管理员指南

iPlanet Directory Server

5.1 版

816-4121-10
2002 年 2 月

目录

表目录

关于本手册

iPlanet Directory Server 5.1 是一个功能强大、伸缩自如的分布式目录服务器，以符合业界标准的轻型目录访问协议 (LDAP) 为基础。 iPlanet Directory Server 是集中化、分布式数据存储库的基础，通过与贸易伙伴间的 Extranet 或公共 Internet，可用于您的内部 Intranet。

本*管理员指南*将介绍维护 iPlanet Directory Server 的目录服务所需的所有管理任务。

有关此版本 iPlanet Directory Server 的新增功能和增强功能的最新信息，请参阅以下网址的在线发行声明：

```
http://docs.iplanet.com/docs/manuals/directory.html
```

开卷必读

本手册介绍如何管理目录服务器及其内容。但书中并未详细介绍基本目录和体系结构的概念，而后者却是成功部署、安装和管理目录服务所必需的。应熟悉这些概念，在 *iPlanet Directory Server 部署指南*中对它们进行了介绍。

有关如何配置适用于 Solaris 9 操作环境的 iPlanet Directory Server 5.1，请参阅 *Solaris 9 System Administration Naming and Directory Services: (DNS, NIS and LDAP)* 中 “iPlanet Directory Server 5.1 Configuration”章节。

同时，*通过 iPlanet Console 管理服务器*中还包含有关如何使用 iPlanet 服务器的常规背景信息。尝试管理 iPlanet Directory Server 之前，应仔细阅读并理解书中的概念。

本书中所用的约定

本部分介绍本书中所用的印刷约定。

等宽字体 — 该字体用于表示在文本中出现的诸如属性和对象类的名称的文字。它也用于表示 URL、文件名及示例。

斜体 — 该字体用于表示强调和新术语，以及必须用实际值替换的文本 （比如路径名称中的占位符）。

大于符号 (>) 用作连续菜单选择之间的分隔符。例如，“对象” > “新建” > “用户”的意思是：下拉 “对象”菜单，拖动鼠标以突出显示 “新建”，然后沿 “新建”子菜单拖动鼠标，选择其中的 “用户”。

注意 “注意”、“警告”和 “提示”突出显示重要的条件或限制。继续某项任务前，请务必阅读这些信息。

本书使用以下格式表示路径和文件名：

```
/var/ds5/slapd-serverID/...
```

serverID 代表配置服务器时赋予该服务器的标识。例如，如果赋予目录服务器的名称 `phonebook`，则实际的路径为：

```
/var/ds5/slapd-phonebook/...
```

相关信息

就面向 iPlanet Directory Server 的文档而言，还包括下列指南：

iPlanet Directory Server 部署指南。概述 iPlanet Directory Server 的部署方案。其中包含部署示例。

iPlanet Directory Server 配置、命令和文件参考指南。提供有关 Directory Server 随带的命令行脚本、配置属性及日志文件的参考信息。

iPlanet Directory Server 模式参考指南。有关 Directory Server 附带的 LDAP 模式和用于客户机应用程序的信息。

其它有用的 iPlanet 信息可到下列 Internet 位置查找：

- iPlanet 产品联机文档：
  `http://docs.iplanet.com/docs/manuals/`
- iPlanet 产品情况：
  `http://www.iplanet.com/support/technical_resources/`
- iPlanet 专业服务信息：
  `http://www.iplanet.com/services/professional_services_3_3.html`
- Sun 企业服务 — Solaris 补丁程序和支持：
  `http://www.sun.com/service/`
- iPlanet 开发人员信息：
  `http://developer.iplanet.com/`
- iPlanet 学习解决方案：
  `http://www.iplanet.com/learning/index.html`
- iPlanet 产品数据表：
  `http://www.iplanet.com/products/index.html`

第 1 章

iPlanet Directory Server 简介

iPlanet Directory Server 产品包括管理多个目录的 iPlanet Directory Server 和通过图形界面管理这两个服务器的 iPlanet Console。本章概述有关 iPlanet Directory Server 的信息以及使用控制台建立目录管理服务所需的最基本的任务。

其中包含以下部分：

- iPlanet Directory Server 管理概述
- 使用 iPlanet Directory Server Console
- 启动和停止 iPlanet Directory Server
- 配置 LDAP 参数
- 在启用 SSL 的情况下启动服务器

iPlanet Directory Server 管理概述

iPlanet Directory Server 是一种功能强大、伸缩自如的服务器软件，设计用于管理整个企业范围内的用户和资源目录。它基于一个称做轻型目录访问协议 (LDAP) 的开放式系统服务器协议。在计算机上，iPlanet Directory Server 以 `ns-slapd` 进程或服务的形式运行。该服务器管理着目录数据库，同时对客户请求作出响应。

大部分 iPlanet Directory Server 管理任务是通过 Administration Server 执行的，该服务器软件是由 iPlanet 提供的、用于帮助管理 iPlanet Directory Server（以及其它所有 iPlanet 服务器）的辅助服务器。 iPlanet Console 是 Administration Server 的图形界面。 *iPlanet Directory Server Console* 是 iPlanet Console 的组成部分，专门设计用于 iPlanet Directory Server。

大部分 iPlanet Directory Server 管理任务都可以通过 iPlanet Directory Server Console 执行。通过编辑配置文件或使用命令行实用程序，也可以手动执行管理任务。有关 iPlanet Console 的详细信息，请参阅*通过 iPlanet Console 管理服务器*。

使用 iPlanet Directory Server Console

iPlanet Directory Server Console 是作为 iPlanet Console 的一个独立窗口访问的界面。 iPlanet Directory Server Console 需从 iPlanet Console 中启动，有关说明见以下步骤。

启动 iPlanet Directory Server Console

1. 检查目录服务器的守护程序 `slapd-`*serverID* 是否处于运行状态。如果未运行，则以 root 用户身份输入下列启动命令：

   ```
   # /usr/sbin/directoryserver start
   ```

2. 检查管理服务器的守护程序 `admin-serv` 是否处于运行状态。如果未运行，则以 root 用户身份输入下列启动命令：

   ```
   # /usr/sbin/directoryserver start-admin
   ```

3. 输入以下命令，启动 iPlanet Console：

   ```
   # /usr/sbin/directoryserver startconsole
   ```

   此时显示控制台登录窗口。或者，如果配置目录（包含 `o=NetscapeRoot` 后缀的目录）储存于一个独立的 Directory Server 实例中，则所显示的窗口要求输入管理员用户 DN、口令及此目录服务器的管理服务器 URL。

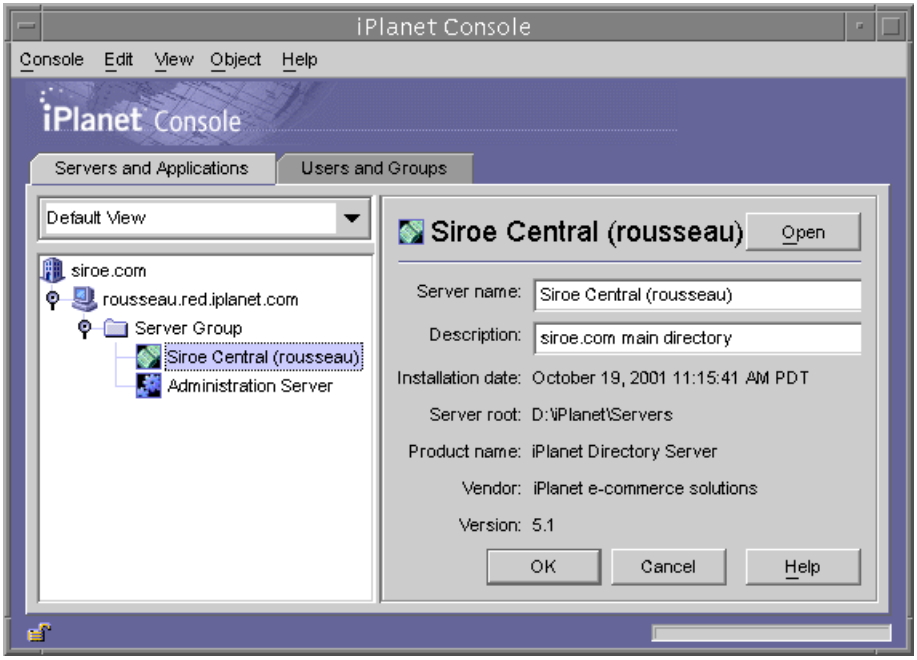
4. 使用满足下列条件的用户绑定 DN 和口令进行登录：该用户对所要执行的操作拥有足够的访问权限。例如，可以使用 `cn=Directory Manager` 及相应的口令。

   此时显示 iPlanet Console。

5. 浏览左侧导航窗口中的目录树，找到为 Directory Server 提供主机服务的计算机，然后单击它的名称或图标以显示它的一般属性。

   要编辑目录服务器的名称和说明，单击 “编辑”按钮。在文本框中输入新的名称和说明。该名称将出现在左侧的目录树中，如下图所示。

图 1-1 iPlanet Console



   单击 “确定”设置新的名称和说明。

6. 双击目录树中您的 Directory Server 的名称，或者单击 “打开”按钮，以显示用于管理该目录服务器的 iPlanet Directory Server Console。

Directory Server Console 概览

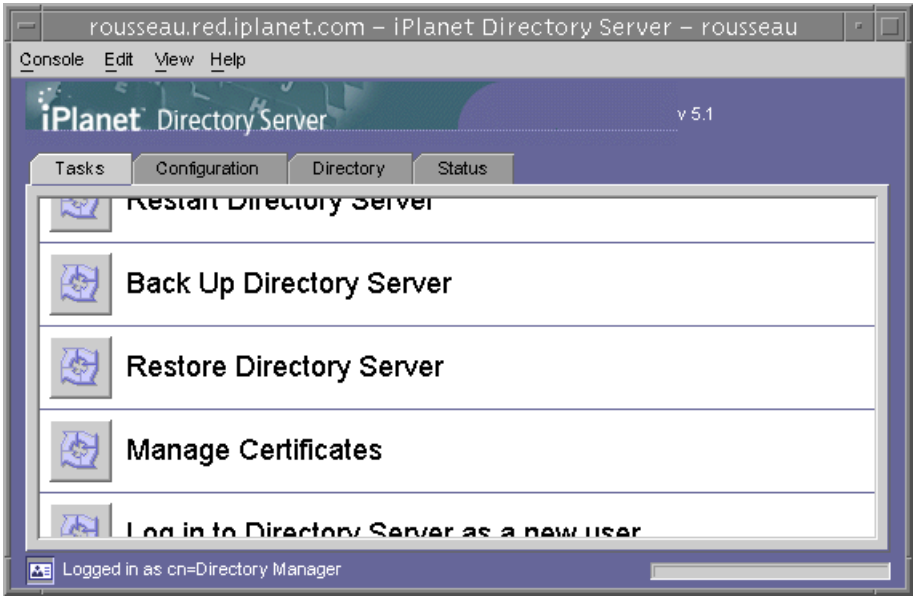
Directory Server Console 提供对 Directory Server 实例进行浏览和执行管理操作的界面。它始终显示四个选项卡，通过这些选项卡可以访问所有的 Directory Server 功能：

- “任务”选项卡
- “配置”选项卡
- “目录”选项卡
- “状态”选项卡

“任务”选项卡

“任务”选项卡是打开 Directory Server Console 后首先看到的界面。如下图所示，它包括用于执行所有重要管理任务 （例如启动或停止 Directory Server）的按钮。要查看所有任务及其对应的按钮，最好调整控制台窗口的大小。

图 1-2 Directory Server Console 的 “任务”选项卡



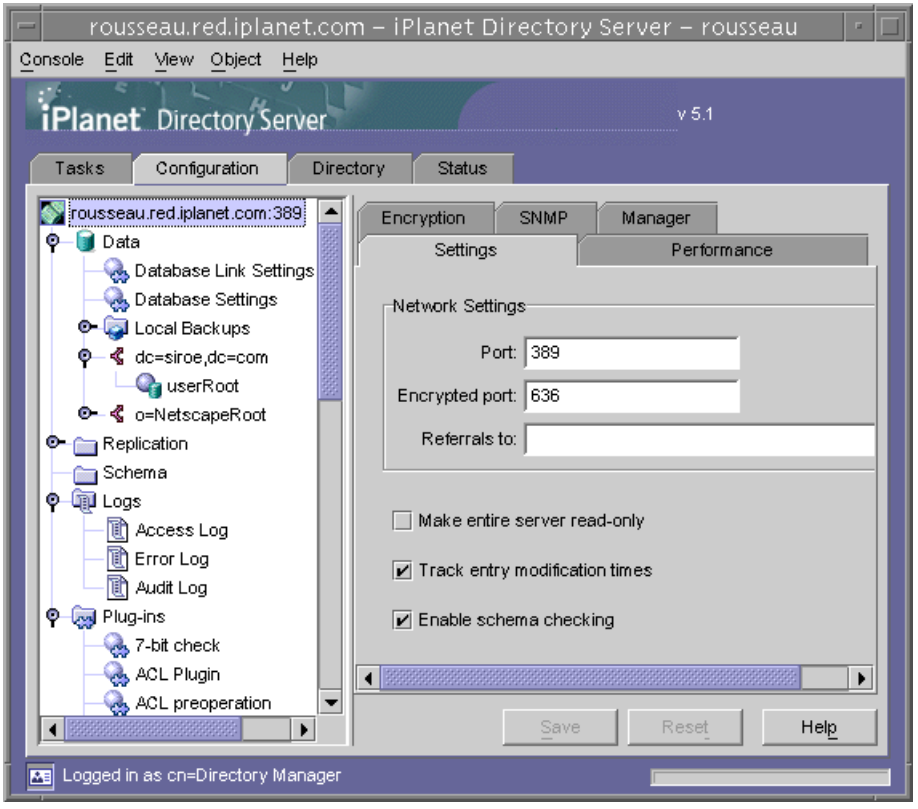
必须以具有目录管理员权限的用户身份登录才可以执行这些任务。如果执行任务时没有足够的权限，控制台就会提示输入目录管理员的 DN 和口令。

“配置”选项卡

Directory Server Console 的 “配置”选项卡提供查看和修改所有目录设置 （如数据库、后缀、复制、模式、日志和插件等设置）的界面和对话框。这些对话框只有在以具有目录管理员权限的用户身份登录后方可用或有效。

该选项卡的左侧窗口中包括所有配置功能的目录树，而右边窗口则显示具体管理各项功能的界面。这些界面通常还包含其它选项卡、对话框或弹出窗口。例如，下图显示整个目录的常规设置。

图 1-3 Directory Server Console 的 “配置”选项卡



在左边目录树中选中一个可配置项时，该项的当前设置将显示在右边窗口的一个或多个选项卡中。根据不同的设置，有些更改在保存后立即生效，而有些更改则在重新启动服务器后方才有效。有关这些设置的说明和相关的操作，请参阅本指南中介绍各项功能的章节。

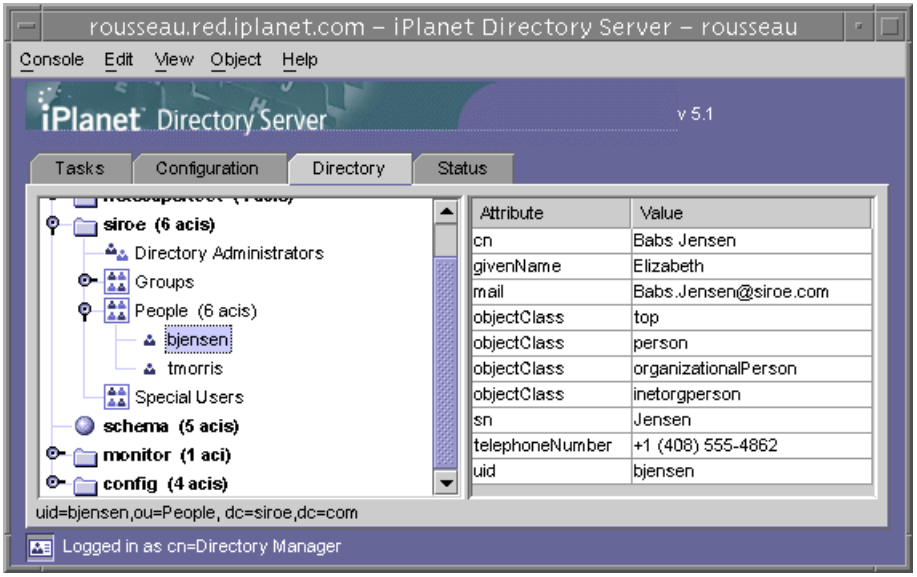
若选项卡中有未保存的更改，则在该选项卡名称旁边用红色的标记指明。即使配置另一项或者对其它主选项卡进行更改，未保存的更改也将一直保留在选项卡上。“保存”和“复位”按钮适用于给定可配置项的所有选项卡，但不会影响其它项的未保存设置。

大多数文本字段要求以正确的语法输入设置值。默认情况下，如果语法不正确，则用红色突出显示该设置的标签和输入的值，直至语法正确。“保存”按钮在所有设置的语法都正确后才可以使用。可以从“编辑”>“首选项”对话框的“其他”选项卡中选择使用斜体字来突出显示不正确的值，或者完全不用突出显示。

“目录”选项卡

控制台的“目录”选项卡以树状形式显示目录条目，以方便用户查找。在该选项卡中，可以浏览、显示和编辑所有条目以及它们包含的属性。

图 1-4 Directory Server Console 的“目录”选项卡



如果在登录期间给定的绑定 DN 有足够的访问权限，则配置条目可以按常规条目进行查看，并且可以直接修改。但是，为安全地更改配置设置，应始终使用通过 “配置”选项卡提供的对话框。

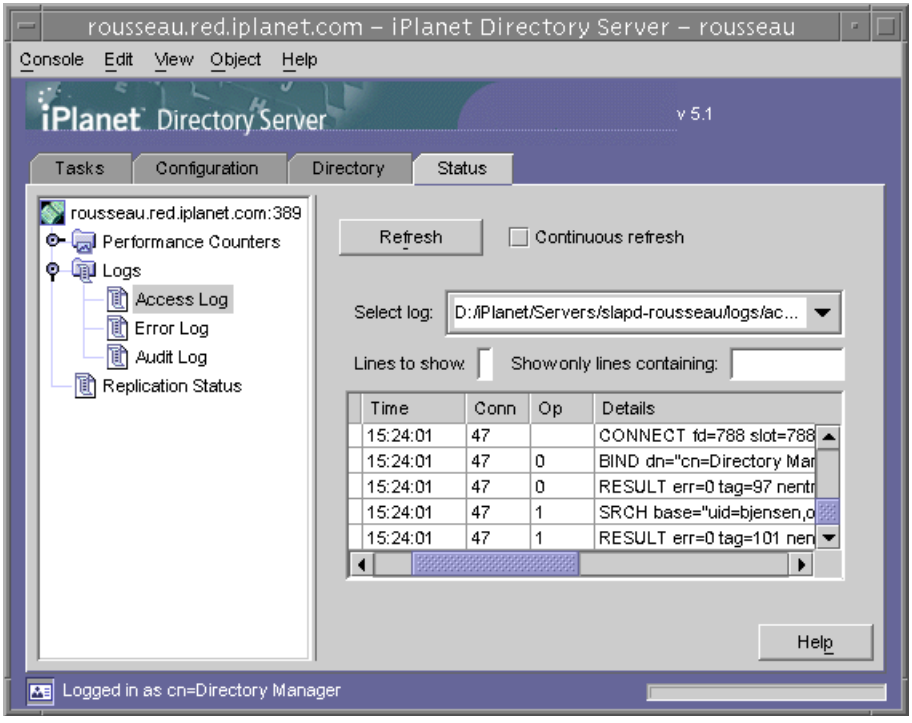
“视图”菜单提供几个选项，用于更改 “目录”选项卡的布局和内容。新的布局选项包括在单个目录树中显示所有条目 （包含叶条目）以及在右边窗口显示属性等。默认的布局是在右边窗口查看叶条目，而不在左边的目录树。

“视图” > “显示”选项启用 ACI 计数、角色计数和去活状态图标用于目录树中的所有条目。在上一幅图中， ACI 计数和叶条目显示在左边的树中，而所选条目的属性值显示在右边窗口中。

“状态”选项卡

“状态”选项卡显示服务器统计数据和日志消息。左边的树列出所有状态项，而右边窗口显示每个所选项的内容。例如，下图显示一个日志条目表格。

图 1-5 Directory Server Console 的 “状态”选项卡



从控制台查看当前绑定 DN

单击显示器左下角的登录图标，可以查看用于登录 iPlanet Directory Server Console 的绑定 DN。如下所示，当前绑定 DN 显示在登录图标的旁边。

更改登录身份

从 iPlanet Directory Server Console 创建或管理条目以及第一次访问 iPlanet Console 时，可以选择通过提供绑定 DN 和口令进行登录。这用于表明要访问目录树的用户身份并确定其是否有执行操作所需的访问权限。

第一次启动 iPlanet Console 时，可以通过目录管理员 DN 进行登录。您可以随时选择以其它用户身份进行登录，而无须停止和重新启动控制台。

要在 iPlanet Console 中更改登录身份：

1. 在 iPlanet Directory Server Console 中，选择 “任务” 选项卡并单击 “作为新用户登录到 iPlanet Directory Server” 标签旁边的按钮。或者，在另一个控制台选项卡中，从控制台菜单中选择 “作为新用户登录” 菜单项。

   此时显示一个登录对话框。

2. 输入新的 DN 和口令，然后单击 “确定”。

   输入要绑定到服务器的条目的完整特异名称。例如，如果想绑定为目录管理员，则在 “特异名称” 文本框中输入以下内容：

   ```
   cn=Directory Manager
   ```

目录管理员 DN 和口令将在后面的部分做进一步的介绍。

配置目录管理员

*目录管理员*是有相应权限的数据库管理员，类似于 UNIX 中的 root 用户。访问控制不适用于定义为目录管理员的条目。缺省值为 `cn=Directory Manager`。

该用户口令在 `nsslapd-rootdn` 属性中定义。

使用 Directory Server Console 来更改目录管理员 DN 和口令，以及该口令使用的加密模式：

1. 以目录管理员的身份登录 Directory Console。

   如果已登录 Directory Console，请参阅第 32 页上的 “更改登录身份”，了解如何以其它用户的身份进行登录。

2. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树中的顶级条目。

3. 在右侧窗口中，选择 “管理员”选项卡。

4. 在 “根目录 DN”字段中输入目录管理员的新特异名称。

   缺省值是 `cn=Directory Manager`。

5. 从 “管理员口令加密”下拉菜单中，选择希望服务器用于存储目录管理员口令的储存模式。

6. 用提供的文本字段输入新的口令并加以确认。

7. 单击 “保存”。

启动帮助系统

iPlanet Directory Server 的帮助系统依赖于 iPlanet Administration Server。如果在 Administration Server 的远程计算机上运行 iPlanet Directory Server Console，则需要确认以下各项：

Administration Server 被授予了客户机 IP 地址。运行 iPlanet Directory Server Console 的计算机需要有对 Administration Server 的访问权限。要将 Administration Server 配置为接受客户机 IP 地址，请在 Administration Server 中执行以下操作：

1. 启动 iPlanet Administration Server Console。该控制台应与 Administration Server 在同一计算机上运行。

2. 单击 “配置”选项卡，然后单击 “网络”选项卡。

3. 在连接限制设置中，从下拉菜单中选择 “允许的 IP 地址”。单击 “编辑”。

4. 将 “IP 地址”字段编辑为：*.*.*.*

   这样即允许所有客户机访问 Administration Server。

5. 重新启动 Administration Server。现在即可通过单击 Directory Server Console 中的帮助按钮来启动在线帮助。

Administration Server 被授予了代理。如果在运行 Directory Server Console 的客户机上使用 HTTP 连接的代理，则需要执行以下操作之一：

- 从运行 Directory Server Console 的计算机上删除代理。这样即允许客户机直接访问 Administration Server。

  要从运行 Directory Server Console 的计算机上删除代理，需要改动用来运行帮助的浏览器的代理配置。在 Netscape Communicator 中，从 “编辑”菜单中选择 “首选项”。依次选择 “高级”和 “代理”以访问代理配置。在 Internet Explorer 中，从 “工具”菜单中选择 “Internet 选项”。

- 将客户机代理 IP 地址添加到 Administration Server 的可接受的 IP 地址列表中。

警告 将客户机代理 IP 地址添加到 Administration Server 可导致系统出现潜在的安全漏洞。

控制台剪贴板

Directory Server Console 使用系统剪贴板来复制、剪切和粘贴文本。另外，它还包含一个有用的减少输入的功能：在 “目录”选项卡内浏览时，可以将条目的 DN 或 URL 生成到剪贴板上：

1. 在 Directory Server Console 上，选择 “目录”选项卡。
2. 浏览目录树并选择 （左键单击）要复制 DN 或 URL 的条目。
3. 从菜单中选择 “编辑” > “复制 DN”或 “编辑” > “复制 URL”。

在打开一个对话框或其它选项卡前执行上述操作，这样可以将 DN 或 URL 文本添加到任何文本字段。

启动和停止 iPlanet Directory Server

如果未使用安全套接层 (SSL)，则可通过下列方法启动和停止 iPlanet Directory Server。如果使用了 SSL，请参阅第 39 页上的 “在启用 SSL 的情况下启动服务器”。

从控制台启动/停止服务器

1. 启动 iPlanet Directory Server Console。

   有关说明，请参阅第 26 页上的 “启动 iPlanet Directory Server Console”。

2. 在 “任务”选项卡中，相应地单击 “启动 iPlanet Directory Server”或 “停止 iPlanet Directory Server”。

从 iPlanet Directory Server Console 成功地启动或停止 iPlanet Directory Server 后，服务器就会显示一个消息框，提示服务器处于启动或关闭状态。

从命令行启动/停止服务器

利用根用户特权，运行以下之一：

```
# /usr/sbin/directoryserver start
```

或

```
# /usr/sbin/directoryserver stop
```

脚本必须利用与 iPlanet Directory Server 相同的 UID 与 GID 来运行。例如，如果 iPlanet Directory Server 作为 `nobody` 运行，则实用程序也必须作为 `nobody` 运行。

配置 LDAP 参数

可以通过 iPlanet Directory Server Console 查看和更改与服务器的网络和 LDAP 设置相关的参数。本部分提供有关下列内容的信息：

- 更改目录服务器端口号
- 将整个 Directory Server 置于只读模式
- 跟踪目录条目的修改

有关模式检查的信息，请参阅第 9 章 “扩展目录模式”。

更改目录服务器端口号

使用 iPlanet Directory Server Console 或通过更改 `cn=config` 条目下的 `nsslapd-port` 属性值，可以修改用户目录服务器的端口或安全端口号。

如果要修改包含 iPlanet 配置信息 （`o=NetscapeRoot` 子树）的 iPlanet Directory Server 的端口或安全端口号，则可通过 iPlanet Directory Server Console 完成。

如果想更改配置目录或用户目录端口或安全端口号，则应注意以下几点：

- 您需要更改 Administration Server 的配置或用户目录端口或者安全端口号。有关信息，请参阅*通过 iPlanet Console 管理服务器*。
- 如果安装有其它指向配置或用户目录的 iPlanet 服务器，则需要更新这些服务器以使其指向新的端口号。

要修改用户或配置目录用于监听输入请求的端口或安全端口：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树中的顶级条目。
2. 选择右侧窗口中的 “设置”选项卡。
3. 在 “端口”文本框中，输入希望服务器用于非 SSL 通讯的端口号。

   缺省值是 389。
4. 在 “加密端口”文本框中，输入希望服务器用于 SSL 通讯的端口号。

   指定的加密端口号必须与普通 LDAP 通讯使用的端口号相同。缺省值是 636。
5. 单击 “保存”，然后重新启动服务器。

   有关信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

将整个 Directory Server 置于只读模式

如果要用目录服务器维护多个数据库，且需要将所有数据库都设为只读模式，则可通过一步操作完成。但要注意：如果 iPlanet Directory Server 包含副本，则不得使用只读模式，因为这将禁用复制功能。

要将 iPlanet Directory Server 置于只读模式：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树中的顶级条目。
2. 选择右侧窗口中的 “设置”选项卡。
3. 选中 “使整个服务器只读”复选框。
4. 单击 “保存”，然后重新启动服务器。

注意 此操作也会使目录服务器配置处于只读状态。因此，当处于只读模式时，不能升级服务器配置，不能启用或禁用插件，甚至不能重新启动目录服务器。

有关将单个数据库置于只读模式的信息，请参阅第 129 页上的 “启用只读模式”。

跟踪目录条目的修改

可以对服务器进行配置，从而为新创建或新修改的条目维护特殊的属性。

- `creatorsName` — 条目最初创建者的特异名称。
- `createTimestamp` — 条目创建的时间戳，以 GMT （格林尼治时间）格式表示。
- `modifiersName` — 条目最终修改者的特异名称。
- `modifyTimestamp` — 最终修改条目的时间戳，以 GMT （格林尼治时间）格式表示。

