第34卷 第14期 计 算 机 工 程 2008年7月
Vol.34 No.14 Computer Engineering July 2008
·网络与通信· 文章编号：1000—3428(2008)14—0138—02 文献标识码：A 中图分类号：TP393.07

基于反向代理的网站群单点登录

王 琦
(浙江大学计算机科学与技术学院，杭州 310027)

摘 要：传统门户网站单点登录解决方案存在着对原有系统改动大、维护代价高的问题，该文利用反向代理机制实现网站群资源的虚拟集中，结合基于角色的访问控制(RBAC)模型及 LDAP 技术，设计并实现基于反向代理的单点登录解决方案。该方案不需要对原有系统进行接口编写和改造，具有优良的性能。

关键词：单点登录；反向代理；基于角色的访问控制；LDAP 技术

Websites Single Sign-on Based on Reverse Proxy

WANG Qi
(College of Computer Science and Technology, Zhejiang University, Hangzhou 310027)

【Abstract】 Aiming at the drawback of general single sign-on solution to the Websites that has onerous coding and maintenance costs, this paper introduces reverse proxy to virtual concentration of Websites resources, combines it with Role-Based Access Control(RBAC) and LDAP, and proposes a new single sign-on model. According to the model, less coding is needed to the original websites, thus it has effective performance.

【Key words】 single sign-on; reverse proxy; Role-Based Access Control(RBAC); LDAP

1 概述

政府门户网站需要整合各所属部门子网站，以便提供便捷的网上服务。但由于各部门子网站间用户信息无法共享和同步形成的“信息孤岛”，导致市民在网上办事过程中需要在不同子网站间多次注册和登录，成为电子政务应用推广过程中的一大障碍，并且子网站间同一用户信息由于注册多次和无法同步，使得信息冗余和不一致，这会严重制约网上服务的发展。

传统的基于Web Service的单点登录(Single Sign-On, SSO)解决方案在部署过程中需要对原有各系统的认证、授权模块进行修改，且无法整合不支持Web Service的业务系统，因此，当存在大量遗留业务系统时，其实际操作性较差。而基于Agent的单点登录解决方案如Sun公司的Sun ONE Indentity Server Policy Agents[1]只需在原有子网站的Web服务器中安装相应的Agent并进行配置，无须修改原有系统的代码即可实现，但由于各子网站采用的系统软件产品、版本不同，在实施过程中很难找到全部对应的Agent产品。本文在对上述两大类单点登录解决方案深入分析的基础上，结合门户网站单点登录建设实际，提出一种基于反向代理的单点登录解决方案。

2 单点登录概述

单点登录是指在多个应用系统中，用户只需登录一次就可以访问所有相互信任的应用系统。单点登录是应用系统整合的基础之一，也是面向服务的架构(SOA)设计在安全方面的重要内容。

2.1 基于 Web Service 的单点登录

基于Web Service的单点登录解决方案[2]一般以LDAP[3]等目录服务作为整个系统用户账号管理的核心，利用Web Service跨平台的特点实现不同子网站应用中用户信息的整合。由于直接在各应用系统开发调用单点登录服务的接口，比较适合新建系统间的整合。缺点是需要对原有系统进行修改，存在因修改、调试导致长时间的停机以及原有系统文档的不完整、熟悉系统的维护人员的缺乏等问题，在涉及整合的系统数量较多的情况下，实际可操作性较差。

2.2 基于 Agent 的单点登录

基于 Agent 的单点登录解决方案分为 2 个部分：

(1)在需要整合的子网站 Web 服务器上安装一个 Agent Filter 以获取访问者对保护资源的访问请求。

(2)一个安装在应用服务器的模块用于提供和认证服务器如 LDAP 的访问。

该方案不改动原有系统，只要安装后对需要控制访问的 URL 资源进行配置即可。但安装的 Agent 与操作系统、Web 服务器、应用服务器的平台以及版本都密切相关，没有对应的 Agent 就很难实施成功，因此，不太适合存在着大量异构业务系统的整合需求。

3 基于反向代理的单点登录系统架构

反向代理服务应用在网站系统中一般有两大用途：

(1)安全反向代理

作为网站 Web 服务器的替身，以提高网站系统的安全性。

(2)Web 内容缓存

多台缓存网站内容的反向代理服务器以服务器负载均衡的方式运行，可以起到降低Web服务器负载，提高网站访问效率的作用[4]。

上述基于 Agent 的单点登录方案可以在不改动原有系统的前提下实现单点登录，这是保障系统整合实际可行的前提，但由于 Agent 是分布式部署到各个子网站的，因此无法应对

作者简介：王 琦(1974－)，男，教授级高级工程师、博士，主研方向：电子政务，计算机网络与信息安全

收稿日期：2007-10-15 **E-mail：**wangchee99@yahoo.com

各个部门复杂多样的平台状况。针对上述情况，借鉴反向代理器在网站服务器群架构中所起的虚拟集中的作用，提出了如图 1 所示的基于反向代理的集中式单点登录解决方案。

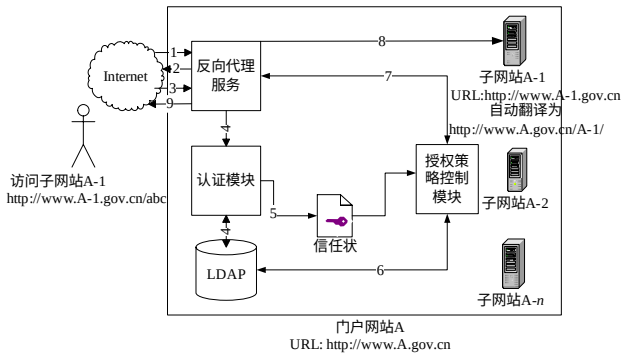

图 1　系统架构

本文利用反向代理服务器的多虚拟主机功能以及 URL 重定向功能，将物理上分散在全市各地的子网站虚拟为门户网站相应的子目录，这样就可以将基于 Agent 解决方案中分散于各子网站的用户登录请求集中到了门户网站的反向代理服务器上，避免了因各部门系统平台不同带来的问题。

