基于SAML的新型单点登陆模型研究

黄 滨，周德俭，卫传征
（桂林工学院，广西 桂林 541004）

摘 要：随着互联网技术的发展和企业信息化建设的进步，用户在一次事务应用中，可能需要访问多个不同的应用系统，这需要用户在各个不同的应用系统中进行注册，这不但造成了系统的资源浪费，同时也增加了用户的工作量。单点登陆技术实现了用户一次性登陆后，即可进行无缝访问。SAML是一项基于XML的交换安全性信息的框架，结合SAML的单点登陆可以很好地实现信息的描述和信息的安全共享。讨论了单点登陆技术，在研究SAML技术规范与传统基于SAML的单点登陆技术的基础上，针对其存在的安全性、可靠性较差问题，引进CA认证技术，提出了一种新型单点登陆模式，提高了单点登陆的可靠性与安全性。

关键词：安全断言标记语言技术；断言；单点登陆；数字加密；数字签名
中图分类号：TP393 **文献标识码**：A **文章编号**：1673-629X(2008)09-0219-03

Research of New Single Sign-on Model Based on SAML

HUANG Bin, ZHOU De-jian, WEI Chuan-zheng
(Guilin University of Technology, Guilin 541004, China)

Abstract: With the development of Internet technology and the progress of information technology in enterprise, user may enter many different application systems to achieve one affair. In the case, user must register in many different application systems which not only cause waste of resources, but also increase the user's work. SAML is a XML-based framework for exchanging security information. SAML-based SSO does well in the description and safe sharing in information. In this thesis, discuss the technology of single sign-on (SSO). It is based on the research with SAML technical specifications and traditional SSO which is made of SAML. In view of the lack of safety and reliability, it develops a new mode of SSO to improve the reliability and security of SSO.

Key words: security assertion markup language; assertion; single sign on; digital encryption; digital signature

0 引 言

为了实现企业的信息化、电子商务和其他需求，越来越多企业的信息系统通过网络实现了互联。而企业之间的传统登陆是基于用户标识/密码机制，这使得用户需要分别记忆各个企业系统的用户标识/密码，同时在每个系统中也要分别建立一个系统来保存用户标识/密码，这给用户造成了困难，同时也造成了系统资源的浪费[1]。

为了解决以上问题，人们提出了单点登陆的概念，以实现将分散的用户管理集中起来，各系统之间依靠相互信赖的关系进行用户身份的认证。文中主要介绍和研究基于SAML的单点登陆技术。

收稿日期：2007-12-02
基金项目：广西自然科学基金（桂科基0575101）；研究生教育创新项目（2007105960812M12）
作者简介：黄 滨（1981-），男，湖南长沙人，硕士研究生，研究方向为制造业信息化；周德俭，教授，研究方向为制造业信息化。

1 单点登陆技术探讨

目前实现单点登陆系统的技术有COOKIES、SESSION和SAML[1]，基于COOKIES的单点登陆系统通过COOKIES记录认证信息，实现用户登陆。而基于SESSION的单点登陆系统通过SESSION共享技术实现认证信息的共享。然而COOKIES和SESSION只在同一个域中起作用，如果系统建立在不同的域，需要使用代理AGENT等方式来辅助实现跨域的单点登陆，大大增加了实现的成本[2]。SAML（安全认证声明标记语言），其开发目的是作为一种基于XML的、用于交换安全性信息的框架，为认证、授权、策略提供标准机制。其能很好地支持实现单点登陆，并且其所实现的单点登陆具有跨平台性和跨域性等特点。

1.1 传统的SAML单点登陆

传统SAML单点登陆模式包括三个模块（见图1）[3]，分别是服务请求方、安全认证方、服务提供方。

（1）服务请求方：向服务提供方请求服务，接受服务提供方提供的服务，其可以是个人、应用系统或者是

服务代理。

(2)服务提供方:对服务提供方的服务请求进行身份权限验证,并以验证结果提供服务响应:拒绝或者提供服务。

(3)安全认证方:对服务请求方进行身份验证,并对服务请求方的身份验证要求"SAML 认证声明请求"进行响应,提供服务请求方以身份验证 SAML 认证声明。

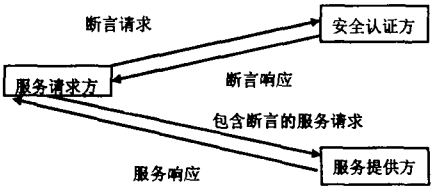

图 1 SAML 单点登陆模式

1.2 单点登陆方式

传统的单点登陆包括 PULL 方式和 PUSH 方式。

1) PULL 方式实现过程(见图 2)[4]:

