在 Java CAPS 中使用 LDAP

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

文件号码 820–5624
2008 年 6 月

080820@20795

目录

◆ ◆ ◆ 第 1 章

在 Java CAPS 中使用 LDAP

此处列出的主题提供了有关如何在 Sun Java™ Composite Application Platform Suite (Java CAPS) 中使用轻量目录访问协议 (Lightweight Directory Access Protocol, LDAP) 的信息。

如果您有任何问题，请参见 `http://goldstar.stc.com/support` 中的 Java CAPS Web 站点。

LDAP 概述

轻量目录访问协议 (Lightweight Directory Access Protocol, LDAP) 是一种标准，用于使客户机可以查询和更新目录服务中的数据。

LDAP 目录中包含一系列的**条目**。条目由**属性**的集合和唯一标识条目的**标识名**组成。

在以下示例中，第一行指定 DN。后续行指定属性。

```
dn: cn=all, ou=Roles, dc=company, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: all
ou: Roles
```

DN 的各组成部分是按层次结构排列的（按具体程度由高到低）。因此，DN 中的最后一个组成部分标识目录的根条目。

每个属性都包含一个类型，以及一个或多个值。例如，属性 `ou: Roles` 的类型为 `ou`（组织单位），值为 `Roles`。**对象类**是一种属性，用于指定条目的必需属性和可选属性。您可以在 RFC 2256 中查找许多对象类的定义。

以上示例是以 LDAP 数据交换格式 (LDAP Data Interchange Format, LDIF) 表示的。还可以使用图形方式表示条目。

搜索 LDAP 目录时，使用**搜索过滤器**指定搜索条件。可以使用星号作为通配符。例如：

```
(cn=John S*)
```

将 LDAP 服务器用于系统信息库用户管理

可以将 Java CAPS 系统信息库配置为使用 LDAP 服务器进行用户管理。

当用户尝试登录系统信息库时，将检查用户名和密码是否与 LDAP 服务器中存储的用户名和密码匹配。此外，可以从服务器检索用户的角色列表，以授权用户访问系统信息库中的各种对象。

支持以下 LDAP 服务器：

- Sun Java™ System Directory Server 版本 5.1、5.2 和 6.*x*
- Microsoft 的 Active Directory（Windows Server 2003 随附的版本）
- OpenLDAP Directory Server 2.*x*

首先，必须配置 LDAP 服务器。请参见相应的章节：

- 第 6 页中的 “配置 Sun Java™ System Directory Server”
- 第 7 页中的 “配置 Active Directory 服务”
- 第 8 页中的 “配置 OpenLDAP Directory Server”

然后，配置系统信息库，使其可以找到 LDAP 服务器，并查找相应的信息（例如目录中包含用户的部分）。请参见第 10 页中的 “配置系统信息库”。

如果要对系统信息库和 LDAP 服务器之间的通信进行加密，请参见第 13 页中的 “SSL 支持”。

《管理 Java CAPS 用户》提供了有关系统信息库用户管理的基本信息。

配置 Sun Java™ System Directory Server

Sun Java System Directory Server 版本 5.1 和 5.2 包含以下主要组件：

- Directory Server
- Administration Server
- Directory Server 控制台

可以使用 Directory Server 控制台执行大多数管理任务。该控制台包含四个顶级选项卡：“任务”、“配置”、“目录”和“状态”。“目录”选项卡将目录条目显示为树。可以通过此选项卡浏览、显示和编辑所有条目和属性。

还可以通过编辑配置文件或使用命令行实用程序手动执行管理任务。

Sun Java System Directory Server 版本 6.*x* 提供以下方式来管理目录中的条目：

- Directory Service Control Center (DSCC)
- 目录编辑器
- `ldapmodify` 和 `ldapdelete` 命令行实用程序

DSCC 已集成到 Sun Java™ Web Console 中。DSCC 包含五个顶级选项卡：“一般任务”、“目录服务器”、“代理服务器”、“服务器组”和“设置”。

依次单击“目录服务器”选项卡、服务器的名称和“条目管理”选项卡后，将显示可以浏览、添加和修改条目的页面。目录信息树 (Directory Information Tree, DIT) 将显示在左侧。

还可以使用“一般任务”选项卡创建新条目或浏览数据。

注 – 有关如何执行以下步骤的详细信息，请参见随 Sun Java System Directory Server 提供的文档。

▼ 配置 Sun Java System Directory Server

1. **在 `People` 目录下创建 `admin` 用户和 `Administrator` 用户。**
2. **在顶层节点下创建角色 `all`、`administration` 和 `management`。**
3. **将创建的角色分配给 `admin` 用户和 `Administrator` 用户。**
4. **转至第 10 页中的“ 配置系统信息库 ”。**

配置 Active Directory 服务

Active Directory 是 Windows 2003 的一个关键部件。它提供了各种可管理性、安全性以及互操作性功能。主要的管理工具是名为“Active Directory 用户和计算机”的管理单元。

