International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

XML Access Control Models

Hitesh Pambhar

[1]Sr. Engineer, SSL Electrical Systems Team, Axis-Cades Inc., Bangaluru

Abstract

Extensible Markup Language (XML) is definedby World Wide Web Consortium (W3C) as astandard for data representation and transmission onthe Web. The disseminated XML documents on theWeb may contain a secret data that must beprotected from ineligible access, so XML accesscontrol is an important issue in Web informationsecurity.XML access control refers to the practice of limiting access to (parts of) XML data to only authorized users. Here we tried to study, compare and analyze different approaches towards Access control and its policies.

Keywords: Data, Access control, policy, PBAC, XML, DAC,MAC, RBAC, ERBAC, RBACo

I. Introduction

XML (Extensible Markup Language) is widely used in many applications as it has the ability to store, exchange, transfer and retrieve data. Much of the research on XML focuses on storage strategies and query performance. Although data storage and retrieval techniques are important, so is security and in comparison, this is a neglected area. XML databases are multi-user systems, meaning they can be accessed by millions of users and can provide a huge amount of data. Much of this data is sensitive and personal. Confidential data need to be protected and saved in a secure environment. Security research for XML databases is crucial in protecting data from unauthorized processes and misuse.

Different models for XML database access control have been proposed and developed. Access control systems for XML databases can be categorized into three core approaches: discretionary access control (DAC), mandatory access control (MAC) and role-based access control (RBAC) [1] [2] [3]. Most traditional access control models protect data from malicious activities of outside users but cannot protect the data from insiders [4]. Research has suggested that damage caused by insiders is more harmful than that of outsiders [5].

Here we evaluated Trust based Access control which depends on a trust management system, which automatically calculates and updates the trust values of users. Policy Based Access control or PBAC defines attribute to describe property of session and issues a method of policy management independent from application logic, then realizes a new independent mode of session decision. a distributed push-basedXML access control model that effectively works with the increased scale by distributing the system and management workloads to different components (or servers) and several administrators, respectively. XML based access control for pervasive computing follow a RBAC model where access policies are defined for each individual role using XPath expressions. Action and Attributes based Access control model avoids the complex structure of multi-attribute and solves the problem that relevant dynamic authorization and permission changes.

II. POLICY BASED ACCESS CONTROL

As we know Role-based Access control, RBAC [11] is the most popular. Though they have made great progress on accesscontrol, there are still some shortages.Firstly, the restraint of session is not comprehensive.Secondly, current models are not very flexible.

Different from RBACcontrolling session with configuring role privilege (as Fig. 2a), PBAC defines attribute to describe property of session,and issues a method of policy management independent fromapplication logic, then realizes a new independent mode ofsession decision (as Fig. 2b). Therefore, PBAC is moreflexible and multi-policy supporting.

PBAC has some elements as follows:

SUBJECT: set of model's subjects.

TARGET: set of model's objects.

ACTION: set of model's actions.

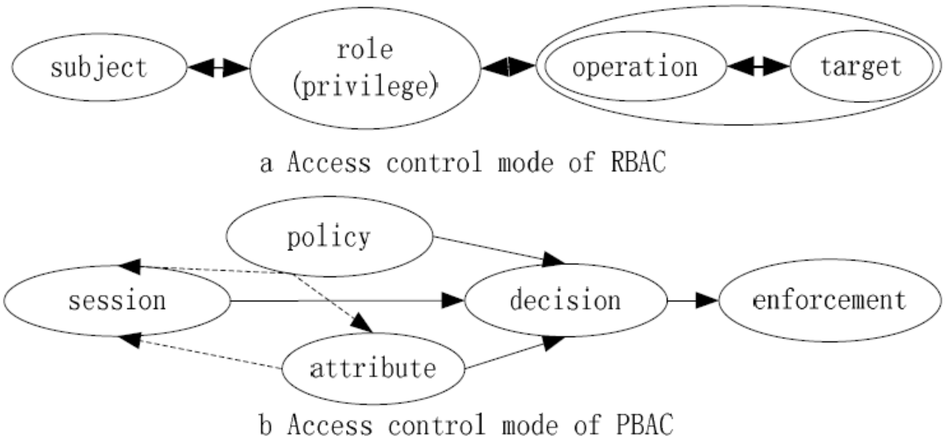


Fig. 2 Control mode comparison of PBAC and RBAC

SESSION: set of model's sessions, which means asubject takes an action on a target. Session has threeelements as subject, target and action,

SESSION ⊆SUBJECT× TARGET×ACTION.

