Efficient Access Control Mechanism for XML Databases Using Web Services

Aye Sandar Myint, May Aye Khine
University of Computer Studies, Yangon
ayesandarmyint.ccg@gmail.com, maya.khine@gmail.com

Abstract

XML document are frequently used in application such as business transaction and medical record involving sensitive information. Typically, parts of xml documents should be visible to users depending on their roles. And then access control on the basis of data location or value in an XML document is essential. Additionally, web services are application components that are designed to support interoperable machine-to-machine interaction over a network. This interoperability is gained through a set of XML-based open standards, such as the Web Services Description Language (WSDL), the Simple Object Access Protocol (SOAP), and Universal Description, Discovery, and Integration (UDDI). These standards provide a common and interoperable approach for defining, publishing, and using web services. This paper describes the design of an Access Control System using Web Services for xml data and access right management. And then we present an overview of the access control mechanism to build access control services around a Web Services model and address how to increase sever throughput using access control rules functions that are management separately from the server database using web service.

1. Introduction

A web service is an application that accepts requests from other systems across the Internet or an Intranet, mediated by lightweight, vendor-neutral communications technologies. These communi-cations technologies allow any network-enabled systems to interact. As technologies mature, a web service will encompass additional special functionality geared towards performing multiparty B2B collaboration. Web services are evolving and beginning to operate in an extremely intelligent and dynamic way. These smart web services will understand the context of each request and produce dynamic results based on each specific situation. The services will adapt their processes based on the user's identity, preferences, location, and reason for the request.

The XML standards which a web services system is built upon allows for an implementation-neutral approach to performing business collaborations. There are many possible implementations developers can use, including a variety of products, platforms, and standards.

XML is rapidly emerging as the new standard for data representation and exchange on the Internet. As large corporations and organizations increasingly exploit the Internet as a means of improving business-transaction efficiency and productivity, it is increasingly common to find operational data and other business information in XML format. In light of the sensitive nature of such business information, this also raises the important issue of securing XML content and ensuring the selective exposure of information to different classes of users based on their access privileges.

Moreover, as a large quantity of information is presented in XML format on the Web, there are increasing demands for XML security. Until now, research on XML security has been focused on the security of data communication using digital signatures or encryption technologies. As XML is also used for a data representation of data storage, XML security comes to involve not only communication security but also managerial security. Managerial security is guaranteed through access control, but most existing XML access control models are considered based on server-side or client side access control evaluation.

2. Related Work

Several XML access control models [4, 8, 9, 10] provide expressive access control over XML documents. These approaches usually support grant or denial access control specifications, a propagation

mechanism whereby descendant elements inherit rules from their parents, and conflict resolution in case the data is covered by multiple access control rules. Since these models perform access control by traversing XML documents at runtime, the enforcement imposes heavy computational costs especially for deeply layered XML documents with large expressive access control rules.

To overcome this problem, Qi et al. [3], in their research paper, they presented a method that performs in near-constant time regardless of the number of access control rules. This is achieved by using an access condition table generated from the access control rules independently of the XML data and XML documents. However, this approach places limitations on the XPath expressions, and does not provide an efficient runtime evaluation mechanism for value-based conditions. Murata et al. [2] optimized the pre-processing steps by minimizing the number of runtime checks for determining the accessibility of nodes in a query with automata. However, the mechanism is limited to XPath-based language (i.e. XQuery [1]) and cannot handle other query languages and primitive APIs such as DOM. XPath-based document filtering systems [4, 5, 6] also provide value-based access control enforcement and independence of XML data through a pre-computed data structure. However, these approaches focus more on data filtering rather than data selection. Other [6] proposed an efficient table-driven access control model that takes into account XML document updates. It provides runtime efficiency but has limitations on access control expressiveness. Function-based access control system [7], they proposed again the access control system to support both large and expressive access control rules and partial rule updates at low cost.

