基于Web的目录服务信息系统设计与实现

洪东明,姚长利,李 凤,吴志远
(中国地质大学,北京 100083)

摘 要:随着信息系统应用领域的不断扩展和目录服务技术的兴起,目录服务已经应用到主流的企业和Internet的环境下。任何一个企业级或面向Internet的应用,在将目录服务领先的数据存储、管理和查询技术集成到信息系统过程中,需要解决一个突出问题是克服信息系统和目录服务两者之间耦合的困难。探讨了LDAP目录服务及相关技术的基础,分析了目录数据的访问操作方式,提出了基于Servlet技术的目录服务Web信息系统建设的可行性方案。通过B/S模式下"目录节点查找操作"的实例,解决具体应用中的问题,明确了采用B/S模式和Servlet技术开发目录服务Web信息系统具有简便实用性。

关键词:目录服务;LDAP;三层B/S结构;Servlet
中图分类号:TP311 **文献标识码**:A **文章编号**:1673-629X(2008)06-0233-03

Design and Implementation of Web-Based Directory Service Information System

HONG Dong-ming, YAO Chang-li, LI Feng, WU Zhi-yuan
(China University of Geosciences, Beijing 100083, China)

Abstract: Along with rapidly increasing expansion of information application software in various fields, dightweight directory access protocol (LDAP) has exploded into the mainstream of enterprise and Internet software environments. One of the requirements of any major enterprise-level or Internet-oriented application is to integrate LDAP's advanced technology (such as data storing, organizing and querying). Therefore there need to resolve their interrelated problems. Explores the base and extensions of LDAP. After analyzing the method to access LDAP data, it puts forward a solution of constructing Web-based directory service information system. Then one code example of "LDAP data searching" in servlet is demonstrated. It turns out that the introduced way is simpler and easier to develop directory service information system by using B/S model and servlet technology.

Key words: directory service; LDAP; 3-tier B/S model; servlet

0 引 言

目录服务将成为下一代网络应用构造的核心组件。目录作为主要的集成点,可以集成身份验证、安全网关和网络管理等服务;目录服务支持分布式环境,而且灵活方便、安全可靠,将是当前网络越来越复杂化的情况下进行高效智能管理的一个不可缺少的应用。当前开发成功的目录服务信息系统产品已经用于政府和企业门户、居民园区宽带计费、校园资源信息服务[1]等方面。目录服务实现单点登陆、统一认证、资源访问控制等,容易集成作为真正安全便捷面向用户的先进信息系统。

收稿日期:2007-09-13
作者简介:洪东明(1978-),男,浙江苍南人,博士研究生,研究方向为网络数据库和信息技术;姚长利,教授,博士生导师,主要从事计算机技术及地球物理软件方法研究。

1 LDAP目录服务简介

LDAP(Lightweight Directory Access Protocol)即轻量级目录访问协议是从X.500协议简化的基础上演变而来的[2]。目录服务的目录一般只执行简单的更新操作,适合于进行大量数据的检索。目录服务是一种按照树状信息组织模式,实现信息管理和服务接口的方法。目录服务系统一般由两部分组成:(1)拥有一个数据规划描述的分布式数据库;(2)访问和处理数据库有关的详细的访问协议。当前,目录服务技术的国际标准主要是较早的X.500标准和正在迅速发展的LDAP标准,作为一个开放的标准,LDAP以后将可能象HTTP和FTP一样成为Internet协议集的不可缺少的部分。

LDAP目录服务的主要模型包括以下4种类型:

(1)信息模型。

信息模型定义了目录中存放信息的基本单位和数

据的类型。目录中信息以树状形式组织,信息的基本单位是条目(Entry),每个条目为一个属性集合,每个属性含有一个属性类型和一个或几个值。从面向对象的概念看,条目相当于现实世界的一个对象,属性则从某一方面反映对象的特征。如图 1 所示。