Active Directory 不支持角色的概念。因此，在 Active Directory 中，必须使用**组**的概念模拟 Java CAPS 角色。

您应在名为 `CAPSRoles` 的新组织单位中创建组，而不是在 `Users` 目录中创建组。

注 – 有关如何执行以下步骤的详细信息，请参见随 Active Directory 提供的文档。

▼ 配置 Active Directory 服务

1. **启动“Active Directory 用户和计算机”管理工具。**
2. **右键单击根节点，然后选择“新建”>“组织单位”。**
   将显示“新建对象 - 组织单位”对话框。
3. **在“名称”字段中，输入一个值（例如 `CAPSRoles`）。**
4. **单击“确定”。**
5. **在该组织单位下，创建以下组：`all`、`administration` 和 `management`。要创建组，请右键单击组织单位，然后选择“新建”>“组”。对组范围和组类型使用默认值。**
   添加组后，这些组将显示在该组织单位下。
6. **将 `admin` 用户和 `Administrator` 用户作为您所创建的所有组的成员添加，方法是：双击每个组，然后从对话框中选择 `admin` 和 `Administrator`。**
7. **转至第 10 页中的“ 配置系统信息库 ”。**

配置 OpenLDAP Directory Server

OpenLDAP 项目提供了 LDAP 协议的开源实现。LDAP 服务器将作为名为 `slapd` 的独立守护进程运行。主要的配置文件名为 `slapd.conf`。此文件包含全局信息、特定于后端的信息以及特定于数据库的信息。可以使用多种方式将条目添加到数据库，例如，可以使用 `slapadd` 程序。要搜索数据库，请使用 `ldapsearch` 程序。

有关详细信息，请参见 `http://www.openldap.org`。

注 – 有关如何执行以下步骤的详细信息，请参见随 OpenLDAP Directory Server 提供的文档。

▼ 配置 OpenLDAP Directory Server

1. **在用户所在的节点下创建 `admin` 用户和 `Administrator` 用户。**
2. **如果您的模式中没有用于角色的节点，请为将在以下步骤中创建的特定于 Java CAPS 的角色创建一个节点。例如：**

```
dn: ou=CAPSRoles, dc=sun, dc=com
objectClass: top
objectClass: organizationalUnit
ou: CAPSRoles
```

3 在角色所在的节点下创建角色 `all`、`administration` 和 `management`。将 `admin` 用户和 `Administrator` 用户作为每个角色的唯一成员添加。例如：

```
dn: cn=all, ou=CAPSRoles, dc=sun, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: all
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=sun, dc=com
uniqueMember: uid=Administrator, ou=People, dc=sun, dc=com

dn: cn=administration, ou=CAPSRoles, dc=sun, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: administration
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=sun, dc=com
uniqueMember: uid=Administrator, ou=People, dc=sun, dc=com

dn: cn=management, ou=CAPSRoles, dc=sun, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: management
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=sun, dc=com
uniqueMember: uid=Administrator, ou=People, dc=sun, dc=com
```

4 根据需要，将其他用户添加到一个或多个角色。例如：

```
dn: cn=all, ou=CAPSRoles, dc=sun, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: all
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=sun, dc=com
uniqueMember: uid=Administrator, ou=People, dc=sun, dc=com
uniqueMember: uid=userA, ou=People, dc=sun, dc=com
uniqueMember: uid=userB, ou=People, dc=sun, dc=com

dn: cn=administration, ou=CAPSRoles, dc=sun, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: administration
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=sun, dc=com
uniqueMember: uid=Administrator, ou=People, dc=sun, dc=com
uniqueMember: uid=userB, ou=People, dc=sun, dc=com

dn: cn=management, ou=CAPSRoles, dc=sun, dc=com
```

```
objectClass: top
objectClass: groupOfUniqueNames
cn: management
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=sun, dc=com
uniqueMember: uid=Administrator, ou=People, dc=sun, dc=com
```

5 转至第 10 页中的“ 配置系统信息库 ”。

配置系统信息库

要将 LDAP 服务器用于系统信息库用户管理，必须将 <Realm> 元素添加到系统信息库的 server.xml 文件中，该文件位于 JavaCAPS-install-dir/repository/repository/server/conf 目录中。

server.xml 文件包含用于指定用户数据库的平面文件实现的默认 <Realm> 元素。平面文件实现将使用 tomcat-users.xml 文件，该文件位于 JavaCAPS-install-dir/repository/repository/data/files 目录中。

下表介绍了 LDAP 版本的 <Realm> 元素所使用的属性。有关所有可能属性的详细介绍，请参见 org.apache.catalina.realm.JNDIRealm 类的 Tomcat 文档。

属性	描述
className	始终使用以下值：org.apache.catalina.realm.JNDIRealm
connectionURL	标识 LDAP 服务器的位置。包括 LDAP 服务器名称和 LDAP 服务器侦听请求的端口。
roleBase	角色搜索的基条目。如果未指定此属性，则搜索基为顶级目录上下文。
roleName	包含角色名称的角色条目中的属性。
roleSearch	用于选择角色条目的 LDAP 搜索过滤器。可以包含模式替换 {0}（对于标识名）和/或 {1}（对于已验证用户的用户名）。对于已验证用户（如 Administrator），在某些情况下应选择 {0} 选项。
roleSubtree	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的角色部分。要启用对整个子树的搜索，请将该值设置为 true。
userBase	包含用户的子树的基条目。如果未指定此属性，则搜索基为顶级上下文。
userPattern	用户目录条目的标识名 (Distinguished Name, DN) 的模式，遵循 java.text.MessageFormat 类所支持的语法，使用 {0} 表示应插入实际用户名的位置。

