Application of Criterion-Based Multilayer Access Control to XML Documents

Leon Pan
Department of Computer Information Systems, University of the Fraser Valley, Abbotsford, BC, Canada

Abstract - An approach is proposed to address the fine-grained multilayer access control requirements of XML files. The system is based on a set of predefined security criteria and security criterion expressions which serve as keys and locks respectively. The security criterion expressions are defined in the XML schema and embedded into individual XML files automatically. The XML files are enhanced to be secure XML files by embedding a set of security criterion expressions while users are enhanced to be secure users by associating with security criterion subsets. The embedded security criterion expressions are evaluated by the elements in the user’s security criterion subset to determine the accessible part(s).

Keywords: Criterion-Based Multilayer Access Control, Security Criterion Expression, Security Criterion Subset, XML security, Secure XML files, Secure users

1 Introduction

XML [17] is becoming increasingly popular in describing, managing, storing, and sharing data. It provides means of defining a vocabulary of element tags and attributes to structure information of interest. Anyone can define his/her own markup tags and attributes which describe the data and are combined with data to form XML documents. Because sharing and distributing data over Internet becomes more and more common, XML document security becomes more important and the needs of the effective and efficient XML security systems arise.

The author proposes a method of introducing security attributes to every element of the XML documents and applying the criterion-based access control to support XML document security. The values of the security attributes, which are defined in XML schema as the fixed values, are in the form of security criterion expressions. These security criterion expressions are evaluated to determine whether a user has the access to the XML document as a whole and its corresponding elements.

The rest of the paper is organized as follows. Section 2 briefly overviews previous works of the XML security. In section 3, the criterion-based access control is reviewed. Many important concepts of this model are discussed as well. Section 4 presents the details of embedding security information into XML documents and defining the security attributes through XML schema. Section 5 provides the further discussions compatible to the access control policy. In [1, 4], queries are rewritten by combining with the access control policy in order to enhance the performance.

2 Overview of XML security

XML security has attracted increasing attention of researchers and many approaches and models [1, 3-8, 12-15, 18-20] has been proposed and developed, including encryption, digital signature, XML key management, authentication and authorization, and access control. Among these approaches, the access control method is very popular because it is convenient to take advantage of the characteristics of XML document when enforcing the authorization rules.

Some XML access control approaches [4, 6, 7, 8, 14, 19] are based on views that are created to enforce the authorization rules. A specific view created for a user specifies the portion of XML document that is accessible to that user. The approach proposed in [13] supports the access control by filtering out the queries that are not
A security criterion is a criterion used to both specify the user’s security attributes and define the object’s (and the sub object’s) security attributes. Each security criterion is represented by a symbol $s_i$. Security criteria are abstracted from authorization rules. From the whole set of authorization rules, a set of security criteria $s_1, s_2, \ldots, s_n$ in an application domain can be abstracted. The collection of all security criteria, their complement counterparts ($s^c_i$), constant false F and true T form a set which is called the security criterion set.

A user may have more than one security attributes. So, several security criteria are often required to specify the user’s security attributes. The set composed of these security criteria is called a security criterion subset (SCSS). When a user is associated with a security criterion subset (SCSS), he/she is enhanced to be a secure user (SU).

To precisely reflect authorization rules, objects (and the sub objects), as well as their security attributes, are defined by security criterion expressions. A security criterion expression (SCE) is a Boolean expression in terms of security criteria. A Boolean expression is considered to be a security criterion expression iff it reflects one or more authorization rules. Following are legal security criterion expressions:

1. A constant true, T, or false, F
2. A Boolean expression derived directly from an authorization rule
3. Logical “OR” of (1) and (2)

The constant true, T, or false, F, represents special cases. When an (sub) object needs unconditional protection, its corresponding security criterion expression should be the constant true, T. On the other hand, when the security criterion expression is the constant false, F, the related (sub) object is accessible in any circumstances.

To support fine-grained multilevel access, in an object, each part (i.e. sub object) with different security attributes and thus of different security levels has an embedded security criterion expression to specify its security attributes. A sub object with an embedded security criterion expression is a secure sub object.

3.2 Security criterion abstraction and secure object and secure user generations

A systematic method has been developed to abstract security criteria from authorization rules, to transform authorization rules into security criterion expressions, and to generate security criterion subset based on the authorization rules [11]. For details, please reference [11].

Example 1: There is a XML database which contains XML documents of patients’ medical history records. Each XML document includes insensitive information such as recuperation information and sensitive information, including personal information, nursing information, diagnosis information, and treatment information. Suppose the following authorization rules are applied to the XML documents.

1. Non-professions who are not registered have no access to the files
2. Non-professions don’t have access to the personal information and professional information
3. Non-professionals and nurses have no permission for diagnosis and treatment information

It is easy to generate the components following the method provided in [11]. Tables 1, 2, and 3 show the generated security criteria, security criterion expressions for the sensitive sub objects, and the subset for each group of users, respectively.