注意 当客户机应用程序利用数据库链接创建或修改条目时， `reatorsName` 和 `modifiersName` 属性无法反映条目的真正创建者或修改者。这些属性包含对远程服务器拥有代理验证权限的管理员名称。有关代理验证的信息，请参阅第 95 页上的 “提供绑定凭证”。

要启用 iPlanet Directory Server 以跟踪这些信息：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树中的顶级条目。
2. 选择右侧窗口中的 “设置”选项卡。
3. 选中 “跟踪项目修改时间”复选框。

   服务器将为每个新创建或修改的条目添加 `creatorsName`、 `createTimestamp`、`modifiersName` 和 `modifyTimestamp` 属性。
4. 单击 “保存”，然后重新启动服务器。

   有关详细信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

在启用 SSL 的情况下启动服务器

您必须从命令行下启动服务器，还可以创建密码文件来存放您的证书密码。通过把证书数据库口令放在文件中，可以从服务器控制台启动服务器，也可以在无人看护运行时允许服务器自动重新启动。

警告 此口令以纯文本形式储存于口令文件中，因此对它的使用会带来重大的安全隐患。如果服务器运行于非安全环境中，则切勿使用口令文件。

口令文件必须放在以下位置：

`/usr/iplanet/ds5/alias/slapd-`*`serverID`*`-pin.txt`

其中 *serverID* 是配置时为服务器指定的标识符。

如下所示，需要在以下文件中包括令牌名称和口令：

```
Token:Password
```

例如：

```
Internal (Software) Token:mypassword
```

要创建证书数据库，则必须使用管理服务器和证书安装向导。有关证书数据库、证书别名、 SSL 和获得服务器证书的信息，请参阅*通过 iPlanet Console 管理服务器*。有关在 iPlanet Directory Server 中使用 SSL 的信息，请参阅第 11 章 “管理 SSL”。

第 2 章

创建目录项

本章介绍如何使用 iPlanet Directory Server Console 和 `ldapmodify` 及 `ldapdelete` 命令行实用程序来修改目录内容。

在目录部署的规划阶段，就应明确目录所要包含的数据类型特征。创建条目及修改默认模式前，应仔细阅读 *iPlanet Directory Server 部署指南*。

本章包括以下几节：

- 从目录控制台管理条目
- 从命令行管理条目
- LDIF 更新语句
- 保持参照完整性

从目录控制台管理条目

您可以使用 iPlanet Directory Server Console 上的 “目录”选项卡和属性编辑器来分别添加、修改或删除各个条目。

有关启动 iPlanet Directory Server Console 和浏览用户界面的信息，请参阅第 26 页上的 “使用 iPlanet Directory Server Console”。

如果希望同时添加多个条目，可以使用第 51 页上的 “从命令行管理条目”中所述的命令行实用程序。

本节提供下列信息：

- 创建根条目
- 创建目录项

- 修改目录项
- 删除目录项

本节假定您已具备有关对象类和属性的基本知识。有关对象类和属性的说明，请参阅 *iPlanet Directory Server 部署指南*。有关定义及使用 iPlanet 服务器产品随带的所有模式的用法信息，请参阅 *iPlanet Directory Server 模式参考指南*。

注意 除非已设置相应的访问控制规则，否则将无法修改目录。有关创建目录之访问控制规则的信息，请参阅第 6 章 “管理访问控制”。

创建根条目

每次创建新数据库时，其关联的后缀都将存储到数据库中。代表该后缀的目录项并不是自动创建的。

要创建数据库的根条目：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 按照第 80 页上的 “创建和维护数据库”中所述，创建新的数据库。
3. 在 “目录”选项卡上，右键单击代表目录服务器的顶端对象，然后选择 “新根对象”。

   “新根对象”下的二级菜单中将显示没有相应条目的后缀列表。
4. 选择对应所要创建的条目的后缀。

   此时显示 “新对象”窗口。
5. 在 “新对象”窗口中，选择对应于新条目的对象类。

   所选的对象类中必须包含用于命名后缀的属性。例如，如果是创建对应于后缀 `ou=people,dc=siroe,dc=com` 的条目，则可以选择 `organizationalUnit` 对象类 （或其它允许使用 `ou` 属性的对象类）。
6. 单击 “新对象”窗口中的 “确定”。

   此时显示新条目的属性编辑器。单击 “确定”可接受当前值，也可以按第 45 页上的 “修改目录项”中的说明对该条目进行修改。

创建目录项

iPlanet Directory Server Console 提供了几种预定义的模板，可用于创建目录项。下列类型的条目具有可用的模板：

- 用户
- 组
- 组织单元
- 角色
- 服务类

表 2-1 显示每个模板所用对象类的类型。

表 2-1 条目模板及对应的对象类

模板	对象类
用户	inetOrgPerson
组	groupOfUniqueNames
组织单元	organizationalUnit
角色	nsRoleDefinition
服务类	cosSuperDefinition

这些模板中包含的字段分别代表着所有必选属性及部分常用的可选属性。要使用其中的某个模板来创建条目，请参阅第 44 页上的 “使用预定义模板创建条目”。要创建其它类型的条目，请参阅第 44 页上的 “创建其它类型的条目”。

使用预定义模板创建条目

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 在左侧窗口中，右键单击要在下面添加新条目的条目，然后选择相应的条目类型：用户、组、组织单元、角色、服务类或其它。

   此时显示相应的创建窗口。
3. 为所有必选属性 （以星号标识）提供相应的值。必要时，还可为可选属性提供值。

   “创建”窗口并不为所有可选属性都提供字段。
4. 要显示属性的完整列表，请单击 “高级”按钮。

   此时显示 “属性编辑器”。有关使用属性编辑器的信息，请参阅第 45 页上的 “修改目录项”。
5. 单击 “确定”以关闭 “创建”窗口。

   新的条目随即显示在右侧窗口中。

创建其它类型的条目

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 在左侧窗口中，右键单击要在下面添加新条目的条目，然后选择 “其它”。

   此时显示 “新对象”窗口。
3. 在对象类列表中，选择定义新条目的对象类。
4. 单击 “确定”。

   如果选择的对象类所关联的条目类型具有可用的预定义模板，则显示相应的创建窗口。（请参阅第 44 页上的 “使用预定义模板创建条目”）。

   其它情况下将显示 “属性编辑器”窗口。窗口中包含必选属性的列表。
5. 为所列的属性提供相应的值。

   注意：有些字段为空，但有些却可能有通用的占位符 （例如 **`New`**）。对于后者，应替换为该条目有意义的值。

   有些对象类可以有几个命名属性。记住：应选择要用于命名新条目的命名属性。

   要为未列出的可选属性提供值，请参阅第 45 页上的 “修改目录项”。
6. 单击 “确定”以保存新条目并关闭 “属性编辑器”窗口。

   新的条目随即显示在右侧窗口中。

修改目录项

要 iPlanet Directory Server Console 从修改目录项，则必须启动属性编辑器。属性编辑器中包含属于该条目的对象类和属性列表。

在属性编辑器中，您可以：

- 向条目中添加对象类
- 删除条目中的对象类
- 向条目中添加属性
- 向条目中添加属性值
- 删除条目中的属性值
- 向条目中添加属性子类型

本节将介绍如何启动属性编辑器，以及如何使用属性编辑器来修改条目的属性和属性值。

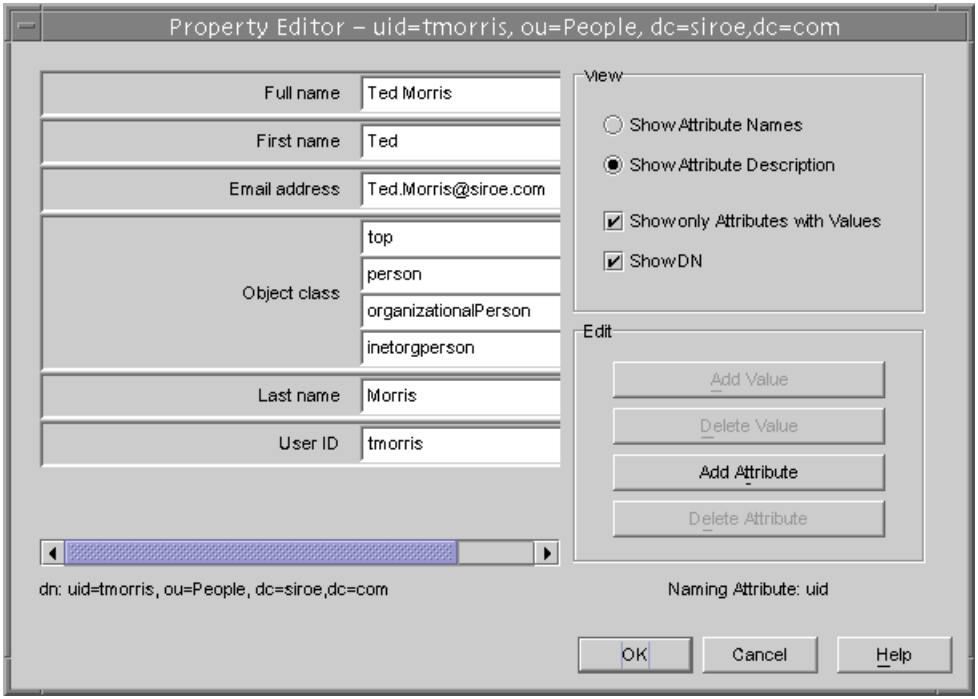
显示属性编辑器

启动属性编辑器的方法有多种：

- 在 “目录”选项卡中，右键单击左侧或右侧窗口中的条目，然后从弹出菜单中选择 “属性”。
- 在 “目录”选项卡中，双击左侧或右侧窗口中的条目。
- 在 “创建用户”、“创建组”、“创建组织单元”、“创建角色”和 “创建服务类”模板中，单击 “高级”按钮 （请参阅第 44 页上的 “使用预定义模板创建条目”）。
- 在 “新对象”窗口中，单击 “确定”（请参阅第 44 页上的 “创建其它类型的条目”）。

有关 “属性编辑器”窗口的图示见图 2-1，其中包括一个说明空间关系的条目示例。

图 2-1 iPlanet Directory Server Console — 属性编辑器



向条目中添加对象类

要向条目中添加对象类：

1. 在 iPlanet Directory Server Console 的 “目录”选项卡中，右键单击所要修改的条目，然后从弹出菜单中选择 “属性”。

   也可以双击该条目。此时显示 “属性编辑器”。

2. 选择对象类字段，然后单击 “添加值”。

   此时显示 “添加对象类”窗口。窗口中将显示可以向条目中添加的对象类列表。

3. 选择要添加的对象类，然后单击 “确定”。

   所选的对象类将出现在属性编辑器的对象类列表中。要关闭 “添加对象类”窗口，请单击 “取消”。

4. 编辑完条目后，单击属性编辑器中的 “确定”。

   此时将关闭 “属性编辑器”。

删除对象类

要删除条目中的对象类：

1. 在 iPlanet Directory Server Console 的 “目录”选项卡中，右键单击所要修改的条目，然后从弹出菜单中选择 “属性”。

   也可以双击该条目。此时显示 “属性编辑器”。

2. 对于显示所要删除的对象类的文本框，单击其中的光标，然后单击 “删除值”。

3. 编辑完条目后，单击属性编辑器中的 “确定”。

   此时将关闭 “属性编辑器”。

向条目中添加属性

向条目中添加属性之前，条目中必须包含要求或允许使用该属性的对象类。有关详细信息，请参阅第 46 页上的 “向条目中添加对象类”和第 9 章 “扩展目录模式”。

要向条目中添加属性：

1. 在 iPlanet Directory Server Console 的 “目录”选项卡中，右键单击所要修改的条目，然后从弹出菜单中选择 “属性”。

   也可以双击该条目。此时显示 “属性编辑器”。

2. 单击 “添加属性”。

   此时显示 “添加属性”对话框。

3. 从列表中选择所需的属性，然后单击 “确定”。

   此时将关闭 “添加属性”对话框，而所选的属性将出现在属性编辑器的属性列表中。

4. 在属性名右侧的文本框中键入新属性的值。

5. 编辑完条目后，单击属性编辑器中的 “确定”。

   此时将关闭 “属性编辑器”。

添加属性值

如果条目中包含多值属性，即可为这些属性提供多个值。

要为多值属性提供属性值：

1. 在 iPlanet Directory Server Console 的 “目录”选项卡中，右键单击所要修改的条目，然后从弹出菜单中选择 “属性”。

   也可以双击该条目。此时显示 “属性编辑器”。

2. 选择要添加值的属性，然后单击 “添加值”。

   新的空白文本字段将显示在右侧栏中。

3. 键入新属性值的名称。

4. 编辑完条目后，单击属性编辑器中的 “确定”。

   此时将关闭 “属性编辑器”。

删除属性值

要删除条目中的属性值：

1. 在 iPlanet Directory Server Console 的 “目录”选项卡中，右键单击所要修改的条目，然后从弹出菜单中选择 “属性”。

   也可以双击该条目。此时显示 “属性编辑器”。

2. 对于包含要删除属性值的文本框，单击其中的光标，然后单击 “删除值”。

   如果想从条目中删除整个属性及其全部值，请选择 “编辑”菜单中的 “删除属性”。

3. 编辑完条目后，单击属性编辑器中的 “确定”。

   此时将关闭 “属性编辑器”。

添加属性子类型

对于条目中包含的属性而言，可以添加三种不同的子类型：语言、二进制及发音。

语言子类型

有时，用非默认语言的字符来表示用户名会更为准确。例如， Noriko 的名字为日语，她喜欢自己的名字尽可能用日语字符表示。您可以选择日语作为 `givenname` 属性的语言子类型，这样其它用户即可搜索其日语名字。

如果为某个属性指定语言子类型，则该子类型将按下列方式添加到属性名中：

attribute`;` `lang-`*subtype*

其中 *attribute* 是要添加到条目中的属性，而 *subtype* 则是两个字符的语种缩写。有关受支持的语言子类型列表，请参阅第 483 页的表 D-2。例如：

```
givenname;lang-ja
```

对于条目中的每个属性实例，只能为它分配一个语言子类型。要分配多个语言子类型，请向条目中添加另一个属性实例，然后分配新的语言子类型。例如，下列子类型非法：

```
cn;lang-ja;lang-en-GB:Smith
```

请使用：

```
cn: lang-ja: ja_value
cn: lang-en-GB: en-GB_value
```

二进制子类型

为属性分配二进制子类型，指示属性值为二进制值。例如 `usercertificate;binary`。

尽管可以将二进制数据存储到不包含二进制子类型的属性中 （例如 `jpegphoto`），但二进制子类型会向客户机指示：该属性类型可能存在多个变体。

发音子类型

为属性分配发音子类型，指示属性值为音标。该子类型将按下列方式添加到属性中：`attribute;phonetic`。

对于有多个字母表，而其中一个为音标表示的语言而言，该子类型常与语言子类型组合使用。

您可以将其与预计包含用户名的属性 （如 `cn` 或 `givenname`）配合使用。例如，`givenname;lang-ja;phonetic` 指示属性值为条目之日语名的音标。

要使用属性编辑器添加子类型：

1. 在 iPlanet Directory Server Console 的 “目录”选项卡中，右键单击所要修改的条目，然后从弹出菜单中选择 “属性”。

   也可以双击该条目。此时显示 “属性编辑器”。

2. 单击 “添加属性”。

   此时显示 “添加属性”对话框。

3. 从列表中选择所需的属性。

4. 要为属性分配语言子类型，请从 “语言”下拉列表中选择该子类型。

5. 从 “子类型”下拉列表中，也可以分配其它两种子类型之一：二进制或发音。

6. 单击 “确定”。

   此时显示 “添加属性”窗口。

7. 定义完条目的信息后，单击属性编辑器中的 “确定”。

删除目录项

要使用 iPlanet Directory Server Console 删除目录项：

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。

2. 右键单击导航窗口或右侧窗口中所要删除的条目，然后从弹出菜单中选择 “删除”。

   要选择多个条目，请在按住 Ctrl 或 Shift 的同时单击所需条目，然后选择 “编辑”菜单中的 “删除”。

   服务器将立即删除所选的条目。该操作无法撤消。

从命令行管理条目

命令行实用程序可用于操作目录的内容。如果想编写脚本以实现目录的批管理，或者想测试 iPlanet Directory Server，即可使用命令行实用程序。例如，在更改访问控制信息后，您可能想确保它返回所期望的信息。

利用命令行实用程序，即可直接从命令行或通过 LDIF 中的输入文件获取相关的信息。

本节提供下列信息：

- 从命令行提供输入
- 从命令行创建根条目
- 使用 LDIF 添加条目
- 使用 ldapmodify 添加和修改条目
- 使用 ldapdelete 删除条目
- 使用特殊字符

注意 除非已设置相应的访问控制规则，否则将无法修改目录。有关创建目录之访问控制规则的信息，请参阅第 6 章 “管理访问控制”。

从命令行提供输入

从命令行直接为 `ldapmodify` 和 `ldapdelete` 实用程序提供输入信息时，必须使用 LDIF 语句。有关 LDIF 语句的详细信息，请参阅第 56 页上的 “LDIF 更新语句”。

`ldapmodify` 和 `ldapdelete` 实用程序将读取所输入的语句，其方式与从文件中读取完全相同。提供完输入信息后，输入可被 shell 识别为 “文件结束” (EOF) 转义序列的字符。实用程序随即根据所提供的输入内容开始操作。

一般情况下， EOF 转义序列为 control-D (^D)。例如，假设想向 `ldapmodify` 中输入一些 LDIF 更新语句。随后，您应执行以下操作：

```
prompt> ldapmodify -D bindDN -w password -h hostname
> dn: cn=Barry Nixon, ou=people, dc=siroe,dc=com
> changetype: modify
> delete: telephonenumber
> -
```

```
> add: manager
> manager: cn=Harry Cruise, ou=people, dc=siroe,dc=com
> ^D
prompt>
```

从命令行或从 LDIF 中添加条目时，确保已创建代表子树的条目，之后才能在此分支下创建新条目。例如，如果想将某个条目放到 People 子树中，则应创建代表该子树的条目，然后再在该子树中创建条目。

例如：

```
dn: dc=siroe,dc=com
dn: ou=People, dc=siroe,dc=com
...
```
People subtree entries.
```
...
dn: ou=Group, dc=siroe,dc=com
...
```
Group subtree entries.
```
...
```

从命令行创建根条目

您可以使用 `ldapmodify` 命令行实用程序在数据库中创建新的根条目。例如，可以按如下所示添加新的根条目：

```
prompt% ldapmodify -a -D "dn=directory manager" -w secret
```

`ldapmodify` 实用程序将绑定到服务器上并准备添加条目。

如下所示，创建新的根对象：

`dn:` *Suffix_Name*
`objectclass:` *newobjectclass*

DN 对应数据库所含的根后缀或子后缀的 DN。 *newobjectclass* 值与要向数据库中添加的对象类的类型有关。根据添加的根对象，有时需要指定其它必选属性。

注意 只有在每个后缀都有一个数据库的情况下，才能使用该方法。如果创建的后缀存储在多个数据库中，则必须将 `ldif2db` 实用程序与 `-n` 选项配合使用，从而指定用于容纳此新条目的数据库。有关信息，请参阅第 134 页上的 “从命令行导入”。

使用 LDIF 添加条目

使用 LDIF 文件可以添加多个条目或导入整个数据库。要使用 LDIF 文件及 iPlanet Directory Server Console 添加条目：

1. 在 LDIF 文件中定义条目。

   LDIF 的说明见附录 A “LDAP 数据交换格式”。

2. 从 iPlanet Directory Server Console 中导入 LDIF 文件。

   有关信息，请参阅第 131 页上的 “从控制台执行导入”。导入 LDIF 文件时，选择 “导入”对话框中的 “附加数据到数据库”，从而使服务器仅导入目录中当前并不存在的条目。

将 `ldapmodify` 命令与 `-f` 选项配合使用，也可从命令行中添加 LDIF 文件中所描述的条目。

使用 ldapmodify 添加和修改条目

使用 `ldapmodify` 命令可以添加条目及修改现有 iPlanet Directory Server 数据库中的条目。`ldapmodify` 命令将利用所提供的特异名称和口令打开到特定服务器的连接，然后根据指定文件中的 LDIF 更新语句对条目进行修改。由于 `ldapmodify` 使用的是 LDIF 更新语句，因此 `ldapmodify` 可以执行 `ldapdelete` 所能执行的任何操作。

使用该实用程序时，如果已打开模式检查功能，服务器就会在条目发生更改时对整个条目执行模式检查：

- 如果服务器检测到条目中有服务器未知的属性或对象类，则达到有错误的条目时，修改操作即告失败。在遇到错误前所处理的所有条目都已予以成功添加或修改。如果运行 `ldapmodify` 时使用 `-c` 选项 （出错时不停止），则在错误条目后所处理的所有正确条目都将予以成功添加或修改。
- 如果所需的属性不存在，修改操作即告失败。即使未修改有冲突的对象类或属性，也会出现这种情况。如果运行 iPlanet Directory Server 时关闭了模式检查功能，然后添加未知的对象类或属性，且之后打开了模式检查功能，就会出现上述情况。

有关详细信息，请参阅第 319 页上的 “打开和关闭模式检查”。

要利用 `ldapmodify` 创建数据库后缀 （例如 `dc=siroe,dc=com`），则必须作为目录管理员绑定到目录上。

使用 ldapmodify 添加条目

下面是有关如何使用 `ldapmodify` 实用程序来向目录中添加条目的典型示例。假定：

- 希望创建文件 **`new.ldif`** 中所指定的条目。
- 已创建数据库管理员。他有权修改条目，且特异名称为 **`cn=Directory Manager, dc=siroe,dc=com`**。
- 数据库管理员的口令为 **`King-Pin`**。
- 服务器位于 **`cyclops`** 处。
- 服务器使用端口号 **`845`**。

本例中， **`new.ldif`** 文件中的 LDIF 语句并不指定更改类型。它们遵从第 446 页上的“LDIF 文件格式”中所定义的格式。

要添加条目，则必须输入下列命令：

```
ldapmodify -a -D "cn=Directory Manager,dc=siroe,dc=com" -w King-Pin
-h cyclops -p 845 -f new.ldif
```

使用 ldapmodify 修改条目

下面是有关如何使用 `ldapmodify` 实用程序来修改目录条目的典型示例。假定：

- 想修改文件 **`modify_statements`** 中指定的条目。
- 已创建数据库管理员。他有权修改条目，且特异名称为 **`cn=Directory Manager,dc=siroe,dc=com`**。
- 数据库管理员的口令为 **`King-Pin`**。
- 服务器位于 **`cyclops`** 处。
- 服务器使用端口号 **`845`**。

要修改条目，则必须首先利用相应的 LDIF 更新语句来创建 `modify_statements` 文件，然后输入下列命令：

```
ldapmodify -D "cn=Directory Manager,dc=siroe,dc=com" -w King-Pin -h
cyclops -p 845 -f modify_statements
```

使用 ldapdelete 删除条目

使用 ldapdelete 命令行实用程序可以删除目录中的条目。该实用程序将利用所提供的特异名称和口令打开到指定服务器的连接，然后删除条目。

您只能删除分支末尾的条目。但不能删除作为目录树分支点的条目。

例如，对于下面三个条目而言：

```
ou=People,dc=siroe,dc=com
cn=Paula Simon,ou=People,dc=siroe,dc=com
cn=Jerry O'Connor,ou=People,dc=siroe,dc=com
```

只能删除最后两个条目。标识 People 子树的条目只有在下面无任何条目的情况下才能予以删除。如果想删除 ou=People,dc=siroe,dc=com，则必须首先删除 Paula Simon 和 Jerry O'Connor 条目，以及该子树下的其它所有条目。

下面是一个有关如何使用 ldapdelete 实用程序的典型示例：假定：

- 想删除由特异名称 cn=Robert Jenkins,ou=People,dc=siroe,dc=com 和 cn=Lisa Jangles,ou=People,dc=siroe,dc=com 标识的条目。
- 已创建数据库管理员。他有权修改条目，且特异名称为 **cn=Directory Manager,dc=siroe,dc=com**。
- 数据库管理员的口令为 **King-Pin**。
- 服务器位于 **cyclops** 处。
- 服务器使用端口号 **845**。

要删除用户 Robert Jenkins 和 Lisa Jangles 的条目，请输入下列命令：

```
ldapdelete -D "cn=Directory Manager,dc=siroe,dc=com" -w King-Pin -h
cyclops -p 845 "cn=Robert Jenkins,ou=People,dc=siroe,dc=com"
"cn=Lisa Jangles,ou=People,dc=siroe,dc=com"
```

下表说明了示例中所用的 ldapdelete 参数：

参数名	说明
-D	指定进行服务器验证时所用的特异名称。此值必须为可被 iPlanet Directory Server 识别的 DN，且还必须有权修改条目。
-w	指定与 -D 参数中所指定的特异名称相关联的口令。
-h	指定运行服务器的主机的名称。
-p	指定服务器所用的端口号。

有关 ldapdelete 参数的完整信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

使用特殊字符

使用 iPlanet Directory Server 命令行客户机工具时，可能需要包含对于命令行解释程序而言有特殊意义的字符 （例如空格 [ ]、星号 [*]、反斜杠 [\] 等）。这种情况下，请将此值括到引号 (“”) 中。例如：

```
-D "cn=Barbara Jensen,ou=Product Development,dc=siroe,dc=com"
```

取决于所用的命令行实用程序，此时应使用单引号或双引号。有关详细信息，请参阅操作系统文档。

此外，如果使用的 DN 中包含逗号，则必须用反斜杠 (\) 对逗号进行转义。例如：

```
-D "cn=Patricia Fuentes,ou=people,o=siroe.com Bolivia\,S.A."
```

要从 siroe.com Bolivia, S.A. 目录树中删除用户 Patricia Fuentes，请输入下列命令：

```
ldapdelete -D "cn=Directory Manager,dc=siroe,dc=com" -w King-Pin -h
cyclops -p 845 "cn=Patricia Fuentes,ou=People,o=siroe.com
Bolivia\,S.A."
```

LDIF 更新语句

使用 LDIF 更新语句可以定义 ldapmodify 更改目录的方式。一般情况下， LDIF 更新语句是一系列具有下列功能的语句：

- 指定所要修改之条目的特异名称。
- 指定用于定义特定条目修改方式 (add, delete, modify, modrdn) 的更改类型。
- 指定一系列属性及其更改值。

除非将 ldapmodify 与 -a 参数配合使用，否则需要指定更改类型。如果指定 -a 参数，则认为是添加操作 (changetype: add)。但是，其它任何更改类型都将覆盖 -a 参数。

如果指定修改操作 (changetype: modify)，则应提供更改操作，以指示条目的更改方式。

如果指定 changetype: modrdn，则需要提供更改操作，以指定如何修改 “相对特异名称” (RDN)。特异名称的 RDN 就是 DN 最左侧的值。例如，特异名称 uid=ssarette,dc=siroe,dc=com 的 RDN 为 uid=ssarette。

LDIF 更新语句的一般格式如下所示：

```
dn: distinguished_name
changetype_identifier
change_operation_identifier
list_of_attributes

-

change_operation_identifier
list_of_attributes
-
```

如果需要指定连续的更改操作，则必须使用破折号 (-) 来指示更改操作的结束。例如，下列语句将向条目中添加电话号码和 manager 属性。

```
dn: cn=Lisa Jangles,ou=People,dc=siroe,dc=com
changetype: modify
add: telephonenumber
telephonenumber: (408) 555-2468
-
add: manager
manager: cn=Harry Cruise,ou=People,dc=siroe,dc=com
```

此外，折行续接运算符为单个空格。因此，下列两个语句相同：

```
dn: cn=Lisa Jangles,ou=People,dc=siroe,dc=com
```

```
dn: cn=Lisa Jangles,
 ou=People,
 dc=siroe,dc=com
```

下列各节将详细介绍更改类型。

使用 LDIF 添加条目

使用 `changetype: add` 可以向目录中添加条目。添加条目时，请务必创建代表分支点的条目，之后再尝试在该分支下创建新条目。也就是说，如果想将条目放到 People 或 Groups 子树中，则在这些子树中创建条目之前，应首先创建这些子树的分支点。