ATTRIBUTE: set of model's attributes, which describesproperty of model's session.

ENTITY: set of entities. The entity refers to subject,target or action.

POLICY: set of policies. Policy is the core of PBAC.

It isused to restrict session request.Different from the exiting models, PBAC represents policy with a direct description method instead of configuring role of subject indirectly.

The relationship of entity and attribute is denoted as:

SA: attribute assigned to subject,

SA ⊆ SUBJECT×ATTRIBUTE

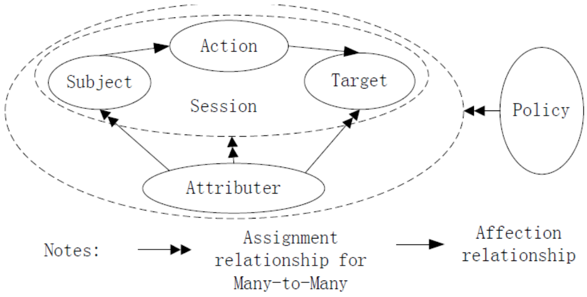
TA: attribute assigned to target,

TA ⊆ TARGET×ATTRIBUTE

SEA: attribute assigned to session circumstance,

SEA⊆SESSION×ATTRIBUTE

The relationship among elements of PBAC is shown inFig. 3, while Fig. 4 shows the framework.



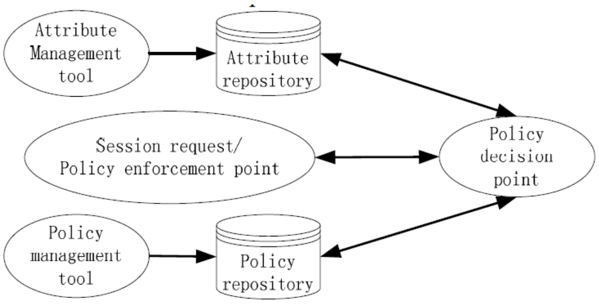


Fig. 3 Relationship among PBAC's element

Fig. 4 Management framework of PBAC

III. TRUST BASED ACCESS CONTROL

Trust-based access control has become an establishedtechnique in many areas, such as networks and virtualorganizations. It depends on a trust management system,which automatically calculates and updates the trust values ofusers. Trust values rely on users behaviors, users histories,userscredit and users Operations.Userscan access resourcesthrough trust values and levels [6] [7] [8] [9] [10].

The architecture of our trust-based access control for XMLdatabases is shown in Fig. 5. It is based on direct trust andignores indirect trust. Direct trust focuses on users operationsand errors.Indirect trust depends on recommendations.