(1)用户"USER"连接安全认证方"IDP",进行身份验证。

(2) IDP 生成 SAML 认证声明,返回代表 SAML 认证声明的辅件 Artifact,并将用户请求重定向于 SP。

(3)用户使用 Artifact 访问 SP 受保护资源。

(4)SP 根据接收到的 Artifact 向 IDP 请求认证需要的该用户的相关 SAML 认证声明。

(5) IDP 根据 Artifact 向 SP 回复所请求的 SAML 认证声明。

(6)SP 根据 SAML 认证声明允许或者拒绝用户访问受保护资源。

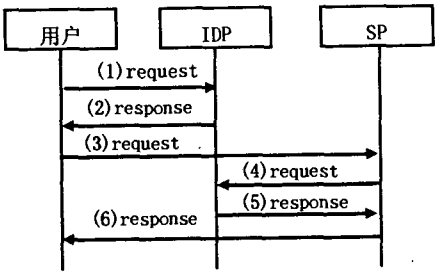

图 2 PULL 方式实现

2) PUSH 方式实现过程(见图 3)[4]:

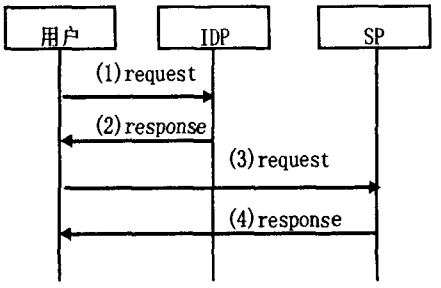

图 3 PUSH 方式实现

(1)用户进行身份验证并请求连接。

(2)IDP 生成身份认证 SAML 认证声明,将 SAML 认证声明发送回用户,并将用户重定向于 SP。

(3)用户使用 SAML 认证声明访问 SP 的受保护资源。

(4)SP 根据 SAML 认证声明允许或者拒绝用户访问受保护资源。

1.3 单点登陆方式存在的问题

传统 SAML 单点登陆模式存在着安全问题,PUSH 和 PULL 方式都面临的威胁有[2]:

(1)攻击 SAML 协议的报文交换。包括窃取 SAML 声明,篡改报文,重话攻击,中间人攻击。

(2)恶意服务提供者 SP 仿冒用户。因为服务提供者从用户处获取 SAML 辅件,恶意的 SP 会在某一新的 SP 处仿冒用户,新的 SP 相信这个恶意的 SP 是用户,并从其处获得 SAML 认证声明。

PULL 方式面临的威胁:

因为 PUSH 方式是使用辅件 Artifact 方式传输,而 PULL 方式是直接传输 SAML 认证声明,因此,PUSH 方式还得防止窃取 SAML 辅件,如偷听者可以复制用户的 SAML 辅件,那么偷听者就可以用用户的 SAML 辅件构造 URL,并且到目的站点冒充用户,而 PULL 方式得防止窃取 SAML 认证声明。

2 对单点登陆方式的改进

单点登陆的两种方式 PULL 方式和 PUSH 方式都存在着各种各样的安全性问题,而 SAML 规范本身对于安全问题的解决没有明确的技术规范要求,可以借助于已经存在的安全解决方案辅助解决 SAML 单点登陆安全问题。因为 SAML 是一种基于 XML 的技术规范,因此关于 XML 的数字加密与 XML 数字签名也同样适合于 SAML 技术。

第三方信任机构 CA 通过颁发数字证书给服务请求方"USER"、安全认证方"IDP"和服务提供方"SP",其中 IDP 的数字证书中包含 IDP 的公钥 Kidp,将私钥 Kidp'从 CA 处通过安全信道传送给 IDP[5]。同样服务请求方的数字证书包含用户的公钥 Kuser,私钥 Kuser'存放在用户处。SP 的公钥为 Ksp,私钥为 Ksp'。CA 处保存有 USER、IDP 和 SP 的公钥,并提供公钥查询功能。利用公钥可以生成数字签名,该签名并不对消息加密,而是证实一条消息的真实性和确认发送者的标识。利用私钥可以生成数字加密,对消息进行加密,防止消息泄露[6,7]。

2.1 新型单点登陆过程

新型单点登陆过程如下(见图 4):