属性	描述
userRoleName	用户目录条目中的属性名称，包含为此用户分配的角色名称的零个或多个值。此外，可以使用 `roleName` 属性指定要从通过搜索目录找到的各个角色条目中检索的属性的名称。如果未指定 `userRoleName`，将从角色搜索中派生用户的所有角色。
userRoleNamePattern	角色目录条目的标识名 (Distinguished Name, DN) 的模式，遵循 `java.text.MessageFormat` 类所支持的语法，使用 `{0}` 表示实际的角色名称。在应插入实际用户名的 Java CAPS 中进行授权时，将使用此模式解析 DN 以获取实际的角色名称。
userSearch	用于在使用 `{0}` 替换用户名之后选择用户条目的 LDAP 搜索过滤器。
userSubtree	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的用户部分。要启用对整个子树的搜索，请将该值设置为 `true`。

▼ 配置系统信息库

1. **打开 `server.xml` 文件，该文件位于 `JavaCAPS-install-dir/repository/repository/server/conf` 目录中。**

2. **删除或注释掉默认的 `<Realm>` 元素。**

3. **如果使用的是 Sun Java System Directory Server，将以下 `<Realm>` 元素添加到 `<Engine>` 标记中。根据需要更改默认值。前面的表对属性进行了介绍。**

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldap://localhost:489"
    userBase="cn=People,dc=sun,dc=com"
    userSearch="(uid={0})"
    userSubtree="true"
    userRoleName="nsroledn"
    userRoleNamePattern="cn={0},dc=sun,dc=com"
    roleSubtree="true"
/>
```

4. **如果使用的是 Active Directory，将以下 `<Realm>` 元素添加到 `<Engine>` 标记中。根据需要更改默认值。前面的表对属性进行了介绍。**

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldap://localhost:389"
    userBase="cn=Users,dc=sun,dc=com"
    userSearch="(cn={0})"
    userSubtree="true"
    roleBase="ou=CAPSRoles,dc=sun,dc=com"
    roleName="cn"
    roleSearch="(member={0})"
```

```
    roleSubtree="true"
/>
```

5 **如果使用的是 OpenLDAP Directory Server，将以下 <Realm> 元素添加到 <Engine> 标记中。根据需要更改默认值。前面的表对属性进行了介绍。**

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldap://localhost:389"
    userBase="ou=People,dc=sun,dc=com"
    userSearch="(uid={0})"
    userSubtree="true"
    roleBase="ou=CAPSRoles,dc=sun,dc=com"
    roleName="cn"
    roleSearch="(uniquemember={0})"
    roleSubtree="true"
/>
```

6 **如果您的 LDAP 服务器未配置为允许匿名读取访问，请将 connectionName 属性和 connectionPassword 属性添加到 <Realm> 元素中。将第一个属性设置为 Administrator 用户的 DN。将第二个属性设置为用户的加密密码。请参阅以下示例。**

Sun Java System Directory Server :

```
connectionName="cn=Directory Manager"
connectionPassword="E451KDVb0OPcH+GN46OZcg=="
```

Active Directory :

```
connectionName="Administrator@sun.com"
connectionPassword="geEiVIbtO+DcH+GN46OZcg=="
```

OpenLDAP Directory Server :

```
connectionName="cn=Manager,dc=sun,dc=com"
connectionPassword="l/ZRt1cfNKc="
```

要对密码进行加密，请使用 encrypt 实用程序，该程序位于 JavaCAPS-install-dir/repository/repository/util 目录中。该实用程序的文件扩展名取决于您的平台。该实用程序使用未加密的密码作为参数。例如 :

```
C:\JavaCAPS6\repository\repository\util>encrypt mypwd
LCUApSkYpuE
```

7 **保存并关闭 server.xml 文件。**

8 **启动 LDAP 服务器。**

9 **关闭并重新启动系统信息库。**

SSL 支持

默认情况下，系统信息库和 LDAP 服务器之间的通信没有加密。

要对系统信息库和 LDAP 服务器之间的通信进行加密，请对本主题中先前所介绍的过程进行以下添加和修改。

配置 LDAP 服务器上的 SSL

确保已将 LDAP 服务器配置为使用安全套接字层 (Secure Sockets Layer, SSL)。有关详细信息，请参见随 LDAP 服务器提供的文档。

在准备下一步时，请将 LDAP 服务器的证书导出为文件。

导入 LDAP 服务器的证书

必须将 LDAP 服务器的证书添加到系统信息库的可信证书列表中。该列表位于名为 `cacerts` 的文件中。

在以下过程中，您将使用 `keytool` 程序。此程序随 Java SDK 一起提供。

▼ 导入 LDAP 服务器的证书

1. **导航至 `JDK-install-dir/jre/bin` 目录。**
   使用在安装系统信息库过程中指定的 JDK。

2. **运行以下命令：**

   ```
   keytool -import -trustcacerts -alias alias -file certificate_filename
   -keystore cacerts_filename
   ```

   对于 `-alias` 选项，您可以指定任何值。

   对于 `-file` 选项，请指定 LDAP 服务器的证书的全限定名称。例如：

   ```
   C:\mycertificate.cer
   ```

   对于 `-keystore` 选项，请指定 `cacerts` 文件的全限定名称。`cacerts` 文件位于 `JDK-install-dir/jre/lib/security` 目录中。例如：

   ```
   C:\Java\jdk1.6.0_06\jre\lib\security\cacerts
   ```

3. **出现提示时，请输入密钥库密码。默认密码为 `changeit`。**

4. **出现是否信任此证书的提示时，请输入 `yes`。**
   将显示以下消息：

   ```
   Certificate was added to keystore
   ```

修改 LDAP 服务器 URL

在 `server.xml` 文件的 `<Realm>` 元素中，按以下方式修改 LDAP 服务器的 URL：

- 将协议设置为 `ldaps`。
- 将端口号设置为 LDAP 服务器侦听 SSL 请求时使用的端口号。此端口号通常为 636。

例如：

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldaps://myldapserver:636"
    ...
```