Moreover, access control systems are important to any system with multiple users. For example, UNIX [12] provides a very simple but powerful form of access control that partially inspires some of our system. Akenti [13] is an access control system for distributed resources based upon Public Key Infrastructure. An important recent development for Web Services is WS-Policy [14], which provides secure environment for Web Services. Also for Web Services, SAML [15] can be used to convey both authentication and access control assertions in SOAP headers.

3. Preliminaries

3.1. Access Control Policy

Various access control policy models have been proposed. We use the one proposed by Murata et al. [11] in which an access control policy contains a set of 3-tuple rules with the syntax1: <Subject, Permission Action, Object>.

3.2. Rule Functions

A rule function is the basic element that performs access control on a specific object or a specific subject. In their model, they define the following three types of rule functions.

Definition Object-driven Rule Function (ORF) is a function indexed by the object and is shared by a group of access control rules. At runtime, ORF receives the relevant attribute(s) of a subject for a given access request and returns an evaluation result to the caller according to the algorithm of the ORF.

Definition Subject-driven Rule Function (SRF) is a function indexed by the subject and is shared by a group of rules. SRF receives the accessed path and returns an evaluation result to the caller on the basis of the algorithm of the SRF.

Definition General Rule Function (GRF) is an extended rule function that is not indexed on either the object or the subject. A GRF may place arbitrary conditions on either the subject or object when determining accessibility and, as such, we use it to encode rules that cannot be bound to a specific ORF or a specific SRF. For instance, we use GRFs to process rules containing //, as these cannot be mapped to a single ORF.

2.3. Evaluation of Accessibility

Given a requested path or the subject of a request, the rule function returns an evaluation result in accordance with both the action permission and the propagation property of the appropriate rule. There are four types of evaluation results as Table 1 shows.

Access Effect	Evaluation Result
+r	GRANT_ON_NODE
+R	GRANT_ON_SUBTREE
-R	DENY
Nothing	UNDECIDED

Table 1. Evaluation results of a rule function

3.4. Mapping Tables

A mapping table is the key component that connects an access request to the appropriate rule function(s) for accessibility evaluation.

We define the following mapping tables.

Definition ORF Mapping Table

The ORF mapping table maps a given access path to one or more applicable ORF functions indexed by a package name and a class name.

Definition: SRF Mapping Table

The SRF mapping table maps a given subject of an access request to one or more applicable SRF functions indexed by a package name and a class name.

4. Access Control Evaluation Algorithm

Based on the defined components (rule functions, evaluation results, and mapping tables), our model computes a decision result of an access request based on the following algorithms. The requirement of this model is: the access control policy must be consistent, so that all of the ancestors of a descendant specified with a grant of accessibility should be accessible as well.

ORF Algorithm

Given the path and subject of a request, the corresponding entry is looked up in the ORF mapping table. If the corresponding entry exists, the ORF name is looked up. Otherwise, the ORF name of its closest ancestor is looked up. Since the accessed path is implicitly controlled by the access to the ancestor, the decision result can be decided on the basis of the propagation information carried with the evaluation result returned from the ancestral ORF.

In accessibility evaluation, a GRF is executed first. If the GRF returns DENY, the decision result is already decided and the ORF method is not executed. Otherwise, the ORF method is invoked and executed for a decision result. According to the access control policy, multiple applicable functions may be bound to a request. To produce the final decision result, we need to combine individual evaluation results. This is done in accordance with the denial-takes-precedence.

SRF Algorithm

The SRF algorithm is much simpler than the ORF algorithm. Given the path and the subject of the request, the corresponding entry is looked up in the SRF mapping table. If the corresponding entry exists, the SRF with that name found in the table is invoked and executed. Otherwise, the decision result is DENY by default

5. Access Control Mechanism Using Web Service

Web services are Web based applications that use open, XML-based standards and transport protocols to exchange data with clients. 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 Figure 3. The end user sends its request in the SOAP message format to the web server via the browser. 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 and executes the request to get the decision. If the access evaluation request returns DENY, the access denied response will be sent back to the user without data retrieval from the database. Otherwise, web service connects to database using JDBC or suitable 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.

