IC’2001
IC SESSION: XML Data Management and Applications
Paper Number: IC 104
Title: Security Model for XML data
Author(s): C. Ilioudis, G. Pangalos and A. Vakali.

Security model for XML data

C. Ilioudis, G. Pangalos
Informatics Laboratory,
Aristotelian University of Thessaloniki,
Thessaloniki 54006, Greece

A. Vakali
Computer Science Department
Aristotelian University of Thessaloniki,
Thessaloniki 54006, Greece

Abstract: *The significance of XML technology for sharing data over the Internet is being rapidly recognised. In this paper, we examine the security problems related to XML data and present our approach, the XML Security model, for enforcing security policies in XML based Information systems. Our methodology has been based on the study of the XML data model, on the identification of the security requirements of XML Information systems and on the survey of security models which have been proposed to support the conventional data models (relational, object-oriented, hypertext etc). The proposed approach takes into account and exploits the specific characteristics of XML data and incorporates the flexibility of Role based Access Control policies.*

Keywords**:** XML Security, Role Based Access Control.

1. Introduction

XML is a data format for structured document interchange on the Web which is used to create data structures that can be shared between and among disparate and otherwise incompatible systems [1]. The IT community agrees today that XML is to be a common meta-language that will enable data to be transformed from one structure to another [2].

No doubt, XML is state of the art technology. But, how secure are XML applications? This question is very important especially for Information Systems that share sensitive data because security is a key issue, which must be taken into account from the very first steps of the design process of those systems [3].

In this paper, we examine the security problems related to XML data and present our approach for enforcing security policies in XML based Information Systems. Our methodology has been based on the study of the XML data model, on the identification of the security requirements of XML Information systems and on the proposal of a suitable security policy, which is based on Role Based Access Control.

This work has taken place in the context of the Intranet Health Clinic (IHC) project which is an international project involving EU (European Union) member countries and Canada. It concerns a deployment of a Secure Internet-based application for patient care using Internet-based advanced multimedia techniques [4]. The IHC is intended to help patients discharged from a tertiary-level health care organization (e.g. a highly specialized hospital) who must be effectively followed-up by the primary-level physician in a geographically remote area like the many small isolated islands of Greece.

The core of IHC information system is the patient data which are metadata, originate from a legacy Hospital Information system and contain structured information. For this reason, a transferring medium that preserves this structure is clearly beneficial. Also, the diversity of the applications and the need for easy adaptation to different content and presentation requirements are important factors for the IHC project. Thus XML has been chosen in order to implement the data model at the mediator level.

1.1 Research review

Several security models have been proposed in the literature to support the conventional data models, (relational, object-oriented, hypertext etc). Most of those are based on discretionary policies with authorizations specifying the accesses the users are to be allowed on the information, or on mandatory policies that govern the access on the basis of the classification of subjects and objects in the system [5], [6], [7]. A more recent approach in access control are the Role Based policies that regulate the access of users to the information on the basis of the activities the users execute in the system [8], [9]. Other efforts have been

devoted to the investigation of flexible models and mechanisms able to support different authorization policies.

Although several projects for the development XML information systems have recently been carried out, the authorization and access control mechanisms available today are at a preliminary stage. Various proposals are under development however by both industry and academia, and commercial products started becoming available which provide security features in XML environment.

The structure of XML documents makes it easier to add digital signatures or encryption to individual parts of a document as well as to the whole document [10]. However these approaches focus on lower level features, such as encryption and digital signatures and they are not able to support a sophisticated access control mechanism.

The W3 consortium has also released the draft proposal "The Platform for privacy Preferences Project" [11]. This proposal is usually a set of specifications in which enable Web sites to express their privacy practices in a standard format (XML based) that can be retrieved automatically and interpreted easily by user agent. This can be regarded as the first context classification in XML documents but it is not sufficient to enforce a complete access control mechanism, in cases where the definition of security subjects and authorizations are essential.

Bertino et al. also studied a set of authorization and dissemination policies for XML documents [12]. In this study, they proposed a discretionary access control policy with propagation rules. In particular they focus on XML documents that partially conform to a DTD file. Also, Damiani et al. investigated an access control policy for semistructured data that takes into consideration their semantics [13], [14]. They proposed a View based Access Control policy which supports subject's location and a set of propagation rules according to whether the XML document is valid or well-formed. However, all these approaches are based on discretionary policies which are not suitable to manage authorizations in complex systems with many users and many resources [8].