用户访问子网站受保护资源的过程可分为 9 个步骤：

(1)用户递交访问门户网站某部门子网站 A-1 受保护资源 abc 的请求 http://www.A-1.gov.cn/abc。

(2)访问 A-1 的请求经 DNS 解析实际递交至门户网站 A 的反向代理服务器。由于反向代理服务器将各个部门子网站以多虚拟主机的形式运行，该请求被自动翻译为访问门户网站的目录 A-1，因访问资源是受保护的，系统要求用户递交认证信息。

(3)用户递交账号、口令等认证信息。

(4)认证模块通过访问 LDAP 服务器检查用户的认证信息是否正确和合法。

(5)当用户的认证信息获通过后，将用户信息经过加密生成一个 cookie 形式的信任状(Credential)，并将其保存在用户的会话中。

(6)授权策略控制模块将递交的用户信任状解密，并检查该用户是否有权访问该资源。

(7)确认有权访问后，通知反向代理服务器。

(8)反向代理服务器将访问请求重定向至子网站 A-1。

(9)访问的应答内容经反向代理服务器转发给用户，访问流程结束。

在用户认证和授权整个过程中，网站系统包括各部门子网站，并只需维护一个 LDAP 服务器上的用户信息，这样避免了维护多个用户信息库所带来的数据同步问题。另外，通过只传递包含用户信息的加密信任状而不是用户口令的机制，降低了口令被截获破解的风险。

4　设计与实现

4.1　反向代理服务

反向代理服务是系统的重要模块，其核心是通过对 URL 的自动翻译，实现用户的透明访问，也就是把所有子网站映射为反向代理服务器的相应目录，然后把用户需要访问子网站服务器的请求自动翻译为对代理服务器子目录的访问。

例如：门户网站(反向代理服务器)的 URL 为 http://www.A.gov.cn，用户访问请求 http://www.A-1. gov.cn/ abc，就被改写为 http:// www.A.gov.cn/ www.A-1.gov.cn/abc。

由于返回的页面资源中如图片、超链接等还可能带有子网站的 URL，因此需要对子网站服务器返回的信息进行过滤，对于那些位于反向代理范围内的 URL 信息要改写为通过反向代理服务器访问的形式。自动翻译采用正则表达式和正则自动机实现，对于一些动态网页内容的翻译，可通过基于正则表达式的模板来完成。

4.2　基于角色的授权

对于受保护资源的授权访问控制是系统的另一个重要模块，系统采用了基于角色的用户访问控制模型(Role-Based Access Control, RBAC)[5]。

$RBAC_0$模型定义如下，其中U,R,P和S分别表示用户、角色、许可和会话：

(1) $PA \subseteq P \times R$，给角色赋予许可权限，是一个多对多的关系；

(2) $UA \subseteq U \times R$，给用户赋予角色，是一个多对多的关系；

(3) $user: S \to U$，将每个会话映射到对应的用户$user(s_i)$的函数；

(4) $roles: S \to 2^R$ 是一个将每个会话 s_i 映射到角色集 $roles(s_i) \subseteq \{r|(user(s_i), r) \in UA\}$ 的函数，其中，会话 s_i 具有许可 $\cup_{r \in roles(s_i)} \{p|(p,r) \in PA\}$。

在实际应用中上述模型还需具有许可继承和约束两大特性，也就是需要实现$RBAC_3$，为了便于灵活、便捷的管理，本文将LDAP引入了用户组。

基于 RBAC 的授权策略控制包括两个阶段的映射：

(1)角色对资源的映射：资源按照允许访问的不同角色加以保护。

(2)用户/用户组对角色的映射：即用户或用户组按照他们的职能等不同分别映射至相应的角色。

资源 resource_1 的公务员邮件用户角色 email user 对资源的映射 resource_1.xml 可定义如下：

```
<security-role id=”SecurityRole_1”>
    <description> civil servant email user</description>
    <role-name>email user<role-name>
</security-role>
```

相应的信息中心用户组 InforCenterGrp 对角色 email user 的映射 resource_1-bnd.xmi 如下：

```
<authorizationTable xmi:id=”AuthorizationTable_1”>
    <authorizations xmi:id=”RoleAssignment_1”>
        <role
href=”META-INF/resource_1.xml#SecurityRole_1”/>
        <group xmi:id=”Group_1” name=”InforCenterGrp”/>
    </authorizations>
</authorizationTable>
```

5　结束语

本文提出的基于反向代理的单点登录解决方案在门户网站的应用表明：反向代理方式对原有系统改动少，停机时间短，用户访问透明。在将反向代理与 RBAC 的授权控制以及信任状机制结合后，该方案为大型松散耦合型系统中用户管理模块的整合提供了安全、可行的解决方法。

今后的研究方向将集中在：利用反向代理的内容缓存特性进一步提高访问性能，缓解子网站访问的高负载问题，利用反向代理实现对用户异常访问行为进行审计和主动应对，以便提高系统整体性能和安全性。 (下转第 142 页)