下列 LDIF 更新语句可用于创建 People 和 Groups 子树，然后在这些目录树中创建条目：

```
dn: dc=siroe,dc=com
changetype: add
objectclass: top
objectclass: organization
o: siroe.com
```

```
dn: ou=People, dc=siroe,dc=com
changetype: add
objectclass: top
objectclass: organizationalUnit
ou: People
ou: Marketing

dn: cn=Pete Minsky,ou=People,dc=siroe,dc=com
changetype: add
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: Pete Minsky
givenName: Pete
sn: Minsky
ou: People
ou: Marketing
uid: pminsky

dn: cn=Sue Jacobs,ou=People,dc=siroe,dc=com
changetype: add
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: Sue Jacobs
givenName: Sue
sn: Jacobs
ou: People
ou: Marketing
uid: sjacobs

dn: ou=Groups,dc=siroe,dc=com
changetype: add
objectclass: top
objectclass: organizationalUnit
ou: Groups

dn: cn=Administrators,ou=Groups,dc=siroe,dc=com
changetype: add
objectclass: top
objectclass: groupOfNames
member: cn=Sue Jacobs,ou=People,dc=siroe,dc=com
member: cn=Pete Minsky,ou=People,dc=siroe,dc=com
cn: Administrators

dn: ou=siroe.com Bolivia\, S.A.,dc=siroe,dc=com
changetype: add
```

```
objectclass: top
objectclass: organizationalUnit
ou: siroe.com Bolivia\, S.A.

dn: cn=Carla Flores,ou=siroe.com Bolivia\, S.A.,dc=siroe,dc=com
changetype: add
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: Carla Flores
givenName: Carla
sn: Flores
ou: siroe.com Bolivia\, S.A.
uid: cflores
```

使用 LDIF 重命名条目

使用 `changetype:modrdn` 可以更改条目的相对特异名称 (RDN)。条目的 RDN 就是特异名称中最左侧的部分。因此，

```
cn=Barry Nixon,ou=People,dc=siroe,dc=com
```

的 RDN 为：

```
cn=Barry Nixon
```

而

```
ou=People,dc=siroe,dc=com
```

的 RDN 为：

```
ou=People
```

因此，此重命名操作将用于更改条目特异名称中最左侧的值。

例如，条目

```
cn=Sue Jacobs,ou=People,dc=siroe,dc=com
```

可以改为：

```
cn=Susan Jacobs,ou=People,dc=siroe,dc=com
```

但不能改为：

```
cn=Sue Jacobs,ou=old employees,dc=siroe,dc=com
```

下面的示例可用于将 Sue Jacobs 重命名为 Susan Jacobs：

```
dn: cn=Sue Jacobs,ou=Marketing,dc=siroe,dc=com
changetype: modrdn
newrdn: cn=Susan Jacobs
deleteoldrdn: 0
```

由于 deleteoldrdn 为 0，因此本例中将把现有的 RDN 继续保留为新条目的值。除了原条目中所包含的其它所有属性外，最终的条目还将有一个通用名 (cn) 属性：既设置为 Sue Jacobs，也设置为 Susan Jacobs。但如果使用的是

```
dn: cn=Sue Jacobs,ou=Marketing,dc=siroe,dc=com
changetype: modrdn
newrdn: cn=Susan Jacobs
deleteoldrdn: 1
```

服务器就会删除 cn=Sue Jacobs，而此时只有 cn=Susan Jacobs 保留在条目中。

有关重命名条目的说明

不能用 modrdn 更改类型来重命名条目，因为这样会将条目移动到完全不同的子树中。要将条目移动到完全不同的分支中，则必须使用旧条目的属性在备用子树中创建新的条目，然后删除旧的条目。

同样的原因，如果某个条目是分支点，则不能删除该条目；如果条目中有子项，则不能重命名该条目。否则，就会孤立目录树中的子项，而这是 LDAP 协议所不允许的。例如，对于下面三个条目而言：

```
ou=People,dc=siroe,dc=com
cn=Paula Simon,ou=People,dc=siroe,dc=com
cn=Jerry O'Connor,ou=People,dc=siroe,dc=com
```

您只能重命名最后两个条目。只有在下面没有任何其它条目的情况下，才能对标识 People 子树的条目进行重命名。

使用 LDIF 修改条目

使用 `changetype:modify` 可以针对条目添加、替换或删除属性和/或属性值。指定 `changetype:modify` 时，也必须同时提供更改操作，以指示条目的修改方式。更改操作可以是：

- `add:` *attribute*

  添加指定的属性或属性值。如果属性类型相对于该条目而言当前不存在，就会创建该属性及其相应的值。如果属性类型相对于该条目而言已经存在，就会将指定的属性值添加到现有的值中。如果该条目中已存在这个特定的属性值，操作即告失败，同时服务器将返回错误。

- `replace:` *attribute*

  指定的值将用于整个替换属性的值。如果属性尚未存在，则予以创建。如果未指定属性的替换值，则删除该属性。

- `delete:` *attribute*

  删除指定的属性。如果条目中存在属性的多个值，则删除条目中该属性的所有值。要想只删除多个属性值中的一个，请在 delete 更改操作后面的行上指定该属性及其相关值。

该部分包含下列主题：

- 使用 LDIF 将属性添加到现有的条目中
- 使用 LDIF 更改属性值
- 使用 LDIF 删除所有属性值
- 使用 LDIF 删除特定属性值

使用 LDIF 将属性添加到现有的条目中

将 changetype:modify 与更改操作配合使用，可以向条目中添加属性和属性值。

例如，下列 LDIF 更新语句将向条目中添加电话号码：

```
dn: cn=Barney Fife,ou=People,dc=siroe,dc=com
changetype: modify
add: telephonenumber
telephonenumber: 555-1212
```

下例将向条目中添加两个电话号码：

```
dn: cn=Barney Fife,ou=People,dc=siroe,dc=com
changetype: modify
add: telephonenumber
telephonenumber: 555-1212
telephonenumber: 555-6789
```

下例将向条目中添加两个 telephonenumber 属性和一个 manager 属性：

```
dn: cn=Barney Fife,ou=People,dc=siroe,dc=com
changetype: modify
add: telephonenumber
telephonenumber: 555-1212
telephonenumber: 555-6789
-
add: manager
manager: cn=Sally Nixon,ou=People,dc=siroe,dc=com
```

使用 LDIF 更改属性值

将 changetype:modify 与替换操作配合使用，即可更改条目中属性的所有值。

例如，下列 LDIF 更新语句将把 Barney 的经理从 Sally Nixon 更改为 Wally Hensford：

```
dn: cn=Barney Fife,ou=People,dc=siroe,dc=com
changetype: modify
replace: manager
manager: cn=Wally Hensford, ou=People, dc=siroe,dc=com
```

如果条目中有属性的多个实例，要更改其中的某个属性值，则必须删除所要更改的属性值，然后添加替换值。例如，不妨考虑下列条目：

```
cn=Barney Fife,ou=People,dc=siroe,dc=com
objectClass: inetOrgPerson
cn: Barney Fife
sn: Fife
telephonenumber: 555-1212
telephonenumber: 555-5678
```

要将电话号码 555-1212 更改为 555-4321，请使用下列 LDIF 更新语句：

```
dn: cn=Barney Fife,ou=People,dc=siroe,dc=com
changetype: modify
delete: telephonenumber
telephonenumber: 555-1212
-
add: telephonenumber
telephonenumber: 555-4321
```

Barney 的条目现在为：

```
cn=Barney Fife,ou=People,dc=siroe,dc=com
objectClass: inetOrgPerson
cn: Barney Fife
sn: Fife
telephonenumber: 555-5678
telephonenumber: 555-4321
```

使用 LDIF 删除所有属性值

将 `changetype:modify` 与删除操作配合使用，即可删除条目中的属性。如果条目中有属性的多个实例，则必须指示所要删除的属性。

例如，下列 LDIF 更新语句将从条目中删除 `telephonenumber` 属性的所有实例，而不管它在条目中出现的次数：

```
dn: cn=Barney Fife,ou=People,dc=siroe,dc=com
changetype: modify
delete: telephonenumber
```

如果想删除特定的 `telephonenumber` 属性实例，则只需删除该特定属性的属性值。下节将介绍如何完成上述任务。

使用 LDIF 删除特定属性值

将 `changetype:modify` 与 `delete` 操作配合使用可以删除条目的属性值。

例如，不妨考虑下列条目：

```
cn=Barney Fife,ou=People,dc=siroe,dc=com
objectClass: inetOrgPerson
cn: Barney Fife
sn: Fife
telephonenumber: 555-1212
telephonenumber: 555-5678
```

要删除该条目中的电话号码 555-1212，请使用下列 LDIF 更新语句：

```
dn: cn=Barney Fife,ou=People,dc=siroe,dc=com
changetype: modify
delete: telephonenumber
telephonenumber: 555-1212
```

Barney 的条目将变成：

```
cn=Barney Fife,ou=People,dc=siroe,dc=com
objectClass: inetOrgPerson
cn: Barney Fife
sn: Fife
telephonenumber: 555-5678
```

使用 LDIF 修改条目

使用 `changetype:delete` 可以删除目录中的条目。您只能删除叶条目。因此，在删除条目时，应确保目录树中该条目的下面没有其它条目。也就是说，除非首先删除属于某个组织单元的所有条目，否则将无法删除该组织单元条目。

例如，对于下面三个条目而言：

```
ou=People,dc=siroe,dc=com
cn=Paula Simon,ou=People,dc=siroe,dc=com
cn=Jerry O'Connor,ou=People,dc=siroe,dc=com
```

只能删除最后两个条目。标识 People 子树的条目只有在下面没有其它条目的情况下才能予以删除。

下列 LDIF 更新语句可用于删除人员条目：

```
dn: cn=Pete Minsky,ou=People,dc=siroe,dc=com
changetype: delete

dn: cn=Sue Jacobs,ou=People,dc=siroe,dc=com
changetype: delete
```

警告 请勿删除后缀 `o=NetscapeRoot`。 iPlanet 管理服务器使用该后缀来存储有关已安装 iPlanet 服务器的信息。删除该后缀会强制用户重新安装所有 iPlanet 服务器，包括目录服务器。

修改国际化目录中的条目

可以用有关属性类型的语言选项卡来指定使用非英语语言的属性值。使用 `ldapmodify` 命令行实用程序来修改具有关联语言标记的属性时，必须确保属性值与语言标记完全匹配，否则修改操作即告失败。

例如，如果想修改语言标记为 `lang-fr` 的属性值，则必须在修改操作中包含 `lang-fr`，如下所示：

```
dn: bjensen,dc=siroe,dc=com
changetype: modify
replace: homePostalAddress;lang-fr
homePostalAddress;lang-fr: 34\, rue de Seine
```

保持参照完整性

*参照完整性*是一种确保维持相关条目之间关系的数据库机制。在 Directory Server 中，参照完整性可用于确保对一个条目的更新将正确反映在可能引用该更新条目的其它所有条目中。

例如，如果从目录中移走某个用户条目且已启用参照完整性，服务器就会同时从该用户所在的所有组中删除该用户。如果未启用参照完整性，则在由管理员进行手动删除前，该用户仍将是组的成员。如果是将目录服务器与其它依赖用户和组管理目录的 iPlanet 产品进行集成，该功能就会较为重要。

参照完整性的原理

启用参照完整性后，它将在删除或重命名操作后立即对指定的属性进行完整性更新。默认情况下，参照完整性插件处于禁用状态。

删除或重命名目录中的用户或组条目时，操作将被记录到参照完整性日志文件中：

```
/var/ds5/slapd-serverID/logs/referint
```

经过指定的一段时间后（称为*更新间隔*），服务器将搜索目录中已启用参照完整性的所有属性，并使搜索到的条目与日志文件中已删除或修改之条目的 DN 相匹配。如果日志文件显示已删除该条目，则说明相应的属性也已被删除。如果日志文件显示条目已进行过更改，则对应的属性值也会相应地进行更改。

默认情况下，如果参照完整性插件处于启用状态，它就会在删除或重命名操作后立即对 member、uniquemember、owner 及 seeAlso 属性执行完整性更新。但可以对参照完整性插件的行为进行配置，使之符合自己的需要。您可以：

- 在复制更改日志中记录参照完整性的更新
- 修改更新间隔
- 选择要应用参照完整性的属性
- 禁用参照完整性

将参照完整性与复制功能配合使用

在复制环境中，参照完整性插件的使用将受到一定的限制：

- 一定不要在专用客户服务器 （一种仅包含只读副本的服务器）上启用该插件。
- 对于包含读写副本及只读副本组合的服务器，一定不要启用上面的参照完整性插件。
- 可以在仅包含读写副本的原版服务器上启用该插件。
- 在多原版复制环境下，应仅启用一个原版上的参照完整性插件。

配置供给服务器

当复制环境满足上述条件时，即可启用参照完整性插件。

1. 启用参照完整性插件。

   有关该任务的说明见第 67 页上的 “启用 / 禁用参照完整性”。

2. 将插件配置为在更改日志中记录所有完整性更新。

   有关该任务的说明见第 67 页上的 “在更改日志中记录更新”。

3. 确保禁用所有客户服务器上的参照完整性插件。

注意 由于供给服务器会将参照完整性插件所做的更改发送给客户服务器，因此没必要在客户服务器上运行参照完整性插件。

启用/禁用参照完整性

从 iPlanet Directory Server Console 或命令行上可以启用或禁用参照完整性。

从 iPlanet Directory Server Console 上

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。

   有关启动 iPlanet Directory Server Console 的信息，请参阅第 26 页上的 “使用 iPlanet Directory Server Console”。

2. 展开导航树中的插件文件夹，然后选择 Referential Integrity Postoperation 插件。

   该插件的设置显示在右侧窗口中。

3. 选中 “启用插件”复选框将启用该插件，而清除该复选框则禁用该插件。

4. 单击 “保存”可保存更改结果。

5. 为使更改生效，请转到 “任务”选项卡，然后选择 “重新启动 Directory Server”。

在更改日志中记录更新

您可以决定在复制更改日志中记录更新内容，而不是将其记录到 `slapd-`*`serverID`*`/logs` 目录下的 `referint` 文件中。在复制环境中，如果想将参照完整性更新内容复制到客户服务器中，则必须进行上述操作。

此更改可以从 iPlanet Directory Server Console 中进行。

从 iPlanet Directory Server Console 上

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。

2. 展开导航树中的插件文件夹，然后选择 Referential Integrity Postoperation 插件。

   该插件的设置将显示在右侧窗口中。

3. 在参数列表中，将文件名 `referint` 替换为到更改日志目录的绝对路径。

4. 单击 “保存”可保存更改结果。

5. 为使更改生效，请转到 “任务”选项卡，然后选择 “重新启动 Directory Server”。

修改更新间隔

默认情况下，服务器将在 `delete` 或 `modrdn` 操作后立即执行参照完整性更新。如果想降低该操作对系统所造成的影响，可以增加更新的时间间隔。尽管没有最大更新间隔的限制，但我们常用的是下列间隔值：

- 立即更新
- 90 秒 （每隔 90 秒更新一次）
- 3600 秒 （每小时更新一次）
- 10,800 秒 （每 3 小时更新一次）
- 28,800 秒 （每 8 小时更新一次）
- 86,400 秒 （每天更新一次）
- 604,800 秒 （每周更新一次）

可以从 iPlanet Directory Server Console 上修改更新间隔。

从 iPlanet Directory Server Console 上

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 展开导航树中的插件文件夹，然后选择 Referential Integrity Postoperation 插件。

   该插件的设置将显示在右侧窗口中。
3. 在参数列表中，将第一个文本框中的值替换为相应的时间间隔。
4. 单击 “保存”可保存更改结果。
5. 为使更改生效，请转到 “任务”选项卡，然后选择 “重新启动 Directory Server”。

修改属性列表

默认情况下，参照完整性将设置为对 `member`、 `uniquemember`、 `owner` 及 `seeAlso` 属性进行更新。可以从 iPlanet Directory Server Console 上添加或删除所要更新的属性。

从 iPlanet Directory Server Console 上

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 展开导航树中的插件文件夹，然后选择 Referential Integrity Postoperation 插件。

   该插件的设置将显示在右侧窗口中。
3. 在 “参数”部分，使用 “添加”和 “删除”按钮来修改列表中的属性。
4. 单击 “保存”可保存更改结果。
5. 为使更改生效，请转到 “任务”选项卡，然后选择 “重新启动 Directory Server”。

注意 为获得最佳性能，还应建立进行更新的属性集的索引。有关索引的信息，请参阅第 8 章 “管理索引”。

第 3 章

配置目录数据库

目录由数据库组成，您可以在这些数据库上分配目录树。本章介绍如何创建*后缀*（目录树的分支点）及如何创建与每个后缀 (suffix) 相关联的数据库。文中同时还将介绍如何创建到远程服务器上参考数据库的数据库链接，以及如何使用引荐将客户机指向目录数据的外部资源。

本章包含以下几部分：

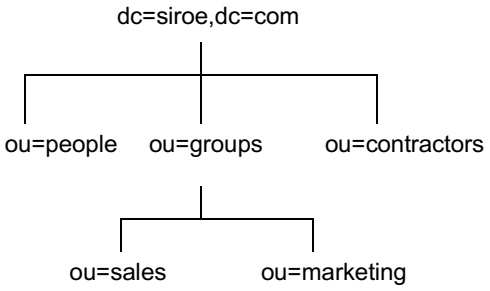
- 创建和维护后缀
- 创建和维护数据库
- 创建和维护数据库链接
- 使用引荐

有关分配目录数据的概念性信息，请参阅 *iPlanet Directory Server 部署指南*。

创建和维护后缀

您可在不同的数据库中存储不同的目录树块，然后跨多个服务器分配这些数据库。目录树中包含称为节点的分支点。这些节点可以与数据库相关联或不相关联。节点是使用 Directory Server Console 中的 “目录”选项卡创建的。从中，您可以对出现在目录树中的条目进行自由编辑。

后缀是与特定数据库相关联的目录树节点。使用 Directory Server Console 上的 “数据库”选项卡可以创建这些特殊的节点。例如，一个简单的目录树可能表现为：



后缀 `ou=people` 及其下面的所有条目和节点均存储于一个数据库中，后缀 `ou=groups` 位于另一个数据库中，而后缀 `ou=contractors` 则在其它数据库中。

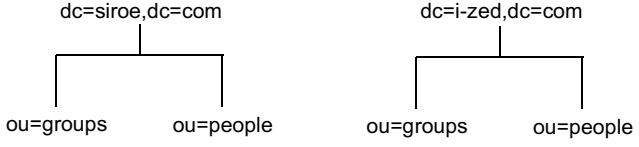
本部分介绍如何在目录服务器上创建后缀并使之与数据库相关联。本部分包含以下过程：

- 第 72 页上的 “创建后缀”
- 第 78 页上的 “维护后缀”

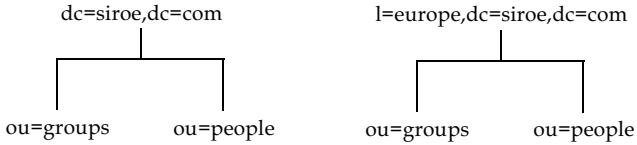
创建后缀

可创建根后缀和子后缀以对目录树的内容加以组织。后缀 (suffix) 是子后缀 (sub suffix) 的父项。它可以是为目录服务器所设计的大目录树的一部分。子后缀是根后缀下面的一个分支。根后缀和子后缀的数据都包含在数据库中。

目录中可能包含多个根后缀。例如，一个 ISP 可能为多个 Web 站点提供主机服务：一个为 siroe.com，而另一个则为 i-zed.com。 ISP 将创建两个根后缀，分别对应于 `dc=siroe,dc=com` 命名环境和 `dc=i-zed,dc=com` 命名环境。如下所示，目录树显示为：

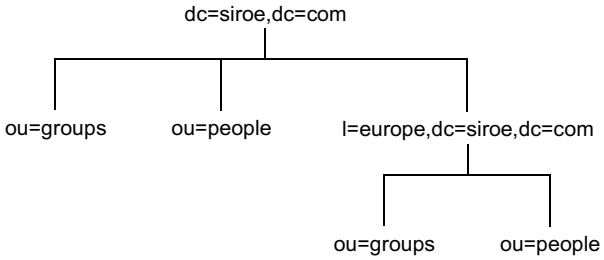


也可创建根后缀以从搜索操作中排除部分目录树。例如， siroe.com Corporation 可能需要从常规 siroe.com Corporation 目录搜索中排除其欧洲办事处。为此，他们需要创建两个根后缀。一个根后缀 `dc=siroe,dc=com` 与常规 siroe.com Corporation 目录树相对应，而另一个根后缀 `l=europe,dc=siroe,dc=com` 则与目录树的欧洲分支相对应。在客户机应用程序看来，目录树形如：



如果客户机应用程序对 siroe.com Corporation 目录的 `dc=siroe,dc=com` 分支执行搜索，则不会返回目录中 `l=europe,dc=siroe,dc=com` 分支下的条目，因为这是一个独立的根后缀。

如果 siroe.com Corporation 决定在常规搜索中包含其目录树欧洲分支中的条目，则需要使欧洲分支成为常规分支的子后缀。为此，他们需要创建 siroe.com Corporation 根后缀 `dc=siroe,dc=com`，然后在其下为欧洲目录条目创建子后缀 `l=europe,dc=siroe,dc=com`。在客户机应用程序看来，目录树形如：



本部分介绍使用 Directory Server Console 或命令行为目录创建根后缀和子后缀的方法。本部分包含下列步骤：

- 第 74 页上的 “使用控制台创建新的根后缀”
- 第 74 页上的 “使用控制台创建新的子后缀”
- 第 75 页上的 “从命令行创建根后缀和子后缀”

使用控制台创建新的根后缀

下列过程介绍创建后缀并使之与数据库相关联的方法：

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 右键单击左侧导航窗口中的 “数据”，然后从弹出菜单中选择 “新根后缀”。

   此时显示 “创建新的根后缀”对话框。
3. 在 “新后缀”字段中输入一个唯一性的后缀名。

   后缀的命名必须遵循域组件 (dc) 命名约定。例如，可以输入名为 `dc=siroe, dc=com` 的新后缀。
4. 如果希望在创建新的根后缀的同时在当前目录中创建一个数据库，则 “自动创建相关联的数据库”复选框应被默认选中。

   如果想在另外一个目录中或稍后为新的根后缀创建数据库，请取消选中该复选框。在创建数据库之前，新的根后缀将处于禁用状态。
5. 如果在步骤 4 中选中 “自动创建相关联的数据库”复选框，则在 “数据库名称”字段中为新数据库输入唯一性的名称。

   该值只可以包含 ASCII （7 位）字母数字字符、连字符 (-) 和下划线 (_)。例如，可以将新数据库命名为 `siroe_2`。
6. 单击 “确定”以创建该新根后缀。

   根后缀将自动出现在左侧导航窗口的 “数据”分支下。

使用控制台创建新的子后缀

下列过程介绍在现有根后缀或子后缀下创建子后缀。

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧导航窗口的 “数据”下，选择想在其下添加新子后缀的后缀。右键单击该后缀，然后从弹出菜单中选择 “新子后缀”。

   此时显示 “创建新子后缀”对话框。
3. 在 “新后缀”字段中输入一个唯一性的后缀名。

   该后缀应遵循其根后缀的命名约定。根后缀将被自动添加到名称中。例如，如果是在 `dc=siroe,dc=com` 根后缀下创建 `ou=groups` 子后缀，则控制台将自动把它命名为 `ou=groups,dc=siroe,dc=com`。

4. 如果希望在创建新的根后缀的同时在当前目录中创建一个数据库，则 “自动创建相关联的数据库”复选框应被默认选中。

   如果想在另外一个目录中或稍后为新的子后缀创建数据库，请取消选中该复选框。在创建数据库之前，新的后缀将处于禁用状态。

5. 如果在步骤 4 中选中 “自动创建相关联的数据库”复选框，则在 “数据库名称”字段中为新数据库输入唯一性的名称。

   该值只可以包含 ASCII （7 位）字母数字字符、连字符 (-) 和下划线 (_)。例如，可以将新数据库命名为 `siroe_sub2`。

6. 单击 “确定”以创建新的子后缀。

   新后缀将自动出现在左侧导航窗口 “数据”树中的根后缀下。

从命令行创建根后缀和子后缀

使用 `ldapmodify` 命令行实用程序可以将新后缀添加到目录配置文件中。后缀配置信息存储在 `cn=mapping tree,cn=config` 条目中。

注意 应避免在 `dse.ldif` 文件中的 `cn=config` 条目下创建条目。 `cn=config` 条目存储在简单的、平面化的 `dse.ldif` 配置文件中，而不象一般条目那样存储在同一个、具有高度伸缩性的数据库中。因此，如果有许多条目，尤其是可能要经常更新的条目储存在 `cn=config` 下面，则性能将会受到严重影响。

然而，虽然由于性能原因不推荐在 `cn=config` 下存储简单的用户条目，但是将诸如目录管理员条目或复制管理器 （供给器绑定 DN）等特殊的用户条目储存在 `cn=config` 下很有用，因为这可以将配置信息集中起来。

例如，假定您想使用 `ldapmodify` 实用程序将新的根后缀添加到配置文件中。按如下所示运行 `ldapmodify`：

```
ldapmodify -a -h host -p port -D "cn=Directory Manager" -w password
```

`ldapmodify` 实用程序将绑定到服务器并准备向配置文件中添加条目。

接着，按如下所示为 siroe.com Corporation 创建根后缀条目：

```
dn: cn="dc=siroe,dc=com",cn=mapping tree,cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsMappingTree
nsslapd-state: backend
nsslapd-backend: UserData
cn: dc=siroe,dc=com
```

要为该根后缀下的组创建子后缀，您需要执行 ldapmodify 操作以添加下列条目：

```
dn: cn="ou=groups,dc=siroe,dc=com",cn=mapping tree,cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsMappingTree
nsslapd-state: backend
nsslapd-backend: GroupData
nsslapd-parent-suffix: "dc=siroe,dc=com"
cn: ou=groups,dc=siroe,dc=com
```

注意 如果想使用 Directory Server Console 来维护后缀，则需要使用与通过命令行来命名根后缀和子后缀时相同的间隔。

例如，如果命名根后缀 ou=groups ,dc=siroe,dc=com （groups 后有两个空格），则在该根后缀下创建的子后缀中也需要在 ou=groups 后面指定两个空格。

下表说明用于配置后缀条目的属性：

表 3-1 后缀属性

属性名	值
dn	定义后缀的 DN。 DN 包含在引号中。所输入的值格式如下：cn=”dc=domain,dc=com”,cn=mapping tree, cn=config 该属性为必需项。
cn	定义条目的相关 DN (RDN)。 该属性为必需项。
objectclass	告知服务器该条目是根后缀条目或子后缀条目。它的值始终为 nsMappingTree。 该属性为必需项。

表 3-1 后缀属性 *（续）*

属性名	值
nsslapd-state	确定后缀如何处理操作。该属性取下列值： • backend：利用后端 （数据库）处理所有操作。 • disabled：数据库不能用于处理操作。服务器响应客户机应用程序请求时，将返回 “无该搜索对象”错误。 • referral：为对该后缀所做的请求返回引荐。 • referral on update：除更新请求 （将接收引荐）外，数据库可用于所有操作。 缺省值为 disabled。
nsslapd-referral	定义由后缀返回的引荐 (referral) 的 LDAP URL。该属性可有多个值，每个值有一个引荐。当 nsslapd-state 属性值是 referral 或 referral on update 时，该属性为必需项。
nsslapd-backend	指定数据库或用于处理请求的数据库链接 (database link) 的名称。该属性可有多个值，每个值有一个数据库或数据库链接。有关数据库链接的详细信息，请参阅第 87 页上的 “创建和维护数据库链接”。 当 nsslapd-state 属性值设置为 backend 或 referral on update 时，该属性为必需项。
nsslapd-distribution-plugin	指定要与自定义分配函数共用的共享库。仅当在 nsslapd-backend 属性中指定多个数据库时，该属性才是必需项。 有关自定义分配函数的详细信息，请参阅第 80 页上的 “创建和维护数据库”。
nsslapd-distribution-funct	指定自定义分配函数的名称。仅当在 nsslapd-backend 属性中指定多个数据库时，该属性才是必需项。 有关自定义分配函数的详细信息，请参阅第 80 页上的 “创建和维护数据库”。
nsslapd-parent-suffix	提供子后缀父项的 DN。默认情况下该属性并不出现。这意味着该后缀被视作根后缀。 例如，假定您想在根后缀 dc=siroe,dc=com 下创建子后缀 o=sales,dc=siroe,dc=com。请将下列值添加到子后缀的 nsslapd-parent-suffix 属性中： nsslapd-parent-suffix: "dc=siroe,dc=com"

维护后缀

本部分介绍以下过程：

在后缀中使用引荐

引荐可用于临时将客户机应用程序指向另一个服务器。例如，您可将引荐添加到后缀，这样当对与后缀有关的数据库进行脱机维护时，后缀可指向另外的服务器。

有关引荐的一般详细信息，请参阅 *iPlanet Directory Server 部署指南*。

要在后缀中设置引荐：

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧窗口的 “数据”下，单击要添加引荐的后缀。
3. 单击 “后缀设置”选项卡。选择 “使用引荐”单选按钮。
4. 单击 “引荐”选项卡。在 “输入新引荐”字段中输入 LDAP URL，或者单击 “构造”，从而在指导下完成 LDAP URL 的创建。