1.2 Contribution of the proposed work

In this paper we proposed a security policy for XML Information systems that takes into account and exploits the specific characteristics of XML data. Our approach incorporates the flexibility of Role based policies and expanding them, using roles - permissions inheritance and propagation of authorizations. In addition it uses constraints in order to cover the negative authorizations, the role cardinality and the dependencies of user location.

We also describe our access control specification language defined in XML, using XML as a security language.

1.3 Outline of the paper

This paper is organized as follows. Section 2 presents the problem of XML security. Section 3 and 4 proposes a suitable security policy which is based on Role Based Access Control and presents an implementation of our approach. Section 5 concludes the paper and outlines future research.

2. XML Security Problem Statement

The proposed so far security policies for the usual relational and object-oriented databases or hypertext documents are not however sufficient to support in a flexible and efficient mode the security requirements in XML Information systems [12]. The above approaches do not consider the particular characteristics of XML data, as for example the partial absence of schema, the existence of connections between data fragments and the data structure of XML documents. XML data is not object-oriented and the data hierarchies represent part of relationships, which require specific techniques different from those applicable to the hierarchies in the object-oriented model. Also, the lack of a management information system able to support access and integrity constraint rules in XML environment introduces new protection requirements [15].

The main protection requirements for XML documents that influence the definition of the policies for their access, are related to the following characteristics of XML and the subjects accessing them:

Access control. There is a need to protect XML resources against unauthorised access. The access control components decide whether

a subject can access a particular resource (object). This functionality is related to both the secrecy and integrity of information. An access control policy in the XML data model is differentiated from the corresponding policies on usual data models because it can be defined at schema level (DTD), or of instances of it, or on specific XML documents. Thus an access control policy for XML Information systems can be enforced on DTD level, which applies to all valid documents that are instances of the DTD. Whereas for well-formed XML documents an access control policy must take into account the fact that a DTD is not available. In this case, it should be possible to define policies based on the classification of a well-formed document, by finding the best matching DTD. Alternatively, explicit policies can be defined for each document separately.

Granularity. XML, in contrast to hypertext, provides a clean separation between the structure and layout of a document. It is therefore possible to define access restrictions in a fine granularity, directly on the structure and content of documents. Granularity considerations in XML Information systems call for the support of an Access Control Policy on individual elements (fine-grained) as well as on the whole portions of a document (coarse-grained).

Propagation. The XML data model is essentially an ordered labelled tree and the data exists in an ordered hierarchy. Thus permissions that specified on an object (e.g. element) can be propagated to the nested objects (e.g. sub-elements and text nodes), too. According to the propagation attribute, policies which specified for a protection object at a given granularity level (e.g. a document) propagate to all protection objects that are semantically related to it through a data hierarchy relationship.

Negative and positive authorizations. The existence of propagated permission in XML documents and the use of authorizations at a coarse-grained granularity level (e.g. on whole document) would prove limited without the support of exceptions. The support of both permissions and denials allows the same security requirements to be represented with two authorizations: a positive authorization on the whole document, and a negative authorization on a specific element/attribute.

Integrity constraints mechanism. In XML Information systems the need to support integrity constraints does not differ fundamentally from systems in the database sector. Thus an XML system must encapsulate entity, referential, and constraints specifying the semantics of object identities. These constraints are useful both for native XML documents and to preserve the semantics of data originating in relational or object databases (Legacy DB). The standard XML schema Language (i.e. DTD) supports only reference constraints, and the lack of a management system able to enforce integrity constraints makes the security problem much harder and complicated.

Presence of other data formats. The fact that XML can be delivered, together with html data, multimedia objects and scripts, causes extra security needs. In the case of multimedia objects (e.g. images, sounds, video), security requirements demand the definition of authorizations, for example on a part of a video object, thus an XML security policy have to integrate the existing security policies for multimedia objects.

Administration of authorizations. The administration of authorizations of a security policy for XML documents is differentiated from those of the conventional data where are based on ownership or centralized administration. In XML information systems, as well in hypertext, the data warehouses on several nodes owned by different users. A flexible administration policy for XML information system must be consider that the system is an interconnecting collection of objects where operates through web.

User authentication. The verification of the identity of users is of crucial importance in XML information systems due to the inherent ability of these systems to allow access to remote resources via physically untrusted communication environments. It essential for XML Systems to support the traditional “One-way authentication”, when a client authenticates itself to a server, and the most restricted “Two-way authentication using Trusted Third Party (TTP)” which provides authentication for clients and servers through the credentials from the Trusted Third Party.

Document and signer authentication. An XML information system, usually, operates over untrusted networks such as the Internet. Thus an effective security policy for XML must

provide evidences that the XML document or fragment is what it claims to be. Signer authentication refers to the ability to identify who signed an XML document, or fragment. The prevailing electronic method for achieving that is the digital certificates technology.