将 LDAP 服务器用于 Sun JMS IQ Manager 用户管理

可以将 Sun JMS IQ Manager 配置为使用 LDAP 服务器进行用户管理。

领域是用于执行安全策略的用户、组和角色的集合。JMS IQ Manager 支持多个 LDAP 领域同时运行。

在执行以下步骤时，仅在连接具有有效的用户名和密码时才允许访问 JMS IQ Manager。

支持以下 LDAP 服务器：

- Sun Java System Directory Server 版本 5.1、5.2 和 6.*x*
- Microsoft 的 Active Directory（Windows Server 2003 随附的版本）
- OpenLDAP Directory Server 2.*x*

《管理 Java CAPS 用户》提供了与 Sun JMS IQ Manager 用户管理有关的基本信息。

配置 LDAP 服务器

在以下过程中，您将在 LDAP 服务器中创建用户和角色。

▼ 配置 LDAP 服务器

1. **创建一个或多个 JMS IQ Manager 用户。**
2. **创建一个或多个以下角色：**

角色	描述
application	允许客户机访问 JMS IQ Manager。
asadmin	允许使用 JMS 控制实用程序 (`stcmsctrlutil`) 或 Enterprise Manager，并允许客户机访问 JMS IQ Manager。

3 **根据需要为用户分配角色。**

配置 Sun JMS IQ Manager

必须配置 JMS IQ Manager，使其能够找到 LDAP 服务器，并查找相应的信息。

可以启用多个 LDAP 服务器。此外，您还可以指定默认领域。

▼ 配置 Sun JMS IQ Manager

1 **如果应用服务器没有运行，请启动应用服务器。**

2 **登录到 Configuration Agent。URL 的格式为** `http://hostname:port-number/configagent`**。将主机名设置为安装了应用服务器的计算机的 TCP/IP 主机名。将端口号设置为应用服务器的管理端口号。例如：**

```
http://localhost:4848/configagent
```

3 **在左侧窗格中，单击 JMS IQ Manager 节点（例如** `IQ_Manager_18007` **）。**

4 **单击“访问控制”选项卡。**

5 **确保选中了“需要验证”标签右侧的复选框。**

6 **如果要启用 Sun Java System Directory Server，则请选中“启用 Sun Java System Directory Server”标签右侧的复选框，然后单击“显示属性”。**

下表介绍了显示的属性。默认值将匹配 Sun Java System Directory Server 的标准模式。查看每个属性的默认值。如有需要，修改默认值。

属性	描述
命名提供者 URL	Java Naming and Directory Interface (JNDI) 服务提供者的 URL。 默认值为 `ldap://IP_address:589`。

属性	描述
命名初始工厂	创建初始上下文的工厂类的全限定名称。初始上下文是 JNDI 命名操作的起点。 默认值为 `com.sun.jndi.ldap.LdapCtxFactory`。
命名安全验证	JNDI 命名操作中所使用的安全级别。 默认值为 `simple`。
命名安全主体	用于连接 LDAP 服务器的安全主体。
命名安全凭证	命名安全主体的密码。 默认值为 `STC`。在您保存该值然后再次查看该值时，该值将被加密。
组中组 DN 属性的名称	组条目中标识名属性的名称。 默认值为 `entrydn`。
组 DN 中的组名称字段	组标识名中组名称字段的名称。 默认值为 `cn`。
组父 DN 下的用户组过滤器	用于检索用户的所有组的 LDAP 搜索过滤器。此属性遵循 `java.text.MessageFormat` 类所支持的语法，使用 `{1}` 表示用户的标识名应该插入的位置。 默认值为 `uniquemember={1}`。
组父 DN	组条目的父标识名。也就是说，此属性指定 LDAP 目录的组部分的根条目。
用户中角色名称属性的名称	用户条目中角色名称属性的名称。 默认值为 `nsroledn`。
角色 DN 中的角色名称字段	角色标识名中角色名称字段的名称。 默认值为 `cn`。
角色父 DN	角色条目的父标识名。也就是说，此属性指定 LDAP 目录的角色部分的根条目。
搜索组子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的组部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。