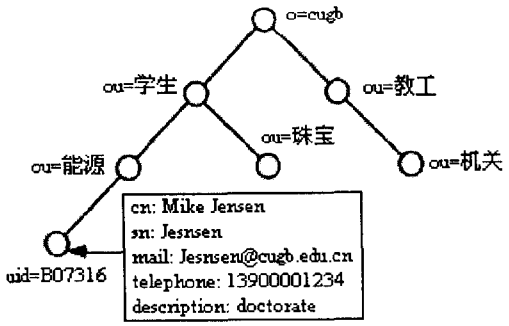

图 1 典型目录信息树

(2)命名模型。

命名模型定义了目录的组织和查询方式。LDAP 指定目录条目应被组织成倒转的树型层次结构——目录信息树。树根(Root)是虚根,树的每个节点都存储信息,每个节点都有一个属性作为相对名 Rdn(Relative distinguished name)。将某个节点回溯到根,所有 Rdn 一起组成该节点的区分名 Dn(Distinguished name)。实际使用过程中,用户通过 Dn 作为查询条件,可以直接找到所需要的条目。

(3)功能模型。

功能模型定义了访问和更新目录的操作。这些操作分为三类:a. 查询操作,包括查找(search)和比较(compare)操作。查询某个条目并返回结果,LDAP 既支持根据区分名查询,又支持根据某一属性检索;b. 更改操作,进行条目或其属性的增加(add)、删除(delete)、重命名(rename)和修改(modify);c. 认证及控制操作,对客户端认证,控制某些交互行为。

(4)安全模型。

安全模型定义了如何保护目录信息,防止未授权用户对目录信息的访问和修改。通过以下几种方式来实现:a. 目录认证。使用 LDAP 功能模型中的认证操作类来防止未经授权的非法访问;在基本认证的基础上,可以使用第三方的安全认证,例如:LDAPv3 把 SASL (Simple Authentication and Security Layer)框架作为协议实现的一部分。b. 目录授权。主要通过 ACL (Access Control Lists),即访问控制列表来实现。在通过目录认证后,对目录的访问操作权限按 ACL 的定义来分配。ACL 主要有两种方式的实现:设置在 LDAP Server 的配置文件中;作为目录信息的一部分存储在目录信息树中。c. 数据传输。使用数据加密手段防止对目录信息的非法窃听,如使用 TLS (Transport Layer Security)来建立加密的 LDAP 会话[3]。

2 系统设计与实现

2.1 系统设计

基于 Web 的目录服务信息系统的实现目标是将用户熟悉的浏览器与复杂的目录服务两者有机结合起来,促进目录服务的开发和应用。实现本系统主要采用三层浏览器/服务器体系结构。浏览器/服务器三层结构是在分布式技术成熟之后从客户机/服务器的多层结构简化而来。本系统拟采用三层 BC/S 分布式计算结构模型,即"浏览器—Web 应用服务器—目录服务器"结构模型。目录服务器采用的是 C/S 结构,在此结构中,Web 应用服务器即是客户访问的服务器,又是目录服务器的客户端。从用户的使用角度看,通过 Web 页面就可以直接访问目录服务器。如图 2 所示。

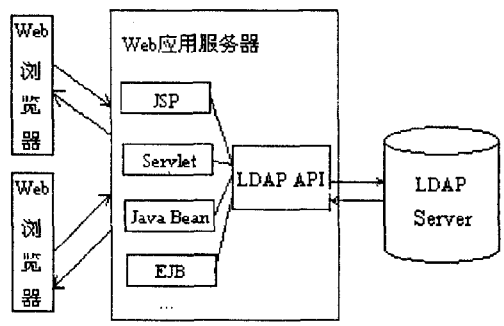

图 2 基于 Web 的目录服务信息系统结构图

Web 应用服务器端主要采用 J2EE 的 Servlet 技术,Servlet 是用 Java 编写的服务端程序,它具有高性能、易开发、可移植、动态加载、与协议和平台无关等优点[4]。基于 Servlet 可以应用面向对象的编程模型 Model - View - Controller(MVC),采用逻辑和表现分层架构 JSP, Servlet, Java Bean 来开发复杂工程。Servlet 的扩展机制很好地满足了对于中间层可扩展性、易管理性、高可用性和高安全性的要求,适合于构造目录访问接口。

2.2 LDAP 开发接口

文中 LDAP 以 Netscape Directory Server 4.1 为例,接口采用 Netscape 自带的 Directory Java SDK 4.1。在 iPlanet 的主页上下载开发包,解压可得到 Java LDAP API 的 JAR 包及详细文档资料。

LDAP 主要开发接口见表 1。

2.3 系统实现

在目录服务 Web 应用中,通常包含各种各样的不同操作,这些操作组合起来完成一个满足特定需要的复杂的用户应用系统。下文以一个典型的目录服务操作来演示基于 Web 的目录服务信息系统开发的一般

