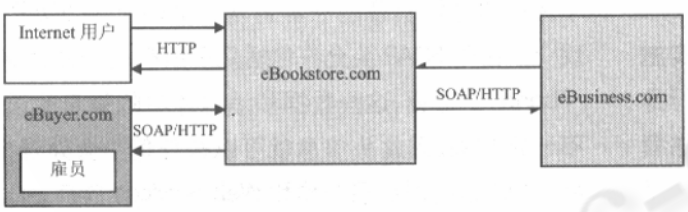
基于安全 Web Services 应用系统中身份验证服务的实现

汤永刚　（湖北职业技术学院 湖北 孝感 432000）

摘　要： 身份验证服务是 Web Services 中需要解决的安全问题之一。在综合考虑 Web Services 中安全问题的基础上，对基于安全 Web Services 应用系统中的身份验证服务进行研究，给出了 ASP. NET 身份验证服务、HTTP 模块、利用 SOAP 头定制身份验证的具体实现方法。实例表明，在基于安全 Web Services 的应用系统中部署有效地身份验证服务，能够保障应用系统的安全，提升系统的安全服务能力。

1　引言

Web Services（万维网服务）是为程序到程序的交互做准备，是网格服务的基础，可以使用 Web Services 来解决异构的分布式计算问题[1]。Web Services 的安全问题一直是人们关注和研究的焦点，是 Web Services 大规模普及的障碍，如何有效地解决 Web Services 中的一揽子安全问题已成为近年来研究人员研究的热点之一。身份验证服务可以证实正在建立一个 Web Services 连接的当事人的身份，并且可以保证一个 Web Services 的请求者就是他（她）自己所声称的那个人，而不是冒名顶替[2]。



注：eBuyer — 电子客户
eBookstore — 电子图书城
eBusiness — 电子事务

图 1　eBuyer、eBookstore 和 eBusiness

2　基于一个安全 Web Services 应用系统实例中身份验证服务需求

基于安全 Web Services 的网上图书城交易系统（如图 1 所示）是集网上订购图书和图书在线信息的一个服务网站，主要提供网上订购图书和图书信息咨询服务。图书城向客户出售图书，而客户则通过由购物车所代表的客户账户，对想要购买的图书产品下电子订单而且在线支付相应消费额。还有两类其他的可能访问这个在线图书城的用户：访问者（站点的偶然浏览者）；职员（管理在线图书城应用程序）。

该系统是由 eBookstore 和 eBusiness 两个企业合作推出的一家在线图书城。由 eBookstore 提供客户服务，由 eBusiness 提供图书信息和订购处理。eBookstore 是用 Microsoft 技术开发的，它依赖于 Windows 和 ASP. NET 技术来保证其 Web Services 应用程序的安全[3]。eBusiness 主要在 Unix 上开发，并使用 BEA WebLogic J2EE 环境来保证其 Web Services 应用程序的安全。eBusiness 使用 Oracle 9i 数据库来存储图书和客户数据。

在基于安全 Web Services 的网上图书城交易系统中，至少可以确定在 eBusiness 和 eBookstore 交互期间四个必须验证的身份，当然可能还存在其他的要求。这四个要求验证的身份如下所描述：

eBookstore 必须知道发起者是谁。

eBusiness 必须确定它接收到的一个 SOAP 请求是来自 eBookstore。

eBusiness 必须可靠地知道发起者是谁。

eBookstore 必须确定是 eBusiness 发送了这个 SOAP 响应。

该系统中的身份验证服务可由 IIS、ASP. NET（有时位于 IIS 之上）和 SOAP 头这三种方法来实现。

3 ASP. NET 身份验证服务

ASP. NET 身份验证功能可位于 IIS 身份验证之上，也可以在没有 IIS 身份验证的时候使用[4]。如果使用了 IIS 身份验证，那么 ASP. NET 可以被配置成接受已验证的身份，从而使得 Web Services 可以访问它。通过 ASP. NET 配置文件 web. config 中的以下元素完成配置：

```
<configuration>
<system. web>
<authentication mode = "Windows" />
</system. web>
</configuration>
```