Communication security. Communication over insecure links is typically the case in an XML information system. Thus there is a need to employ mechanisms that provide the required communication secrecy and integrity. The provision of this functionality is based on encryption.

3. A security approach under XML Information Systems

3.1 The XML security model

A security policy for an XML Information system, which is distributed over the Internet, must encapsulate flexibility, decrease the security administration overhead and enhanced protection from unauthorised information disclosure. As seen earlier, the three major types of security policies that have been proposed and are usually used in computer systems are not sufficient to support the security requirements in XML Information system in a flexible and efficient mode.

There is therefore a need for a new security policy suitable for XML environments. The proposed approach, the XML Security model, takes into account and exploits the specific characteristics of XML data. The proposed XML security model incorporates the flexibility of Role based policies, using roles, inheritance and permissions on objects and enforces negative permission and subject location constraints and propagation of authorizations. In the rest of the section we define the basic principles of our XML security policy.

3.2 Principles of XML security policy

The development of an access control system requires the definition of the subjects and objects against which authorizations must be specified and access control must be enforced. In order to define the XML security model it is necessary to define a number of essential aspects. We have developed the following DTD for representing the policy schema:

```
<! --
        Document Type Definition for Role Based
Access Control policy adapted for XML documents
-->

<!-- element  security objects. Security Objects are
identified by XPath expression -->
<!ELEMENT security_objects (object)+>
<!ELEMENT security_object EMPTY>
<!ATTLIST security_object path CDATA
#REQUIRED>
<!-- element type operation. There are four types-->

<!ELEMENT operation EMPTY>
<!ATTLIST operation type (read|write|create|delete)
#REQUIRED>

<!-- element  privileges-->
<!ELEMENT privileges (privilege)+>
<!ELEMENT privilege (operation,security_object)+>
<!ATTLIST  privilege name CDATA #REQUIRED>

<!-- element type role_hierarchy-->
<!ELEMENT role_hierarchy (role)+>
<!ELEMENT role (name,cardinality?,(parent_role?)*,
(child_role?)*)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT cardinality (#PCDATA)>
<!ELEMENT parent_role (#PCDATA)>
<!ELEMENT child_role (#PCDATA)>
<!ATTLIST role role_id ID #REQUIRED>

<!--element users -->
<!ELEMENT users (user)*>
<!ELEMENT user (#PCDATA)>

<!--user - role assigments with location constraints.
                Locations are expressed in Internet
address format,
                e.g. med.auth.gr and declare only
permissible locations-->
<!ELEMENT security_subjects (security_subject)+>
<!ELEMENT security_subject
(user,(allowed_role,location_constraint)*)>
<!ELEMENT allowed_role EMPTY>
<!ATTLIST allowed_role r IDREF #REQUIRED>
<!ELEMENT location_constraint EMPTY>
<!ATTLIST location_constraint Internet_address
CDATA #REQUIRED>
<!--End of the DTD -->
```

3.2.1 Security objects

Security *objects* are the targets of the security protection. Our model supports different levels of granularity. Thus as seen below security objects are the elements and attributes or the entire document. The multimedia objects (e.g. images, sounds, video) are considered as passive components and forms a security objects, too. We use the Xpath Language in order to identify the security objects within a document [16].

3.2.2 Permitted actions

An action (or operation) indicates the operation to be performed on one or more objects. We

limit our consideration to read operations. The support of other operations, like write, update, and delete does not complicate the authorization model. However, the support for such actions arises integrity constraints problems, which have not yet been defined.

3.2.3 Permissions

Permission indicates the right to perform a specific operation on a particular object. Permissions in our model can be fine-grained (e.g. at the element level) or coarse-grained (e.g. at the level of entire document). They can be defined in terms of the permitted operations such as read, write, update, and delete.

As already mentioned, the XML data model can be simulated as an ordered labeled tree and the data exists in an ordered hierarchy. Thus the permissions can be propagated. More specifically, permissions specified on an object (e.g. element) can be defined as applicable to the specific object only (*exclusive* permissions) or to the nested objects e.g. sub-elements and attributes (*propagate* permissions). The explicit permissions on an element apply only to this element and not to those of its sub-elements. The propagated permissions are delegated to all nested elements from the tree hierarchy.

In order to add flexibility in our model we introduce the *negative permissions*. The RBAC variations are based on positive permissions that confer the ability to do something on holders of the permission. Also the use of negative permissions can be very confusing, especially in presence of general hierarchies. Thus in our model we choose to cover the negative permissions using constraints.