   有关 LDAP URL 结构的详细信息，请参阅附录 C “LDAP URL”。
5. 单击 “添加”以将引荐 (referral) 添加到列表中。

   您可以输入多个引荐。为响应客户机应用程序的请求，目录将返回整个引荐列表。
6. 单击 “保存”。

仅在更新操作期间启用引荐

您可以对目录进行配置，以便将客户机应用程序的更新和写入请求重定向到只读数据库。

例如，如果有目录数据 （非您所拥有）的本地副本，则可为更新操作启用引荐。假定您想使该数据可用于搜索，但不可用于更新。为此，只需在更新请求期间启用引荐。当客户机应用程序请求更新条目时，客户机将被引荐到拥有该数据的服务器，从中继续处理修改请求。

要想仅在更新操作期间启用引荐：

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧窗口的 “数据”下，单击要添加引荐的后缀。
3. 单击 “后缀设置”选项卡。选择 “使用有关更新的引荐”单选按钮。
4. 单击 “引荐”选项卡。在 “输入新引荐”字段中输入 LDAP URL，或者单击 “构造”，从而在指导下完成 LDAP URL 的创建。

   有关 LDAP URL 结构的详细信息，请参阅附录 C “LDAP URL”。
5. 单击 “添加”以将引荐 (referral) 添加到列表中。

   您可以输入多个引荐。为响应客户机应用程序的请求，目录将返回整个引荐列表。
6. 单击 “保存”。

禁用后缀

有时可能需要关闭数据库以进行维护，但该数据库中的数据不会被复制。您可禁用负责该数据库的后缀，而非返回引荐。

禁用后缀后，当客户机应用程序执行 LDAP 操作 （例如搜索、添加和修改）时，它们将看不到与该后缀相关的数据库内容。

要禁用后缀：

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧导航窗口的 “数据”下，单击要禁用的后缀。
3. 单击 “后缀设置”选项卡。取消选中 “启用该后缀”复选框。

   此时 “后缀设置”选项卡上将出现一个红点，提醒用户有需要进行保存的更改内容。
4. 单击 “保存”。

   后缀将不再处于启用状态。

删除后缀

下列过程介绍如何删除后缀：

警告 删除后缀时，将同时删除与该后缀有关的所有数据库条目和复制信息。

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧导航窗口的 “数据”下，选择要删除的后缀。
3. 选择 “对象”菜单中的 “删除 ...”。

   也可右键单击后缀，然后从弹出菜单中选择 “删除 ...”。
4. 如果想删除后缀及其下面的所有后缀，请选择 “删除该后缀以及它所有的子后缀”。

   如果只想删除该特定后缀，而不删除其子后缀，请选择 “只删除该后缀”。
5. 单击 “确定”以删除该后缀。

   此时将显示进度对话框，告诉您控制台正在完成相应的操作。

创建和维护数据库

完成后缀创建以对目录数据加以组织后，请创建包含目录数据的数据库。数据库用于存储目录数据。

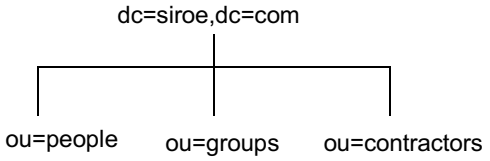
本部分中包括有关创建用于包含目录数据的数据库、删除数据库和将数据库设置为临时只读的信息。

创建数据库

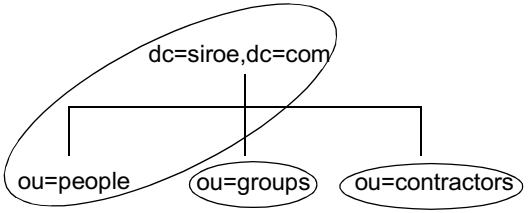
iPlanet Directory Server 5.0 支持使用多个数据库：您可以将目录树中的内容分配到这些数据库中。跨多个数据库分配数据的方式有两种：

- 每个后缀一个数据库。

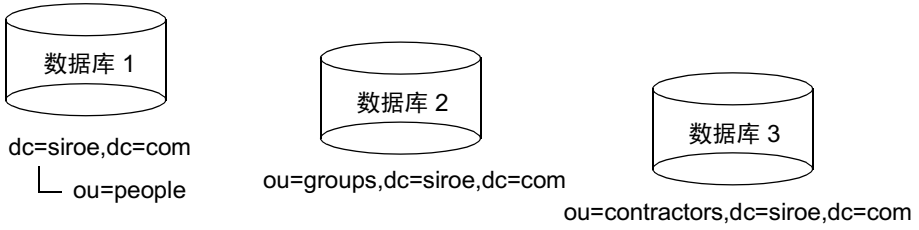
  每个后缀的数据都包含在独立的数据库中。例如，目录树形如：



  您可以按下列方式添加三个数据库，用于存储包含在独立后缀中的数据：



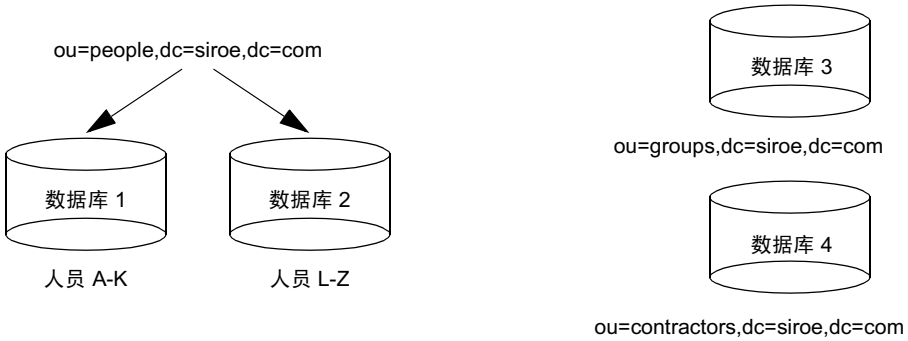
  目录树的这种分割对应于以下三个数据库：



  数据库 1 包含 `ou=people` 的数据外加 `dc=siroe,dc=com` 的数据，这样客户机即可基于 `dc=siroe,dc=com` 执行搜索。数据库 2 包含 `ou=groups` 的数据，而数据库 3 包含 `ou=contractors` 的数据。

- 一个后缀多个数据库。

例如，假设目录树的 ou=people 分支中条目数很大，因此需要用两个数据库进行存储。这种情况下，可将 ou=people 中包含的数据跨两个数据库进行分配。如下所示：



数据库 1 包含姓名从 A 到 K 的人，数据库 2 则包含姓名从 L 到 Z 的人。数据库 3 包含 ou=groups 数据，而数据库 4 包含 ou=contractors 数据。

您需要使用自定义分配插件将单个后缀的数据跨多个数据库进行分配。有关如何为目录服务器创建分布逻辑的信息，请联系 iPlanet 专业服务。有关 iPlanet 专业服务的详细信息，请访问 http://www.iplanet.com/services/。

使用控制台为现有的后缀创建新数据库

下列过程介绍如何将数据库添加到已创建的后缀中：

1. 在 iPlanet Directory Server Console 中，选择 “配置”选项卡。
2. 在左侧窗口中，展开 “数据”，然后单击要添加新数据库的后缀。
3. 右键单击该后缀，然后从弹出菜单中选择 “新数据库”。

   此时显示 “创建新数据库”对话框。
4. 在 “创建新数据库”对话框中，为该数据库输入一个唯一性的名称。

   该值不能包含逗号、制表符、等号 (=)、星号 (*)、反斜杠 (\)、斜杠 (/)、加号 (+)、单引号 (')、双引号 (") 或问号 (?)。例如，可以将新数据库命名为 `siroe2`。
5. 在 “创建数据库位置”字段中，输入用于存储新数据库的目录路径。也可单击 “浏览”以查找本机目录。

   默认情况下，该目录将新数据库储存在以下目录：

   `/var/ds5/slapd-serverID/db`
6. 单击 “确定”。单击确认对话框中的 “是”以创建该新数据库。

注意 要在 “目录”选项卡中查看新后缀，首先需要创建与该后缀关联的根条目。请参阅第 43 页上的 “创建目录项”。

从命令行为单个后缀创建新数据库

使用 `ldapmodify` 命令行实用程序可以将新数据库添加到目录配置文件中。数据库配置信息存储在 `cn=ldbm database,cn=plugins,cn=config` 条目中。

例如，假定您想将新数据库添加到服务器 siroe1 中。如下所示，通过执行 `ldapmodify` 将新条目添加到配置文件中：

```
ldapmodify -a -h host -p port -D "cn=Directory Manager" -w password
```

`ldapmodify` 实用程序将绑定到服务器并准备向配置文件中添加条目。

接着，按如下所示为新数据库创建条目：

```
dn: cn=UserData,cn=ldbm database,cn=plugins,cn=config
objectclass: extensibleObject
objectclass: nsBackendInstance
nsslapd-suffix: ou=people,dc=siroe,dc=com
```

所添加的条目对应于名为 `UserData` 的数据库，其中包含用于 `ou=people, dc=siroe,dc=com` 根后缀或子后缀的数据。

要从命令行创建根后缀或子后缀，请参阅第 75 页上的 “从命令行创建根后缀和子后缀”。 DN 属性中给出的数据库名称必须与后缀条目 `nsslapd-backend` 属性中的值相对应。

为单个后缀添加多个数据库

可以跨多个数据库分配单个后缀。但是，若要分配后缀，需要创建自定义分配函数以扩展目录。有关创建自定义分配函数的详细信息，请联系 iPlanet 专业服务。有关 iPlanet 专业服务的详细信息，请访问 http://www.iplanet.com/services/l。

注意 分配完条目后，将无法再对其进行重新分配。下列限制条件适用：

- 部署完条目的分布方式后，将无法更改分配函数。
- 如果使用 LDAP `modrDN` 操作来重命名条目会导致这些条目被分配到另一个数据库中，请不要使用该操作。
- 您不能复制已分配的本地数据库。
- 如果使用 `ldapmodify` 操作来更改条目会导致这些条目被分配到另一个数据库中，请不要使用该操作。

违背这些限制条件将妨碍 iPlanet Directory Server 进行正确的条目查找和返回。

一旦 iPlanet 专业服务帮您创建完自定义分布逻辑插件后，您需要将其添加到目录中。下面介绍如何将分布逻辑添加到目录后缀的步骤。

向后缀中添加自定义分配函数

分布逻辑是在后缀中声明的函数。任何到达该后缀的操作 （包括从该后缀以上开始的子树搜索操作）都会调用该函数。使用控制台和命令行可以将分配函数插入到后缀中。

有关创建自定义分布逻辑的详细信息，请联系 iPlanet 专业服务。

使用控制台添加自定义分配函数

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧导航窗口中的 “数据”。选择要对其应用分配函数的后缀。
3. 在右侧窗口中，选择 “数据库”选项卡。
4. 单击 “添加”以使附加数据库与后缀关联。

   此时显示 “数据库列表”对话框。从列表中选择数据库，然后单击 “确定”。
5. 在 “分配库”字段中输入到分配库的路径，或者单击 “浏览”以查找本机上的分配库。
6. 在 “函数名称”字段中，输入分配函数的名称。
7. 单击 “保存”可保存更改结果。

从命令行添加自定义分配函数

使用 `ldapmodify` 命令行实用程序可以将下列属性添加到后缀条目中。

```
nsslapd-backend: Database1
nsslapd-backend: Database2
nsslapd-backend: Database3
nsslapd-distribution-plugin: /full/name/of/a/shared/library
nsslapd-distribution-funct: distribution-function-name
```

`nsslapd-backend` 属性指定与该后缀关联的所有数据库。
`nsslapd-distribution-plugin` 属性指定插件所用的库名称。
`nsslapd-distribution-funct` 属性提供分配函数自身的名称。

有关使用 `ldapmodify` 命令行实用程序的详细信息，请参阅第 53 页上的 “使用 ldapmodify 添加和修改条目”。

维护目录数据库

本部分介绍与维护目录数据库有关的作业。它包括以下过程：

- 第 86 页上的 “将数据库置于只读模式”
- 第 87 页上的 “删除数据库”

将数据库置于只读模式

当数据库处于只读模式时，将无法创建、修改或删除任何条目。例如，如果要手动初始化客户服务器，则必须将数据库置于只读模式。

如果目录服务器管理着多个数据库，则通过将整个服务器置于只读模式，即可将所有数据库同时置于只读模式下。有关详细信息，请参阅第 37 页上的 “将整个 Directory Server 置于只读模式”。

本部分包含下列过程：

- 第 86 页上的 “使用控制台将数据库设为只读”
- 第 86 页上的 “从命令行将数据库设为只读”

使用控制台将数据库设为只读

要从服务器控制台将数据库设为只读模式：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧窗口中的 “数据”。展开包含要置于只读模式的数据库的后缀。
3. 选择要设置为只读模式的数据库。
4. 选择右侧窗口中的 “数据库设置”选项卡。
5. 选中 “数据库为只读”复选框。
6. 单击 “保存”。

从命令行将数据库设为只读

如果要手动将数据库设为只读模式，则必须将只读属性 `nsslapd-readonly` 更改为 `on`。为此，请使用 `ldapmodify` 命令行实用程序。特定数据库的 `nsslapd-readonly` 属性位于 `cn=database_name,cn=ldbm database, cn=plugins,cn=config` 条目中 （`database_name` 是数据库名）。

注意 默认情况下，安装时所创建的数据库名为 `userRoot`。

删除数据库

下列步骤介绍如何使用 Directory Server Console 来删除目录数据库。删除数据库时，删除的只是该数据库的配置信息和条目，而非实际的物理数据库。

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧导航窗口，查找所要删除的数据库并将其选定。
3. 从 “对象”菜单中，选择 “删除”。

   也可右键单击数据库，然后从弹出菜单中选择 “删除”。

   此时显示要求您确认删除数据库的对话框。
4. 单击 “是”以确认删除数据库。

   此时将显示进度对话框，告诉您删除过程中目录服务器所完成的步骤。

   删除后，数据库就不会再出现在右侧窗口中。

创建和维护数据库链接

所谓链接，就是服务器代表客户机应用程序与其它服务器进行联系，然后返回组合结果的一种方法。该方法是通过数据库链接 (database link) 实现的。数据库链接指向远程存储的数据。当客户机应用程序从数据库链接请求数据时，数据库链接将从远程数据库检索数据，然后将其返回给客户机。

下列部分介绍如何创建和配置数据库链接。有关链接的一般信息，请参阅 *iPlanet Directory Server 部署指南*中的 “设计目录拓扑结构”。

您可以使用 Directory Server Console 或命令行来创建和配置数据库链接。下列部分介绍创建和维护数据库链接的过程。

- 第 88 页上的 “配置链接策略”
- 第 93 页上的 “创建新数据库链接”
- 第 102 页上的 “使用 SSL 链接”
- 第 102 页上的 “维护数据库链接”
- 第 104 页上的 “数据库链接和访问控制评估”
- 第 105 页上的 “高级功能：调整数据库链接性能”
- 第 109 页上的 “高级功能：配置级联链接”

有关监控数据库活动的详细信息，请参阅第 380 页上的 “监控数据库链接活动”。

配置链接策略

下面部分介绍目录服务器如何将客户机应用程序的请求链接到包含数据库链接的目录服务器的配置过程。链接 (chaining) 策略适用于目录服务器上所创建的所有数据库链接。

本部分包含下列信息：

- 第 88 页上的 “链接组件操作”
- 第 91 页上的 “链接 LDAP 控件”

链接组件操作

组件可以是服务器中使用内部操作的任何功能单元。例如，插件可视为组件，就象前端功能那样。但是，实际上，一个插件可能由多个组件组成 （例如 ACI 插件）。

有些组件向服务器发送内部 LDAP 请求，希望只访问本地数据。对此类组件，您需要控制链接策略，以确保它们能成功地完成自己的操作。例如，不妨考虑证书查验功能。如果链接该功能所生成的 LDAP 请求以检查证书，则意味着您信任远程服务器。如果远程服务器不被信任，则存在安全问题。

默认情况下，所有内部操作都无法进行链接。但是，可使用控制台或命令行来指定所要链接的组件，从而忽略该默认情况。默认情况下不允许链接组件。

同时，还必须在远程服务器上创建 ACI，从而允许指定的插件在远程服务器上执行操作。 ACI 是在分配给数据库链接的后缀 (suffix) 中创建的。

下表列出了组件名、允许将组件链接到内部操作的潜在副作用，以及组件在远程服务器上创建的 ACI 中所需的权限。

表 3-2 允许链接的组件

组件名称	说明	权限
ACI 插件	该插件执行访问控制功能。不要链接用于检索和更新 ACI 属性的操作，这是因为将本地及远程 ACI 属性混合在一起并不安全。但是，可以链接用于检索用户条目的请求。在 `nsActiveChainingComponents` 属性中指定下列值： `nsActiveChainingComponents: cn=ACI Plugin,cn=plugins,cn=config`	读取、搜索和比较

表 3-2 允许链接的组件

组件名称	说明	权限
4.0 插件	该组件名代表所有的 Directory Server 4.0 插件。4.0 插件共享相同的链接策略。在 nsActiveChainingComponents 属性中指定下列内容： `nsActiveChainingComponents: cn=old Plugin,cn=plugins,cn=config`	取决于允许链接的 4.0 插件
资源限制组件	该组件将根据用户绑定 DN (bind DN) 来设置服务器限制。如果允许链接到资源限制组件，则可对远程用户应用资源限制。要链接该组件的操作，请指定下列内容： `nsActiveChainingComponents: cn=resource limits,cn=components,cn=config`	读取、搜索和比较
基于证书的验证检查组件	如果使用 SASL 外部绑定方法，则使用该组件。它将从远程服务器上的数据库中检索用户证书。如果允许该组件进行链接，则可以将基于证书的验证与数据库链接一起使用。要链接该组件的操作，请指定下列内容： `nsActiveChainingComponents: cn=certificate-based authentication,cn=components,cn=config`	读取、搜索和比较
参照完整性插件	该插件可确保对包含 DN 的属性所做的更新将被传播到包含指向该属性之指针的所有条目中。例如，如果所删除的条目是某个组的成员，则该条目将会自动从该组中被删除。如果组成员是静态组定义的远程对象，则将该插件和链接一起使用将有助于简化静态组的管理。 要链接该组件的操作，请指定下列内容： `nsActiveChainingComponents: cn=referential integrity postoperation,cn=plugins,cn=config`	读取、写入、搜索和比较
UID 唯一性插件	该插件将检查所指定的 uid 属性的所有属性值是否唯一（无重复）。如果允许该插件进行链接，则即使通过数据库链接更改 uid 属性的值，该插件也将确认其是否具有唯一性。要链接该组件的操作，请指定下列内容： `nsActiveChainingComponents: cn=uid uniqueness,cn=plugins,cn=config`	读取、搜索和比较

注意 不能链接到下列组件：

- 角色插件
- 口令策略组件
- 复制插件

有关围绕 ACI 和链接的限制的详细信息，请参阅第 180 页上的 “ACI 限制”。

修改允许链接的组件后，必须重新启动服务器以使修改生效。

下列部分介绍如何使用控制台和命令行指定允许链接的组件。

使用控制台链接组件操作

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧窗口中的 “数据”，然后单击 “数据库链接设置”。
3. 在右侧窗口中，选择 “设置”选项卡。要将组件添加到 “允许链接的组件”列表中，请单击 “添加”。

   此时显示 “选择要添加的组件”对话框。从列表中选择组件，然后单击 “确定”。
4. 要从列表中删除组件，请选中该组件，然后单击 “删除”。
5. 修改组件列表后，选项卡上将出现一个红点，同时文件名将变为灰色。单击 “保存”可保存更改结果。

   重新启动服务器以使更改生效。

允许组件进行链接后，必须在操作将要链接的远程服务器后缀中创建 ACI。例如，可以为参照完整性插件创建下列 ACI：

```
aci: (targetattr
 "*")(target="ldap:///ou=customers,l=us,dc=siroe,dc=com")
 (version 3.0; acl "RefInt Access for chaining"; allow
 (read,write,search,compare) userdn = "ldap:///cn=referential
 integrity postoperation,cn=plugins,cn=config";)
```

从命令行链接组件操作

可以使用配置文件 cn=config,cn=chaining database,cn=plugins,cn=config 条目中的 nsActiveChainingComponents 属性来指定要包含在链接中的组件。

例如，如果想允许参照完整性组件链接某些操作，则可将下列内容添加到数据库链接配置文件中：

```
nsActiveChainingComponents: cn=referential integrity postoperation,
 cn=components,cn=config
```

有关允许链接的组件列表，请参阅第 88 页的表 3-2。

修改完 nsActiveChainingComponents 属性后，必须重新启动服务器以使更改生效。

允许组件进行链接后，必须在操作将要链接的远程服务器后缀中创建 ACI。例如，可以为参照完整性组件创建下列 ACI：

```
aci: (targetattr
 "*")(target="ldap:///ou=customers,l=us,dc=siroe,dc=com")
 (version 3.0; acl "RefInt Access for chaining"; allow
 (read,write,search,compare) userdn = "ldap:///cn=referential
 integrity postoperation,cn=plugins,cn=config";)
```

链接 LDAP 控件

您可以选择不链接 LDAP 控件的操作请求。默认情况下，下列控件的请求将被数据库链接转发给远程服务器：

- 受管理的 DSA — 该控件将智能引荐作为条目返回，而非遵从引荐。这样即允许修改或删除智能引荐自身。
- 循环检测 — 该控件跟踪服务器与其它服务器进行链接的次数。当计数达到配置的数目时，就会检测到循环并通知客户机应用程序。

  有关使用该控件的详细信息，请参阅第 115 页上的 “检测循环”。
- 服务器端排序 — 该控件根据条目的属性值对条目进行排序。
- 虚拟列表视图 (VLV) — 该控件提供包含部分结果的列表，而不是一次返回搜索结果的所有条目。

注意 服务器端排序和 VLV 控件只在搜索范围是单个数据库时通过链接获得支持。当客户机应用程序向多个数据库发出请求时，数据库链接将无法支持 VLV 控件。

下列部分介绍如何使用控制台和命令行来改变数据库链接所转发的控件。

使用控制台链接 LDAP 控件

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧窗口中的 “数据”文件夹，然后单击 “数据库链接设置”。
3. 在右侧窗口中，选择 “设置”选项卡。要将 LDAP 控件添加到列表中，请单击 “添加”。

   此时显示 “选择要添加的控件 OID”对话框。选择要添加到列表中的控件 OID，然后单击 “确定”。
4. 要从列表中删除控件，请从 “转发给远程服务器的 LDAP 控件”列表中选定该控件，然后单击 “删除”。
5. 修改完组件列表后，选项卡上将出现一个红点，同时组件字段名称将变为灰色。单击 “保存”可保存更改结果。

从命令行链接 LDAP 控件

通过更改 `cn=config,cn=chaining database,cn=plugins,cn=config` 条目的 `nsTransmittedControls` 属性，可以改变数据库链接所转发的控件。例如，要转发虚拟列表视图控件，请将下列内容添加到配置文件的数据库链接条目中：

```
nsTransmittedControls: 2.16.840.1.113730.3.4.9
```

另外，如果目录服务器的客户机还创建有自己的控件，而您也希望将其操作链接到远程服务器上，则需要将自定义控件的 OID 添加到 `nsTransmittedControls` 属性中。

下表列出可链接的 LDAP 控件及其 OID：

表 3-3 LDAP 控件及其 OID

控件名称	OID
虚拟列表视图 (VLV)	2.16.840.1.113730.3.4.9
服务器端排序	1.2.840.113556.1.4.473
受管理的 DSA	2.16.840.1.113730.3.4.2
循环检测	1.3.6.1.4.1.1466.29539.12

有关 LDAP 控件的详细信息，请参阅 http://docs.iplanet.com/docs/manuals/directory.html 上的 LDAP C-SDK 文档。

创建新数据库链接

数据库链接的基本配置提供下列信息：

后缀信息。您需要在由数据库链接 （而不是常规数据库）管理的目录树中创建后缀。该后缀与包含此数据的远程服务器后缀相对应。

绑定凭证。当数据库链接绑定到远程服务器时，它将扮演用户。对于每个数据库链接，您需要指定希望其在绑定远程服务器时使用的 DN 和凭证。

LDAP URL。提供数据库链接所连接的远程服务器的 LDAP URL。

故障替换服务器列表。可以为数据库链接提供一个备用服务器列表，以便在出现故障时进行连接。该配置项为可选项。

注意 在安全链接使用 SSL 的情况下绝对不能使用此选项。

下列部分介绍从 Directory Server Console 和命令行创建新数据库链接。

使用控制台创建新的数据库链接

要使用 Directory Server Console 创建新的数据库链接：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡
2. 右键单击左侧导航窗口中的 “数据”，然后从弹出菜单中选择 “新根后缀”或 “新子后缀”。

   此时显示 “创建新后缀”对话框。
3. 在 “新后缀”字段中，输入所要链接的远程服务器的后缀名。

   后缀的命名必须遵循 dc 命名约定。例如，可以输入名为 `dc=siroe,dc=com` 的新后缀。
4. 取消选中 “自动创建相关联的数据库”复选框。

   之所以取消该复选框，是因为您无法将数据库链接添加到与数据库相关联的后缀中。该后缀仅供数据库链接使用。
5. 单击 “确定”以创建新的后缀。

   后缀将自动出现在左侧导航窗口的 “数据”分支下。
6. 在左侧窗口中，右键单击刚创建的后缀，然后从弹出菜单中选择 “新数据库链接”。

   此时显示 “创建新数据库链接”对话框。

7. 在 “数据库链接名称”字段中，输入新数据库链接的名称。

   命名数据库链接时只能使用 ASCII （7 位）字符。该值不能包含逗号、制表符、等号 (=)、星号 (*)、反斜杠 (\)、斜杠 (/)、加号 (+)、单引号 (')、双引号 (") 或问号 (?)。例如，可以将新数据库链接命名为 `siroelink1`。

8. 在 “绑定 DN”字段中，输入数据库链接绑定到远程服务器时所用的 DN。

   例如，可在 “绑定 DN”字段中输入 `cn=dblink`。

9. 在 “口令”字段中，输入数据库链接绑定到远程服务器时所用的口令。

10. 如果希望数据库链接使用 SSL 来与远程服务器进行通信，请选中 “使用服务器之间的安全 LDAP 连接”复选框。

11. 在 “远程服务器”字段中输入远程服务器的名称。在 “远程服务器端口”字段中，输入绑定所用的服务器端口号。缺省端口号为 `389`。

12. 在 “故障替换服务器”字段中输入故障替换服务器的名称，同时在 “端口”字段中指定端口号。缺省端口号为 `389`。单击 “添加”以将故障替换服务器添加到列表中。

    您可以指定多个故障替换服务器。如果主远程服务器出现故障，则数据库链接将与 “故障替换服务器”列表中的第一个服务器进行连接。如果失败，则连接列表中的下一个服务器，依此类推。

13. 单击 “确定”以创建新的数据库链接。单击 “确定”以关闭创建完数据库后出现的成功对话框。

    新的数据库链接将出现在左侧导航窗口的后缀下。

提示 控制台提供了一个信息校验表，要成功绑定数据库链接，远程服务器上需要有该表。要查看该校验表，请单击新数据库链接，然后单击 “验证”选项卡。校验表出现在 “远程服务器校验表”框中。

从命令行创建数据库链接

使用 `ldapmodify` 命令行实用程序可以从命令行创建新的数据库链接。

新的实例必须位于 `cn=chaining database,cn=plugins,cn=config` 条目中。

缺省配置属性包含在 `cn=default config,cn=chaining database,cn=plugins, cn=config` 条目中。这些配置属性将在创建期间应用于所有数据库链接。对缺省配置所做的更改将仅影响新的数据库链接。您无法更改现有数据库链接上的缺省配置信息。

每个数据库链接包含自己的特定配置信息。该信息与数据库链接条目自身 cn=database_link_name,cn=chaining database,cn=plugins,cn=config 一起存储。有关配置属性的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

本部分包含有关从命令行配置数据库链接的以下过程：

- 第 95 页上的 “提供后缀信息”
- 第 95 页上的 “提供绑定凭证”
- 第 97 页上的 “提供 LDAP URL”
- 第 97 页上的 “提供故障替换服务器列表”
- 第 116 页上的 “级联链接配置属性概要”
- 第 99 页上的 “数据库链接配置示例”

提供后缀信息

使用 nsslapd-suffix 属性可以定义数据库链接所管理的后缀。例如，如果想使数据库链接指向公司远程站点的人员信息，则输入下列后缀信息：

```
nsslapd-suffix: l=Zanzibar,ou=people,dc=siroe,dc=com
```

后缀信息存储在 cn=database_link_name,cn=chaining database,cn=plugins,cn=config 条目中。

注意 创建后，对 nsslapd-suffix 属性所做的任何修改都将在重新启动包含该数据库链接的服务器后才生效。

提供绑定凭证

对于来自要链接到远程服务器的客户机应用程序的请求，可以为该客户机应用程序提供特殊的绑定凭证。该操作赋予远程服务器链接操作所需的代理验证权限。如果未指定绑定凭证，则数据库链接将以匿名方式绑定到远程服务器。