Table 1 Generated security criteria

<table>
<thead>
<tr>
<th>Security criterion symbol</th>
<th>Security criterion meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_1$</td>
<td>Non-professionals</td>
</tr>
<tr>
<td>$s_2$</td>
<td>Not registered users</td>
</tr>
<tr>
<td>$s_3$</td>
<td>Nurses</td>
</tr>
</tbody>
</table>

Table 2 Security criterion expressions, corresponding authorization rules, and the applied (sub) objects

<table>
<thead>
<tr>
<th>Security criterion expression</th>
<th>Authorization rules</th>
<th>(Sub) Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_1 \cap s_2^c$</td>
<td>Authorization rule 1</td>
<td>Whole file</td>
</tr>
<tr>
<td>$s_2$</td>
<td>Authorization rule 2</td>
<td>Personal information</td>
</tr>
<tr>
<td>$s_3$</td>
<td>Authorization rule 2</td>
<td>Professional information</td>
</tr>
<tr>
<td>$s_2 \cup s_3^c$</td>
<td>Authorization rule 3</td>
<td>Diagnosis information</td>
</tr>
<tr>
<td>$s_1 \cup s_3^c$</td>
<td>Authorization rule 3</td>
<td>Treatment information</td>
</tr>
</tbody>
</table>
*Note: The authorization rules determine whether intersection or union is used. For example, authorization rule 1 indicates that non-professionals ($S_1$) who are not registered ($S_2$) do not have access. On the other hand, $S_1 \cup S_3$ means either non-professionals ($S_1$) or nurses ($S_3$) have access to the diagnosis information, which reflects authorization rule 3 accurately.

Table 3 Security criterion subset and the corresponding user groups

<table>
<thead>
<tr>
<th>Security criterion subset</th>
<th>User group</th>
</tr>
</thead>
<tbody>
<tr>
<td>{S_1}</td>
<td>non-professionals</td>
</tr>
</tbody>
</table>
| \{S_1, S_2\}             | Not registered non-
                           | professionals          |
| \{S_3\}                  | Nurses                   |

3.3 Achieving fine-grained access control

In the Criterion-Based Access Control model, an (sub) object’s security attributes and security level are implied by indicating users who do not have access rather than explicitly defining them. The system becomes simpler because one mechanism is used to define both the user’s security attributes and the (sub) object’s security attributes. The security criterion expressions embedded in a secure object can be regarded as locks, while the security criteria in the security criterion subset can be considered as keys. When a secure user accesses a secure object, he/she uses the available keys to actuate the locks. Whether the secure user is allowed to access the secure sub object depends on the state of the corresponding locks.

A security criterion expression is evaluated in the following two steps. First, substitute all the security criteria in the security criterion expression with true, T, or false, F. If replace those security criteria in the security criterion expression with true if they also appear in the secure user’s security criterion subset. Otherwise, replace them with false. Second, the security criterion expression is evaluated according to the normal evaluation procedure in Boolean algebra. The evaluation value T of a security criterion expression implies that users with security attributes specified by these security criteria are not allowed to access the corresponding secure sub object. On the contrary, a false evaluation value, F, of the security criterion expression implies that the security criterion expression does not prevent these secure users from accessing this sub object.

4 Criterion-Based fine-grained XML document access control

An XML file forms a tree structure that starts at the root element and branches to the leaf elements. Each file has a unique root element which is the parent or grandparent of all other elements. To support fine-grained access control, security information must be embedded into every element. This paper proposes a novel idea to achieve fine-grained access control by making use of the properties of XML.

4.1 Introducing security attributes

Security information can be expressed in form of security criterion expressions which reflect the relevant security policy. Security criterion expressions are embedded into every element to specify the security features of the element and sensitive attributes (Sensitive attributes access control will be discussed in another paper). If an element and its descendants are accessible to public, a constant false, F is embedded. Otherwise, a specific security criterion expression is generated and embedded according to the security policy. The embedded security criterion expression specifies the users who do not have access to this element.

Figure 1 shows a simplified medical file which includes both sensitive and insensitive information about a patient’s medical records. A specific security criterion expression, which extracted from relevant authorization rules, is embedded into each node in the file. When a user proposes a request to the file, his/her security criterion subset is used to evaluate these embedded security criterion expressions. If the evaluation value is true, the protection conditions for the corresponding element are satisfied. Thus, the user has no access to the node. If the evaluation is false, the user has access to the current element. However, the descendants’ security criterion expressions are still need to be evaluated to decide the accessibility of each descendant element.

For example, if a user’s security criterion subset is \{$S_3$\} (Therefore, he/she is a nurse), only the evaluation values of diagnosis and treatment are true, which results in inaccessibility of these nodes. This conforms to the authorization rule 3.

