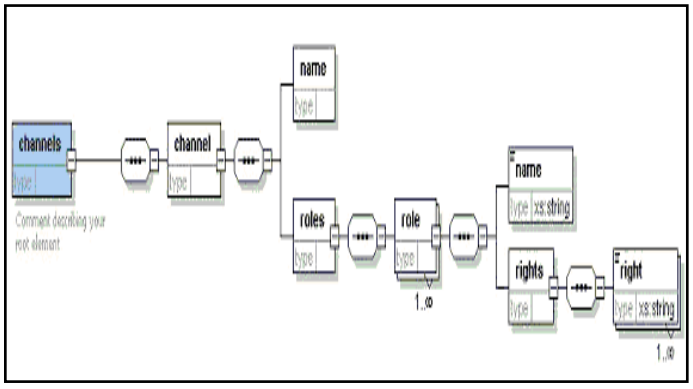

Fig. 2: The Data Model for Roles and Privileges is Expressed in XML

V. User Request and Confirmation Model

The current system has an option for selection mechanism for channels. We define roles for message channel. Admin role, user role, and super administrator role are defined in the newsgroup system, for example. In the access control system, super administrators control the channel administrators, and channel administrators control the channel users. Each channel users has been assigned to the role and group. Second group can modify, remove user rights from the message channel group by having these control structure.

A. Channel Mechanism

The current system has an option for selection mechanism for channels. Each channel has its own administrator, who can grant or deny the requests permissions and services for a channel. This is shown in fig. 2. These are used by individuals to request changes to privileges.

We assign all the channels a unique, hierarchical URI [12], which provides a method of inheriting permissions, similar to the UNIX file structure. Consider a message group for researchers in Signal Image Processing (SIP) this channel should have, [Online] Available: http://[(unique)destination]/Organization/Newsgroup/SIP.

This system can be integrated to the XML Messaging System. If user want to create a new group derived from the SIP message channel, the new URI for the new channel is [Online] Available: http://[(unique)destination]/Organization/Newsgroup/SIP/SIPArchive.

The administrator of the SIP channel will be automatically assigned to the SIP Archive channel. However, if the owner of channel doesn't want to have the upper level administrator to be assigned to the new channel, only the initiator of the channel becomes administrator of the new channel.

B. User Roles and Access Control

Our message system requires access control rights for security. This also governs the dynamic creation of displays in which users see only the message channels that they are entitled to see. Users can have several different roles with associated privileges. We present here some specific role-based access levels. We do not address login and authentication here. These are currently assumed to be inherited from some other system using the messaging system. Access controls are based on this external proof of identity.

Users are allowed to post and receive messages. Users have privileges to read and optionally write to one or more message channels. They also have additional options with regard to the choice of message delivery mechanism. That is, a user may request message notification by push (email), by pull (through a web interface), or both.

C. Message Channel Administrators

have the authority to assign users access to a specific message channel. An individual may have administration privilege over more than one message channel, and a specific channel has one or more administrators. A channel administrator may also modify the access rights of a user, denying a user the privilege of writing to a particular channel, for example.

D. Top Administrators Administer the Entire Messaging System

In addition to the administrator authorities, this role has the authority to create new messages channels and assign administrators to them.

In the access control system, the top administrator controls the channel administrators, and channel administrators control the channel users. Each channel user has been assigned to a role, and message channels define groups. Members of a channel administrator group can modify, add, and remove user rights from the message channel group by having these control structure.

Individuals can have different roles in different groups. For instance, an individual may have only user privileges for one group and administration privileges for another.

