第37卷 第14期 计 算 机 工 程 2011年7月
Vol.37 No.14 Computer Engineering July 2011

·安全技术· 文章编号：1000—3428(2011)14—0123—02 文献标识码：A 中图分类号：N945

基于BLP模型的XML访问控制研究

叶春晓，尉法文

(重庆大学计算机学院，重庆 400044)

摘 要：在高安全领域，XML文档中可能包含不同程度的敏感信息，需要受到强制访问控制策略的保护。为确保高敏感数据的完整性，在BLP模型的基础上对主体和客体的安全标签进行改进，提出EBLP模型，讨论在该模型下的安全标签分配问题，研究该模型的体系结构并给出XML文档的访问控制算法。

关键词：可扩展标记语言；安全标签；强制访问控制；EBLP模型；访问控制

Research on XML Access Control Based on BLP Model

YE Chun-xiao, YU Fa-wen

(College of Computer Science, Chongqing University, Chongqing 400044, China)

【Abstract】In the high security fields, XML documents may include information at different levels of sensitivity. It should be protected by Mandatory Access Control(MAC) policy. In order to maintain the integrality of data at high levels of sensitivity, the security labels of subjects and objects are improved. An extended MAC model called EBLP is proposed on the basis of BLP model. Security label assignment are discussed. The architecture and the access control arithmetic used to implement the fine-grained EBLP model are discussed.

【Key words】eXtensible Markup Language(XML); security label; Mandatory Access Control(MAC); EBLP model; access control

DOI: 10.3969/j.issn.1000-3428.2011.14.040

1 概述

可扩展标记语言(eXtensible Markup Language, XML)[1]作为描述互联网上结构化信息和内容的标准，由于其简单性、标准性以及丰富的数据结构，在电子商务、网络出版和移动通信等领域得到广泛应用。越来越多的信息以XML文档的形式存储，如何实现XML文档的安全访问控制也成为一项重要研究内容。

到目前为止，XML文档安全性的研究主要集中在自主访问控制、基于角色访问控制和基于视图访问控制技术。但是自主访问控制模型有内在的缺陷，不能抵御木马的攻击。基于角色的访问控制模型[2-3]在多安全级别环境中不能很好地对信息进行访问控制。基于视图的访问控制模型[4]在XML文档结构发生变化时，所有存储的视图都需更新，降低了系统的性能。而对XML文档的强制访问控制的研究较少，BLP[5]模型是一种基于规则的强制访问控制模型，能满足对数据保密性要求特别高的需求，但不能保证敏感数据的完整性。本文在BLP模型的基础上对主体和客体的安全标签进行改进，使其能够支持系统完整性，给用户提供更大的灵活性。

2 扩展的BLP模型(EBLP)

强制访问控制(Mandatory Access Control, MAC)策略一般是根据主体和客体的安全属性来管理用户对信息的访问。主体是那些访问信息的主动实体，客体是存放信息的被动实体。强制访问控制主要包括BLP模型、Dion模型等，其中BLP模型是高安全等级系统中最常用的强制访问控制模型。

2.1 BLP模型

在BLP模型中，系统中的主体和客体均被赋予一个安全标签，安全标签由密级和范围组成。密级是密级集合中的一个元素，最早模型定义的时候密级共分4级：绝密(Top Secret)，机密(Secret)，保密(Confidential)，公共(Public)，但是可以扩展成更多密级，只要这些密级形成一个可以比较的序列。范围是一些没有继承关系的元素集合的子集，它的元素依赖于所考虑的环境和应用领域。

BLP模型可以看作是矩阵模型的扩展，其基本思想是确保信息不向下流动，从而保证系统内的信息是安全的。BLP模型的信息不向下流动是通过下面2个规则来保证的。

(1)简单安全特性：当且仅当$L(s) \geqslant L(o)$时，读操作被允许；

(2)星(*)特性：当且仅当$L(s) \leqslant L(o)$时，写操作被允许。

2.2 BLP模型的改进

BLP模型是一个很安全的模型，能满足对数据保密性要求特别高的需求，但低安全级用户能够篡改高级别数据，数据的完整性难以保证。用户需要的读写权限往往不相同，一般要求读权限要比写权限大。因此，对主体和客体的安全标签进行改进以提高系统的可用性。