Although it is possible to embed the security criterion expressions manually, it is very time-consuming, labor-intensive, and error prone to do so. To enforce the security policy more efficiently and consistently, an approach is proposed in this paper which defines the XML documents’ security attributes by using enhanced XML schema.
4.2 Defining security attribute in XML schema

An XML schema [2, 16, 21] describes the structure of an XML file. It defines the legal building blocks of an XML file, including elements, attributes, child elements of an element, the number and order of the child elements, element with or without content, data types for elements and attributes, and default and fixed values for elements and attributes. In an enhanced schema, a security criterion expression can be defined similarly for every element or attribute as a security attribute with the fixed value which is inserted into corresponding element of an XML file automatically when the XML file is created. Again, only the method of defining security attributes for elements will be discussed in this paper.

Therefore, each element of the enhanced schema is accompanied with a security attribute whose value is defined as fixed value of the security criterion expression extracted from the relevant authorization rules. This security attribute specifies the protection condition for the current element and its descendants. A special security criterion expression, constant false $F$, is embedded if the corresponding element does not contain any sensitive information and thus is accessible to everyone according to the security policy.

```
<?xml version="1.0"?>
<smr:medical_history security="*" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.secure.medical.records/redords smredords.xsd"
xmllns:smr="http://www.secure.medical.records">
  <smr:personal_info security="*">
    <smr:patientname security="*">John Smith</smr:patientname>
    <smr:address security="*">1234 66th Ave. Hammond IN</smr:address>
    <smr:phone security="*">219-555-5555</smr:phone>
  </smr:personal_info>
  <smr:medical_records security="*">
    <smr:medical_record security="*">
      <smr:non_professional_info security="*">
        <smr:visitdate security="*">02-28-2012</smr:visitdate>
        <smr:recuperation security="*">Exercise regularly</smr:recuperation>
      </smr:non_professional_info>
      <smr:professional_info security="*">
        <smr:diagnosis security="*">diagnosis info</smr:diagnosis>
        <smr:treatment security="*">treat info</smr:treatment>
        <smr:nursing security="*">nursing info</smr:nursing>
      </smr:professional_info>
    </smr:medical_record>
  </smr:medical_records>
</smr:medical_history>
```

Figure 1 An XML document of a patient’s medical records with security information embedded

The schema shown in figure 2 defines the elements and their security attributes for XML files of simplified patient’s medical records discussed in 4.1 subsection. By using this schema, the required security criterion expressions are embedded into every element for any relevant XML documents.
<xs:element name="medical_history">

  <xs:complexType>
    <xs:sequence>
      <xs:element name="personal_info">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="patientname" type="string1Type"/>
            <xs:element name="address" type="string1Type"/>
            <xs:element name="phone" type="string1Type"/>
          </xs:sequence>
        </xs:complexType>

        <xs:attribute name="security" type="xs:string" fixed="s1"/>
      </xs:element>

      <xs:element name="medical_records">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="medical_record" maxOccurs="unbounded">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="non_professional_info">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="visitdate" type="visitdateType"/>
                        <xs:element name="symptom" type="string2Type"/>
                        <xs:element name="recuperation" type="string2Type"/>
                      </xs:sequence>
                    </xs:complexType>

                    <xs:attribute name="security" type="xs:string" fixed="s1 ∩ s2"/>
                  </xs:element>

                  <xs:element name="professional_info">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="diagnosis" type="string3Type"/>
                        <xs:element name="treatment" type="string3Type"/>
                        <xs:element name="nursing" type="string1Type"/>
                      </xs:sequence>
                    </xs:complexType>

                    <xs:attribute name="security" type="xs:string" fixed="s1"/>
                  </xs:element>

                  <xs:element/>
                  <xs:element/>
                </xs:sequence>
              </xs:complexType>

              <xs:attribute name="security" type="xs:string" fixed="s1 ∩ s2"/>
            </xs:element>
          </xs:sequence>
        </xs:complexType>

        <xs:attribute name="security" type="xs:string" fixed="s1 ∩ s2"/>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
5 Discussions and Conclusion

This paper proposes an approach to support fine-grained XML document access control by using a set of predefined security criteria. Security criterion expressions are extracted from the system authorization rules and are embedded into every element of the document. These security criterion expressions serve as locks. On the other hand, every user is assigned a security criterion subset the elements of which serve as keys. Keys are used to actuate the locks and the accessibility for a user to an XML document as well as its elements is determined by the status of the locks. To enforce the security policy systemically and consistently, the author developed a method in which security attributes are defined in enhanced XML schema and the security criterion expressions are embedded into XML documents automatically.

There are several advantages to define the security attributes in an enhanced schema. First, by embedding security criterion expressions automatically, security policy can be enforced consistently and errors are prevented. Secondly, security officers are relieved from the heavy burden of embedding the security information for every XML file and thus system performance is improved. Finally, the security level of the XML documents is further improved because any malicious modification to the security attributes (security criterion expressions) will be discovered when the documents are validated. As a result, the XML databases enhanced by this method enjoy both high level of security and high performance.

6 References


XML Database Symposium (XSym), Berlin, Germany, 2003.