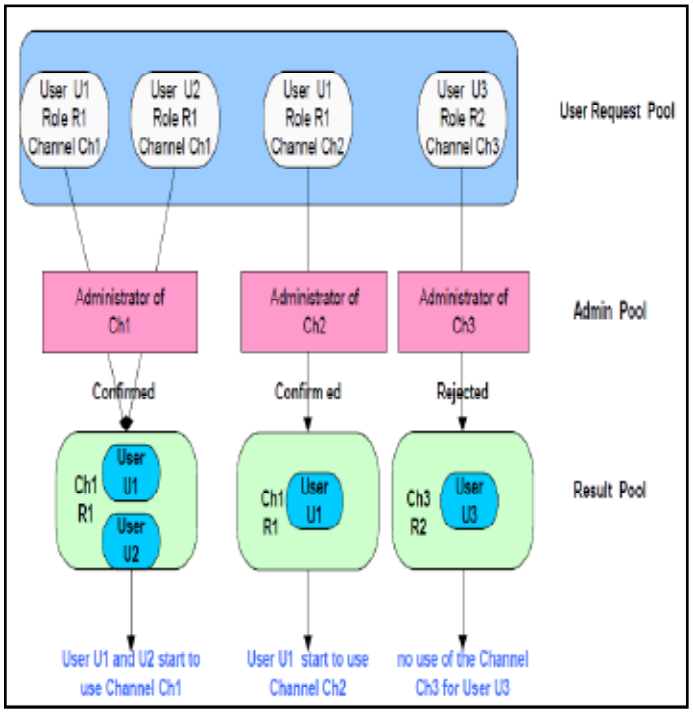

Fig. 3: User Request and Confirmation Model

User might have different roles for each news channels. In the figure, for example, Administrator of Ch1 confirms the user request for both User U1 and User U2 for the channel Ch1. Then, both User U1 and User U2 may use the channel by having Role R1 in the channel system. However, for the channel Ch2, Administrator of Ch2 confirms only the User U1's request having a role R1. Administrator of Ch3 rejects User U3's request for channel Ch3. In summary, different administrators can handle user request objects, which have different roles and different channels. Its depend upon the user who is authenticated or not if he or she is not the authenticated user than the administrator of that channel may reject the request of such users , if authenticated user is requested than channel allocation is accepted and hence the administrator will allow to use that channel.

VI. Web Service for Access Control

Web services have many advantages over others: protocol independent services, well-defined interfaces for distributed services, separation of interface from implementation (transparency). Due to transparent services capability of web service, the underlying data-storage implementation may be XML database, relational databases, or flat file system. The system architecture is shown in fig. 3. The end user sends its request in the SOAP message format. The web server forwards incoming requests to the web service engine, which executes the proper web service in the web service pool. The invoked web service ingests the SOAP message, connects to database using JDBC connection, and performs the incoming service request. If there are responses for the incoming service request, it wraps it into a SOAP message and passes it to the Web service engine. The Web service engine delivers the output to requestor by passing it to web server. Let us now examine services needed for manipulating our data model.

Access Rights Request Service: This service allows a user to make a request on a desired message channel to have proper access rights. For instance, a new user makes a request for "read" and "write" access rights on the "java beginners" message channel by calling this service transparently using her browser.

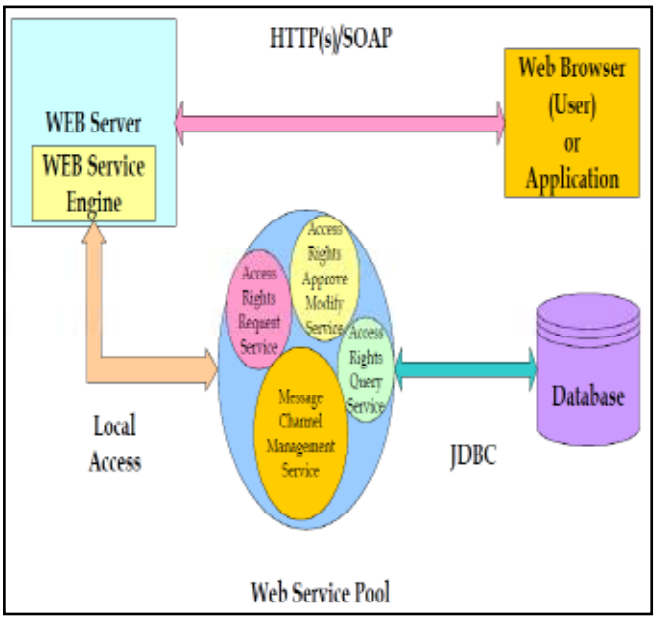

Fig. 4: Interaction of Users with the Access Control Service

When Access Rights Request Service receives this incoming request, it uses the JDBC to connect to database enters user's request.

A. Access Rights Approve, Modify Service

Channel administrators or super administrators use this service. A Channel administrator may approve, modify or reject any request. In addition, she may modify the current access rights of the subscribed users by using this web service. Super administrators assign users as a message channel administrators or remove such privileges by using this service.