A fragment of the XML document which conforms to the above DTD policy schema is given below. It says, for example, that the privilege named "Personal data" has read permission on element /Patient_Record/Personal_data.

```
<?xml version='1.0'?>
<privileges>
 <privilege name= "Personal data">
   <operation type="read"/>
   <security_object path =
         "/Patient_Record/Personal_data"/>
 </privilege>
<privilege name= "Clinical test">
         <operation type="read"/>
         <security_object path
="/Patient_Record/Complaint/Diagnosis"/>
         <operation type="write"/>
         <security_object path
="/Patient_Record/Clinical_test"/>
</privilege>

</privileges>
```

3.2.4 Security subjects

In our model, *subjects* can be referred to on the basis of their identities and on the associated Role. Roles are associated with each individual who might have a need to access information. Each role defines a specific set of permissions that the individual acting in that role may perform. We support partial order roles hierarchies, thus senior role inherits the permissions from the junior role and so on. Also, a session is a mapping between a user and an activated subset of the set of roles the user is assigned to. Once an individual has been properly identified and that identification authenticated, the individual chooses a role that has been assigned through session and accesses information according to the privileges assigned to the role. A fragment of the XML document which conforms to the above DTD policy schema is given below. It says for example that the user “chr” can operate as “doctor” from all the network *.med.auth.gr.

```
<?xml version='1.0'?>
 <security_subjects>
  <security_subject>
   <user>chr</user>
   <allowed_role r="doctor"/>
    <location_constraint Internet_address
         ="*.med.auth.gr"/>
  <user>chr</user>
  <allowed_role r="Head_Dep"/>
   <location_constraint Internet_address
         ="office1.med.auth.gr"/>
   </security_subject>
 </security_subjects>
```

3.2.5 Security Constraints

Constraints in our model may be associated with the user-role assignment, or with the permission to role assignment. We use the constraints in order to cover the negative authorizations, the Role cardinality, the user to object relationships and the dependencies of user location.

Negative permissions Our model supports negative permissions. For example a positive authorization is defined on the whole document,

and a negative authorization on a specific sub-element. This type of constraints is enforced on permission-to-role assignment relation PA, and mentioned as an exception of a set of object.

User location Constraint. We consider that our system operates over a distributed Internet environment thus authorizations is important to expressed on host location, too. The location can be expressed using IP address e.g. ahepa.med.gr or patterns, by using the wild card character *, e.g *.med.gr or *.gr. The location constraint is enforced on user-to-role assignment UR.

Role Cardinality constraint. Another type of constraint supported from our model is the cardinality of a role. Some roles in an organization may be occupied by a certain number of employees at any given time. For example, consider the role of head of the clinic; only one individual may assume the responsibilities of the head.

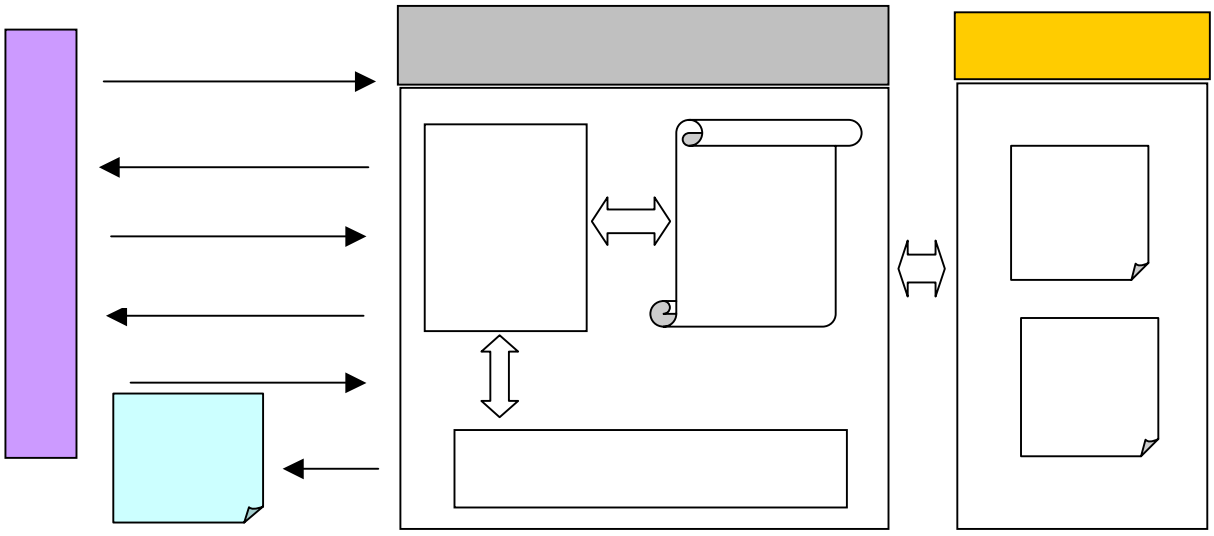
4. Application – Implementation Guidelines.