属性	描述
搜索角色子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的角色部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。
搜索用户子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的用户部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。
用户中用户 DN 属性的名称	用户条目中标识名属性的名称。 默认值为 `entrydn`。
用户中用户 ID 属性的名称	用户条目中用户 ID 属性的名称。 默认值为 `uid`。
用户父 DN	用户条目的父标识名。也就是说，此属性指定 LDAP 目录的用户部分的根条目。

7 如果要启用 Active Directory，则请选中“启用 Microsoft Active Directory Server”标签右侧的复选框，然后单击“显示属性”。

下表介绍了显示的属性。默认值将匹配 Active Directory 的标准模式。查看每个属性的默认值。如有需要，修改默认值。

属性	描述
命名提供者 URL	Java Naming and Directory Interface (JNDI) 服务提供者的 URL。 默认值为 `ldap://IP_address:389`。
命名初始工厂	创建初始上下文的工厂类的全限定名称。初始上下文是 JNDI 命名操作的起点。 默认值为 `com.sun.jndi.ldap.LdapCtxFactory`。
命名安全验证	JNDI 命名操作中所使用的安全级别。 默认值为 `simple`。
命名安全主体	用于连接 LDAP 服务器的安全主体。
命名安全凭证	命名安全主体的密码。 默认值为 `STC`。在您保存该值然后再次查看该值时，该值将被加密。

属性	描述
用户父 DN	用户条目的父标识名。也就是说，此属性指定 LDAP 目录的用户部分的根条目。
用户中用户 DN 属性的名称	用户条目中标识名属性的名称。 默认值为 `distinguishedName`。
用户中用户 ID 属性的名称	用户条目中用户 ID（即登录 ID）的名称。 默认值为 `sAMAccountName`。
角色父 DN	角色条目的父标识名。也就是说，此属性指定 LDAP 目录的角色部分的根条目。
角色中角色 DN 属性的名称	角色条目中标识名属性的名称。 默认值为 `cn`。
角色父 DN 下的用户角色过滤器	用于检索用户的所有角色的 LDAP 搜索过滤器。此属性遵循 `java.text.MessageFormat` 类所支持的语法，使用 `{1}` 表示用户的标识名应该插入的位置。 默认值为 `(&(member={1})(objectclass=group))`。
组父 DN	组条目的父标识名。也就是说，此属性指定 LDAP 目录的组部分的根条目。
组中组 DN 属性的名称	组条目中标识名属性的名称。 默认值为 `distinguishedName`。
组 DN 中的组名称字段	组标识名中组名称字段的名称。 默认值为 `cn`。
组父 DN 下的用户组过滤器	用于检索用户的所有组的 LDAP 搜索过滤器。此属性遵循 `java.text.MessageFormat` 类所支持的语法，使用 `{1}` 表示用户的标识名应该插入的位置。 默认值为 `(&(member={1})(objectclass=group))`。
搜索组子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的组部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。

属性	描述
搜索用户子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的用户部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。
搜索角色子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的角色部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。

8 如果要启用 OpenLDAP Directory Server，则请选中“启用通用 LDAP 服务器”标签右侧的复选框，然后单击“显示属性”。

下表介绍了显示的属性。查看每个属性的默认值。如有需要，修改默认值。

属性	描述
命名提供者 URL	Java Naming and Directory Interface (JNDI) 服务提供者的 URL。 默认值为 `ldap://IP_address:489`。
命名初始工厂	创建初始上下文的工厂类的全限定名称。初始上下文是 JNDI 命名操作的起点。 默认值为 `com.sun.jndi.ldap.LdapCtxFactory`。
命名安全验证	JNDI 命名操作中所使用的安全级别。 默认值为 `simple`。
用户父 DN	用户条目的父标识名。也就是说，此属性指定 LDAP 目录的用户部分的根条目。
用户中用户 ID 属性的名称	用户条目中用户 ID 属性的名称。 默认值为 `uid`。
角色父 DN	角色条目的父标识名。也就是说，此属性指定 LDAP 目录的角色部分的根条目。
角色中角色名称属性的名称	用户条目中角色名称属性的名称。 默认值为 `cn`。

属性	描述
角色父 DN 下的用户角色过滤器	用于检索用户的所有角色的 LDAP 搜索过滤器。此属性遵循 `java.text.MessageFormat` 类所支持的语法，使用 `{1}` 表示用户的标识名应该插入的位置。 默认值为 `uniquemember={1}`。
组 DN 中的组名称字段	组标识名中组名称字段的名称。 默认值为 `cn`。
组父 DN	组条目的父标识名。也就是说，此属性指定 LDAP 目录的组部分的根条目。
组父 DN 下的用户组过滤器	用于检索用户的所有组的 LDAP 搜索过滤器。此属性遵循 `java.text.MessageFormat` 类所支持的语法，使用 `{1}` 表示用户的标识名应该插入的位置。 默认值为 `uniquemember={1}`。
搜索组子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的组部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。
搜索用户子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的用户部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。
搜索角色子树	默认情况下，将仅在根条目的下一级搜索 LDAP 目录的角色部分。要启用对整个子树的搜索，请将该值设置为 `true`。 默认值为 `false`。