B. Access Rights Query Service

service is designated for users who do not have the right to enter any information (access rights request, modification of access rights, etc.) into database but need to access rights information in the database. For example, a message-brokering publisher may need this service to verify the current message sender has write access on given message channel before publishing his/her message.

The following are the current methods of this service:getReadableTopics,getWritableTopics,getAllUsers,getUsersHaveReadAccess,hasReadAccess,hasWriteAccess,getUsersHaveWriteAccess,getAllTopics,hasReadWriteAccess

C. Message Channel Management Service

Super administrators use this service. They can add new message channel into system, remove expired message channels from the system and manage their channel administrators by calling this service.

VII. Conclusion

Here we have presented a data model for access control definitions on an information channel. We implement services that manipulate instances of this data model using a Web services approach. Access controls are part of larger security framework. Authorization must typically be coupled with two other security concepts: authentication and transport level security. In authentication part, the correctness of the user identity is verified. Data integrity and privacy are typically provided by transport level security mechanisms such as SSL. The access control system we have presented here depends on an external authentication method. Currently, we implement only HTTP-based authentication. Future versions may incorporate other authentication systems such as PKI, Kerberos or Shibboleth.

References

[1] V. Gambiroza, B. Sadeghi, E. Knightly,"End-to-End Performance and Fairness in Multihop Wireless Backhaul Networks", In Proceedings of MobiCom, 2004.

[2] M. Kodialam, T. Nandagopal,"Characterizing the Capacity Region in Multi-Radio Multi-Channel Wireless Mesh Networks", In Proceedings of MobiCom, 2005.

[3] R. Anderson, M. Kuhn,"Tamper Resistance - a Cautionary Note", In The Second USENIX Workshop on Electronic Commerce Proceedings, 1996.

[4] A. Seshadri, A. Perrig, L. van Doorn, P. Khosla. "SWATT: SoftWare-based ATTestation for Embedded Devices", In Proceedings of IEEE Symposium on Security and Privacy, 2004.

[5] B. Parno, A. Perrig, V. Gligor,"Distributed Detection of Node Replication Attacks in Sensor Networks", In Proceedings of IEEE Symposium on Security and Privacy, 2005.

[6] Y.-Ch. Hu, A. Perrig,"A Survey of Secure Wireless Ad Hoc Routing", IEEE Security and Privacy, special issue on Making Wireless Work, Vol. 2, No. 3, 2004.

[7] W. Xu, W. Trappe, Y. Zhang, T. Wood,"The Feasibility of Launching and Detecting Jamming Attacks in Wireless Networks", In Proceedings of MobiHoc, 2005.

[8] N. Ben Salem, J.-P. Hubaux,"A Fair Scheduling for Wireless Mesh Networks", In Proceedings of WiMesh, 2005.

[9] M. Raya, J.-P. Hubaux,"The Security of Vehicular Ad Hoc Networks", In Proceedings of SASN, 2005.

[10] J. Bicket, D. Aguayo, S. Biswas, R. Morris,"Architecture and evaluation of an Unplanned 802.11b Mesh Network", In Proceedings of MobiCom, 2005.

[11] M. Felegyhazi, J.-P. Hubaux,"Wireless Operators in a Shared Spectrum", In Proceedings of InfoCom, 2006.

Vinaya. V received his B.E., Degree in Computer Science & Engineering from Kuvempu University at UBDT College of Engineering Davangere Karnataka 2009. At present he is pursuing M.tech Degree in Computer Science & Engineering in Visvesvaraya University at Canara Engineering College Manglore, India.

Praveen M.Sanjeevannanavar received his B.E. Degree in Information Science & Engineering from VTU at STJIT Ranebennur Karnataka. 2010. At present he is pursuing M.tech Degree in Computer Science & Engineering in Visvesvaraya University at Canara Engineering College Manglore, India.

Pradeep G. S received his B.E, Degree in Computer Science & Engineering from VTU at VCET Puttur Karnataka. 2010. At present he is pursuing M.tech Degree in Computer Science & Engineering in Visvesvaraya University at Canara Engineering College Manglore, India

Jeevan vijay received his B.E, Degree in Computer Science & Engineering from VTU at KVG College of Engineering, Sullia Karnataka. 2011. At present he is pursuing M.tech Degree in Computer Science & Engineering in Visvesvaraya University at Canara Engineering College Manglore, India.