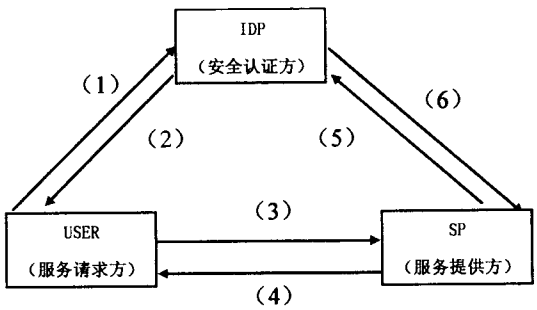

图4　新型单点登陆模式

1)如果用户想访问SP的受保护的资源,首先发送request消息访问IDP,消息中包含了用户的用户名、密码等基本信息。IDP处理此request消息,如果发现用户还没有认证过,则通过request消息认证。如果该用户已经过认证,IDP生成一个代表认证SAML认证声明的辅件Sa,并用Kidp对Sa签名得到Sa'。

2)IDP将Sa和Sa'返回给用户。

3)用户产生随机数m,并将IDP返回的Sa、Sa'用Kuser签名得到(Sa,Sa',m)',将其发送给SP。

4)SP收到(Sa,Sa',m)'后,验证Kuser签名,如果正确证明是用户发送过来的,再验证Sa的签名,证实辅件是由IDP发送的且未被篡改过。SP用自己的私钥Ksp'对Sa加密,产生Ksp'(Sa),并将其发送到IDP。

5)IDP收到Ksp'(Sa)后,用SP公钥Ksp解密得到Sa,根据Sa得到SAML认证声明S,用IDP的私钥加密,产生Kidp'(S),并发送回SP。

6)SP收到Kidp'(S)后,用IDP的公钥Kidp解密,得到SAML认证声明S,SP根据此SAML认证声明,允许或拒绝该用户最初的访问其资源的请求。

```
<! --SAML 身份认证-->
<saml:Assertion>
  <saml:Conditons
  NotBefore="2007-10-11T15:20:01"                    (1)
  NotOnOrAfter="2007-10-11T15:25:01"/>
  <saml:AttributeStatement
    AuthenticationMethod="Password"                  (2)
    AuthenticationInstant="2007-10-11T15:21:00">     (3)
<saml:Subject>
  <saml:NameIdentifier NameQualifier
    SecurityDomain="www.security.com"
    Name="huang"/>                                   (4)
  </saml:Subject>
</saml:AttributeStatement>
<saml:Signature>…</saml:Signature>                   (5)
</saml:Assertion>
```

(1)为此SAML身份认证的认证有效期。

(2)为认证方式,为用户/密码认证。

(3)为认证时间。

(4)身份认证的认证用户名,为"huang"。

(5)为对此认证的数字签名。

2.2　安全性分析

如果攻击者采用重放攻击,因为SAML认证声明是有时效的,可以在IDP处根据网络的实际情况设置合适的声明有效时间,SP会检查声明的有效期,如果在有效期外,SP会拒绝为其服务[8]。另外,SP处保存有随机数m,重放攻击者不知道随机数m,所以重放攻击无效。

如果攻击者截获辅件Sa,因为在第(2)步骤中,Sa被数字签名生成Sa',可以通过验证数字签名,证明Sa的正确性,在第(4)步骤中,SP用自己的私钥对Sa进行加密,生成Ksp'(Sa),保证传输的安全性。

如果攻击者截获认证声明。在第(5)步骤,IDP对认证声明S加密,生成Kidp'(S),保证传输安全性。

3　结束语

SAML规范中,只为单点登陆系统提供一个基础框架,其对单点登陆的具体实现方式没有具体的规则要求,而传统的PUSH方式和PULL方式在安全性等方式都存在着很多问题,文中通过对传统SAML单点登陆进行改造,结合CA数字认证,数字签名、数字加密,构建了一个新型的、较安全的单点登陆系统原型。经实验验证具有实用性和有效性,对单点登陆的工程实践应用具有一定的指导意义。

参考文献:

[1] 高　喆.SAML在单点登录中的安全性分析、改进及实现[D].上海:上海大学,2005.

[2] 林满山,郭荷清.单点登录技术的现状及发展[J].计算机应用,2004,24(6):248-250.

[3] 鲁耀杰.基于SAML和属性证书的单点访问系统的设计与实现[D].南京:南京理工大学,2004.

[4] Chao Yuen-Yan. Weakest Link Attack on Single Sign-On and Its Case in SAML V2.0 Web SSO[J]. Computational Science and Its Applications,2006,3982:507-516.

[5] Bret H, Donald J F. 全面掌握Web服务安全性[M].杨硕,译.北京:清华大学出版社, 2004.

[6] 史创明,王立新.数字签名及PKI技术原理与应用[J].微计算机信息,2005(8):122-124.

[7] 肖长水,高觐悦.保密信息传输的数字签名方案研究[J].苏州市职业大学学报,2007(3):61-63.

[8] Dournace B. XML Security[M]. 北京:清华大学出版社,2003.