9. **如果要更改默认领域，请从“默认领域”下拉列表中选择该领域。**
10. **单击“保存”。**

将 LDAP 服务器用于 Enterprise Manager 用户管理

可以将 Enterprise Manager 配置为使用 LDAP 服务器进行用户管理。

支持以下 LDAP 服务器：

- Sun Java System Directory Server 版本 5.1、5.2 和 6.*x*
- Microsoft 的 Active Directory（Windows Server 2003 随附的版本）

- OpenLDAP Directory Server 2.*x*

首先，应配置 LDAP 服务器。然后，配置 Enterprise Manager 服务器，使其可以找到 LDAP 服务器，并查找相应的信息（例如目录中包含用户的部分）。

《管理 Java CAPS 用户》提供了有关 Enterprise Manager 用户管理的基本信息。

配置 Sun Java System Directory Server

Sun Java System Directory Server 版本 5.1 和 5.2 包含以下主要组件：

- Directory Server
- Administration Server
- Directory Server 控制台

可以使用 Directory Server 控制台执行大多数管理任务。该控制台包含四个顶级选项卡：“任务”、“配置”、“目录”和“状态”。“目录”选项卡将目录条目显示为树。可以通过此选项卡浏览、显示和编辑所有条目和属性。

还可以通过编辑配置文件或使用命令行实用程序手动执行管理任务。

Sun Java System Directory Server 版本 6.*x* 提供以下方式来管理目录中的条目：

- Directory Service Control Center (DSCC)
- 目录编辑器
- `ldapmodify` 和 `ldapdelete` 命令行实用程序

DSCC 已集成到 Sun Java™ Web Console 中。DSCC 包含五个顶级选项卡：“一般任务”、“目录服务器”、“代理服务器”、“服务器组”和“设置”。

依次单击“目录服务器”选项卡、服务器的名称和“条目管理”选项卡后，将显示可以浏览、添加和修改条目的页面。目录信息树 (Directory Information Tree, DIT) 将显示在左侧。

还可以使用“一般任务”选项卡创建新条目或浏览数据。

注 – 有关如何执行以下步骤的详细信息，请参见随 Sun Java System Directory Server 提供的文档。

▼ 配置 Sun Java System Directory Server

1. **在 `People` 目录下创建 `admin` 用户和 `Administrator` 用户。**
2. **在顶部节点下创建以下角色：**
   - Deployment

- User Management
- Read-Only Monitor
- Controlling Monitor
- JMS Read-Only Monitor
- JMS Read-Write Monitor
- Manager

3 **将创建的角色分配给 `admin` 用户和 `Administrator` 用户。**

4 **转至第 24 页中的“ 配置 Enterprise Manager Server ”。**

配置 Active Directory 服务

Active Directory 是 Windows 2003 的一个关键部件。它提供了各种可管理性、安全性以及互操作性功能。主要的管理工具是名为“Active Directory 用户和计算机”的管理单元。

Active Directory 不支持角色的概念。因此，在 Active Directory 中，必须使用**组**的概念模拟 Enterprise Manager 角色。

注 – 有关如何执行以下步骤的详细信息，请参见随 Active Directory 提供的文档。

▼ 配置 Active Directory 服务

1 **启动“Active Directory 用户和计算机”管理工具。**

2 **右键单击根节点，然后选择“新建”>“组织单位”。**
将显示“新建对象 - 组织单位”对话框。

3 **在“名称”字段中，输入一个值（例如 `EntMgrRoles`）。**

4 **单击“确定”。**

5 **在该组织单位下，创建以下组：**

- Deployment
- User Management
- Read-Only Monitor
- Controlling Monitor
- JMS Read-Only Monitor
- JMS Read-Write Monitor
- Manager

添加组后，这些组将显示在该组织单位下。

6 将 admin 用户和 Administrator 用户作为您所创建的所有组的成员添加，方法是：双击每个组，然后从对话框中选择 admin 和 Administrator。

7 转至第 24 页中的“配置 Enterprise Manager Server”。

配置 OpenLDAP Directory Server

OpenLDAP 项目提供了 LDAP 协议的开源实现。LDAP 服务器将作为名为 slapd 的独立守护进程运行。主要的配置文件名为 slapd.conf。此文件包含全局信息、特定于后端的信息以及特定于数据库的信息。可以使用多种方式将条目添加到数据库，例如，可以使用 slapadd 程序。要搜索数据库，请使用 ldapsearch 程序。

有关详细信息，请参见 http://www.openldap.org。

注 – 有关如何执行以下步骤的详细信息，请参见随 OpenLDAP Directory Server 提供的文档。

▼ 配置 OpenLDAP Directory Server

1 在用户所在的节点下创建 admin 用户和 Administrator 用户。

2 如果您的模式中没有用于角色的节点，请为将在以下步骤中创建的 Enterprise Manager 角色创建一个节点。

3 在角色所在的节点下创建以下角色：

- Deployment
- User Management
- Read-Only Monitor
- Controlling Monitor
- JMS Read-Only Monitor
- JMS Read-Write Monitor
- Manager