过程。

表1　LDAP主要开发接口列表

函数	描述
connect(String host,int port)	指定ip和端口连接主机
authenticate(String dn,String passwd)	使用用户名密码授权访问
bind(String dn,String passwd)	简单认证目录绑定函数
bind(String dn,Hashtable props,bject cbh)	SASL的目录绑定函数
disconnect()	断开与LDAP主机的连接
search(String base,int scope,String filter, String[] attrs,boolean attrsOnly)	按指定条件过滤查询
modify(String DN, LDAPModification mod)	修改某个条目的单个值
modify(String DN, LDAPModification[] mods)	修改某个条目的一组值
rename(String DN,String newRDN, boolean deleteOldRDN)	重命名某条目的名称
add(LDAPEntry entry)	增加目录中的条目
delete(String DN)	删除目录中的条目
abandon(int id)	通过id号取消进行的操作
getLDAPErrorMessage()	返回一条结果错误信息
getCount()	计算查询返回条目个数
next()	处理查询到的下一条条目
hasMoreElements()	返回是否还有查询结果
nextElement()	返回下一个查询结果
sort(LDAPEntryComparator compare)	对查询结果排序
getDN()	获得条目的DN值
getAttributeSet()	返回条目的一组属性集合
getAttributes()	属性集合转化为枚举类型
getName()	返回一条属性名
getStringValues()	返回一条属性值

查询操作是目录服务中最常用、功能最强大的一项操作,既可以很简单也可以很复杂。用户指定查询的起始位置(DN)、查询范围(深度)和查询过滤器,还可以限定返回条目的数量和返回属性的列表,服务器根据客户的条件返回DIT中满足客户需要的条目的部分或全部信息。最基本的查询必须提供5个基本的参数,如表2所示。

表2　搜索操作的参数

参　数	描　述
Search Base	搜索的开始节点
Search Scope	搜索的深度
Search Filter	搜索的过滤条件
Attribute list	要返回的属性列表
Type Only	布尔值,决定是否只返回属性类

其中,Search Scope有3个可选值:

.SCOPE_BASE:只搜索该节点本身。

.SCOPE_ONE:搜索该节点的直接下属节点。

.SCOPE_SUB:搜索该节点的所有下属节点。

一般一个应用调用LDAP API有以下5步:

(1)建立一个目录服务连接对象ld。

LDAPConnection ld = new LDAPConnection()

(2)使用方法打开LDAP服务器连接。调用ld.connect()函数。

(3)使用方法认证LDAP服务器。调用ld.authenticate()函数。

(4)执行其它LDAP操作并获得其值。

(5)关闭连接。调用ld.disconnect()断开LDAP服务器[5]。

代码如下:

```
import netscape.ldap.*;
public class SearchServlet extends HttpServlet {
public void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    LDAPConnection ld = new LDAPConnection();
    try{
     ld.connect( ldapHost, ldapPort );
       ld.authenticate(ldapDN, ldapPW);
       }catch(LDAPException e){
         out.println("Error number: " + e.getLDAPResultCode() ); }
         out.println("成功! Connected to " + ldapHost + "at port " + ldapPort );
         String[] myAttrs = {"uid"};
    //以搜索所有(objectclass=*)为过滤条件从根节点dn:o=cugb开始查找
    try{
      LDAPSearchResults myResults = ld.search("o=cugb", LDAPv3.SCOPE_SUB,"(objectclass=*)",myAttrs, false);
      }catch(LDAPException e){
      out.println("LDAPException: return code:" + e.getLDAPResultCode() ); }
    //对查找结果遍历循环,直到取到属性名和属性值
    while (myResults.hasMoreElements()){
    LDAPEntry myEntry=null;
    /*分析每个返回的条目*/
    try {
    myEntry = myResults.next();
    }catch (LDAPException e1){
    e1.printStackTrace();}
    String nextDN = myEntry.getDN();
    //输出对象的dn
    out.println( nextDN );
    LDAPAttributeSet entryAttrs = myEntry.getAttributeSet();
    Enumeration attrsInSet = entryAttrs.getAttributes();
    /*分析条目下每个属性*/
    while(attrsInSet.hasMoreElements()){
    LDAPAttribute nextAttr = (LDAPAttribute)attrsInSet.nextElement();
    String attrName = nextAttr.getName();
```

(下转第239页)