Our access control enforcement is performed on the application layer, according to the three-tier architecture. It operates as a secure mediator between the client-tier and the data-tier. It mediates all the requests to XML documents and evaluates them against the access control policy. For each request it produces a subset of the document composed only of the data that the requester can access, through the user-to-role, the role-to-permission assignment and the corresponding constraints. This implementation prevents the accidental transfer to the client of information he is not allowed to see. However, our approach increases a bit the throughput of the network.

The computation of the privileges of each role is done through a procedure which traverses the tree according to the permissions and the negative permission constraints. We enforce the principle of 'least privilege', and implement a restricted policy that if an authorization is not explicitly permitted, then prohibited. The end result of this procedure is a filtering tree which is the authorized tree of the XML document.

This work has taken place in the context of the Intranet Health Clinic (IHC) project. The security objects in the IHC information system are the targets of the security protection. These are the XML objects (elements or attributes) contained in the XML data files. A DTD file accomplishes the XML files (valid XML files). In the IHC access control policy, permissions are authorised for roles and roles are authorised for users. Permission is an approval of a particular operation to be performed on one or more objects. For example the Role doctor can read all the patient data except the administrative data.

The IHC information system accomplishes the 3-tier architecture and the access control is enforced on the application tier. The figure 1 shows the information flow and the interaction between the tiers. After the identification and authentication of the user he (or she) has to



choose an appropriate role. A privilege is assigned to role and after the transformations through the DOM API, the XML parser return to the user the authorized XML fragment of the XML document.

5. Conclusions

In this paper, we study the XML data model, the security requirements and the security policies for XML Information systems and finally we proposed a suitable security policy which is defined in XML, using XML as a security language.

We achieved to implement a secure XML based Information system with flexible security administration. The next step of this study is to enrich our system with entity and referential integrity constraint mechanism and to enforce the digital signature technology in order to achieve, document, user and signer authentication.

6. References

[1] T.Bray, J. Paoli, and C. Spenger-McQueen. Extensible Markup language (XML) 1.0 W3C Recommendation. Available: www.w3.org/TR/1998/REC-xml-19980210, February 1998.

[2] Land, Steve. XML:the ASCII of the future. Microsoft, May 1999.

[3] Pangalos, Gritzalis, Khair, Bozios. Improving security of medical database Systems. IFIP/SEC 1995.

[4] IHC project. Available: http://www.biomed.ntua.gr/intraclinic, 1999.

[5] Castano S., Fugini M., Martella G., Samarati P.. Database Security. ACM Press Books, 1994, ISBN: 0201593750

[6] M.B. Thuraisingham. Mandatory Security Object-Oriented Databases. Proceedings of OOPSLA 1989.

[7] P. Samarati, E. Bertino, S. Jajodia. An Authorization Model for a Distributed Hypertext System. IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No 4, August 1996.

[8] R. Sandhu, E.J. Coyne, H.L. Feinstein, and C.E. Youman. Role Based Access Control Models. IEEE Computer, 29(2), February 1996.

[9] D. Ferraiolo, J. Barkley, and R. Kuhn. A Role Based Access Control Model and Reference Implementation within a Corporate Intranet. ACM Transactions on Information Systems Security, Volume 1, Number 2,1999.

[10] Joseph, Reagle. W3C recommendation XML-Signature. Available: http://www.w3.org/Signature , 1999.

[11] Marchiori M., The Platform for Privacy preferences 1.0 Specification. W3C May 2000. Available at http://www.w3.org/TR/P3P/.

[12] Bertino, Castano, Ferrari, Mesiti. Controlled Access and Dissemination of XML Documents. WIDM 1999.

[13] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi and P. Samarati. Securing XML Documents. In proc. of EDBT 2000, Germany. Springer Verlag, in LNCS 1777, 2000.

[14] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi and P. Samarati. Regulating Access to semistructured information on the Web. WCC-SEC 2000, China, 2000.

[15] Fan W., Simeon J., Integrity Constraints for XML, Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, 2000.

[16] J. Clark,S DeRose, World Wide Web Consortium (W3C). XML path Language (Xpath) ver. 1.0, Noevember 1999, http://www.w3.org/TR/1999/REC-xpath-19991116.