4 将 admin 用户和 Administrator 用户作为每个角色的唯一成员添加。

5 根据需要，将其他用户添加到一个或多个角色。

6 转至第 24 页中的“配置 Enterprise Manager Server”。

配置 Enterprise Manager Server

配置 LDAP 服务器后，可配置 Enterprise Manager Server，使其可以找到 LDAP 服务器，并查找相应的信息。

必须编辑以下 Enterprise Manager 文件：`web.xml` 和 `ldap.properties`。

▼ 配置 Enterprise Manager Server

1. **关闭 Enterprise Manager 的服务器组件。**
2. **打开 `web.xml` 文件，该文件位于 `JavaCAPS-install-dir/emanager/server/webapps/sentinel/WEB-INF` 目录中。**
3. **找到以下行：**

```
<param-name>com.stc.emanager.sentinel.authHandler</param-name>
<param-value>com.stc.cas.auth.provider.tomcat.TomcatPasswordHandler</param-value>
```

4. **将参数值更改为：**

```
<param-value>com.stc.cas.auth.provider.ldap.LDAPHandler</param-value>
```

5. **保存 `web.xml` 文件。**
6. **打开 `ldap.properties` 文件，该文件位于 `JavaCAPS-install-dir/emanager/server/webapps/sentinel/WEB-INF/classes` 目录中。**
7. **下表介绍了 `ldap.properties` 文件中出现的所有属性。针对您的 LDAP 服务器编辑该部分中的属性，并确保未注释掉这些属性。**

属性	描述
com.stc.sentinel.auth.ldap.serverType	LDAP 服务器的类型。
com.stc.sentinel.auth.ldap.serverUrl	LDAP 服务器的 URL。
com.stc.sentinel.auth.ldap.searchFilter	用户条目中用户 ID 属性的名称。
com.stc.sentinel.auth.ldap.searchBase	Enterprise Manager 将在其中搜索用户的 LDAP 目录部分的根条目。
com.stc.sentinel.auth.ldap.searchScope	当前未使用此属性。
com.stc.sentinel.auth.ldap.bindDN	用于连接 LDAP 服务器的安全主体。
com.stc.sentinel.auth.ldap.bindPassword	安全主体的密码。

属性	描述
com.stc.sentinel.auth.ldap.referral	LDAP 引用策略。默认值为 `follow`，表示将自动执行 LDAP 引用。请注意，必须在 LDAP 服务器中启用引用。其他有效值为 `throw`（对于引用异常）和 `ignore`。 此属性为可选属性。 此属性仅出现在 Active Directory 和 OpenLDAP 属性集中。
com.stc.sentinel.auth.ldap.roleAttribute	用户条目中角色名称属性的名称。
com.stc.sentinel.auth.ldap.roleBaseDN	Enterprise Manager 将在其中搜索角色的 LDAP 目录部分的根条目。 此属性仅出现在 OpenLDAP 属性集中。
com.stc.sentinel.auth.ldap.rolePattern	使用此属性可以为角色名称配置模式匹配。在 LDAP 目录中，您可以将 Enterprise Manager 用户放置在与其他用户不同的单独业务范围中。 此属性仅出现在 Active Directory 属性集中。

8. **保存 `ldap.properties` 文件。**

9. **启动 Enterprise Manager 的服务器组件。**

动态指定应用程序配置属性

要指定应用程序配置属性，可以使用静态方法，也可以使用动态方法。

若使用静态方法，请在设计时在 NetBeans IDE 中指定属性值。此属性值包含在应用程序文件中。如果需要在部署后更改该值，则必须在 NetBeans IDE 中更改该值，重新生成应用程序文件，并重新部署应用程序文件。

若使用动态方法，请在设计时指定 LDAP URL。此 URL 必须指向 LDAP 服务器中的属性。部署应用程序文件时，实际值是从 LDAP 服务器检索的。部署后可以在 LDAP 服务器中更改该值，而不执行静态方法的步骤。但是，为了使更改生效，必须先禁用此应用程序文件，然后再将其重新启用。

可以对接受字符串值（包括密码）、数字值或布尔值的属性使用此功能。

注 – 另一种用于更新属性值的方法不需要使用 LDAP。在 asadmin 工具中，运行 extract-caps-application-configuration 命令。指定应用程序文件的配置属性将被提取为属性文件。更新一个或多个属性值，然后运行 import-caps-configuration 命令。重新启动应用程序。