(1)主体安全标签的改进。将主体的读写密级分开，分别标记为C_r(读密级)和C_w(写密级)。

XML文档是半结构化的数据，为了实现主体的最小特权原则，将主体的范围改进为{XML文档名称:XPath[6]}的集合。例如范围可以是{salary.xml:/salary/Employee}，表示的范围是根节点为“salary”的salary.xml文档中Employee的节点或节点集以及salary节点。可以通过比较主体请求的客体是否在主体的操作范围内决定主体是否有权限对客体进行操作。

(2)客体安全标签的改进。将客体安全标签中的范围删除，使其只包含密级。

EBLP模型采用“上级读，平级写”的读写规则来保证

基金项目：重庆市科委自然科学基金资助项目(CSTC,2008BB2320)

作者简介：叶春晓(1973－)，男，教授、博士，主研方向：网络安全，数据库技术，软件工程；尉法文，硕士研究生

收稿日期：2011-03-09 **E-mail：**yecx@cqu.edu.cn

敏感数据的机密性、完整性。因此，只有主体的读密级不小于客体的密级主体才能读取客体的数据；主体的写密级等于客体的密级，主体才能向客体中写数据，这里的写操作是广义的，包括更新、删除、新建操作。

EBLP 模型可以满足高密级用户向非敏感客体写数据的合理要求，但是也可能造成高密级的信息泄漏给低密级的主体，这就要求管理员合理地分配安全标签。主客体安全标签的改进，给用户提供了更大的灵活性，增强了系统的可用性，同时也增加了实现的复杂性。

3 安全标签的分配

安全标签的分配包括 XML 文档、XML 模式文件及用户安全标签的分配。用户安全标签的分配包括用户读写密级及范围的分配。为了方便说明安全标签的分配，引入文献[7]中 XML 文档模型的定义。由于 XML 模式文件符合 XML 的语法，因此 XML 文档模型的定义同样适用于 XML 模式文件。

3.1 XML 文档模型的定义

定义 添加了标签的 XML 文档是一个九元组 $XDoc=(V_e, v_r, V_a, N_s, L_s, subelem, attrs, n\text{-}ame, label)$，其中，$V_e$ 是文档中所有元素的集合；v_r 文档的根元素，因为 v_r 也是文档中的元素，所以 $v_r \in V_e$；V_a 是文档中所有属性的集合；N_s 是名字集合，既包括元素的名字，也包括属性的名字；L_s 是所有标签的集合；$subelem$ 是一个二元关系，$subelem \subseteq V_a \times V_e$,$(e_1, e_2) \in subelem$ 表示 e_1 是 e_2 的子元素；$attrs$ 是一个二元关系，$attrs \subseteq V_a \times V_e$,如果 $a_1 \in V_a$ 且 $e_1 \in V_e$,那么$(a_1, e_1) \in attrs$ 表示 a_1 是 e_1 的属性；$name$ 是一个二元关系，$name \subseteq N_s \times (V_a, V_e)$，如果 $n_1 \in N_s, v_1 \in (V_a \cup V_e)$，那么$(n_1, v_1) \in name$ 表示 n_1 是 v_1 的名称；$label$ 是一个二元关系，$label \subseteq L_s \times (V_a \cup V_e)$，如果 $l_1 \in L_s$, $v_1 \in (V_a \cup V_e)$，那么$(l_1, v_1) \in label$ 表示 l_1 是 v_1 的标签，也可以表示为 $l_1 = \lambda(v_1)$。

3.2 用户标签的分配

在 EBLP 模型中，管理员给主体分配 2 个密级，即读密级和写密级。为了防止低密级用户恶意篡改敏感信息，规定在用户的安全标签 $L=(C_r, C_w, G)$中，C_r(读密级)必须不小于 C_w(写密级)。