提供绑定凭证包含下列步骤：

1. 在远程服务器上，需要执行下列操作：

   a. 为数据库链接创建管理级用户。

      有关添加条目的信息，请参阅第 41 页上的 “创建目录项”。

   b. 在数据库链接所链接的子树上，为上面步骤中创建的管理级用户提供代理访问权限。

      有关配置 ACI 的详细信息，请参阅第 177 页上的 “管理访问控制”。

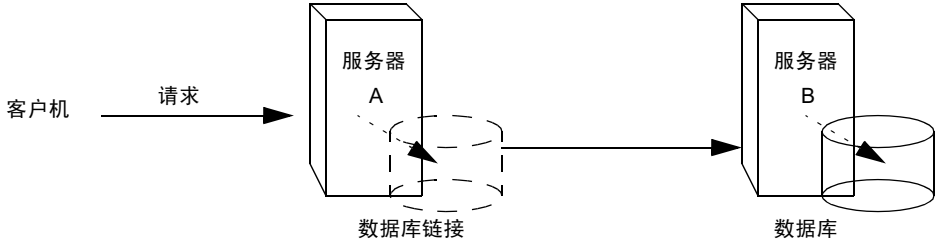
2. 在包含数据库链接的服务器上，需要执行下列操作：

   a. 使用 ldapmodify 为 cn=database_link_name,cn=chaining database, cn=plugins,cn=config 条目的 nsMultiplexorBindDN 属性中的数据库链接提供用户 DN。

警告 nsMultiplexorBindDN 不能为目录管理员所用。

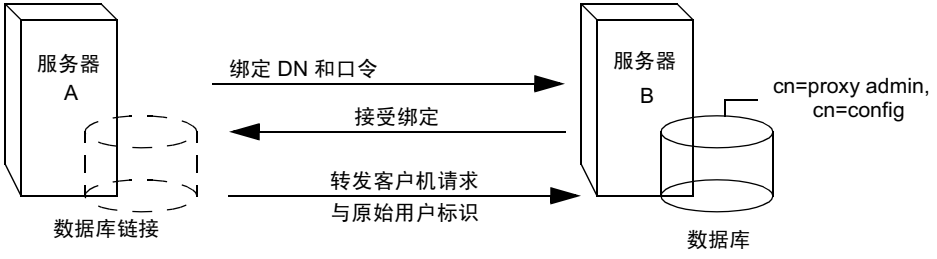
   b. 使用 ldapmodify 为 cn=database_link_name,cn=chaining database, cn=plugins,cn=config 条目的 nsMultiplexorCredentials 属性中的数据库链接提供用户口令。

例如，假定客户机应用程序向服务器 A 发送请求。服务器 A 中包含将该请求链接到服务器 B 之数据库的数据库链接。



服务器 A 上的数据库链接使用在 nsMultiplexorBindDN 属性中定义的特殊用户及在 nsMultiplexorCredentials 属性中定义的用户口令绑定到服务器 B。在本例中，服务器 A 使用下列绑定凭证：

```
nsMultiplexorBindDN: cn=proxy admin,cn=config
nsMultiplexorCredentials: secret
```



服务器 B 必须包含与 nsMultiplexorBindDN 相对应的用户条目，且您必须为该用户设置代理验证权限。要设置代理验证权限，需要与设置其它任何 ACI 一样设置代理 ACI。

警告 启用链接时，请仔细检查访问控制，以免赋予对目录受限区域的访问权。例如，如果在分支上创建默认代理 ACI，则通过该数据库链接进行连接的用户将能看到该分支下的所有条目。有时您并不希望用户看见所有子树。为避免出现安全漏洞，可以创建附加的 ACI，用于限制对子树的访问。

有关 ACI 的详细信息，请参阅第 177 页上的 “管理访问控制”。有关代理验证控件的详细信息，请参阅 http://developer.iplanet.com/docs/manuals/directory.html 上的 C-SDK 文档。

注意 如果客户机应用程序使用数据库链接来创建或修改条目，则属性 creatorsName 和 modifiersName 将不会反映该条目的实际创建者或修改者。这些属性中包含远程数据服务器上被授予代理验证权限的管理级用户的姓名。

提供 LDAP URL

在包含数据库链接的服务器上，必须标识数据库链接使用 LDAP URL 来进行连接的远程服务器。与标准 LDAP URL 格式不同，远程服务器的 URL 不指定后缀。其格式如下所示：

```
ldap://servername:portnumber/
```

使用配置文件的 cn=database_link_name,cn=chaining database,cn=plugins,cn=config 条目中的 nsFarmServerURL 属性可以指定远程服务器的 URL。例如，nsFarmServerURL 格式可能为如下所示：

```
nsFarmServerURL: ldap://siroe.com:389/
```

不要忘记 URL 的末尾使用尾随斜杠 (/)。

如果想使用 SSL 环境下的 LDAP 将数据库链接连接到远程服务器上，则远程服务器的 LDAP URL 格式如下所示：

```
ldaps://servername:portnumber/
```

有关链接和 SSL 的详细信息，请参阅第 102 页上的 “使用 SSL 链接”。

提供故障替换服务器列表

您可以提供出现故障时服务器所用的附加 LDAP URL。为此，请将备用服务器添加到 nsFarmServerURL 属性中，二者之间以空格分隔。例如，可以键入下列内容：

```
nsFarmServerURL: ldap://siroe.com us.siroe.com:389
 africa.siroe.com:1000/
```

在该 LDAP URL 示例中，数据库链接将首先连接标准端口上的 siroe.com 服务器，从而执行相应的操作。如果没有响应，则数据库链接将转而连接端口 389 上的服务器 us.siroe.com。如果该服务器出现故障，则连接端口 1000 上的 africa.siroe.com 端口。

数据库链接配置属性概要

下表列出了可用于配置数据库链接的属性。其中的某些属性已在前面的章节中讨论过。

标有星号 (*) 的属性既可能是全局属性，也可能是实例属性。所有实例属性均在 cn=database_link_name,cn=chaining database,cn=plugins,cn=config 条目中
定义。

这两个全局配置属性位于 cn=config,cn=chaining database,cn=plugins, cn=config 条目中。全局属性是动态的，即所做的任何更改都将自动在目录的所有数据库链接实例中生效。

为特定数据库链接所定义的值将优先于全局属性的值。

表 3-4 数据库链接配置属性

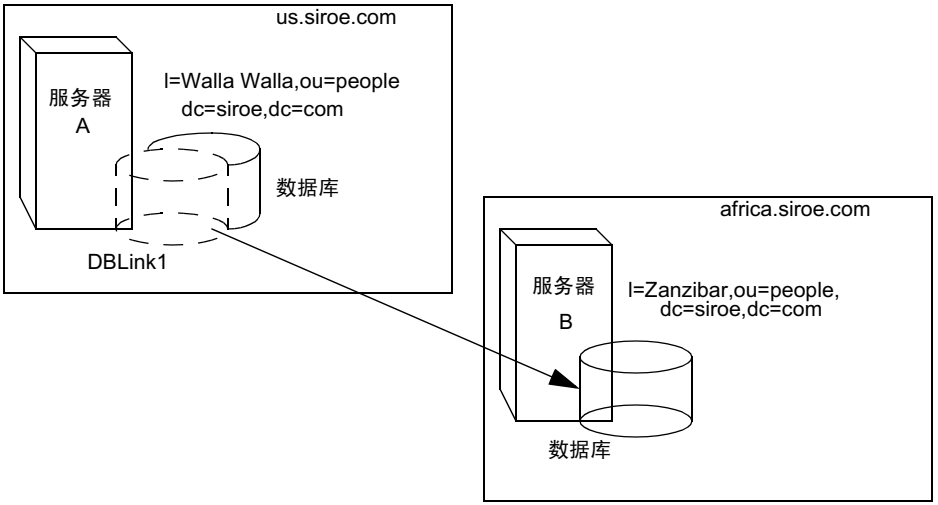
属性	值
nsTransmittedControls*	给出由数据库链接转发给远程数据服务器的 LDAP 控件的 OID。
nsslapd-suffix	由数据库链接管理的后缀。条目创建后，对该属性所做的任何更改都将在重新启动包含该数据库链接的服务器后才会生效。
nsslapd-timelimit	数据库链接的默认搜索时间限制 （以秒计）。默认值为 3600 秒。
nsslapd-sizelimit	数据库链接的默认大小限制 （以条目数计）。默认值为 2000 个条目。
nsFarmServerURL	给出包含数据的远程服务器 （或 FARM 服务器）的 LDAP URL。该属性可包含用于故障替换的可选服务器，二者之间以空格分隔。如果使用级联链接，则该 URL 可指向其它数据库链接。
nsMultiplexorBindDN	用于和远程服务器进行通信的管理条目的 DN。属性名称中的 *multiplexor* 一词表示包含数据库链接并与远程服务器进行通信的服务器。 该绑定 DN 不能是目录管理员。如果未指定该属性，则数据库链接将以匿名方式绑定。

表 3-4 数据库链接配置属性 *（续）*

属性	值
nsMultiplexorCredentials	管理级用户的口令 （纯文本形式）。如果未提供口令，则表示用户可以匿名方式绑定。口令在配置文件中进行加密。
nsCheckLocalACI	只为高级使用保留。控制 ACI 是否既在数据库链接也在远程数据服务器上进行评估。取值为 on 或 off。 对该属性所做的更改将在服务器重新启动后生效。默认值为 off。
nsProxiedAuthorization	只为高级使用保留。允许禁用代理验证。值取 off 时，意味着禁用代理验证。默认值为 on。
nsActiveChainingComponents*	列出使用链接的组件。组件指服务器中的任何功能性装置。该属性在数据库链接实例中的值将取代全局配置属性中相应的值。要在特定的数据库实例上禁用链接，请使用值 none。 默认策略为不允许链接。有关详细信息，请参阅第 88 页上的“链接组件操作”。
nsReferralOnScopedSearch	控制按范围搜索时是否返回引荐。该属性旨在优化目录，因为按范围进行搜索时，返回引荐更为有效。取值为 on 或 off。默认值为 off。
nsHopLimit	某个请求可在数据库链接之间转发的最大次数。默认值为 10。

数据库链接配置示例

假设您拥有 us.siroe.com 域中的某台服务器，且该服务器中包含数据库上的 l=Walla Walla,ou=people,dc=siroe,dc=com 子树，同时您希望将 l=Zanzibar,ou=people,dc=siroe,dc=com 的操作请求链接到 africa.siroe.com 域中的另一台服务器上。该操作图示如下：



首先，使用 `ldapmodify` 命令行实用程序将数据库链接添加到服务器 A 中。

```
ldapmodify -a -h us.siroe.com -p port \
          -D "cn=Directory Manager" -w password
```

然后为数据库链接指定配置信息：

```
dn: cn=DBLink1,cn=chaining database,cn=plugins,cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsBackendInstance
nsslapd-suffix: l=Zanzibar,ou=people,dc=siroe,dc=com
nsfarmserverurl: ldap://africa.siroe.com:389/
nsmultiplexorbinddn: cn=proxy admin,cn=config
nsmultiplexorcredentials: secret
cn: DBLink1

dn: cn="l=Zanzibar,ou=people,dc=siroe,dc=com",cn=mapping
 tree,cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsMappingTree
nsslapd-state: backend
nsslapd-backend: DBLink1
nsslapd-parent-suffix: "ou=people,dc=siroe,dc=com"
cn: l=Zanzibar,ou=people,dc=siroe,dc=com
```

在第一部分中，`nsslapd-suffix` 属性内包含要从服务器 A 链接的服务器 B 上的后缀。`nsFarmServerURL` 属性中包含服务器 B 的 LDAP URL。

第二部分将创建新后缀，允许服务器为面向新数据库链接所发出的请求进行路由选择。 cn 属性内包含的后缀与数据库链接的 nssalpd-suffix 属性所指定的相同。 nsslapd-backend 属性包含数据库链接的名称。 nsslapd-parent-suffix 属性则指定该新后缀的父项：ou=people,dc=siroe,dc=com。

接着，在服务器 B 上按如下所示创建管理级用户：

```
dn: cn=proxy admin,cn=config
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: proxy admin
sn: proxy admin
userPassword: secret
description: Entry for use by database links
```

警告 不要将 “目录管理员”用户当作远程服务器上的代理管理级用户使用。这会造成安全漏洞。

将下列代理验证 ACI 添加到服务器 B 的 l=Zanzibar,ou=people,dc=siroe,dc=com 条目中：

```
aci: (targetattr = "*")(version 3.0; acl "Proxied authorization for
 database links"; allow (proxy) userdn = "ldap:///cn=proxy
 admin,cn=config";)
```

该 ACI 只允许代理管理级用户对 l=Zanzibar,ou=people,dc=siroe,dc=com 子树中远程服务器上所含的数据进行只读访问。

注意 当用户绑定到数据库链接时，用户的标识将被发送给远程服务器。访问控制的评估总是在远程服务器上进行。为确保用户能成功地修改数据或将数据写入到远程服务器中，需要在远程服务器上设置正确的访问控制。

有关如何在链接操作环境中评估访问控制的详细信息，请参阅第 104 页上的 “数据库链接和访问控制评估”。

使用 SSL 链接

可以对数据库链接进行配置，从而使用 SSL 与远程服务器进行通讯。要在链接时使用 SSL，请执行下列步骤：

- 在远程服务器上启用 SSL。

  有关启用 SSL 的详细信息，请参阅第 350 页上的 “启用 SSL：步骤摘要”。

- 以 SSL 格式指定远程服务器的 LDAP URL。

  在 `nsFarmServerURL` 属性中指定 LDAP URL。有关该属性的详细信息，请参阅第 97 页上的 “提供 LDAP URL”。

  例如，可以指定下列 LDAP URL：

  ```
  nsFarmServerURL: ldaps://africa.siroe.com:636/
  ```

- 在包含数据库链接的服务器上启用 SSL。

  有关启用 SSL 的详细信息，请参阅第 350 页上的 “启用 SSL：步骤摘要”。

将数据库链接和远程服务器配置为使用 SSL 进行通讯时，并不表示发出操作请求的客户机应用程序也必须使用 SSL 进行通讯。客户机可使用正常的端口进行绑定。

维护数据库链接

本部分介绍如何更新和删除现有的数据库链接。其中包括以下过程：

- 第 102 页上的 “更新远程服务器验证信息”
- 第 103 页上的 “删除数据库链接”

更新远程服务器验证信息

要更新数据库链接连接到远程服务器时所用的绑定 DN 和口令：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧窗口中，展开 “数据”，然后查找某个后缀下要更新的数据库链接。选定该数据库链接。
3. 在右侧导航窗口中，单击 “验证”选项卡。

4. 要更新远程服务器信息，请在 “远程服务器 URL”字段中输入新的 LDAP URL。

   与标准 LDAP URL 格式不同，远程服务器的 URL 不指定后缀。其格式如下所示：

   `ldap://`*`servername:portnumber`*`/`

5. 在 “数据库链接绑定 DN”字段中，输入新的 DN，从而更新数据库链接与远程服务器绑定时所用的绑定 DN。

6. 在 “数据库链接口令”字段中，输入新的口令，从而更新数据库链接与远程服务器绑定时所用的口令。在 “确认数据库链接口令”字段中，请重新输入口令以进行确认。

   远程服务器检验表框将列出管理级用户条目、后缀，同时还将列出为确保数据库链接成功绑定而需要的远程服务器 ACI。

7. 单击 “保存”可保存更改结果。

删除数据库链接

要删除数据库链接：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。

2. 在左侧导航窗口，查找所要删除的数据库链接并将其选中。

3. 从 “对象”菜单中，选择 “删除”。

   也可右键单击数据库链接，然后从弹出菜单中选择 “删除”。

   此时显示 “删除数据库链接”确认对话框。

4. 单击 “是”以确认要删除数据库链接。

   此时将显示进度对话框，告诉您删除过程中目录服务器所完成的步骤。

   删除后，数据库链接就不会再出现在右侧窗口中。

数据库链接和访问控制评估

当用户绑定到包含数据库链接的服务器时，数据库链接将把用户的标识发送给远程服务器。访问控制的评估总是在远程服务器上进行。远程服务器上所评估的每个 LDAP 操作都使用由代理验证控件传递的客户机应用程序原始标识。只有在用户对远程服务器上所含的子树具有正确的访问控制权限时，对远程服务器的操作才能成功进行。这就意味着在向远程服务器添加常规访问控制时，需要设置一定的限制条件：

- 不能使用所有类型的访问控制。

  例如，基于角色或基于过滤器的 ACI 需要具有对用户条目的访问权。因为是在通过数据库链接来访问数据，所以只能验证代理控件中的数据。设计目录时，应确保用户条目位于与用户数据相同的数据库中。

- 由于客户机的原始域在链接期间丢失，因此所有基于客户机 IP 地址或 DNS 域的访问控制都可能无法正常工作。

  在远程服务器看来，客户机应用程序与数据库链接的 IP 地址和 DNS 域相同。

对于旨在与数据库链接一起使用的 ACI 而言，下列限制条件适用：

- ACI 必须与它所用的组放在一起。如果组是动态的，则组内的所有用户都必须与 ACI 和组放在一起。如果组为静态的，则它可能指向远程用户。
- ACI 必须与所用的任何角色 (role) 定义及计划使用这些角色的用户放在一起。
- 如果用户为远程用户，则指向该用户条目值的 ACI （例如 `userattr` 对象规则）将生效。

尽管访问控制始终在远程服务器上进行评估，但也可选择在包含数据库链接的服务器上及在远程服务器上进行评估。这会产生一些限制条件：

- 在访问控制评估期间，用户条目的内容不一定可用 （例如，如果是在包含数据库链接的服务器上评估访问控制，而条目位于远程服务器上）。

  由于性能方面的原因，客户机不能执行远程查询和评估访问控制。

- 对于客户机应用程序正在修改的条目，数据库链接不一定具有访问该条目的权限。

  修改过程中，对于存储于远程服务器上的条目而言，数据库链接并没有对全部条目的访问权限。删除过程中，数据库链接将只知道条目的 DN。如果访问控制指定了特殊的属性，则当通过数据库链接执行删除操作时，该操作将失败。

注意 默认情况下，系统将不对在包含数据库链接的服务器上所设的访问控制进行评估。要覆盖该默认设置，请使用 cn=database_link_name, cn=chaining database,cn=plugins,cn=config 条目中的 nsCheckLocalACI 属性。不过，除非使用级联链接，否则我们建议不要对包含数据库链接的服务器进行访问控制的评估。

高级功能：调整数据库链接性能

下列部分提供有关通过连接和线程管理来调整数据库链接性能的信息。其中包含以下部分：

- 第 105 页上的 “管理到远程服务器的连接”
- 第 107 页上的 “正常处理过程中检测错误”
- 第 108 页上的 “管理线程操作”

管理到远程服务器的连接

每个数据库链接都维护一个到远程服务器的连接池。您可以对连接进行配置，从而优化目录资源。

可以使用 Directory Server Console 或通过命令行来更改连接属性。

使用控制台管理到远程服务器的的连接

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧窗口中的 “数据”文件夹，然后查找要更改的数据库链接。单击数据库链接，然后单击右侧导航窗口中的 “限制和控制”选项卡。
3. 在 “连接管理”部分中，修改下面各字段：

   最大 TCP 连接数。数据库链接与远程服务器建立的 TCP 连接的最大数量。默认值为 3 个连接。

   绑定超时。数据库链接的绑定尝试在超时前的时间 （以秒计）。默认值为 15 秒。

   每个连接的最多绑定数。每个 TCP 连接未完成的绑定操作的最大数量。默认值为每个连接 10 个未完成的绑定操作。

   放弃前的超时时间 （秒）。服务器在决定是否放弃已超时连接前的秒数。默认值为 2 秒。

最大 LDAP 连接数量。数据库链接与远程服务器建立 LDAP 连接的最大数量。默认值为 10 个连接。

最大绑定条目数。数据库链接尝试绑定到远程服务器的次数。值取 0 时，表示数据库链接将仅尝试绑定一次。默认值为尝试 3 次。

每个连接的最多操作数量。每个 LDAP 连接未完成操作的最大数量。默认值为每个连接 10 个操作。

连接使用期（秒）。数据库链接和远程服务器间连接持续打开的时间。可以使数据库链接与远程数据库之间的连接始终处于打开状态，也可使之在指定时间后关闭。

使连接保持打开状态可以提高操作速度，但会占用更多的资源。例如，如果正在使用拨号连接，则可能希望限制连接时间。

值取 0 时，表示没有限制。默认情况下，此值设为 0。

4. 单击“保存”可保存更改结果。

通过命令行管理到远程服务器的连接

使用 `ldapmodify` 可以将连接属性添加到数据库链接条目中。

默认的连接管理属性存储在下列条目中：`cn=default instance config, cn=chaining database,cn=plugins,cn=config`。

特定数据库链接的连接管理属性存储在下列条目中：`cn=database_link_name, cn=chaining database,cn=plugins,cn=config`，其中 *database_link_name* 是该数据库链接的名称。该条目中指定的连接管理属性优先于 `cn=default instance config` 条目中指定的属性。

下表列出了与连接管理有关的属性：

表 3-5 数据库链接的连接管理属性

属性名	说明
`nsOperationConnectionsLimit`	数据库链接与远程服务器建立 LDAP 连接的最大数量。默认值为每个数据库链接实例有 10 个连接。
`nsBindConnectionsLimit`	数据库链接与远程服务器建立 TCP 连接的最大数量。默认值为 3 个连接。
`nsConcurrentOperationsLimit`	每个 LDAP 连接未完成操作的最大数量。默认值为每个连接 10 个操作。
`nsConcurrentBindLimit`	每个 TCP 连接未完成的绑定操作的最大数量。默认值为 10 个未完成的绑定操作。

表 3-5 数据库链接的连接管理属性 *（续）*

属性名	说明
nsBindRetryLimit	数据库链接尝试绑定到远程服务器的次数。值取 0 时，表示数据库链接将仅尝试绑定一次。默认值为尝试 3 次。
nsConnectionLife	连接使用期 （以秒计）。可以使数据库链接与远程数据库之间的连接始终处于打开状态，也可使之在指定时间后关闭。 使连接保持打开状态可以提高操作速度，但会占用更多的资源。例如，如果正在使用拨号连接，则可能希望限制连接时间。 值取 0 时，表示没有限制。默认情况下，此值设为 0。如果值取 0，且在 nsFarmServerURL 属性中提供故障替换服务器列表，则在备份服务器出现故障时将不连接主服务器。 默认值为 0 秒。
nsBindTimeout	尝试绑定超时前的时间 （单位：秒）。默认值为 15 秒。
nsAbandonedSearchCheckInterval	服务器确认放弃操作前等待的秒数。默认值为 2 秒。

有关数据库链接配置属性的列表，请参阅第 98 页上的 “数据库链接配置属性”。

正常处理过程中检测错误

通过在正常链接操作期间检测数据库链接和远程服务器之间的错误，可有助于保护服务器的性能。同时使用数据库链接的两个属性，即可确定远程服务器是否不再予以响应。

第一个属性即 nsMaxResponseDelay，用于设置完成 LDAP 操作的最长持续时间。如果操作的时间超过此属性中指定的值，则数据库链接服务器将怀疑远程服务器是否仍然在线。

达到 nsMaxResponseDelay 的时间后，数据库链接将对远程服务器执行 ping 操作。在 ping 操作过程中，数据库链接将发出另一个 LDAP 请求：一个针对远程服务器中并不存在的对象的简单搜索请求。使用 nsMaxTestResponseDelay 可以设置 ping 的持续时间。

如果远程服务器在超出 nsMaxResponseDelay 时间限制后仍未响应，系统就会返回错误信息，同时将连接标记为关闭。数据库连接和远程服务器之间的所有连接都将阻塞 30 秒，以防止服务器的性能降低。 30 秒后，数据库链接向远程服务器发出的操作请求将恢复常规状态。

这两个属性都存储于 cn=config,cn=chaining database,cn=plugins,cn=config 条目中。下表介绍这两个属性的详细信息：

表 3-6 数据库链接处理错误检测参数

属性名	说明
nsMaxResponseDelay	在怀疑出错以前，远程服务器对数据库链接发出的 LDAP 操作请求做出响应的最长时间。该期限以秒计。默认的延迟时间为 60 秒。 达到该延时标准后，数据库链接将测试与远程服务器的连接。
nsMaxTestResponseDelay	由数据库链接发出的、用于检查远程服务器是否仍在响应的持续时间测试。如果远程服务器在该时间段后仍未做出响应，则数据库链接假定远程服务器处于关闭状态，该连接将不会用于后续操作。 该期限以秒计。默认的测试响应延迟时间为 15 秒。

管理线程操作

一般情况下， iPlanet Directory Server 在使用有限的线程来执行处理操作时性能最佳。如果线程数目有限，执行操作时通常就会较快，从而可防止等待可用线程的操作队列过长。

不过，数据库链接会将操作转发给远程服务器进行处理。数据库链接将连接远程服务器，转发操作，等待结果，然后再将结果发回给客户机应用程序。整个操作可能比本地操作的时间长得多。

数据库链接在等待远程服务器的结果时，可同时处理其它操作。默认情况下，服务器使用的线程数为 20 个。但在使用数据库链接时，可以通过增加处理操作的可用线程数来改善性能。在本机 CPU 等待远程服务器的响应时，它可处理其它操作，而非一直处于空闲状态。

为更改处理操作所用的线程数，请更改 `cn=config` 条目中的 `nsslapd-threadnumber` 全局配置属性。默认的线程数为 20。例如，可以将线程数增加到 50，从而提高性能。更改线程数后，请重新启动服务器以使更改生效。

高级功能：配置级联链接

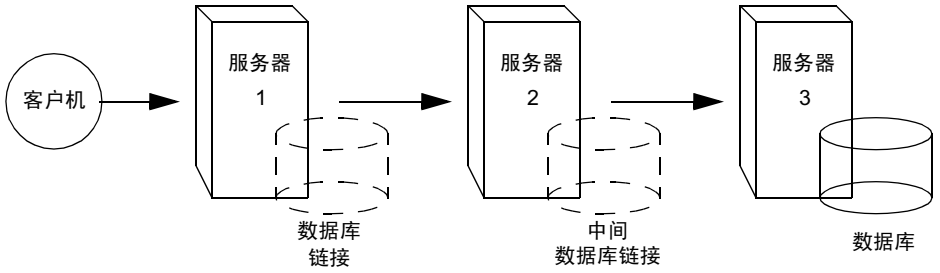
可以对数据库链接进行配置，使之指向另一个数据库链接，从而创建级联链接操作。如果要求多个路由段以访问目录树中的所有数据，即可随时建立级联链接。

本部分包含下列主题：

级联链接概述

如果目录要求多个路由段以处理客户机应用程序的请求，就会产生级联链接。

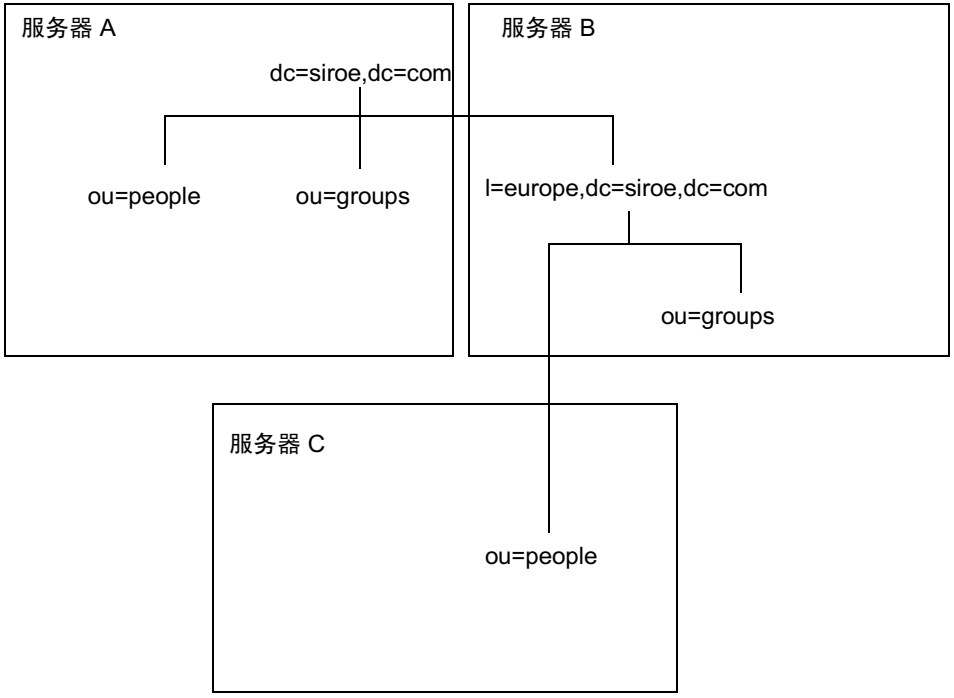
例如，考虑下列环境：



假定客户机应用程序向服务器 1 发送修改请求。服务器 1 中包含将操作转发给服务器 2 的数据库链接，而服务器 2 则包含了另一个数据库链接。服务器 2 上的数据库链接将把操作转发到服务器 3，而服务器 3 则包含客户机所要修改的数据库数据。如果想访问客户机所要修改的数据块，则需要两个路由段。