启用应用服务器以访问 LDAP 服务器

在此任务中，将编辑用于指定应用服务器如何访问 LDAP 服务器的属性。

▼ 启用应用服务器以访问 LDAP 服务器

1. **启动 Sun Java System Application Server 中随附的 asadmin 工具。**

2. **运行 export-caps-ldap-configuration 命令。您必须指定要用于存储 LDAP.properties 文件的目录。**

   ```
   asadmin> export-caps-ldap-configuration --capsconfigdir c:\temp
   ```

   将生成 LDAP.properties 文件。

3. **使用文本编辑器打开 LDAP.properties 文件。**

4. **设置以下属性的值，这些属性用于指定如何访问 LDAP 服务器。**

   - host
   - port
   - sslport
   - password
   - loginDN

   ldapVersion 是可选属性。可以将此属性设置为任何数字值。

5. **保存 LDAP.properties 文件。**

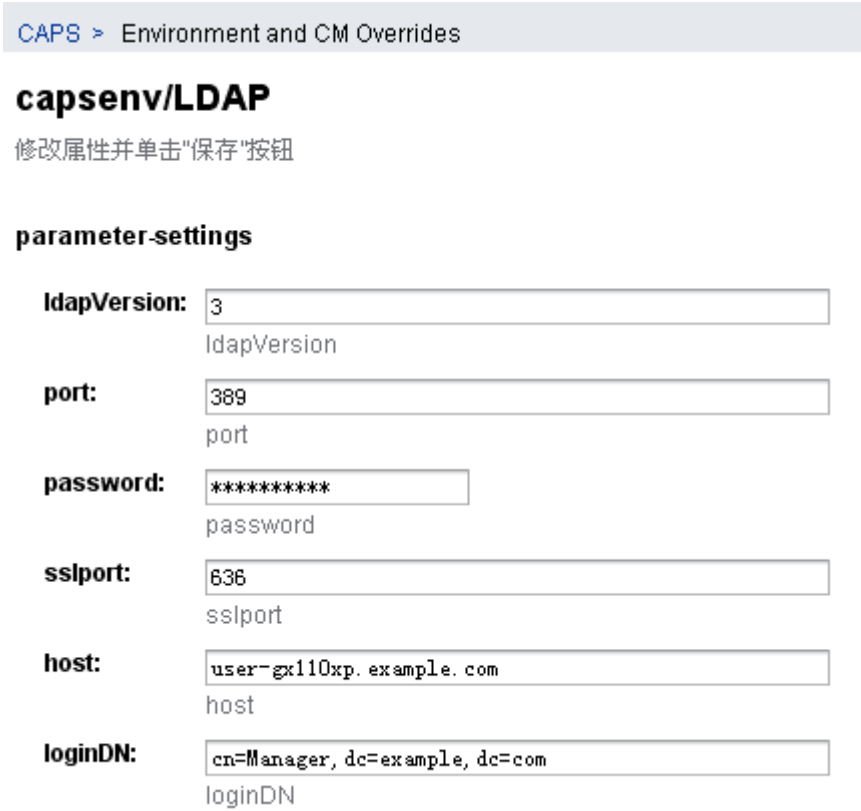
6. **运行 import-caps-configuration 命令。必须指定包含 LDAP.properties 文件的目录。**

   ```
   asadmin> import-caps-configuration c:\temp
   ```

7. **启动 Sun Java System Application Server 中随附的管理控制台。**

8. **在左窗格中，依次展开 CAPS 节点、Environment and CM Overrides 节点和 Environment Overrides 节点。选择 capsenv/LDAP 节点。**

   属性字段将显示在右窗格中。现在即可从管理控制台更新这些属性。也可以更新 LDAP.properties 文件，然后再次运行 import-caps-configuration 命令。



为属性指定 LDAP URL

以下是可以在 Java CAPS 中使用的 LDAP URL 的两个示例：

```
ldap://uid=BatchFTP_TargetFileName,ou=Batch_Adapter,dc=Adapters,dc=sun,dc=com?cn
ldap://uid=BatchFTP_Password,ou=Batch_Adapter,dc=Adapters,dc=sun,dc=com?cn
```

指向 LDAP 服务器中属性值的正确路径取决于目录结构。

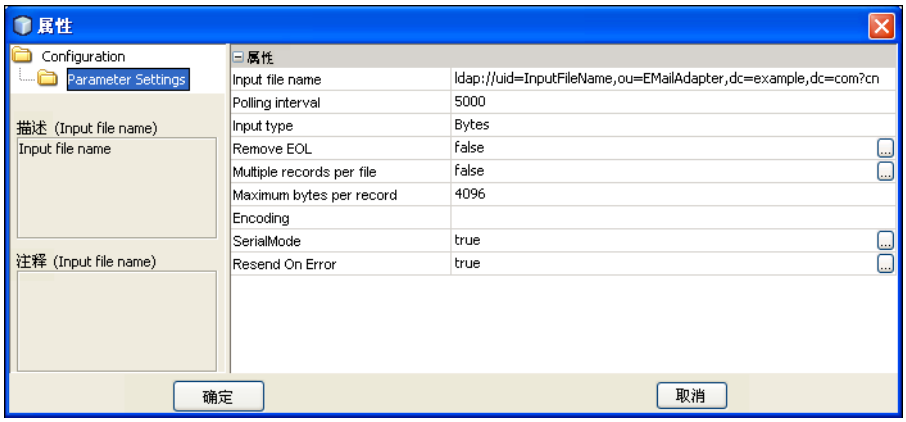
请勿在 LDAP URL 中包含反斜线字符 ()。

RFC 2255 定义了 LDAP URL 的格式。您可以在 `http://www.ietf.org/rfc.html` 中查看 RFC。

▼ 为属性指定 LDAP URL

1 在 NetBeans IDE 中，访问包含此属性的“属性”对话框。

2 输入指向 LDAP 服务器中对应属性的 LDAP URL。
在以下屏幕捕获中，Input File Name（输入文件名）属性设置为 LDAP URL。



3 转至 LDAP 服务器，然后输入实际值。

4 部署应用程序文件时，请确保 LDAP 服务器正在运行。如果未运行 LDAP 服务器，则部署将不会成功。

索引

组