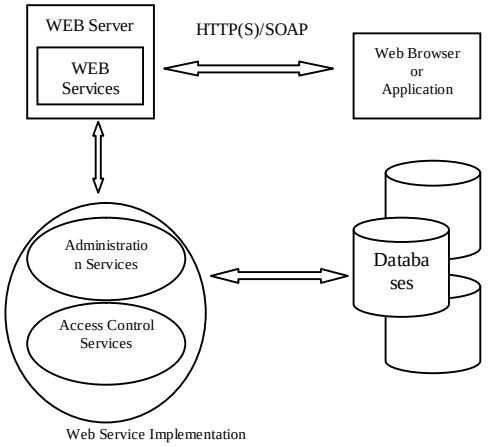

Figure 1: System Architecture of Access Control Mechanism

Now we present the services of the access control mechanism:

Access Control Service:

This service allows a user to make a request to have proper access rights. For instance, a user makes a request for “read” and “write” access rights on the order.xml document by calling this service transparently using her browser. When Access Control Service receives this incoming request, it executes the service to get the decision result. If the access evaluation returns DENY, the access denied response will be sent back to the user without data retrieval from the database. Otherwise, the services sends this request to the XML database server and retrieve data and send back to the user.

Administration Service:

The System Administrator uses this service. The administrator may approve, modify or reject any

request based on their access right. In addition, she may modify the current access rights of the subscribed users by using this web service.

6. Conclusion

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

7. References

[1] UC BOUGANIM, FRANCOIS DANG NGOC, and PHILIPPE PUCHERAL. Dynammic Access-Control Policies on XMl Encryted Data ACM Transactions on Information and System Security, Vol. 10, No. 4, Article 16, Pub. date: January 2008.

[2] M. Murata, A. Tozawa, M. Kudo and H. Satoshi: XML Access Control Using Static Analysis. ACM CCS, 2003.

[3] N. Qi and M. Kudo: Access-condition-table-driven access control for XML databases. ESORICS (2004).

[4] M. Altinel and M. Franklin: Efficient filtering of XML documents for selective dissemination of information. VLDB (2000) pp.53-64.

[5] C.-Y. Chan, P. Felber, M. Garofalakis, and R. Rastogi: Efficient filtering of XML documents with XPath expressions. ICDE (2002) pp.235-244.

[6] Y. Diao, P. Fischer, M. Franklin, and R. To.: YFilter: Efficient and scalable filtering of XML documents. Demo at ICDE (2002) pp.341

[7] Naizhen Qi, Michiharu Kudo, Jussi Myllymaki, Hamid Pirahesh . A Function-Based Access Control Model for XML Databases. CIKM'05, October 31-November 5, 2005, Bremen, Germany.

[8] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati: Design and Implementation of an Access Control Processor for XML documents. WWW 9 (2000).

[9] A. Gabillon and E. Bruno: Regulating Access to XML Documents. Working Conference on Database and Application Security (2001) pp.219-314.

[10] M. Kudo and S. Hada: XML Document Security based on Provisional Authorization. ACM CCS (2000) pp.87-96.

[11] M. Murata, A. Tozawa, M. Kudo and H. Satoshi: XML Access Control Using Static Analysis. ACM CCS, 2003.

[12] F. Grampp and R. Morris, "UNIX Operating System Security", BSTJ, Vol. 62, No. 8, 1984.

[13] S.S. Mudumbai, W. Johnston, M. R. Thompson, A. Essiari, G. Hoo, K. Jackson "Akenti- A Distributed Access Control System".

[14] B. Atkinson, G.Della-Libera, S.Hada,M.Hondo, P. Hallam-Baker, C.Kaler, J.Klein, B.LaMacchia, P.Leach, J.Manferdelli, H. Maruyama "Web Services Security (WS-Security) Version 1.0" April 5, 2002.

[15] J. Hodges, C. Knouse, J. Moreh, R. Philpott "Metadata for SAML 1.0 Web Browser Profiles" Working Draft 01, 1 February 2003.