XML 数字签名及其在成绩管理中的应用

Application of XML Signature in the Grade Management

虞　歌　（杭州师范学院 计算机教育与应用研究所　310036）

摘要:XML 正逐渐成为分布式计算的通用语言。随着 XML 的广泛应用,XML 数据的安全问题已成为关注的焦点。本文分析了 XML 签名规范,在.NET 平台上实现了 XML 签名,并应用 XML 签名替代成绩管理中的手工签名,实现了成绩管理无纸化。

关键词:XML　XML 数字签名　成绩管理

1　引言

可扩展标记语言 XML 是由 W3C 于 1998 年 2 月发布的一种标准规范,是标准通用标记语言 SGML 的一个简化子集,它将 SGML 的丰富功能与 HTML 的易用性结合到了 Web 应用中。

随着越来越多的企业通过网络使用 XML 来交换数据,XML 数据的安全问题也越来越重要。XML 的优势来自于它的语义和结构的灵活性和可扩展性,但是正是这些优点引入了一些重要的安全问题,XML 没有实现数据的安全保护。因此,XML 数据的安全问题已成为 Web 应用的瓶颈之一,只有解决了 XML 数据的安全问题,XML 才能得到更广泛的应用。

信息安全的目标是保护信息的机密性、完整性、不可否认性和可用性。XML 数据安全除了上述要求外,还有自己如下的特殊要求。

(1) 灵活性。可以对 XML 文档或者文档的部分内容加密和签名。

(2) 一致性。XML 文档是结构化的数据,加密和签名不能破坏文档的结构,即加密和 签名后的 XML 文档仍然是格式良好的 XML 文档。

(3) 通用性。加密和签名后的 XML 文档既可应用于消息传输,也可应用于文档数据的存储。

XML 需要的是消息层的安全。现有的安全协议无法满足 XML 数据安全的特殊要求。例如 IPSec 和 SSL 协议是网络层和传输层的安全协议,提供了点到点的数据传输安全,无法保证数据存储安全;而且 IPSec 和 SSL 协议只能对整个数据加密,无法实现部分数据加密;加密后的 XML 文档不是合法的 XML 文档。因此,W3C 和 IETF 共同制定了 XML 加密和 XML 签名规范。

2　XML 签名规范

要了解 XML 签名规范的所有信息,可以在 http://www.w3.org/TR/xmldsig-core 站点上查阅《XML-Signature Syntax and Processing》。

XML 签名提供了灵活的数字签名机制,不仅支持对网络资源和消息整体的签名,也支持对 XML 文档或消息的部分进行签名;既支持公钥数字签名,也支持对称密钥的密钥散列验证。

XML 签名语法使用 <Signature> 元素表示签名,<Signature> 元素的基本结构如下:

```
<Signature ID? >
  <SignedInfo >
    <CanonicalizationMethod/ >
    <SignatureMethod/ >
    ( <Reference URI? >
      ( <Transforms > )?
      <DigestMethod >
      <DigestValue >
    </Reference > ) +
  </SignedInfo >
  <SignatureValue >
  ( <KeyInfo > )?
  ( <Object ID? > ) *
</Signature >
```

在该基本结构中,"?"表示 0 次或 1 次出现,"+"表示 1 次或多次出现,"*"表示 0 次或多次出现。

(1) Signature 元素是 XML 签名的最外层元素(根元素),密封了签名数据。ID 属性允许文档包含多个签名。

(2) SignatureValue 元素包含实际签名,由于签名是二进制值,通常被编码为 Base64 格式。

(3) SignedInfo 元素含有规范化(Canonicalization)算法、签名算法以及一个或多个引用。

(4) CanonicalizationMethod 元素指定在执行签名操作前应用于 SignedInfo 元素的规范化算法。规范化指的是使数据遵循一个已建立的标准格式,它是生成一致的数字签名结果的必要条件。这样就允许交叉平台之间的差异,如表示回车的编码等。

(5) SignatureMethod 元素指定签名算法,是散列算法、公钥算法、MAC、填充等的组合。

(6) Reference 元素引用其它元素,包括内容摘要、如何生成摘要的指示以及在生成摘要之前转换内容时所用的规范。URI 指向被引用的实际内容。由于是 URI,因此可以获得 Web 的所有功能。还可以引用 XML 文档中的内容。DigestMethod 指定散列算法,而 DigestValue 则是内容散列的 Base64 值。Reference 元素中作用最大的部分是可能出现的转换集。Transforms 就是 Transform 元素的列表,描述了如何获取签名的数据对象,每个元素指定一个处理步骤。每个 Transform 的输出作为下一个 Transform 的输入。第一个 Transform 的输入来自于 URI,最后一个 Transform 的输出是 DigestMethod 的输入。