3.3 XML 文档标签分配及规则

XML 文档中有大量的元素和属性，如果给每一个元素和属性都指定一个标签，那么管理人员的工作量是巨大的。可以利用 XML 的特点来减少管理人员的工作量。管理人员只需要给模式文件和 XML 文档中的一些元素指定标签，这些标签就会被模式文件对应的 XML 文档中的元素或者属性所继承，或者这些标签从祖先节点传播给子孙节点和属性。

规则 1 标签从模式文件到 XML 文档的传播。

在 $XDoc$ 和模式文件 $XSch$ 中，文档实例 $i \in XDoc.V_a \cup XDoc.V_e$，模式实例 $s \in XSch.V_a \cup XSch.V_e$。如果 i 是 s 的一个实例，那么 $L(i)=L(s)$。

规则 2 标签从元素到子元素或属性的传播。

在一个 $XDoc$ 中，$e_1, e_2 \in V_e$, $a_1 \in V_a$,$(e_1, e_2) \in subelem$, $(a_1, e_1) \in attrs$，那么 $L(e_1)=L(e_2)$，$L(a_1)=L(e_1)$。

规则 1、规则 2 降低了管理者的工作量，但是会导致几个不同的标签分配给同一个客体。为了确保 XML 文档中的客体只能有一个标签，引入规则 3 来解决标签分配中的冲突。

规则 3 标签指定规则

在一个 $XDoc$ 中，安全标签 L_1, L_2 通过直接指定或者间接指定给同一个客体，如果 $L_1 \geqslant L_2$，那么该客体的安全标签为：$L_3=L_1$。

4 体系结构及实现

EBLP 模型通过安全标签来控制主体对客体的访问。为了有利于扩展现有的 XML 文档系统，用户的安全标签及客体的安全标签信息都以 XML 文档的形式存储。图 1 是实现这种访问控制模型的体系结构，其中，Ulable.xml 存储用户的标签信息，客体的标签存储在信息安全文件库中。MAC 模块是处理 XML 文档访问请求的核心模快。下面给出该模块的访问控制算法。

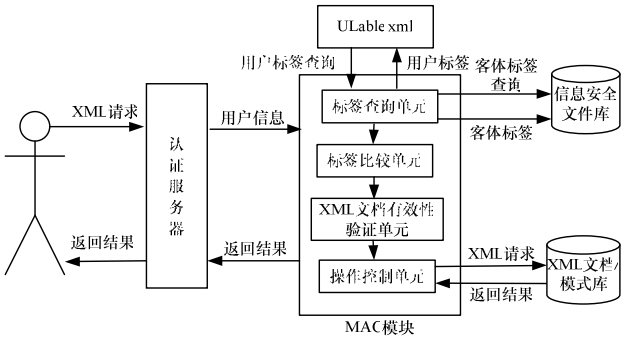

图 1 XML 文档访问控制模型的体系结构

算法 accessControl

输入 访问请求 request=<subject, target, path, type>

输出 访问者得到的视图

处理流程如下：

Step1 从 Ulabel.xml 中取出 subject 的安全标签：L_1= getSubLabel(subject);

Step2 如果是写请求，执行 Step5。如果是读请求，t=parse(target, L_1);

//得到 target 内主体操作范围的文档树 range.xml，该树以 rootnode //为根节点；

Step3 对 range.xml 的文档树进行深度优先搜索，将 subject 有权限访问的节点加入到 response.xml 中，

```
recursive_read(t, rootnode)
{
if (L1.Cr≥getObjLabel (rootnode))
add(rootnode, response.xml);
if (rootnode 不是叶子节点)
for(t 的每个子树 subtree)
//该子树以 subrootnode 为根节点；
recursive_read(subtree, subrootnode);
}
```

Step4 if (response.xml!=null)
then return (“Access Denied”);
else return (response.xml);

Step5 If (InRange(target, path,G_1)
&&Isequal ($L_1.C_w$,target))

```
{
if (IsValid(操作后的文档是有效的)
{
update(xml 文档);
return( 返回更新后的文档树视图);
}
else if (subject==administration)
//只有管理员才有权限修改模式文件；
{
update (schema.xml);
update(xml 文档);
return (返回更新后的文档树视图);
```

(下转第 127 页)