序共享,使用完后将连接对象返还给连接池,避免了因频繁使用数据库造成的数据库效率下降,大大提高了程序的使用效率,同时还可以通过连接池自身的管理机制来监视数据库的数量、使用情况等。

上述两种JSP数据库连接技术在处理数据库操作时各有特点,为此在加油站管理信息系统中,根据不同的信息处理采取了不同的连接技术。当对数据库的访问量很大时,可以采用数据库连接池技术,利用其重用内存资源、提高服务器效率、支持多用户访问的特点,提高数据库系统的使用效率;当对数据库的内容需要经常进行操作时,例如插入、修改、删除数据库记录等,可以采用JDBC(JavaBean)数据库连接技术,发挥JavaBean的开发效率高、使用简单方便的特点。这样,在不同的应用场合使用不同的数据库连接技术,可以充分发挥JDBC(JavaBean)技术和连接池(Connection Pool)技术各自的优点,取得更好的使用效果。

2.6 系统的安全设置

基于Web的数据库应用系统的开发,数据库的安全性无疑是最重要的。为此,除了Web应用系统应具有的防火墙等防护能力外,数据库中还设置了多种用户角色,系统管理员、数据库管理员、数据库操作员、一般用户,系统通过不同级别的角色实现分级访问控制。用户登录时,系统通过用户名和密码来确认用户身份及角色,并根据用户身份决定数据库使用权限,并将有权限对数据库内容进行修改和更新的用户登录信息存入日志文件中备查。另外,系统运用Microsoft SQL Server 2000的数据转换服务(Data Transformation Services)建立了相应的每日数据备份功能,保证了数据库在受损或者遭到攻击时,能够立即恢复,不至于造成整个系统的瘫痪。通过上述几项策略,使软件本身的安全措施颇具特色。

3 结束语

通过构建基于MVC模式的JSP体系结构,结合JDBC(JavaBean)技术和连接池(Connection Pool)技术进行数据库操作,可以充分发挥Java语言所独有的易用性、跨平台性和安全性。本系统已于2004年9月在银府加油城投入使用,运行状况良好,性能稳定,为公司提供了可靠的信息化手段,显著改善了工作效率,增强了公司的管理水平,提高了公司在本行业中的竞争力。

参考文献:

[1] Deepak A,John C.J2EE核心模式[M].刘天北译.北京:机械工业出版社,2005.

[2] 冯相忠.基于J2EE技术的电子商务系统的开发[J].计算机技术与发展,2007,17(8):33-36.

[3] 王启才.用Servlet/JSP构建基于WEB的管理信息系统[J].北京建筑工程学院学报,2004,20(4):71-74.

[4] Avedal K,Ayers D,Briggs T.JSP编程指南[M].黎　文,等译.北京:电子工业出版社,2001.

[5] 齐鲲鹏,顾　宏,唐　达.JSP数据库连接技术在构建信息网站中的研究[J].控制工程,2002,9(5):17-20.

(上接第235页)

```
    //输出每个属性名
    out.println("\t" + attrName +":");
    Enumeration valsInAttr = nextAttr.getStringValues();
    while(valsInAttr.hasMoreElements()){
    String nextValue = (String)valsInAttr.nextElement();
    //输出每个属性名的值
    out.println("\t\t" + nextValue );}
    }
    }
    try{
    ld.disconnect();
    }catch(LDAPException ee){
    out.println(ee.toString());}
    }
}
```

3 结束语

基于Web的目录服务信息系统提供给客户使用网页浏览器直接访问目录服务这样一种简单的方式,使得目录服务的信息访问、存储和检索非常方便快捷。文中介绍的设计和实现方法及应用实例,在中国地质大学(北京)校园网基于Web的目录服务网络计费系统应用过,获得了较好的效果。

参考文献:

[1] 张　蓓,李笑难,马　皓.基于目录服务的校园网用户管理系统[J].计算机工程,2000,26(增刊):291-292.

[2] Johnson S. LDAPman RFC2251[S/OL]. 2004. http://www.ldapman.org/ldap-rfcs.html.

[3] 焦　静.基于LDAP统一身份认证系统的设计与实现[D].西安:西北工业大学,2007:16-17.

[4] 曹元大,岳治宇.基于Servlet的Web数据库接口系统的设计与实现[J].北京理工大学学报,2000,20(4):452-453.

[5] Weltman R,Dahbura T.LDAP Programming with Java[M].Massachusetts,USA:Addison-Wesley Pub Co,2005:429-440.