(7) KeyInfo 元素是可选的,用于获取验证签名所使用的密钥。该元素可能含有密钥、证书或其他公钥管理信息。

(8) Object 元素是可选的,该元素可能出现一次或多次,可以含有任何数据。

在一个 XML 签名文档中,可以定义三种签名方式。如图1所示。

(1) 分离签名(detached signature)。签名数据与 <Signature> 元素没有父子关系。

(2) 封外签名(enveloping signature)。签名数据与 <Signature> 元素紧密结合在一起,签名数据是 <Signature> 元素的子元素。

(3) 封内签名(enveloped signature)。签名数据与 <Signature> 元素紧密结合在一起,签名数据是 <Signature> 元素的父元素。

图 1　XML 签名方式

3　XML 签名在成绩管理中的应用

成绩管理是高校教务管理中一项重要而烦琐的工作。目前,许多高校开始使用基于校园网的教务管理系统,成绩的录入、查询、统计分析和报表的处理效率大大提高,使用更加方便。但是如何证明成绩的有效性、真实性和完整性现在仍然依赖于任课教师在纸质成绩登记表上的手工签名。如何利用 XML 签名取代手工签名,实现成绩管理无纸化,具有一定的现实意义。

3.1　基于 XML 签名的成绩管理传递流程

(1) 根据任课教师所选择的授课名称和班级,从数据库中抽取数据并转换成 XML 格式的成绩登记表。为了保证该表只能填写成绩不能更改其他内容,需要用教务部门的私钥对该表进行排除填写成绩部分的签名。

(2) 任课教师在 XML 格式的成绩登记表上填写成绩并检查无误后,用自己的私钥对它进行签名,上传该表到教务管理系统。

(3) XML 格式的成绩登记表上传到教务管理系统后,首先进行签名验证,包括教务部门自身签名的验证和任课教师签名的验证,如果验证成功,将成绩数据导入相应的数据库,XML 格式的成绩登记表保存到安全的目录下。

(4) 如果学生对成绩有疑问或需要查询成绩情况,教务管理人员可以调出由任课教师数字签名的 XML 格式的成绩登记表,通过验证签名来证实成绩的有效性、真实性和完整性。实现了成绩管理的无纸化。

3.2　数据库与 XML 格式文档的相互转换

在 XML 文档到数据库的转换方法中,可以采用基于数据库表格的映射,将 XML 文档作为一个或多个数据库表格。

```
<database>
  <table>
    <row>
```

```
        <column1 >… </column1 >
        <column2 >… </column2 >
        ……
      </row >
      ……
    </table >
    <table >
      ……
    </table >
    ……
  </database >
```

对于如下的 XML 文档。

```
<? xml version = " 1. 0"  encoding = " UTF - 8" ?
>
<student >
  <name > ZhangSan </name >
  <age > 20 </age >
  <sex > male </sex >
</student >
```

<name >、<age > 和 <sex > 元素可以映射到数据库表格中的 name、age 和 sex 字段。这种映射方法对于关系数据库比较适用。

从数据库生成 XML 文档主要通过查询语言来实现。目前大多数从数据库中查询生成 XML 文档的查询语言都是基于模板的，这些语言没有预先定义 XML 文档和数据库之间的映射，而是将 select 语句嵌入到模板中，在进行数据交换时由应用程序完成查询数据库，将得到的结果插入到数据所需要的位置。

```
<student >
  < selectItem > select name, age, sex from
student… </selectItem >
  <name > $ name </name >
  <age > $ age </age >
  <sex > $ sex </sex >
</student >
```

转换后的结果如下。

```
<student >
  <name > ZhangSan </name >
  <age > 20 </age >
  <sex > male </sex >
</student >
```

基于模板的查询语言灵活，查询结果可以放在任意位置，可以运用复杂的查询语句，包括定义变量和函数等，是关系数据库查询生成 XML 文档比较好的方法。

3.3 多重签名

在成绩管理中，一张已经由教务部门签名的空白成绩登记表，需要任课教师填写相关的内容后签名。这样某些已经签名的部分就改变了，为了使原来的签名继续有效，就应该在签名前将可能会改变的部分排除。

根据 XML 签名规范，通过 XPath 变换，可以方便地将一个 XML 文档变换成包含一个元素或元素集的多个子文档，在多重签名生成过程中，每个签名者只需对自己负责的子文档进行签名，最后由签名收集者收集产生一个多重签名作为团体的签名。验证者通过团体的公钥可以验证多重签名。假设待签名的 XML 文档如下。

```
<? xml version = " 1. 0"  encoding = " UTF - 8" ? >
<GradeTable xmlns = " urn: grade - table" >
  <Student >
    <Name > YuGe </Name >
    <StudentId > 788335 </StudentId >
  </Student >
  <Item coursetNum = " C763" >
    <CourseName > </CourseName >
    <Grade > 88 </Grade >
    <Date > 2005 - 12 - 15 </Date >
  </Item >
</GradeTable >
```