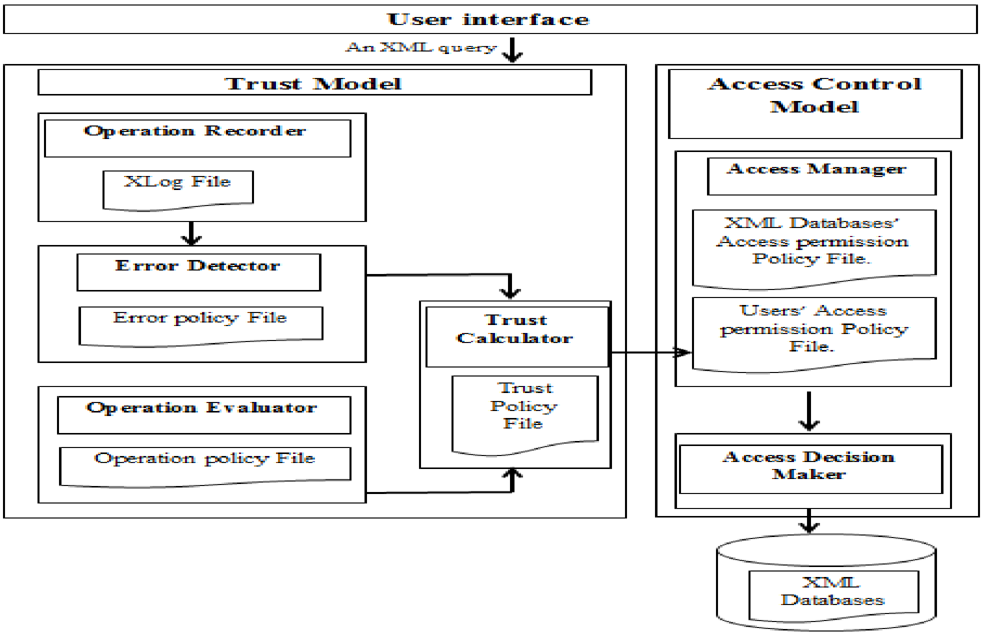


Fig. 5The trust-based access control system for XML databases

The model consists of three main parts: the user interface,the trust model and the access control model. The userinterface sends the access request to XML databases as anXML query. The trust model evaluates users' activities andcalculates the trust value for them. The evaluation processdepends on the recording of some operations in the Xlog file.These specific operations are defined in the operation policyfile. The error detector then finds errors in the Xlog file andassigns a weight to them according to the error policy file.Likewise, the operation evaluator assigns the appropriateweight to misused operations defined in the operation policyfile and recorded in the Xlog file.

The trust calculator determines the trust value, as in
TV = ETV * ETVW + AV * AVW + GOV * GOVW. (1)
In (1), TV = Trust Value, ETV = Existing Trust Value,ETVW = Existing Trust Value Weight, AV = Accuracy Value= 1 - Error Weight (EW), AVW = Accuracy Value Weight,GOV = Good Operations Value = 1 - Bad Operations Value(BOV) and GOVW = Good Operations Value Weight.
All ETVW, AVW and GOVW percentagecan changeaccording to the organization’s policies. For example, AVWcan be 10% if the organization does not consider the errorfactor as an important element or AVW can be 50% if theorganization considers the error rate to be highly significant.The trust value is updated automatically in the users’ accesspermission policy file. The XML database’s access permissionpolicy file describes the trust values required to access XMLnodes. The access manager sub-model matches two files tomanage a user’s right to access requested data. The accessdecision maker allows or denies users access to XMLdatabases according to the results.

IV. ACCESS CONTROL BASED ON DITRIBUTED PUSH

Extensible Markup Language (XML) is definedby World Wide Web Consortium (W3C) as astandard for data representation and transmission onthe Web. The disseminated XML documents on theWeb may contain a secret data that must beprotected from ineligible access so XML accesscontrol is an important issue in Web informationsecurity [12].The existing XML access control approaches areclassified based on their policy enforcement intotwo categories [13, 14]: pull-based and push-based (orpublish-subscribe) approaches. In the pull-basedapproaches, a server receives a request from a clientand responds with an appropriate view. While, inthe push-based approaches, the server periodicallyencrypts the document portions with different keysand then publishes the encrypted document to allusers.Moreover, in push-based XML access control,the policies related to each user must be selectedand distributed for enabling the user to only decryptthe related encrypted portions. But, the policiesdistribution is a complex task because all thepolicies, related to all users, are specified in thesame XML-based file(s) and stored in a securecentralized server.So,distributing the updated/newpolicies to users requires generating a view for eachuser, and then protecting this view for ensuring itsintegrity and authenticity.Since XML document is becoming a standard fordata exchanging on the Web, therefore proposing ascalable XML access control is a crucial needed foraddressing the network effects [15], which is meanshere meeting the increased number of the specifiedpolicies, subscribed users, and publisheddocuments attracted by increasing the number of theprovided services.

Methodology

Apolicy specification language is for decentralizingthe management workload to several trustedadministrators. Then, the model architecture isdistributed into several servers by exploiting thedistributed Client/server architecture as shown inFig. 6 and Fig. 7.