4 HTTP 模块

从"保护 Web Services"的角度看，HTTP 是一个实施各种安全策略（如访问控制、数据保护和审计）和执行身份验证的便利场所。事实上，Forms 和 Passport 提供者在 ASP. NET 中都是通过在 System. web. Security 命名空间预先建立的身份验证模块—— FormsAuthentication-Module 和 PassportAuthenticationModule 来实现的，这两个模块是随着 ASP. NET 的每个安装一起提供的，可以通过配置文件中的 authentication mode 元素打开或者关闭。

每个 ASP. NET 身份验证提供者处理一个在身份验证过程中发生的 OnAuthenticate 事件，这个事件的主要目的是将实现. NET IPrincipal 接口的定制对象附加到请求的上下文中。

假设 ASP. NET 被配置成使用 HTTP 模块，在运行时 Web Services 定制的身份验证模块需要执行的内容是：

（1）模块分析 HTTP 消息，检查是否是 SOAP 消息；

（2）如果模块检测到是 SOAP 消息，则检查请求中的身份验证数据，根据身份验证模式，身份验证数据可以位于请求中的不同部分，包括 HTTP 头、SOAP 头，甚至还可以在 SOAP 体内，例如 SOAP 消息的数字签名的形式，但是并不限制在这些部分；

（3）如果具有 SOAP 消息的 HTTP 请求中包含了身份验证数据，那么模块将执行身份验证，并分派包含已验证身份的新事件；

（4）事件实现使用身份来创建和初始化主体对象的一个定制的或者默认的实例；

（5）主体然后在为消息服务的运行时上下文中设置。

5 利用 SOAP 头定制身份验证

IIS 身份验证机制对 HTTP 请求进行身份验证，或者在客户证书的情况中，对底层的 SSL/TLS 信道进行身份验证，因此它们和特定的 HTTP 传输相关，然而 SOAP 是传输独立的。有时可能需要从一个正在处理的实体向另一个实体转发 SOAP 请求，同时保持对 SOAP 请求的发出者和 SOAP 消息一起进行身份验证。或者，在其他情况下，相同的 SOAP 消息可以以不同的传输方法进行传输，从而需要使用独立于传输的身份验证。实现独立于传输的身份验证的一种方法是通过使用 SOAP 头实现[4]。由于 SOAP 头允许包括几乎所有的带外（out - of - band）的数据，这些数据和消息体中的信息的语义并不一定相关，因此头信息就可以被任何中介处理，包括 ASP. NET 运行时底层结构和 Web Services 自身。

为对 ASP. NET Web Services 的发起方进行身份验证而专用 SOAP 头的方法是要求客户按照下面的方式将身份验证的数据包含在每个请求的头部分：

```
using System. Web. Services;
using System. Web. services. Protocols;
public calss AuthenticationHeader:SoapHeader{
  public AuthenticationData AuthData;
}
public class FooService:WebService{
  public AuthenticationHeader AuthHeader;
  [SoapHeader("AuthHeader",Required = true)]
```

（下转第 81 页）

(上接第 108 页)

```
        public string foo( ) {
        if ( AuthHeader = = null) }
      return" ERROR: Please supply
authentication data";
      } else{
        AuthenticateRequestor ( Au-
thHeader. AuthData) ;
    }
    //Perform business
    }
    }
```

结合上述实例经分析可知:根据实现 Web Services 代理所使用的技术不同,有不同的设置定制的 SOAP 头内容的方法,为了维持身份验证和业务逻辑之间的分离,并实现 SOAP 请求的独立于传输的身份验证,HTTP 模块技术非常适合在 SOAP 头中传送身份验证数据。

参考文献

1 Mark O' Neill. Web 服务安全技术与原理[M],北京:清华大学出版社,2003:27.

2 应宏、王自全、陈晓峰,网格与 Web 服务的融合[J],重庆三峡学院学报,2005,21(3):41~44.

3 汤永刚,Web Services 实现技术研究[J],福建电脑,2006,15(3):34~35,29.

4 王继梅、金连甫,Web 服务安全问题研究和解决[J],北京:计算机应用与软件,2004,(2):15.

5 肖道举、杨 剑、陈晓苏,Web 服务安全保障机制研究[J],武汉:华中科技大学学报(自然科学版),2004,(12):32.