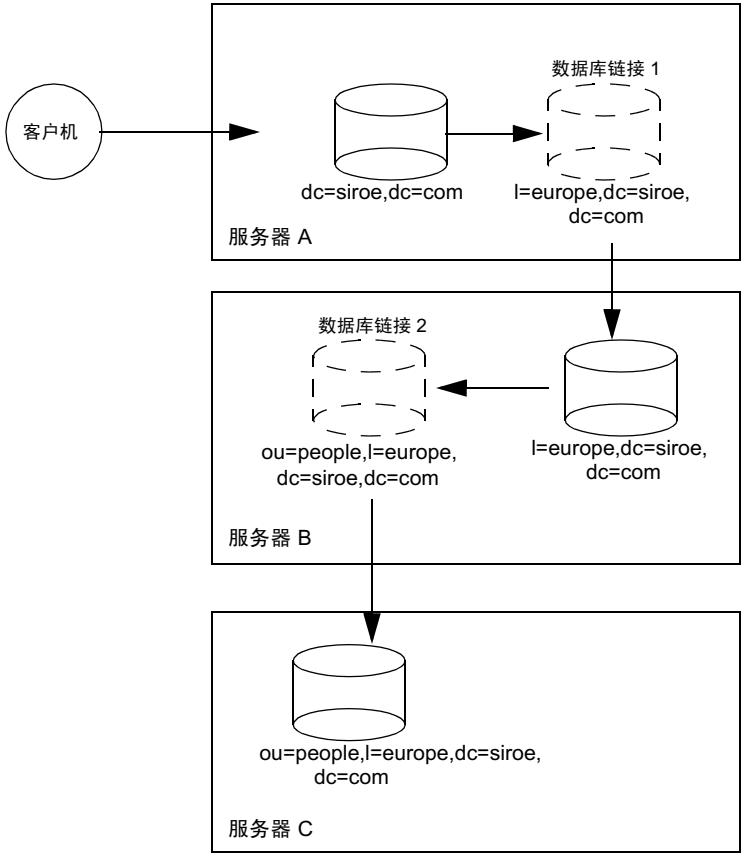
正常操作请求期间，客户机将绑定到服务器，然后评估应用于该客户机的任何 ACI。使用级联链接，客户机绑定请求可在服务器 1 上进行评估，但应用于客户机的 ACI 将仅在该请求已被链接到目标服务器 （上例中为服务器 2）后才进行评估。

请考虑下例环境。在服务器 A 上，假定目录树按如下所示进行拆分：



根后缀 `dc=siroe,dc=com`、 `ou=people` 和 `ou=groups` 子后缀储存在服务器 A 上。`l=europe,dc=siroe,dc=com` 和 `ou=groups` 后缀存储在服务器 B 上， `l=europe, dc=siroe,dc=com` 后缀的 `ou=people` 分支则存储在服务器 C 上。

利用在服务器 A、 B 和 C 上配置的级联，目录将以下列方式路由以 `ou=people, l=europe,dc=siroe,dc=com` 条目为目标的客户机请求：



首先，客户机绑定到服务器 A 并通过数据库链接 1 链接到服务器 B。然后服务器 B 通过数据库链接 2 链接到服务器 C 上的目标数据库，以访问 `ou=people,l=europe,dc=siroe,dc=com` 分支中的数据。因为目录要求至少两个路由段来处理客户机请求，因此我们将其视为级联链接。

使用控制台配置级联链接的默认值

要为目录服务器中的所有数据库链接设置级联链接默认值：

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧窗口中的 “数据”文件夹，然后单击 “数据库链接设置”。单击 “默认创建参数”选项卡。
3. 如果想在级联链接所涉及的中间数据库上进行本地 ACI 的评估，请选中 “检查本地 ACI”复选框。如果选中该复选框，则需要将相应的本地 ACI 添加到包含中间数据库链接的服务器数据库中。

   这属于高级功能。有关详细信息，请参阅第 114 页上的 “启用本地 ACI 评估”。
4. 在 “最大路由段”字段中，输入数据库链接可指向其它数据库的最多次数。

   默认情况下，最大值为 10 个路由段。在 10 个路由段后，服务器检测到循环并向客户机应用程序返回错误。
5. 单击 “保存”可保存更改结果。

注意 对数据库链接默认设置所做的更改是不可逆的。只有在保存完对默认设置所做的更改后，所创建的数据库链接才会反映这些更改。

使用控制台配置级联链接

要为特定的数据库链接集配置级联链接，请执行下列操作：

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧窗口中的 “数据”文件夹，然后查找要包含在级联链接中的数据库链接。单击数据库链接，然后单击右侧导航窗口中的 “限制和控制”选项卡。
3. 如果想在级联链接所涉及的中间数据库链接上启用本地 ACI 评估，请选中 “检查本地 ACI”复选框。如果选中该复选框，则可能需要将相应的本地 ACI 添加到数据库链接中。

   这属于高级功能。有关详细信息，请参阅第 114 页上的 “启用本地 ACI 评估”。
4. 在 “最大路由段”字段中，输入数据库链接可指向其它数据库的最多次数。

   默认情况下，最大值为 10 个路由段。在 10 个路由段后，服务器检测到循环并向客户机应用程序返回错误。
5. 单击 “保存”可保存更改结果。

从命令行配置级联链接

通过命令行配置数据库级联链接时涉及下列步骤：

- 将一个数据库链接指向包含中间数据库链接的服务器的 URL。
- 配置中间数据库链接 （示例中为服务器 2）以传送代理验证控件。
- 在所有中间数据库链接上创建代理管理级用户 ACI。为此，需要在包含中间数据库链接的每台服务器上创建数据库。
- 在所有中间数据库链接上进行本地 ACI 评估。
- 在所有中间数据库链接和最终的目标数据库上创建客户机 ACI。

该部分包含下列主题：

- 第 113 页上的 “指向另一个数据库链接”
- 第 113 页上的 “传送代理验证控件”
- 第 114 页上的 “创建代理管理级用户 ACI”
- 第 114 页上的 “启用本地 ACI 评估”
- 第 115 页上的 “创建客户机 ACI”
- 第 115 页上的 “检测循环”

指向另一个数据库链接

要创建级联链接，其中一个数据库链接的 `nsFarmServerURL` 属性必须有包含另一个数据库链接的服务器的 URL。例如，假设名为 `siroe1.com` 的服务器上的数据库链接指向名为 `africa.siroe.com` 的服务器上的数据库链接。服务器 1 上数据库链接中的 `cn=database_link_name,cn=chaining database,cn=plugins,cn=config` 条目将包含下列内容：

```
nsFarmServerURL: ldap://africa.siroe.com:389
```

传送代理验证控件

默认情况下，数据库链接不传送代理验证控件。不过，当一个数据库链接连接另一个数据库链接时，该控件将用于传送最终的目标服务器所需的信息。中间数据库链接需要传送此控件。要配置数据库链接以传送代理验证控件，请将下列内容添加到中间数据库链接的 `cn=config,cn=chaining database,cn=plugins,cn=config` 条目中：

```
nsTransmittedControls: 2.16.840.1.113730.3.4.12
```

OID 值代表代理验证控件。有关链接 LDAP 控件的详细信息，请参阅第 91 页上的 “链接 LDAP 控件”。

创建代理管理级用户 ACI

在包含中间数据库链接的服务器上需要创建 ACI，其中的数据库链接在将请求转换到另一个服务器之前会检查第一个数据库链接的权限。例如，如果服务器 2 不检查服务器 1 的凭证，则任何人都可以匿名的方式进行绑定并通过代理验证控件的验证，从而使自己获得更多并不适宜的管理特权。

为防止这种安全漏洞，需要在包含中间数据库链接的服务器上创建 ACI。要创建 ACI，您需要执行下列操作：

1. 如果现在没有数据库，则在包含中间数据库链接的服务器上创建数据库。该数据库将包含管理级用户条目和 ACI。有关创建数据库的详细信息，请参阅第 81 页上的 “创建数据库”。
2. 创建对应于数据库中管理级用户的条目。
3. 为那些目标后缀正确的管理级用户创建 ACI。该操作可确保管理员仅能访问数据库链接的后缀。将下列 ACI 添加到管理级用户的条目中：

   ```
   aci: (targetattr = "*")(version 3.0; acl "Proxied authorization
    for  database links"; allow (proxy) userdn = "ldap:///cn=proxy
    admin,cn=config";)
   ```

   该 ACI 与在配置简单链接时在远程服务器上创建的 ACI 类似。

警告 启用链接时，请仔细检查访问控制，以免赋予对目录受限区域的访问权。例如，如果在分支上创建默认代理 ACI，则通过该数据库链接进行连接的用户将能看到该分支下的所有条目。有时您并不希望用户看见所有子树。为避免出现安全漏洞，可以创建附加的 ACI，用于限制对子树的访问。

启用本地 ACI 评估

为确认使用了代理管理 ACI，需要在链接所涉及的所有中间数据库链接上启用本地 ACI 评估。为此，请将下列属性添加到每个中间数据库链接的 `cn=database_link_name,cn=chaining database,cn=plugins,cn=config` 条目中：

```
nsCheckLocalACI: on
```

如果在 `cn=default instance config,cn=chaining database,cn=plugins,cn=config` 条目中将该属性设置为 “开”，则意味着所有新数据库链接实例都会在其 `cn=database_link_name,cn=chaining database,cn=plugins,cn=config` 条目中将 `nsCheckLocalACI` 属性设置为 “开”。

创建客户机 ACI

因为您已启用本地 ACI 评估，所以需要在所有中间数据库链接和最终的目标数据库上创建相应的客户机应用程序 ACI。

要在中间数据库链接上创建相应的 ACI，您需要首先创建一个包含后缀的数据库，其中的后缀表示最终目标后缀的根后缀。

例如，如果是在链接对远程服务器上的 `c=africa,ou=people,dc=siroe,dc=com` 后缀发出的客户机请求，则所有中间数据库链接都需要包含与 `dc=siroe,dc=com` 后缀相关联的数据库。

然后，您需要将所有客户机 ACI 添加到上一级后缀条目中。例如，可以添加下列内容

```
aci: (targetattr = "*")(version 3.0; acl "Client authentication for
 database link users"; allow (all) userdn = "ldap:///uid=*
 ,cn=config";)
```

如果客户机应用程序在服务器 1 的 `cn=config` 条目中具有 `uid`，则此 ACI 将允许该客户机应用程序对服务器 3 上 `ou=people,dc=siroe,dc=com` 后缀下的数据执行任何操作。

检测循环

随 Directory Server 提供的 LDAP 控件可防止出现循环问题。首次尝试链接时，服务器将把该控件设置为所允许的最大路由段数 （或称最大链接数）。后续的每台服务器将使计数减 1。如果服务器接收到的计数为 0 时，就会断定已检测到循环，然后通知客户机应用程序。

允许的路由段数是使用 `nsHopLimit` 属性定义的。如果未指定，则默认值为 10。

要使用该控件，请将下列 OID 添加到 `cn=config,cn=chaining database, cn=plugins,cn=config` 条目的 `nsTransmittedControl` 属性中。

```
nsTransmittedControl: 1.3.6.1.4.1.1466.29539.12
```

如果该控件并非在每个数据库链接的配置文件中都存在，则不会进行循环检测。

级联链接配置属性概要

下表介绍用于配置级联链接内中间数据库链接的属性：

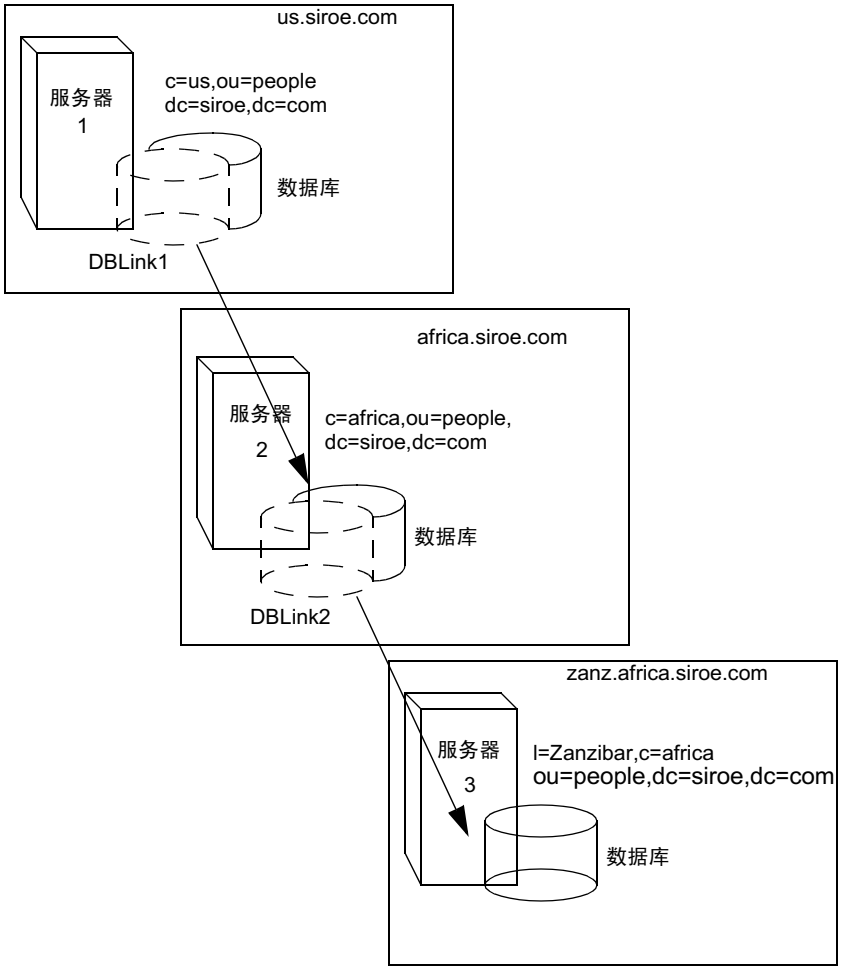
表 3-7 级联链接配置属性

属性	说明
nsFarmServerURL	包含级联链接中下一个数据库链接的服务器的 URL。
nsTransmittedControls	为级联链接所涉及的数据库链接输入下列 OID： `nsTransmittedControls: 2.16.840.1.113730.3.4.12` nsTransmittedControls: 1.3.6.1.4.1.1466.29539.12 第一个 OID 对应于代理验证控件。第二个 OID 对应于循环检测控件。
aci	该属性必须包含下列 ACI： `aci: (targetattr = "*")(version 3.0; acl "Proxied authorization for database links"; allow (proxy) userdn = "ldap:///cn=proxy admin,cn=config";)`
nsCheckLocalACI	要在链接所涉及的所有数据库链接上启用本地 ACI 评估，请按如下所示打开本地 ACI 评估： `nsCheckLocalACI: on`

级联链接配置示例

要创建包含三台服务器的级联链接，必须在这三台服务器上配置级联组件，如下图所示。本部分介绍创建包含三台服务器的级联链接的配置步骤，并分为以下几个部分分别说明：

- 第 118 页上的 “配置服务器 1”
- 第 119 页上的 “配置服务器 2”
- 第 121 页上的 “配置服务器 3”


us.siroe.com
服务器
1
c=us,ou=people
dc=siroe,dc=com
数据库
DBLink1
africa.siroe.com
服务器
2
c=africa,ou=people,
dc=siroe,dc=com
数据库
DBLink2
zanz.africa.siroe.com
服务器
3
l=Zanzibar,c=africa
ou=people,dc=siroe,dc=com
数据库

配置服务器 1

首先，使用 ldapmodify 命令行实用程序将数据库链接添加到服务器 1 中。如下所示，运行该实用程序：

```
ldapmodify -a -h host -p port -D "cn=Directory Manager" -w password
```

然后按如下所示，为服务器 1 上的数据库链接 DBLink1 指定配置信息：

```
dn: cn=DBLink1,cn=chaining database,cn=plugins,cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsBackendInstance
nsslapd-suffix: l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com
nsfarmserverurl: ldap://africa.siroe.com:389/
nsmultiplexorbinddn: cn=server1 proxy admin,cn=config
nsmultiplexorcredentials: secret
cn: DBLink1
nsCheckLocalACI:off

cn="l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com",cn=mapping
tree,cn=config
objectclass=nsMappingTree
nsslapd-state=backend
nsslapd-backend=DBLink1
nsslapd-suffix: l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com
cn: l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com
```

第一部分将创建与 DBLink1 关联的条目。第二部分则创建新后缀，从而使服务器可将向数据库链接发出的请求转发给正确的服务器。无需配置 nsCheckLocalACI 属性以检查本地的 ACI，因为这只在服务器 2 的数据库链接 （即 DBLink2）中才需要。

由于要实现循环检测，因此需要在 nsTransmittedControl 属性中指定循环检测控件的 OID，该属性存储于服务器 1 的 cn=config,cn=chaining database, cn=plugins,cn=config 条目中。按如下方式指定 OID：

```
dn: cn=config,cn=chaining database,cn=plugins,cn=config
changeType: modify
add: nsTransmittedControl
nsTransmittedControl: 1.3.6.1.4.1.1466.29539.12
```

因为 nsTransmittedControl 属性一般通过循环检测控件 OID 1.3.6.1.4.1.1466.29539.12 值被默认配置，所以不管该配置是否已经存在，事先对其进行检查是明智的做法。如果已经存在，则无需执行这一配置步骤。

配置服务器 2

接下来，在服务器 2 上创建代理管理级用户。该管理级用户将用于支持服务器 1 绑定到服务器 2 并为其提供验证。请记住，为服务器 1 选择一个专用的代理管理级用户名称是很有用的，因为只有代理管理用户才允许服务器 *1* 绑定到服务器 2。如下所示，创建代理管理级用户：

```
dn: cn=server1 proxy admin,cn=config
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: server1 proxy admin
sn: server1 proxy admin
userPassword: secret
description: Entry for use by database links
```

警告 不要将目录管理员或管理员 ID 用户当作远程服务器上的代理管理级用户使用。这会造成安全漏洞。

接着，在服务器 2 上配置数据库链接 DBLink2。利用 `ldapmodify`，按如下所示指定 DBLink2 的配置信息：

```
dn: cn=DBLink2,cn=chaining database,cn=plugins,cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsBackendInstance
nsslapd-suffix: l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com
nsfarmserverurl: ldap://zanz.africa.siroe.com:389/
nsmultiplexorbinddn: cn=server2 proxy admin,cn=config
nsmultiplexorcredentials: secret
cn: DBLink2
nsCheckLocalACI: on

dn: cn="l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com",cn=mapping
tree,cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsMappingTree
nsslapd-state: backend
nsslapd-backend: DBLink2
nsslapd-parent-suffix:"c=africa,ou=people,dc=siroe,dc=com"
cn: l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com
```

由于数据库链接 DBLink2 是级联链接配置中的中间数据库链接，因此需要将 `nsCheckLocalACI` 设置为 on，以便让服务器检查它是否应允许客户机和代理管理级用户访问数据库链接。

服务器 2 上的数据库链接必须配置为可传送代理验证控件和循环检测控件。为使用代理验证控件和循环检测控件，需要指定它们对应的 OID。将以下信息添加到服务器 2 上的 cn=config,cn=chaining database,cn=plugins,cn=config 条目中：

```
dn: cn=config,cn=chaining database,cn=plugins,cn=config
changeType: modify
add: nsTransmittedControl
nsTransmittedControl: 2.16.840.1.113730.3.4.12
nsTransmittedControl: 1.3.6.1.4.1.1466.29539.12
```

其中，nsTransmittedControl: 2.16.840.1.113730.3.4.12 是代理验证控件的 OID，而 nsTransmittedControl: 1.3.6.1.4.1.1466.29539.12 则是循环检测控件的 OID。

同样，请记住要事先检查循环检测控件是否已经配置，然后相应地采用上述命令。

下一步是配置 ACI。在服务器 2 上，必须确保在 l=Zanzibar,c=africa, ou=people,dc=siroe,dc=com 后缀之上的现有后缀上有一个后缀，以便完成以下任务：

- 添加数据库链接后缀
- 添加本地代理验证 ACI，用于支持通过在服务器 2 上创建的管理级用户与服务器 1 的连接
- 添加本地客户机 ACI，它可以保证客户机操作在服务器 2 上成功进行，以便可以传递到服务器 3。因为已经为 DBLink2 数据库链接打开了本地 ACI 检查，所以这个本地 ACI 是必需的。

这两个 ACI 放置在包含 c=africa,ou=people,dc=siroe,dc=com 后缀的数据库中。

注意 为创建这些 ACI，假定已经存在与 c=africa,ou=people, dc=siroe,dc=com 后缀相对应的数据库以容纳该条目。该数据库需要与每个数据库链接的 nsslapd-suffix 属性中所指定后缀的上级后缀相关联。也就是说，最终目标服务器上的后缀应该是中间服务器上所指定后缀的子后缀。

将本地代理验证 ACI 添加到 c=africa,ou=people,dc=siroe,dc=coml 条目中：

```
aci:(targetattr="*")(target="l=Zanzibar,c=africa,ou=people,
dc=siroe,dc=com")(version 3.0; acl "Proxied authorization for
 database links"; allow (proxy) userdn = "ldap:///cn=server1 proxy
 admin,cn=config";)
```

然后添加本地客户机 ACI，在 ACI 检查打开的情况下，它可以保证客户机操作在服务器 2 上成功进行。该 ACI 与将在目标服务器上创建的 ACI 一样，以便提供对 l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com 分支的访问权限。您可以决定让 c=us,ou=people,dc=siroe,dc=com 的所有用户对服务器 3 上 l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com 中的条目都具有更新访问权限。下列 ACI 是为实现上述目标需要在服务器 2 的 c=africa,ou=people, dc=siroe,dc=com 后缀上创建的 ACI：

```
aci:(targetattr="*")(target="l=Zanzibar,c=africa,ou=people,
dc=siroe,dc=com")(version 3.0; acl "Client authorization for
 database links"; allow (all) userdn =
"ldap:///uid=*,c=us,ou=people,dc=siroe,dc=com";)
```

如果客户机在服务器 1 的 c=us,ou=people,dc=siroe,dc=com 条目中具有 uid，则该 ACI 将允许此客户机对服务器 3 的 l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com 后缀树中执行任何操作。如果服务器 2 中另一个后缀下的用户需要对服务器 3 具有额外的权限，则需要在服务器 2 上添加额外的客户机 ACI。

配置服务器 3

在该级联链接示例中，最后一个配置步骤是配置服务器 3。首先，在服务器 3 上为服务器 2 创建管理级用户，以供代理验证使用：

```
dn: cn=server2 proxy admin,cn=config
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: server2 proxy admin
sn: server2 proxy admin
userPassword: secret
description: Entry for use by database links
```

然后，需要按照与服务器 2 相同的操作步骤，将同样的代理验证 ACI 添加到服务器 3 中。将下列代理验证 ACI 添加到 l=Zanzibar,ou=people,dc=siroe,dc=com 条目中：

```
aci: (targetattr = "*")(version 3.0; acl "Proxied authorization for
 database links"; allow (proxy) userdn = "ldap:///cn=server2 proxy
 admin,cn=config";)
```

该 ACI 仅授予服务器 2 对 l=Zanzibar,ou=people,dc=siroe,dc=com 子树中远程服务器 （即服务器 3）上的数据拥有代理管理级只读访问的权限。

随即，您需要在与原始客户机应用程序相对应的 l=Zanzibar,ou=people, dc=siroe,dc=com 子树上创建本地客户机 ACI。使用与为服务器 2 上客户机所创建的 ACI 相同的 ACI：

```
aci: (targetattr = "*")(target="l=Zanzibar,c=africa,ou=people,
 dc=siroe,dc=com")(version 3.0; acl "Client authentication for
 database link users"; allow (all) userdn =
 "ldap:///uid=*,c=us,ou=people,dc=siroe,dc=com";)
```

完成上述全部步骤后，级联链接配置就设置好了。该级联配置将允许用户绑定到服务器 1 并修改服务器 3 的 l=Zanzibar,c=africa,ou=people,dc=siroe,dc=com 分支中的信息。根据安全需要，可以决定是否提供更详细的访问控制。

使用引荐

使用引荐可以通知客户机应用程序在查看特定信息时所要连接的服务器。如果客户机应用程序所请求的目录项在本地服务器上不存在，或者数据库已进行脱机维护，就会出现该重定向现象。本部分包含有关引荐的下列信息：

- 设置默认引荐
- 创建智能引荐
- 创建后缀引荐

有关如何在目录中使用引荐的概念性信息，请参阅 *iPlanet Directory Server 部署指南*。

设置默认引荐

对于并不包含于目录所维护的任何后缀中的 DN 而言，默认引荐将被返回给提交该 DN 操作的客户机应用程序。下列过程介绍如何使用控制台和命令行实用程序设置目录的默认引荐。

使用控制台设置默认引荐

如下所示，为目录设置默认引荐：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧窗口的导航树中选择顶级条目。
3. 选择右侧窗口中的 “设置”选项卡。
4. 在 “引荐”文本框中输入 LDAP URL，然后单击 “确定”。

   例如：

   ```
   ldap://directory.siroe.com:389/dc=siroe,dc=com
   ```

   也可以输入多个引荐 URL，它们之间由空格分隔并位于引号中，如下所示：

   ```
   "ldap://d1.siroe.com:389/dc=siroe,dc=com" "ldap://d2.siroe.com/"
   ```

有关 LDAP URL 的详细信息，请参阅附录 C “LDAP URL”。

从命令行设置默认引荐

使用 `ldapmodify` 命令行实用程序可以将默认引荐添加到目录配置文件中的 `cn=config` 条目中。

例如，要从 siroe.com 目录服务器将新的默认引荐添加到名为 Zanzibar.com 的服务器上，则在 `cn=config` 条目中新增一行。如下所示，运行 `ldapmodify` 实用程序：

```
ldapmodify -a -h host -p port -D "cn=Directory Manager" -w password
```

`ldapmodify` 实用程序绑定到服务器上并准备更改配置文件中的条目。

接着，将默认引荐添加到 Zanzibar.com 服务器中：

```
dn: cn=config
changetype: modify
replace: nsslapd-referral
nsslapd-referral: ldap://zanzibar.com/
```

将默认引荐添加到目录的 `cn=config` 条目中后，该目录在响应客户机应用程序的请求时将返回默认引荐。此时无须重新启动服务器。

创建智能引荐

智能引荐允许将目录条目或目录树映射到特定的 LDAP URL 上。利用智能引荐，可以将客户机应用程序指向特定的服务器或特定服务器上的特定条目。

例如，假定客户机应用程序请求下列目录项：`uid=bjensen,ou=people,dc=siroe,dc=com`。那么，指向 `directory.europe.siroe.com` 服务器上 `cn=Babs Jensen,o=people,l=europe,dc=siroe,dc=com` 条目的智能引荐会返回至该客户机。

目录使用智能引荐的方式符合 RFC 2251 中 4.1.11 一节所指定的标准。有关详细信息，请参阅 RFC，网址是 `http://www.ietf.org/rfc/rfc2251.txt`。

下列过程介绍如何使用控制台和命令行实用程序创建智能引荐。

使用控制台创建智能引荐

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 浏览目录树，找到要添加引荐的条目。
3. 右键单击该条目，并从下拉菜单中选择 “设置引荐”。

   此时显示 “编辑引荐”对话框。如果这是为该条目创建的第一个引荐，则引荐列表为空白。
4. 选中 “启用引荐”复选框。
5. 在 “输入新引荐”字段中输入一个 LDAP URL，或者单击 “构造”显示一个对话框以帮助您创建正确的 URL。

   URL 的组成元素包括保存引荐条目的目录服务器的主机名和 LDAP 端口号，以及引荐条目的 DN （目标 DN）。该 DN 可以是后缀、子树或叶条目。
6. 在 “编辑引荐”对话框中，单击 “添加”将新的 LDAP URL 添加到引荐列表中。
7. 仍然在 “编辑引荐”对话框中，单击 “验证”显示一个对话框，提示当前服务器需要使用凭证来进行绑定，从而遵循到远程服务器的引荐。
8. 输入已授权访问该引荐 DN 的用户的 DN 和口令，然后单击 “确定”退出该对话框。
9. 在 “编辑引荐”对话框中，单击 “确定”退出此窗口。

   在导航树中，应该可以在为其创建了引荐的原始条目位置处看到该引荐子树或条目。如果发现原始条目仍存在，则会在该条目旁边有一个警告图标。该图标表示还没有执行步骤 7，或者提供的绑定 DN 和口令没有获得访问引荐 DN 的权限。