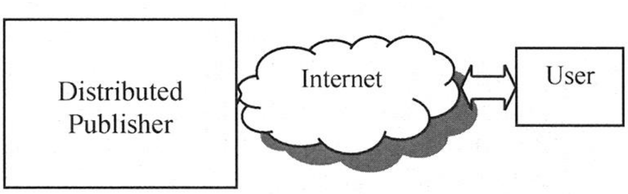


Fig. 6Overall Model Architecture

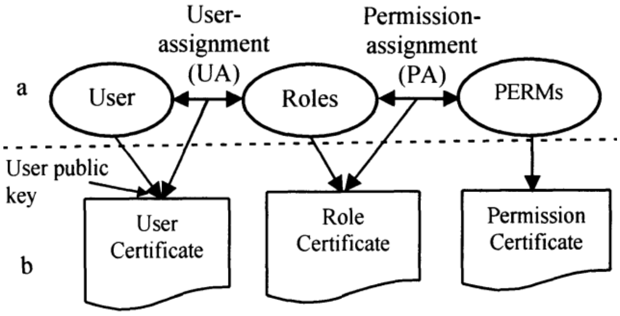
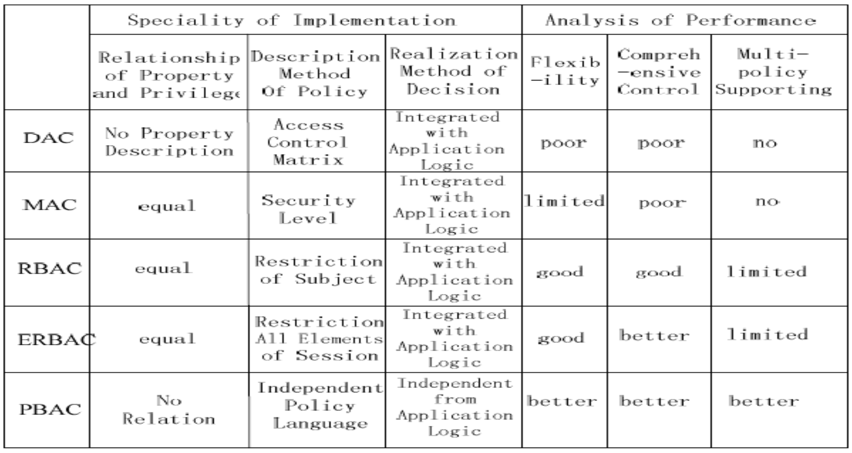


Fig. 7Specifying the RBACO components

In addition, a symmetric (secret) cryptography isused for encrypting the publisheddocument portions. Also, an asymmetric (public key)cryptography is used for digitally signing thedistributed policies andpublished document alongwith securely distributing the secret encryption keysto the right users.

The Decision of PBAC

In order to ensure the consistent and effectiveness ofpolicy, PBAC sets several decision rules of access control.



	Speciality of Implementation			Analysis of Performance		
	Relationship of Property and Privilege	Description Method Of Policy	Realization Method of Decision	Flexib -ility	Compreh -ensive Control	Multi-policy Supporting
DAC	No Property Description	Access Control Matrix	Integrated with Application Logic	poor	poor	no
MAC	equal	Security Level	Integrated with Application Logic	limited	poor	no
RBAC	equal	Restriction of Subject	Integrated with Application Logic	good	good	limited
ERBAC	equal	Restriction All Elements of Session	Integrated with Application Logic	good	better	limited
PBAC	No Relation	Independent Policy Language	Independent from Application Logic	better	better	better

Fig.8Comparison of Access Control Model

Rule 1: If none of policies regulates the request related tothe session, which means the system does not restrict thesession, the session request will be denied.

Rule 2: If there is not less than one policy that prohibitsthe session request, the request will be denied.

Rule 3: When all policies about the session allow it to becarried out, it will be performed.

In view of above, PBAC improves session controlmechanism and its implementation mode. The comparison ofimplementation specialty and performance about DAC,MAC, RBAC, ERBAC and PBAC is shown in Fig. 8.

V. ACCESS CONTROL BASED ON ATTRIBUTES

With the development of the internet in web services, Webservices developing are restricted by communication securityand access control. As we know, traditional Access Control cannot solve those problems. So, it is increasingly important tostudy Access Control Model in web services.on subject (e.g. post, role), resource (e.g. owner, servicesquality) and environment (e.g. currently time, safely level)attributes.