对待签名的 XML 文档应用如下 XPath 变换。

```
< Transform  Algorithm = http: //www. w3. org/
TR/1999/REC - xpath - 19991116 >
  <XPath > ancestor = or - self: : Item </XPath >
</Transform >
```

变换的结果如下。

```
<Item coursetNum = " C763" >
  <CourseName > </CourseName >
  <Grade > 88 </Grade >
  <Date > 2005 - 12 - 15 </Date >
</Item >
```

4 XML 签名的. NET 编程实现

微软公司在其. NET 平台中对 XML 签名提供了很好的支持。可以使用. NET Framework 的 System. Secu-

rity. Cryptography. Xml 命名空间中的类和 System. Security. Cryptography 命名空间中的类对 XML 文档进行 RSA 签名。

XML 签名和验证签名的基本步骤用 C#语言描述如下。

(1) 生成密钥。

```
RSA key = RSA. Create( );
```

(2) 生成待填充的 XML 签名对象。

```
SignedXml signedXml = new SignedXml( );
signedXml. SigningKey = key;
```

(3) 对待填充的 XML 签名对象添加要签名的 XML 文档。这里要签名的是成绩登记表 OriginalGrade. xml,采用封外签名。

```
XmlDocument xmlDoc = new XmlDocument( );
xmlDoc. Load( " OriginalGrade. xml" );
DataObject dataObject = new DataObject( );
dataObject. Data = xmlDoc. ChildNodes;
dataObject. Id = " MyDataObjectID" ;
signedXml. AddObject( dataObject) ;
Reference reference = new Reference( );
reference. Uri = " #MyDataObjectID" ;
signedXml. AddReference( reference) ;
```

(4) 生成 KeyInfo 元素并签名。

```
KeyInfo keyInfo = new KeyInfo( );
keyInfo. AddClause( new RSAKeyValue( key) );
signedXml. KeyInfo = keyInfo;
signedXml. ComputeSignature( );
```

(5) 将 XML 签名输出到指定的文件,这里是 SingedGrade. xml。

```
XmlElement xmlSignature = signedXml. GetXml
( );
xmlDoc = new XmlDocument( );
XmlNode xmlNode = xmlDoc. ImportNode ( xml-
Signature, true) ;
xmlDoc. AppendChild( xmlNode) ;
xmlDoc. Save( " SingedGrade. xml" );
```

(6) 验证签名。

```
XmlDocument xmlDoc = new XmlDocument( );
xmlDoc. Load( " SingedGrade. xml" );
SignedXml signedXml = new SignedXml ( xml-
Doc) ;
XmlNodeList nodeList = xmlDoc. GetElementsBy-
TagName( " Signature" ,
" http://www. w3. org/2000/09/xmldsig#" ) ;
signedXml. LoadXml( ( XmlElement) nodeList[0] ) ;
if ( signedXml. CheckSignature( ) )
  ……
else
  ……
```

5 结束语

本文根据 XML 应用对数据安全的需求,对 XML 签名规范、签名和签名验证过程进行了探讨并做了一定的实现工作。针对高校教务管理系统中对成绩管理无纸化的需求,用 XML 签名替代手工签名,实现了基于 XML 格式的成绩登记表的电子数字签名。XML 在电子商务和电子政务等领域有广泛的应用,因此对基于 XML 的安全问题的研究很有意义。在满足使用的便利性、可靠性和稳健性方面,XML 安全技术还有很多路要走。这些问题在不久的将来一定会解决,XML 安全技术的应用一定有广阔的发展空间。

参考文献

1 D Eastlake, J Reagle, D Solo. XML – Signature Syntax and Processing W3C Recommenda tion [EB/OL]. http://www. w3. org/TR/xmldsig – core, 2002 – 02 – 12.

2 D Eastlake, J Reagle. XML Encryption Syntax and Processing W3C Recommendation [EB/OL]. http://www. w3. org/TR/xmlenc – core, 2002 – 12 – 10.

3 Peter Thorsteinson, G. GnanaArun Ganesh. . NET Security and Cryptography [M]. Pearson Education, Inc. , 2003:43 – 143, 261 – 291.

4 林学练、刘旭东、怀进鹏,XML 数据安全系统的研究与实现[J],北京航空航天大学学报,2003,29(4):362 – 365。

5 何永忠、王晓京,用 XML 实现电子公文的签名和加密,计算机应用,2002,22(8):85 – 88。

6 李凤银,电子公文中多人签名的设计与实现,计算机应用研究,2005,(6):113 – 115。

7 郭竟乐、赵正德、于清华等,XML 数字签名技术的研究与研究,计算机工程与设计,2005,26(5):1211 – 1213。