从命令行创建智能引荐

使用 `ldapmodify` 命令行实用程序可以从命令行创建智能引荐。

要创建智能引荐，请创建相关目录条目并添加 `Referral` 对象类。该对象类允许使用单个属性 `ref`。 `ref` 属性中应包含 LDAP URL。

例如，添加下列内容可返回现有条目 `uid=bjensen` 的智能引荐：

```
dn: uid=bjensen,ou=people,dc=siroe,dc=com
objectclass: referral
ref: ldap://directory.europe.siroe.com/cn=babs%20jensen,ou=people,
 l=europe,dc=siroe,dc=com
```

注意 服务器将忽略 LDAP URL 中空格后面的任何信息。因此，在要用作引荐的所有 LDAP URL 中，必须使用 `%20` 来替代空格。

要将带有引荐的条目 `uid=ssarette,ou=people,dc=siroe,dc=com` 添加到 `directory.europe.siroe.com` 中，请在导入前在 LDIF 文件中包含下列内容：

```
dn: uid=ssarette, ou=people, dc=siroe,dc=com
objectclass: top
objectclass: person
objectclass: organizationalperson
objectclass: inetOrgPerson
objectclass: referral
cn: somi sarette
sn: sarette
uid: ssarette
ref: ldap://directory.europe.siroe.com/cn=somi%20sarette,ou=people,
 l=europe,dc=siroe,dc=com
```

有关智能引荐的详细信息，请参阅 *iPlanet Directory Server 部署指南*。有关 `ldapmodify` 实用程序的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

创建后缀引荐

下列过程介绍如何在后缀 (suffix) 中创建引荐。这意味着后缀将使用引荐 （而非数据库或数据库链接）来处理操作。有关引荐的详细信息，请参阅 *iPlanet Directory Server 部署指南*。

警告 将后缀配置为返回引荐后，将忽略与该后缀相关联的数据库中所含的 ACI。

使用控制台创建后缀引荐

要使用控制台创建后缀引荐：

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧窗口的 “数据”下，单击要添加引荐的后缀。
3. 在 “后缀设置”选项卡上，选择以下一个单选按钮：

   使用引荐。这意味着当该后缀收到客户机应用程序的任何请求时，都将返回引荐。

   使用有关更新的引荐。这意味着当该后缀收到客户机应用程序的更新请求时，将返回引荐。该选项将用于将客户机应用程序发出的更新和写入请求重定向到某个只读数据库。

4. 单击 “引荐”选项卡。在 “输入新引荐”字段中输入 LDAP URL，或者单击 “构造”以在指导下完成 LDAP URL 的创建。

   有关 LDAP URL 结构的详细信息，请参阅附录 C “LDAP URL”。

5. 单击 “添加”可将引荐添加到列表中。

   您可以输入多个引荐。为响应客户机应用程序的请求，目录将返回整个引荐列表。

6. 单击 “保存”。

从命令行创建后缀引荐

使用 ldapmodify 命令行实用程序可以将后缀引荐添加到目录配置文件的条目中。后缀引荐信息将被添加到 cn=mapping tree,cn=config 分支下的根后缀和子后缀条目中。

例如，要将新后缀引荐添加到 ou=people,dc=siroe,dc=com 根后缀中，需要执行 ldapmodify。按如下所示运行 ldapmodify：

```
ldapmodify -a -h host -p port -D "cn=Directory Manager" -w password
```

ldapmodify 实用程序将绑定到服务器上并准备向配置文件中添加信息。

接着，按如下所示将后缀引荐添加到 ou=people,dc=siroe,dc=com 根后缀中：

```
dn: cn="ou=people,dc=siroe,dc=com",cn=mapping tree,cn=config
objectclass: extensibleObject
objectclasss: nsmappingtree
nsslapd-state: referral
nsslapd-referral: ldap://zanzibar.com/
```

将 nsslapd-state 属性设置为 referral 意味着将为对该后缀发起的请求返回引荐。 nsslapd-referral 属性包含由后缀返回的引荐的 LDAP URL。本例中为指向 Zanzibar.com 服务器的引荐。

也可将 nsslapd-state 属性设置为 referral on update。这意味着该数据库可用于除更新请求以外的所有操作。当客户机应用程序对设置为 referral on update 的后缀发出更新请求时，客户机会收到引荐。

有关后缀配置属性的详细信息，请参阅第 76 页上的 “后缀属性”。

第 4 章

填充目录数据库

数据库中包含受目录服务器管理的目录数据。本章介绍下列填充目录数据库的操作：

- 启用和禁用只读模式
- 导入数据
- 导出数据
- 备份和恢复数据

启用和禁用只读模式

在 iPlanet Directory Server 上执行某些导出或备份操作之前，可以启用任何数据库上的只读模式，从而确保在给定时刻充分了解这些数据库的状态。

执行导出或备份操作之前，iPlanet Directory Server Console 及命令行实用程序不会自动将目录置于只读模式，因为这样会使目录无法进行更新。但如果为多原版配置，就不会出现上述问题。

启用只读模式

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后展开导航树中的 “数据”文件夹。
2. 选择要置于只读模式的数据库，然后单击右侧窗口中的 “数据库设置”选项卡。

3. 选中 “数据库为只读”复选框。
4. 单击 “保存”。

   所做更改将立即生效。

执行导入或恢复操作之前，应确保受该操作影响的数据库*未*处于只读模式下。如果确实为只读模式，请利用下列操作使之处于可更新状态下：

禁用只读模式

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后展开 “数据”目录树。
2. 选择要让其可更新的数据库，然后单击右侧窗口中的 “数据库设置”选项卡。
3. 清除 “数据库为只读”复选框。
4. 单击 “保存”。

   所做更改将立即生效。

导入数据

iPlanet Directory Server 提供了三种数据导入方法：

- 从 iPlanet Directory Server Console 导入。

  可以使用 iPlanet Directory Server Console 将数据追加到所有数据库中，包括数据库链接。

- 初始化数据库。

  可以使用 Directory Server Console 将数据导入一个数据库中。该方法将覆盖数据库中所含的任何数据。

- 从命令行导入数据。

  可以使用命令行实用程序导入数据。

注意 所导入的所有 LDIF 文件都必须使用 UTF-8 字符集编码方式。

下表说明导入和初始化数据库之间的区别：

表 4-1 导入数据与初始化数据库之间的比较

比较范围	导入数据	初始化数据库
覆盖数据库	否	是
LDAP 操作	添加、修改、删除	只添加
性能	更为耗时	快速
分区特殊要求	适用于所有分区	仅本地分区
对服务器故障的响应	尽力 （出现故障前所做的所有更改都可保留下来）	极少 （所有更改在出现故障后都将丢失）
LDIF 文件位置	从本地到控制台	从本地到控制台或从本地到服务器
导入配置信息 (`cn=config`)	是	否

下列部分介绍导入数据：

- 第 131 页上的 “从控制台执行导入”
- 第 133 页上的 “从控制台初始化数据库”
- 第 134 页上的 “从命令行导入”

警告 所有导入的 LDIF 文件也都必须包含根后缀。

从控制台执行导入

从 iPlanet Directory Server Console 执行导入操作时，系统可执行 `ldapmodify` 操作，从而进行数据的追加及条目的修改和删除。操作将面向所有受 iPlanet Directory Server 管理的数据库，同时还包括 iPlanet Directory Server 对其配置有数据库链接的远程数据库。

为执行导入，必须以目录管理员 (Directory Manager) 的身份进行登录。

要从 iPlanet Directory Server Console 导入数据：

1. 在 iPlanet Directory Server Console 中，选择 “任务”选项卡。滚动到屏幕的底端，选择 “导入数据库”。

   也可以转到 “配置”选项卡并选择 “控制台”菜单中的 “导入”，同样可以进行导入。

   此时显示 “导入数据库”对话框。

2. 在 “LDIF 文件”字段中，输入所要导入的 LDIF 文件的完整路径，或者单击 “浏览”以选择所要导入的文件。

   如果是在目录的远程计算机上运行控制台，则字段名将显示为 “LDIF 文件（位于运行控制台的计算机上）”。这是提醒用户：所浏览的目录并非当前目录。相反，此时所浏览的是运行控制台的计算机的文件系统。

3. 在 “选项”框中，可选择下列一个或多个选项：

   只添加。除了默认的添加指令外， LDIF 文件可能还包含修改和删除指令。如果希望服务器忽略除添加操作以外的其他操作，请选中 “只添加”复选框。

   出错时继续。如果希望服务器即使在出现错误的情况下仍继续导入，请选中 “出错时继续”复选框。例如，如果导入的 LDIF 文件中除了新的条目外，还含有数据库中已有的某些条目，即可使用该选项。服务器会在拒绝文件中记录下已有的条目，同时添加所有新条目。

4. 在 “拒绝的文件”字段中，输入某个文件的完整路径，用于供服务器记录不能导入的条目；或者单击 “浏览”以选择容纳被拒绝条目的文件。

   例如，服务器无法导入数据库中已有的条目或无父对象的条目。控制台将把服务器发送来的错误信息写入拒绝文件中。

   如果将该字段空置，服务器将不会记录被拒绝的条目。

5. 单击 “确定”。

   服务器将执行导入并同时创建索引。

从控制台初始化数据库

您可以覆盖数据库中现有的数据。以下部分介绍利用控制台来初始化数据库。

为初始化数据库，则必须以目录管理员 (Directory Manager) 身份进行登录。原因是：除非您作为目录管理员绑定到目录上 （根 DN），否则将无法导入包含根条目的 LDIF 文件。只有目录管理员才有权访问根条目 （例如，根条目可能为 `dc=siroe,dc=com`）。

警告 从 LDIF 文件初始化数据库时，除非是要恢复数据，否则应注意不要覆盖 `o=NetscapeRoot` 后缀。如果覆盖该后缀，就会删除某些重要信息，并因此要求重新安装所有 iPlanet 服务器。

要使用 iPlanet Directory Server Console 初始化数据库：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧导航窗口中的 “数据”目录树。展开所要初始化的数据库的后缀，然后单击该数据库。
3. 右键单击数据库，然后选择 “初始化数据库”。

   也可以从 “对象”菜单中选择 “初始化”数据库。
4. 在 “LDIF 文件”字段中，输入所要导入的 LDIF 文件的完整路径，或者单击 “浏览”并在计算机上进行查找。
5. 如果是在所导入文件的本机上运行控制台，请跳到步骤 6。对于包含 LDIF 文件的服务器而言，如果是从其远程计算机运行控制台，请选择下列选项之一：

   从本地机器。表明 LDIF 文件位于本地计算机上。

   从服务器。表明 LDIF 文件位于远程服务器上。默认情况下，控制台将查找下列目录中的文件：

   `/var/ds5/slapd-`*`serverID`*`/ldif`
6. 单击 “确定”。

从命令行导入

有三种通过命令行导入数据的方法：

- 使用 `/usr/sbin/directoryserver ldif2db`

  该导入方法将覆盖数据库的内容，同时要求停止服务器。

- 使用 `/usr/sbin/directoryserver ldif2db-task`

  该导入方法将覆盖数据库的内容，而服务器仍将处于运行状态。

- 使用 `/usr/sbin/directoryserver ldif2ldap`

  该方法将通过 LDAP 向 LDIF 文件中追加数据。使用该方法可以将数据追加到所有数据库中。

使用 ldif2db 命令进行导入

`/usr/sbin/directoryserver ldif2db` 命令将覆盖所指定的数据库中的数据。该命令要求在导入前关闭服务器。

默认情况下，该命令首先保存任何现有的 `o=NetscapeRoot` 配置信息，然后将其与所导入文件中的 `o=NetscapeRoot` 配置信息进行合并。

警告 该命令将覆盖数据库中的数据。

要在停止服务器的情况下导入 LDIF：

1. 从命令行上作为 root 用户通过以下命令停止服务器：

   ```
   # /usr/sbin/directoryserver stop
   ```

2. 使用 `ldif2db` 子命令：

   ```
   # /usr/sbin/directoryserver ldif2db
   ```

下面的示例使用命令将两个 LDIF 文件导入一个单一的数据库中。

警告 如果在 `-n` 选项中指定的数据库并不对应于 LDIF 文件包含的后缀，则该数据库中的所有数据都将被删除，而导入操作将失败。请务必不要将数据库名称拼写错误。

```
#!/bin/sh
/usr/sbin/directoryserver ldif2db -n Database1 \
 -i /usr/iplanet/servers/slapd-siroe/ldif/demo.ldif \
 -i /usr/iplanet/servers/slapd-siroe/ldif/demo2.ldif
```

表 4-2 示例中所用的 ldif2db 选项说明

选项	说明
-n	指定用于接收导入数据的数据库名称。
-i	指定所要导入的 LDIF 文件的完整路径名。该选项为必需项。可以使用多个 -i 变量一次导入多个 LDIF 文件。导入多个文件时，服务器将按命令行中指定的顺序导入 LDIF 文件。

使用 ldif2db-task 命令进行导入

同上， /usr/sbin/directoryserver ldif2db-task 将覆盖所指定的数据库中的数据。执行导入时，该命令要求服务器处于运行状态。

警告 该命令将覆盖数据库中的数据。

以下示例导入一个 LDIF 文件。运行该脚本无需 root 权限，但是必须以目录管理员进行身份验证。

```
#!/bin/sh
/usr/sbin/directoryserver ldif2db-task \
  -D "cn=Directory Manager" -w password -n Database1 \
  -i /usr/iplanet/servers/slapd-siroe/ldif/demo.ldif
```

表 4-3 示例中所用的 ldif2db-task 选项说明

选项	说明
-D	指定目录管理员的 DN。
-w	指定目录管理员的口令。
-n	指定用于接收导入数据的数据库名称。
-i	指定所要导入的 LDIF 文件的完整路径名。该选项为必需项。可以使用多个 -i 变量一次导入多个 LDIF 文件。导入多个文件时，服务器将按命令行中指定的顺序导入 LDIF 文件。

使用 ldif2ldap 命令进行导入

`/usr/sbin/directoryserver ldif2ldap` 命令通过 LDAP 向 LDIF 文件中追加数据。使用该命令可同时向所有目录数据库中导入数据。使用该命令导入时，服务器必须处于运行状态。

以下示例执行导入。运行该命令无需 root 权限，但必须在命令行中提供目录管理员凭证。最后一个参数是要导入的 LDIF 文件的名称。

```
#!/bin/sh
/usr/sbin/directoryserver ldif2ldap "cn=Directory Manager" password \
  /usr/iplanet/servers/slapd-siroe/ldif/demo.ldif
```

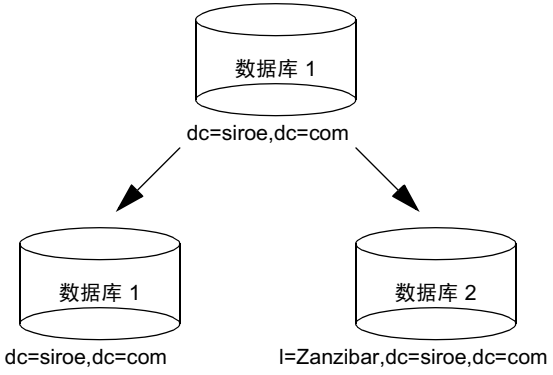
导出数据

使用 LDAP 数据交换格式 (LDIF) 可以导出数据库中的数据库条目。 LDIF 是一种 RFC 2849 “LDAP 数据交换格式 (LDIF) — 技术规格”中所介绍的标准格式。

导出数据功能可用于：

- 备份数据库中的数据
- 将数据复制到另一个目录服务器中
- 将数据导入另一个应用程序
- 在目录拓扑结构变化后重新填充数据库

例如，假设目录位于某个数据库中，而您决定将其内容拆分到两个数据库中，如下所示：



填充新数据库时，要求将数据库 1 的内容导出，然后将其导入新数据库 1 和 2 中。

可以使用 iPlanet Directory Server Console 或命令行实用程序来导出数据。下列部分将详细介绍这些方法：

- 第 137 页上的 “使用控制台将目录数据导出到 LDIF”
- 第 138 页上的 “使用控制台将单个数据库导出到 LDIF”
- 第 139 页上的 “从命令行导出到 LDIF”

导出操作并不导出配置信息 (`cn=config`)。

警告 请勿在导出操作期间停止服务器。

使用控制台将目录数据导出到 LDIF

您可以将部分或全部目录数据导入 LDIF 中，而这与最终导出文件的位置有关。如果 LDIF 文件位于服务器上，则可以仅将本地数据库中所含的数据导出到服务器上。如果 LDIF 文件位于服务器的远程位置，则可以导出数据库全部内容及数据库链接。

要在服务器运行状态下从 iPlanet Directory Server Console 将目录数据导入 LDIF：

1. 在 iPlanet Directory Server Console 中，选择 “任务”选项卡。滚动到屏幕的底端，然后单击 “导出数据库”。

   要导出数据库全部内容，也可选择 “配置”选项卡，然后选择 “控制台”菜单中的 “导出”。

   此时显示 “导出数据库”对话框。

2. 在 “LDIF 文件”字段中输入 LDIF 文件的完整路径和文件名，或者单击 “浏览”以查找该文件。

   如果是在远程服务器上运行控制台，则不启用 “浏览”按钮。如果未启用 “浏览”按钮，文件将默认存储到以下目录中：

   `/var/ds5/slapd-`*`serverID`*`/ldif`

3. 如果是在服务器的远程计算机上运行控制台，则在 LDIF 文件字段的下面将出现两个单选按钮。选择 “对于本地机器”则指示导出到控制台所在计算机的 LDIF 文件中。选择 “对于服务器”则指示导出到服务器的 LDIF 文件中。

4. 如果想导出整个目录，请选择 “整个数据库”单选按钮。

   如果只想导出数据库中所含后缀的单个子树，请选择 “子树”单选按钮，然后在 “子树”文本框中输入后缀的名称。该选项可用于导出包含在多个数据库中的某个子树。

   也可单击 “浏览”以选择后缀或子树。

5. 单击 “确定”以导出文件。

使用控制台将单个数据库导出到 LDIF

要在服务器运行状态下从 iPlanet Directory Server Console 将一个数据库导入 LDIF：

1. 在 Directory Server Console 上，选择 “配置”选项卡。
2. 展开左侧导航窗口中的 “数据”目录树。展开所要导出的数据库所维护的后缀。选择到要导出的后缀下的数据库。
3. 右键单击该数据库，然后选择 “导出数据库”。

   也可从 “对象”菜单中选择 “导出数据库”。

   此时显示 “导出分区”对话框。
4. 在 “LDIF 文件”字段中，输入 LDIF 文件的完整路径，或者单击 “浏览”以在计算机上查找该文件。

   如果未启用 “浏览”按钮，文件将默认存储到以下目录中：

   `/var/ds5/slapd-`*`serverID`*`/ldif`
5. 单击 “确定”以导出文件。

从命令行导出到 LDIF

使用 `/usr/sbin/directoryserver db2ldif` 命令可以将数据库导出到 LDIF。该命令可以将所有数据库内容或其中的一部分导出到一个 LDIF 文件，而不管服务器是否处于运行状态。

要将数据库内容导出到 LDIF 文件，请使用以下命令：

```
# /usr/sbin/directoryserver db2ldif
```

下面的示例将两个后缀下的数据库导出到一个 LDIF 文件中：

```
/usr/sbin/directoryserver db2ldif -n database1 -a output.ldif \
        -s "dc=siroe,dc=com" -s "o=NetscapeRoot"
```

下表说明了示例中所用的选项：

表 4-4 示例中所用的 db2ldif 选项说明

选项	说明
`-n`	指定用于接收导入数据的数据库名称。
`-a`	定义供服务器保存所导出的 LDIF 的输出文件的名称。默认情况下，该文件储存在 `/var/ds5/slapd-`*serverID* 目录中。
`-s`	指定导出中所含的后缀。可以使用多个 `-s` 参数指定多个后缀。

备份和恢复数据

使用 iPlanet Directory Server Console 或命令行实用程序可以备份和恢复数据库。

下列部分介绍备份和恢复数据的过程：

警告 备份和恢复期间，请勿停止服务器。

备份所有数据库

下列过程介绍如何利用 iPlanet Directory Server Console 及命令行备份目录中所有的数据库内容。

注意 对于使用数据库链接功能所链接的远程服务器而言，使用该备份方法无法备份其数据库中所含的数据。

从服务器控制台备份所有数据库

从 iPlanet Directory Server Console 备份数据库时，服务器将把数据库的全部内容及相关索引文件都复制到备份位置。执行备份时，服务器可以处于运行状态下。

要从服务器控制台备份数据库：

1. 在 iPlanet Directory Server Console 上，选择 “任务”选项卡。
2. 单击 “备份目录服务器”。

   此时显示 “备份目录”对话框。

3. 在 “目录”文本框中，输入用于存储备份文件的完整路径。如果是在目录所在的计算机上运行控制台，则单击 “浏览”以查找某个本地目录。

   或者单击 “使用默认值”将备份文件储存在以下目录中：

   /var/ds5/slapd-*serverID*/bak/*YYYY_MM_DD_hh_mm_ss*

   其中， *serverID* 是目录服务器的名称。

4. 单击 “确定”以创建备份。

从命令行备份所有数据库

可以使用 /usr/sbin/directoryserver db2bak 命令从命令行备份数据库。无论服务器是否处于运行状态，该命令都可以工作。

使用该备份方法无法备份配置信息。有关备份配置信息的说明，请参阅第 142 页上的 “备份 dse.ldif 配置文件”。

要备份目录，请使用以下命令：

```
# /usr/sbin/directoryserver db2bak backupDir
```

backupDir 参数指定储存备份的目录。默认的备份目录名利用当前的日期生成：YYYY_MM_DD_hh_mm_ss。

下面的示例将所有数据库备份到指定的目录中：

```
# /usr/sbin/directoryserver db2bak /var/ds5/slapd-sv/bak/checkpoint
```

备份单个数据库

如果符合以下条件，可以使用本部分介绍的方法：

- 目录服务器已关闭。
- 要制作的备份将用于恢复同一服务器上的数据库。

注意 使用该备份方法无法备份远程服务器上的数据库 （由数据库链接所链接的数据库）所含的数据；使用备份数据也无法初始化客户或中枢副本。

要备份单个数据库：

1. 从命令行上作为 root 用户通过以下命令停止服务器：

   ```
   # /usr/sbin/directoryserver stop
   ```

2. 变换到包含所要备份的数据库的目录：

   ```
   # cd /var/ds5/slapd-serverID/db
   ```

3. 将目录中的所有文件都复制到所创建的备份目录中。请勿在 `slapd-serverID/bak/` 下创建目录，因为 iPlanet Directory Server Console 假定该目录所含的备份为全局备份。

备份 dse.ldif 配置文件

Directory Server 会自动备份 `dse.ldif` 配置文件。启动目录服务器时，它将自动在以下目录中以 `dse.ldif.startOK` 文件名创建 `dse.ldif` 文件的备份：

```
/var/ds5/slapd-serverID/config
```

对 `dse.ldif` 文件进行修改时，该文件将被首先备份到 `config` 目录下名为 `dse.ldif.bak` 的文件中，之后，服务器将写入对 `dse.ldif` 文件的修改内容。如果需要保存配置，请制作上述文件副本之一。

恢复所有数据库

下列过程介绍如何利用 iPlanet Directory Server Console 及命令行恢复目录中所有的数据库内容。

注意 恢复数据库时，服务器必须处于运行状态。但在恢复过程中，数据库将无法用于处理操作。

从控制台恢复所有数据库

如果数据库已损坏，则可使用 iPlanet Directory Server Console 从以前生成的备份中进行恢复。该过程包括：停止服务器，然后将数据库及相关的索引文件从备份位置复制到数据库目录中。

警告 恢复数据库时将覆盖任何现有的数据库文件。

要从以前创建的备份中恢复数据库：

1. 在 iPlanet Directory Server Console 上，选择 “任务”选项卡。
2. 单击 “恢复 Directory Server”。

   此时显示 “恢复目录”对话框。
3. 从 “可用备份”列表中选择备份，或者在 “目录”文本框中输入到有效备份的完整路径。

   “可用备份”列表中给出以下默认目录中的所有备份：

   `/var/ds5/slapd-`*`serverID`*`/bak`
4. 单击 “确定”以恢复数据库。

从命令行恢复数据库

在命令行中，可以使用下列命令来恢复数据库：

- 使用 `/usr/sbin/directoryserver bak2db` 命令。该命令要求关闭服务器。
- 使用 `/usr/sbin/directoryserver bak2db-task` 命令。使用该命令时，要求服务器处于运行状态。

使用 bak2db 命令

要在关闭服务器的情况下从命令行恢复目录：

1. 从命令行上作为 root 用户通过以下命令停止服务器：

   ```
   # /usr/sbin/directoryserver stop
   ```

2. 使用 bak2db 命令和备份目录的完整路径：

   ```
   # /usr/sbin/directoryserver bak2db backupDir
   ```

警告 恢复数据库时将覆盖任何现有的数据库文件。

下面的示例从默认的备份目录中恢复备份：

```
# /usr/sbin/directoryserver bak2db /var/ds5/slapd-sv/bak/2001_07_01_11_34_00
```

使用 bak2db-task 命令

要在服务器运行状态下从命令行恢复目录，请使用以下命令：

```
/usr/sbin/directoryserver bak2db-task
```

警告 恢复数据库时将覆盖任何现有的数据库文件。

以下示例导入一个 LDIF 文件。

```
#!/bin/sh
/usr/sbin/directoryserver bak2db-task -D "cn=Directory Manager" \
   -w password -a /usr/iplanet/servers/slapd-siroe/bak/checkpoint
```

表 4-5 示例中所用的 bak2db-task 选项说明

选项	说明
-D	指定目录管理员的 DN。
-w	指定目录管理员的口令。
-a	定义备份目录的完整路径。

恢复单个数据库

如果符合以下条件，可以使用本部分介绍的方法：

- 目录服务器已关闭。
- 要从以前创建的备份文件中恢复的数据库将用于同一服务器上的同一数据库。

要恢复单个数据库：

1. 从命令行上作为 root 用户通过以下命令停止服务器：

   ```
   # /usr/sbin/directoryserver stop
   ```

2. 变换到包含所要恢复之备份的目录。
3. 对于包含要用备份来予以覆盖的数据库的目录而言，请将所有文件都复制到其中。数据库目录位于：

   `/var/ds5/slapd-`*`serverID`*`/db`

   例如，可以键入：

   `cp` *`backupDir`*`/* /var/ds5/slap-siroe/db/`*`databaseDir`*

恢复包含复制条目的数据库

本部分介绍如何在供给服务器和客户服务器上恢复数据库，以及如何确保在恢复操作后保持供给服务器和客户服务器同步。

恢复供给器副本

如果是要恢复向其它服务器提供条目的数据库 （供给器副本），则必须重新初始化所有从恢复数据库中接收更新数据的服务器 （例如客户服务器、中枢服务器及多原版复制环境中的其它供给服务器）。

恢复过程中，将擦除与恢复数据库相关联的更改日志。供给服务器的日志文件中将记录一条消息，指示需要重新进行初始化。

有关初始化客户服务器的信息，请参阅第 8 章 “管理复制”。

恢复客户副本

如果所恢复的数据库中包含从供给服务器 (supplier server) 接收来的数据，就会出现下列两种情况之一：

- 更改日志条目在供给服务器上尚未过期。

  只有当备份时间小于所设更改日志有效期属性的最大值时，才会出现这种情况。该属性称为 `nsslapd-changelogmaxage`，位于 `cn=changelog5,cn=config` 条目中。有关该选项的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

  可恢复本地客户 (consumer) 然后继续正常操作。但是，在恢复客户副本时，客户服务器*必须*停止。如果在恢复客户副本时继续复制，则复制过程将出现许多错误。

- 自本地备份以来，供给服务器上的更改日志条目已过期。

  需要重新初始化客户服务器。有关重新初始化客户服务器的详细信息，请参阅第 294 页上的 “初始化客户”。

有关管理复制的信息，请参阅第 8 章 “管理复制”。

恢复 dse.ldif 配置文件

要恢复 `dse.ldif` 配置文件，请停止服务器，然后使用第 145 页上的 “恢复单个数据库”中所述的操作过程将 `dse.ldif` 文件的备份副本复制到自己的目录中。复制完数据后，重新启动服务器。

目录将在以下目录中创建 `dse.ldif` 文件的两个备份副本：

`/var/ds5/slapd-`*`serverID`*`/config`

其中 `dse.ldif.startOK` 文件记录服务器启动时 `dse.ldif` 文件的副本。而 `dse.ldif.bak` 文件则包含对 `dse.ldif` 文件所做最新更改的备份。将带有最新更改信息的文件复制到自己的目录中。

第 5 章

高级条目管理

除在目录中建立数据层次结构外，管理诸如用户等条目通常还需要创建组和共享公用属性值。 iPlanet Directory Server 通过组、角色和服务类 (CoS) 提供高级条目管理功能。