Action and Attribute-based Access Control

With the support of SAML, XACML, and AOP, this papermakes the action a limited condition to access service methodsby combining attributes in user resource and environment andassigns permission through using relation of authorities. It getsa result whether it can access object successfully. The formaldefinition of model is presented by introducing scalable factorsfor ABAC, and then the model is constructed through thosedefinitions and the structure diagram of current access controlmodel.

The ABAC Architecture of Web services is establishedbased on SAML and XACML language model. At the sametime, XACML has a good scalability makes it supportparameterized Policy Description. This model introduces themethod which is one of the elements in action as referencemakes the AOP intercept the invoking method effectively.

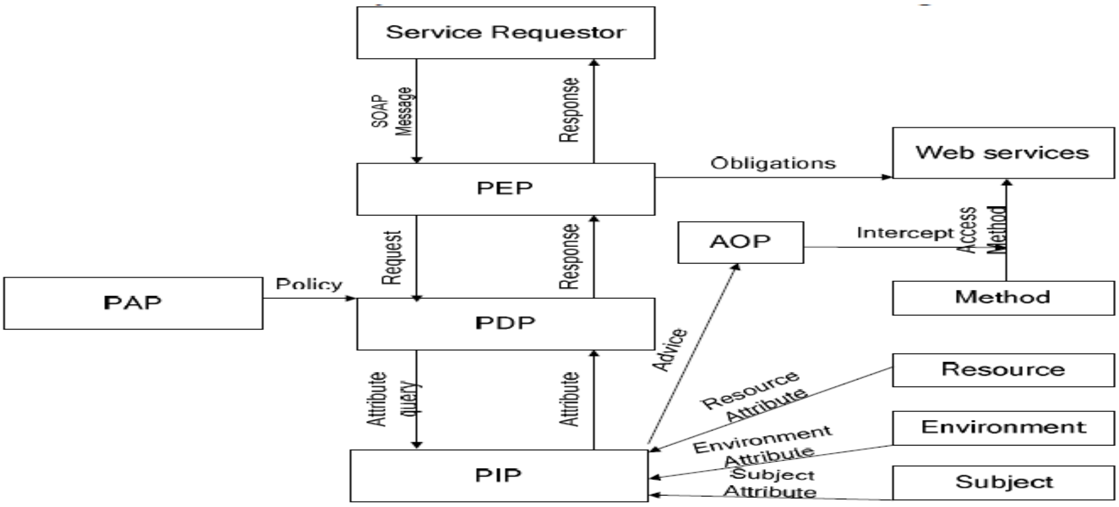


Fig 9. AABAC system Diagram

The process of the model shown in Fig. 9 is described as follows:[16], [17]

1) PEP extracts the attribute information of subject from theSAML statement of message head by processing thereceived SOAP message which is requested by therequester.

2) PEP sends the generated request of access authorization tothe PDP (Policy Decision Point).

3) PAP (Policy Access Point) takes responsible for writingstrategy and the set of strategy. PDP indexes the storagesthe strategy according to restriction conditions.

4) PDP asks PIP (Policy Information Point) for invokingservice and indexing the related attribute value of subject,resources, environment, and method.

5) Web Services calls the method which is associated withthe request, AOP intercepts the method, and crosscuttingConcerns, and verifies the provided attribute value.

6) PDP sends the decision request to the PEP, and then PEPexecutes the assignment. Finally, PDP decides whetherthe requester is allowed to access according to the request.

VI. CONCLUSION

Trust Based Access Control, combines detecting insider threats andimproving access control. This framework calculates trust values depending on usererrors and operations. The access decision depends onmatching the node trust value and the user trust value.

A policy-based access control model(PBAC)comprehensively describes session property withattribute, realizes an independent policy managementmethod, and issues a new session decision mechanism.Comparative analysis indicates that PBAC is more flexibilityand has the ability of multi-policy supporting.

Distribute push is a scalable push-basedXML access control model. The scaling strategyused is distributing the increasedsystem and management workloads to differentservers and several administrators, respectively.Also, a trust-based policy language is proposed forspecifying access control policies in decentralizedtrust management.

Action and Attribute based Access Control Model analyzeand builds language modelby combining ABAC with SAML and XACML. Comparedwith other Access Control Model, this new Model has someadvantages, like a proper order relation, improved speed to access web services and solving problem of dynamic authorization.

An access control model defines the typeof behavior to solve the problem to the access model toincrease operator efficiency may arise. At the same time,also makes it easier for users to manage authorizationinformation, and by removing unnecessary parsing andDOM retrieval to provide quick access.

VII. REFERENCES

[1] M. Hitchens and V. Varadharajan, “RBAC for XML Document Stores,” in Information and Communications Security, Lecture Note in Computer Science, vol. 2229, S. Qing, T. Okamoto and J.Zhou, Eds. Springer Berlin/Heidelberg, 2001, pp. 131-143.

[2] J. Wang and S. L. Osborn, “A role-based approach to access control for XML databases,” in the Proceedings of the Ninth ACM Symposium on Access Control Models and Technologies, Yorktown Heights, New York, USA, 2004, pp. 70-77.

[3] H. Zhu, R. Jin and K. Lu, “A flexible mandatory access control policy for XML databases,” in the Proceedings of the Second International Conference on Scalable Information Systems, Suzhou, China, 2007.

[4] M. Chagarlamudi, B. Panda and Y. Hu, “Insider Threat in Database Systems: Preventing Malicious Users' Activities in Databases,” in 2009 *Sixth International Conference on Information Technology: New Generations, ITNG '09*, 2009, pp. 1616-1620.

[5] J. S. Park and J. Giordano, “Role-based profile analysis for scalable and accurate insider-anomaly detection,” in *25th IEEE International Performance, Computing, and Communications Conference, IPCCC 2006*, 2006, pp. 463-470.

[6] A. Lin, E. Vullings and J. Dalziel, “A Trust-based Access Control Model for Virtual Organizations,” in *Fifth International Conference on Grid and Cooperative Computing Workshops, GCCW '06*, 2006, pp. 557-564.

[7] F. Almenarez, A. Marin, D. Diaz and J. Sanchez, “Developing a model for trust management in pervasive devices,” in *Pervasive Computing and Communications Workshops*, 2006. *PerCom Workshops 2006. Fourth Annual IEEE International Conference on*, 2006, pp. 267-271.

[8] X. Ma, Z. Feng, C. Xu and J. Wang, “A Trust-Based Access Control with Feedback,” in *International Symposiums in Information Processing (ISIP)*, 2008, pp. 510-514.

[9] X. Han-fa, C. Bing-liang and X. Li-lin, “A mixed access control method based on trust and role,” in *2010 Second IITA International Conference on Geoscience and Remote Sensing (IITA-GRS)*, 2010, pp. 552-555.

[10] S. Singh, “Trust Based Authorization Framework for Grid Services,” *Journal of Emerging Trends in Computing and Information Sciences*, vol. 2, pp. 136-144, 2011.

[11] R S. Sandhu, E J. Coync, H L Fcinstcin et al. “Role-based access control models”, IEEE Computer., vol. 29(2), pp.38~47, 1996.

[12] E. Bertino, S. Castano, and E. Ferrari, "Author-X: AComprehensive system for securing XML documents"IEEE Internet Computing, vol. 5, pp. 21-31, May-June2001.

[13] E. Bertino, B. Carminati, and E. Ferrari, "Access controlfor XML documents and data," Journal of InformationSecurity: Technical Report, vol. 9, pp. 19-34,2004.

[14] G. Miklau and D. Suciu, "Controlling Access to PublishedData Using Cryptography," in 29th InternationalConference on Very Large Data Ba

[15] A. D. Keromytis and M. Smith, "Requirements forScalable Access Control and Security ManagementArchitectures," ACM Transactions on InternetTechnologies" vol. 7, pp. 1-22,2007.

[16] Kraft R, “Research and design issues of access control for net-workservices on the Web,” The 3th International Conference on InternetComputing (IC2002), vol. 3, pp.542-548.

[17] Sandhu R , Ferraiolo D and Kuhn R, “The NIST Model for Role-basedAccess Control: Towards A Unified Standard,” The 5th ACM Workshopon Role Based Access Control, July 2000,pp.47-63.