组是以成员列表方式或成员过滤器方式命名其它条目的条目。角色不仅提供相同的功能，而且通过一种机制生成有关每个角色成员的 `nsrole` 属性。 CoS 还生成一种虚拟属性，从而允许条目共享公用属性值，而无需将该属性值存储在各个条目中。

本章介绍下列分组机制及其操作过程：

- 管理组
- 分配角色
- 定义服务类 (CoS)

为充分利用角色和服务类的功能，最好在目录部署的规划阶段就确定目录的拓扑结构 (topology)。有关详细信息，请参阅 *iPlanet Directory Server 部署指南*。

管理组

组是一种建立条目关联以便于管理的机制，例如定义 ACI。该机制在先前版本的 iPlanet Directory Server 中已经提供，主要用于与旧服务器版本实现兼容。有关创建等价角色定义的过程，请参阅第 150 页上的 “分配角色”。

下面部分将介绍静态组和动态组的管理。有关组的概念性说明，请参阅 *iPlanet Directory Server 部署指南*。有关管理组的详细信息，请参阅*通过 Directory Console 管理服务器*。

组定义是特殊的条目，用于在静态列表中命名组成员，或者提供定义一组动态条目的过滤器。一个组所能包含的成员的范围是整个目录，而不管组定义项在什么位置。为简化管理，所有组定义项通常储存在一个位置，一般是根后缀下的 `ou=Groups`。

定义静态组的条目继承 groupOfUniqueNames 对象类。组成员按它们的 DN 列出，作为 uniqueMember 属性的多个值。

定义动态组的条目继承 groupOfUniqueNames 和 groupOfURLs 对象类。组成员资格由 memberURL 属性指定的过滤器定义。动态组的成员是每次评估过滤器时与过滤器匹配的条目。

条目编辑器管理这两种类型的组条目。该对话框用于给组命名，然后创建或修改成员列表或成员过滤器。本部分包含下列创建和修改组的操作步骤：

- 第 148 页上的 “添加新静态组”
- 第 149 页上的 “添加新动态组”
- 第 149 页上的 “修改组定义”
- 第 149 页上的 “删除组定义”

添加新静态组

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 右键单击目录树中要在其中添加新组的条目。选择 “新建” > “组”。

   也可转到 “对象”菜单，选择 “新建” > “组”。
3. 单击左侧窗口中的 “常规”。在 “组名”字段中，键入新组的名称。

   组名为必需项。
4. 在 “说明”字段中输入对新组的说明。
5. 单击左侧窗口中的 “成员”。在右侧窗口中，选择 “静态组”选项卡。单击 “添加”以将新成员添加到组中。

   此时显示标准的 “搜索用户和组”对话框。
6. 在 “搜索”下拉列表中，选择所要搜索的条目类型 （用户、组或上述二者），然后单击 “搜索”。在搜索结果中选择一个或多个条目，然后单击 “确定”。

注意 由于链接原因，静态组成员可能在远程位置上。可以使用参照完整性插件来确保所删除的成员条目也将自动从静态组条目中删除。有关将参照完整性与链接配合使用的信息，请参阅第 88 页上的 “配置链接策略”。

---

7. 单击左侧窗口中的 “语言”，从而为组添加特定语言的信息。
8. 单击 “确定”以创建新组。它将作为创建位置上的一个子项出现。

添加新动态组

1. 执行第 148 页上的 “添加新静态组”中的步骤 1-4。
2. 单击左侧窗口中的 “成员”。在右侧窗口中，选择 “动态组”选项卡。单击 “添加”以创建用于数据库查询的 LDAP URL。

   此时显示标准的 “构造及测试 LDAP URL”对话框。
3. 在文本字段中输入一个 LDAP URL，或者选择 “构造”，从而在引导下完成一个包含组过滤器的 LDAP URL 的构造。构造 URL 后单击 “确定”。
4. 单击左侧窗口中的 “语言”，从而为组添加特定语言的信息。
5. 单击 “确定”以创建新组。

   新组出现在目录树中。

修改组定义

1. 在 Directory Server Console 上，选择 “目录”选项卡。
2. 在目录树中，双击要修改的组条目，或者从 “对象”菜单中选择 “打开”。

   用于修改组定义项的 “编辑项目”对话框出现。
3. 在 “常规”、“成员”或 “语言”类别中修改组信息。单击 “确定”。

   要查看更改结果，请转到 “视图”菜单，选择 “刷新”。

删除组定义

要删除任一类型的组，只需删除定义该组的条目即可。

分配角色

角色是一种新的分组机制，设计用于提高效率和便于应用程序使用。角色的定义和管理都与组相似，但此外，角色成员条目还有一个生成的属性，该属性指明其担任的角色。例如，应用程序可以直接读取条目 (entry) 的角色，而不用先选择组然后再浏览成员列表。

本部分包含下列主题：

关于角色

每个角色都有自己的*成员*，即拥有该角色的条目。与组一样，角色成员可以明确指定，也可以动态指定。角色机制自动生成 `nsRole` 属性，它包含该条目所属的所有角色定义项的 DN。

指定角色成员资格的方式与所要使用的角色类型有关。 iPlanet Directory Server 支持三类角色：

- 受管理的角色 — 明确向成员条目分配角色。
- 已过滤的角色 — 如果条目与指定的 LDAP 过滤器相匹配，则条目为角色成员。因此，该角色与每个条目中包含的属性有关。
- 嵌套角色 — 可用于创建包含其它角色的角色。

在受管理的角色中，管理员通过将 `nsRoleDN` 属性添加到条目来分配角色。属性的值就是该角色定义项的 DN。受管理的角色相当于一个静态组，不同之处在于静态组的成员资格在每个条目中定义，而不在角色定义项中定义。

已过滤的角色相当于动态组：它们都在其 nsRoleFilter 属性中定义过滤器字符串。但是，已过滤的角色涵盖的范围是其所在的子树，即在其定义项的父项的下面。每当服务器返回一个在过滤角色范围内的条目时 （即它与过滤器字符串相匹配），则该条目将包含标识角色的 nsRole 生成属性。

nsRole 属性属于计算属性，它并不与条目一起存储，而是作为操作结果中的常规属性返回到客户机应用程序。但由于是服务器为客户机应用程序执行的操作，因此评估角色比评估组需要占用更大量的资源。然而，检查角色成员资格的方法是一致的，且在服务器端透明进行。

注意 1. nsRole 属性只供角色机制使用，并且受到保护以防被修改。但是，该属性可以被读取，您可以定义访问控制从而保护其不被读取。

2. nsRole 属性不可用于搜索过滤器。如果希望某个应用程序读取 nsRole 属性，必须使用其它过滤器执行搜索，然后在搜索操作返回的条目中读取 nsRole 属性的值。

有关如何在目录中使用角色的详细信息，请参阅 *iPlanet Directory Server 部署指南*。

角色限制

创建用于支持目录服务的角色时，需要注意下列限制条件：

角色和链接。如果目录树是利用链接功能在多个服务器上分布的，则定义角色的条目必须与拥有这些角色的条目位于同一服务器上。如果一个服务器 （比如服务器 A）通过链接功能接收另一个服务器 （比如服务器 B）的条目，则这些条目将包含在 B 上定义的角色，但不会被分配 A 上定义的任何角色。

已过滤的角色不可使用 CoS 的生成属性。已过滤的角色的过滤器字符串不可基于 CoS 虚拟属性的值 （请参阅第 160 页上的 “关于 CoS”）。但是， Cos 定义中的说明符属性可以引用由角色定义生成的 nsRole 属性 （请参阅第 174 页上的 “创建基于角色的属性”）。

使用控制台管理角色

本部分包含下列创建和修改角色的操作步骤：

创建角色时，需要决定用户能否将自身添加到角色中，以及能否从角色中删除自身。有关角色和访问控制的详细信息，请参阅第 158 页上的 “安全使用角色”。

创建受管理的角色

受管理的角色可用于创建明确枚举的成员列表。向条目中添加受管理的角色可通过为条目添加 `nsRoleDN` 属性来完成。

要创建成员并将其添加到受管理的角色中：

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 浏览目录树，选择新角色的父项。
3. 转到 “对象”菜单，然后选择 “新建” > “角色”。也可右键单击条目并选择 “新建” > “角色”。

   此时显示 “创建新角色”对话框。
4. 单击左侧窗口中的 “常规”。在 “角色名”字段中，键入新角色的名称。

   角色名为必需项。
5. 在 “说明”字段中输入新角色的说明。
6. 单击左侧窗口中的 “成员”。
7. 在右侧窗口中，选择 “受管理的角色”。单击 “添加”，将新条目添加到成员列表中。

   此时显示标准的 “搜索用户和组”对话框。

8. 在 “搜索”下拉列表中，选择 “用户”，然后单击 “搜索”。选择所返回的条目之一，然后单击 “确定”。
9. 完成向角色中添加条目后，单击 “确定”。

   新的角色将显示在目录中，并带有已过滤的角色的图标。

创建已过滤的角色

根据各个条目所含的特定属性，可以将条目分配给已过滤角色。这是通过指定 LDAP 过滤器来实现的。与过滤器相匹配的条目即被视为拥有该角色。

要创建成员并将其添加到已过滤角色中：

1. 执行第 152 页上的 “创建受管理的角色”的步骤 1-5。
2. 单击左侧窗口中的 “成员”。
3. 在右侧窗口中，选择 “已过滤的角色”。
4. 在文本字段中输入一个 LDAP 过滤器，或者单击 “构造”，从而在指导下完成一个 LDAP 过滤器的构造过程。
5. 如果单击 “构造”，则会显示标准 LDAP URL 构造对话框。忽略 LDAP 服务器主机、端口、基本 DN 和搜索等字段 （因为无法为过滤角色的定义指定搜索区域）。

   a. 从搜索内容下拉列表中选择要过滤的条目类型。

      可以选择用户、组或选择两者。

   b. 从搜索范围下拉列表中选择属性。后面的两个字段可用于精简搜索过程，方法是从下拉列表中选择限定符 （例如包含、不包含、是、非）并在文本框中输入属性名称。要添加其它过滤器，请单击 “更多”。要删除不必要的过滤器，请单击 “较少”。

   c. 单击 “确定”以保存过滤器。
6. 单击 “测试”可试用过滤器。

   “过滤器测试结果”对话框将显示与过滤器相匹配的条目。
7. 单击 “确定”。

   新的角色将显示在目录中，并带有已过滤的角色的图标。

创建嵌套角色

嵌套角色可用于创建包含其它角色的角色。创建嵌套角色前，必须已经存在另一个角色。创建嵌套角色时，控制台将显示可用于嵌套的角色列表。嵌套角色中所嵌套的角色是用 `nsRoleDN` 属性指定的。

要创建成员并将其添加到嵌套角色中：

1. 执行第 152 页上的 “创建受管理的角色”的步骤 1-5。
2. 单击左侧窗口中的 “成员”。
3. 在右侧窗口中，选择 “嵌套角色”。
4. 单击 “添加”，将角色添加到列表中。嵌套角色的成员是其它现有角色的成员。

   此时显示 “角色选择器”对话框。
5. 从 “可用角色”列表中选择角色，然后单击 “确定”。
6. 单击 “确定”。

   新的角色将显示在目录中，并带有嵌套角色的图标。

查看和编辑条目的角色

1. 在 iPlanet Directory Server Console 中，选择 “目录”选项卡。
2. 浏览目录树，然后选择要查看或编辑其角色的条目。选择 “对象”菜单中的 “设置角色”。

   此时显示 “角色”对话框。
3. 选择 “受管理的角色”选项卡，显示该条目所属的受管理角色。
4. 要添加新的受管理角色，请单击 “添加”，然后从 “角色选择器”窗口中选择可用的角色。单击 “确定”。

   要删除某个受管理角色，请将其选定，然后单击 “删除”。

   要编辑与条目相关联的受管理角色，请单击 “编辑”。此时显示 “编辑项目”对话框。对常规信息或成员进行更改，然后单击 “确定”。
5. 选择 “其它角色”选项卡，查看该条目所属的已过滤角色或嵌套角色。
6. 单击 “编辑”可对与条目相关联的已过滤角色或嵌套角色进行更改。单击 “确定”按钮，保存修改结果。
7. 修改完角色并保存好更改结果后，单击 “确定”。

修改角色项

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 浏览导航树，查找现有角色的定义项。角色是创建该角色所在的条目的子项。双击角色。

   此时显示 “编辑项目”对话框。
3. 单击左侧窗口中的 “常规”以更改角色名和说明。
4. 单击左侧窗口中的 “成员”以更改受管理角色和嵌套角色的成员，或者更改已过滤角色的过滤器。
5. 单击 “确定”按钮，保存修改结果。

去活角色

通过去活成员所属的角色，可以临时禁用角色的成员。去活角色将去活角色所拥有的条目，而非角色本身。如果角色成员条目代表目录用户，则在去活用户条目时，他们将无法访问目录。

要临时禁用角色的成员：

1. 在 Directory Server Console 上，选择 “目录”选项卡。
2. 浏览导航树，查找角色的定义项。角色是创建该角色所在的条目的子项。
3. 选择角色。选择 “对象”菜单中的 “去活”。

   也可以右键单击角色并选择菜单中的 “去活”。

   角色被去活。

   要查看去活的条目，请从菜单中选择 “视图” > “去活状态”。贯穿角色成员图标的红色斜杠表明这些成员已处于去活状态。

重新激活角色

1. 在 Directory Server Console 上，选择 “目录”选项卡。
2. 浏览导航树，查找角色的定义项。角色是创建该角色所在的条目的子项。
3. 选择角色。选择 “对象”菜单中的 “激活”。

   也可右键单击角色，然后选择菜单中的 “激活”。

   角色随即被重新激活。

   要查看被重新激活的条目，请选中 “视图”菜单中的 “去活状态”。正常显示的角色图标指示角色处于活动状态。

删除角色

删除角色将仅删除角色的定义项，而不会删除角色的成员。

要删除角色：

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 浏览导航树，查找角色的定义项。角色是创建该角色所在的条目的子项。
3. 右键单击角色，然后选择 “删除”。

   此时显示的对话框将要求对删除予以确认。单击 “是”。
4. 此时显示 “已删除的项目”对话框，通知用户已成功删除角色。单击 “确定”。

注意 删除角色将仅删除角色条目，而不会为每个角色成员删除 nsRoleDN 属性。要删除该属性，请启用参照完整性插件并将其配置为管理 nsRoleDN 属性。有关详细信息，请参阅第 65 页上的 “保持参照完整性”。

使用命令行管理角色

在目录管理员可以通过命令行实用程序访问的条目中定义角色。创建角色后，按如下所示分配成员：

- 对于受管理角色的成员，其条目中有 nsRoleDN 属性。
- 已过滤角色的成员是与 nsRoleFilter 属性中所指定的过滤器相匹配的条目。
- 嵌套角色的成员是嵌套角色定义项的 nsRoleDN 属性中所指定的角色成员。

所有角色定义继承 LDAPsubentry 和 nsRoleDefinition 对象类。下表列出了针对每种角色类型的附加对象类和关联属性。

角色类型	对象类	属性
受管理的角色	nsSimpleRoleDefinition nsManagedRoleDefinition	Description （可选）
已过滤的角色	nsComplexRoleDefinition nsFilteredRoleDefinition	nsRoleFilter Description （可选）
嵌套角色	nsComplexRoleDefinition nsNestedRoleDefinition	nsRoleDN Description （可选）

注意 有些情况下需要用 ACI 来保护 nsRoleDN 属性的值，因为该属性是可写的。有关安全性和角色的信息，请参阅第 158 页上的 “安全使用角色”。

受管理的角色定义示例

要创建将分配给所有营销人员的角色，请运行以下 ldapmodify 命令：

```
ldapmodify -a -D "cn=Directory Manager" -w secret -h host -p 389
dn: cn=Marketing,ou=people,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: nsRoleDefinition
objectclass: nsSimpleRoleDefinition
objectclass: nsManagedRoleDefinition
cn: Marketing
description: managed role for marketing staff
```

注意：nsManagedRoleDefinition 对象类继承 LDAPsubentry、nsRoleDefinition 和 nsSimpleRoleDefinition 对象类。

要将该角色分配给营销人员 Bob，请用以下 ldapmodify 命令更新其条目：

```
ldapmodify -D "cn=Directory Manager" -w secret -h host -p 389
dn: cn=Bob,ou=people,dc=siroe,dc=com
changetype: modify
add: nsRoleDN
nsRoleDN: cn=Marketing,ou=people,dc=siroe,dc=com
```

条目中的 nsRoleDN 属性指示该条目是某个受管理角色的成员，该角色由其角色定义的 DN (cn=Marketing,ou=people,dc=siroe,dc=com) 标识。

已过滤的角色定义示例

要为销售经理设置已过滤的角色，请运行以下 ldapmodify 命令：

```
ldapmodify -a -D "cn=Directory Manager" -w secret -h host -p 389
dn: cn=SalesManagerFilter,ou=people,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: nsRoleDefinition
objectclass: nsComplexRoleDefinition
objectclass: nsFilteredRoleDefinition
cn: SalesManagerFilter
nsRoleFilter: o=sales managers
Description: filtered role for sales managers
```

注意：nsFilteredRoleDefinition 对象类继承 LDAPsubentry、nsRoleDefinition 和 nsComplexRoleDefinition 对象类。 nsRoleFilter 属性指定在同一子树中其 o （组织）属性的属性值为 sales managers 的所有条目都将是该角色的成员。

嵌套角色定义示例

如果要创建一个角色，该角色同时包含上面示例中创建的角色中的营销人员和销售经理，请使用以下 ldapmodify 命令：

```
ldapmodify -a -D "cn=Directory Manager" -w secret -h host -p 389
dn: cn=MarketingSales,ou=people,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: nsRoleDefinition
objectclass: nsComplexRoleDefinition
objectclass: nsNestedRoleDefinition
cn: MarketingSales
nsRoleDN: cn=SalesManagerFilter,ou=people,dc=siroe,dc=com
nsRoleDN: cn=Marketing,ou=people,dc=siroe,dc=com
```

注意：nsNestedRoleDefinition 对象类继承 LDAPsubentry、nsRoleDefinition 和 nsComplexRoleDefinition 对象类。 nsRoleDN 属性中包含受管理角色 marketing 和已过滤角色 sales managers 的 DN。

上例中的两个用户 （Bob 和 Pat）将成为此新嵌套角色的成员。

安全使用角色

并非每个角色都适于在安全环境中使用。创建新角色时，不妨考虑如何简便地将角色分配给条目或从条目中删除角色。有时用户会倾向于能简便地向角色中添加自身或从角色中删除自身。例如，如果有一个名为 “山地自行车”的兴趣小组角色，您可能希望感兴趣的用户能方便地将自己添加到其中，或者从角色中删除自身。

但在安全环境下，这样开放的角色并不适用。例如，请考虑帐户去活角色。默认情况下，帐户去活角色中包含为其后缀定义的 ACI （有关帐户去活的详细信息，请参阅第 252 页上的 “去活用户和角色”）。创建角色时，服务器管理员将决定用户是否能将自身分配给角色，或者从角色中删除自身。

例如，用户 A 拥有受管理角色 MR。 MR 角色已通过命令行的帐户去活功能予以锁定。这就意味着用户 A 无法绑定到服务器，因为该用户的 nsAccountLock 属性将被计算为 “真”。但是，假设该用户已经绑定且已通过 MR 角色予以锁定。如果没有相应的 ACI 进行阻止，该用户即可删除自己条目中的 nsRoleDN 属性并将自己解锁。

为防止用户删除 nsRoleDN 属性，根据所用角色的类型，请使用下列 ACI。

受管理的角色。对于作为受管理角色成员的条目，使用下列 ACI 可以防止用户通过删除相应的 nsRoleDN 而解锁自身：

```
aci: (targetattr="nsRoleDN")
     (targattrfilters="
add=nsRoleDN:(!(nsRoleDN=cn=AdministratorRole,dc=siroe,dc=com)),
del=nsRoleDN:(!(nsRoleDN=cn=nsManagedDisabledRole,dc=siroe,dc=com)
     ")
     (version3.0;aci "allow mod of nsRoleDN by self
        except for critical values";
        allow(write)
          userdn="ldap:///self";)
```

已过滤的角色。应对作为过滤器一部分的属性进行保护，从而确保用户无法通过修改属性而丢弃已过滤的角色。用户不得添加、删除和修改已过滤角色所用的属性。如果过滤器属性的值为计算值，则应采取一定的方式对能修改过滤器属性值的所有属性加以保护。

嵌套角色。嵌套角色中包括已过滤的角色和受管理的角色，因此对构成嵌套角色的各个角色都应考虑上述几点。

定义服务类 (CoS)

服务类 (COS) 机制使您可以创建不存储在条目中的虚拟属性。相反， Cos 机制在条目被发送到客户机应用程序时生成虚拟属性。 CoS 可以简化条目的管理，同时减少对存储空间的要求。

与组和角色一样， CoS 依赖于目录中的帮助程序条目，并可以通过控制台或通过命令行进行配置。以下部分将介绍 CoS，并提供通过上述两种途径来管理 CoS 的操作步骤：

- 第 160 页上的 “关于 CoS”
- 第 164 页上的 “CoS 限制”
- 第 165 页上的 “使用控制台管理 CoS”
- 第 168 页上的 “从命令行管理 CoS”
- 第 174 页上的 “创建基于角色的属性”
- 第 175 页上的 “保护 CoS 安全”

关于 CoS

CoS 为其所有*目标项*（即 CoS 范围内的任何条目）定义虚拟属性及其属性值。每个 CoS 都由目录中的下列条目组成：

- CoS 定义项 — 标识所用的 CoS 类型和将要生成的 CoS 属性的名称。与角色定义项类似，它也继承 `LDAPsubentry` 对象类。 CoS 的范围包括 CoS 定义项父项下面的整个子树。同一 CoS 属性存在多个定义，因此该属性为多值属性。
- 模板项 — 包含一个或多个虚拟属性的值。 CoS 范围内的所有条目都将使用此处定义的值。可以有多个模板项，在这种情况下，生成的属性可以是多值属性。

CoS 有三种类型，每种类型分别对应于 CoS 定义项和模板项的不同交互作用：

- 指针 CoS — CoS 定义项使用模板 DN 直接标识模板项。对于 CoS 属性而言，所有目标项的属性值都与模板中给定的值相同。
- 间接 CoS — CoS 定义项标识称为间接说明符的属性，该属性在目标项中的值将决定用于该条目的模板。目标项中的属性必须包含 DN。利用间接 CoS，每个目标项可以使用不同的模板，因此有不同的 CoS 属性值。
- 典型 CoS — CoS 定义项标识模板的基本 DN 和作为目标项属性名称的说明符。包含 CoS 值的模板是由目标项中说明符属性的 RDN （相对域名）值和模板的基本 DN (base DN) 的共同确定的。

注意 对于 LDAP 搜索请求而言，如果其中所含的过滤器引用了 CoS 虚拟属性，则服务器不支持该搜索请求。 LDAP 搜索过滤器只支持存储在条目中的实际属性，而不包含 CoS 或 `nsRole` 属性。决定利用 CoS 定义项生成哪些属性时，请务必小心。

要查找基于虚拟属性值的条目，目录客户机必须检索条目的超集 （例如整个分支），然后筛选出感兴趣的条目。

以下部分将详细介绍 CoS 定义项和模板项，并给出每种 CoS 类型的示例。

CoS 定义项和模板项

CoS 定义项是 `cosSuperDefinition` 对象类的一个实例。 CoS 定义项也继承下列其中一个对象类，以指定 CoS 类型：

- `cosPointerDefinition`
- `cosIndirectDefinition`
- `cosClassicDefinition`

CoS 定义项包含每种 CoS 类型的特定属性，用于根据需要命名目标项中的虚拟 CoS 属性、模板 DN 和说明符属性。默认情况下， CoS 机制不会覆盖与 CoS 属性同名的现有属性的值。但是，可以使用 CoS 定义项 (CoS definition entry) 的语法来控制上述行为。

CoS 模板项是 `cosTemplate` 对象类的一个实例。 CoS 模板项 (CoS template entry) 中包含由 CoS 机制所生成的一个或多个属性值。给定 CoS 的模板项存储在目录树中与 CoS 定义同级的位置上。

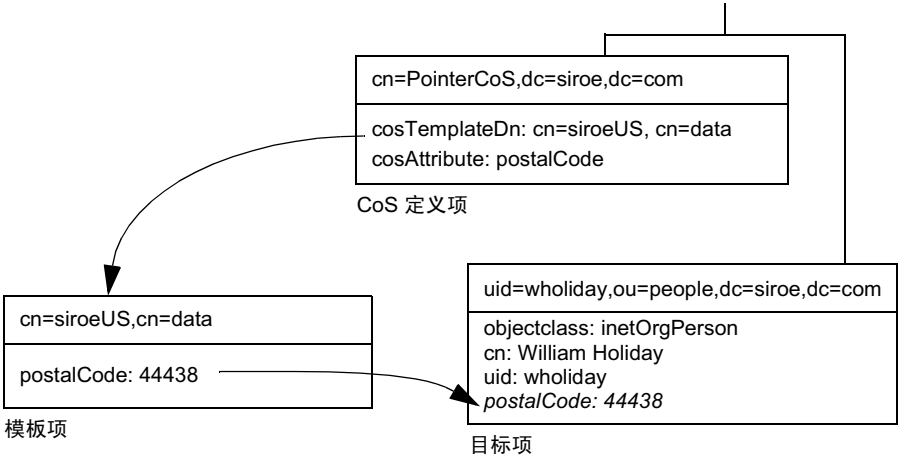
如有可能，定义、模板和模板项应位于同一位置以方便管理。并且，应使用暗示它们所提供的功能的名称来对它们进行命名。例如，定义项 DN `"cn=classicCosGenerateEmployeeType,ou=People,dc=siroe,dc=com"` 比 `"cn=ClassicCos1,ou=People,dc=siroe,dc=com"` 的说明性更强。

有关与每种 CoS 相关联的对象类和属性的详细信息，请参阅第 168 页上的 “从命令行管理 CoS”。

指针 CoS 示例

下例显示一个为存储在 dc=siroe,dc=com 下的所有条目定义共用邮政编码的 CoS。该示例中的三个条目如下图所示：

图 5-1 指针 CoS 定义和模板示例

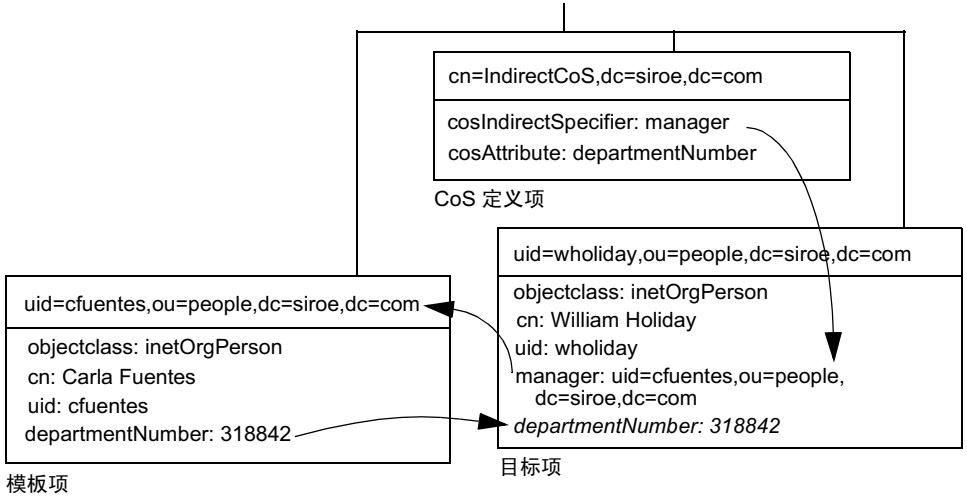


模板项 (template entry) 由 CoS 定义项 (CoS definition entry) 中的模板 DN cn=siroeUS,cn=data 标识。每次在位于 dc=siroe,dc=com 下的条目上查询 postalCode 属性时， iPlanet Directory Server 都会返回模板项 cn=siroeUS,cn=data 中可用的值。因此，邮政编码将与条目 uid=wholiday,ou=people,dc=siroe,dc=com 一起出现，但不存储在该条目中。当 CoS 为数千或数百万个条目生成多个共享属性时，该机制可以节省大量的存储空间。

间接 CoS 示例

在本示例中，间接 CoS (indirect CoS) 使用目标项 (target entry) 的 manager 属性来标识模板项 (template entry)。利用此方法， CoS 机制可以为某个部门的所有员工生成与他们的经理相同的部门号，并且确保不断对其更新。该示例中的三个条目如下图所示：

图 5-2 间接 CoS 定义和模板示例



间接 CoS 定义项命名说明符属性，在本例中，即为 `manager` 属性。 William Holiday 的条目是此 CoS 的目标项之一，他的 `manager` 属性包含 `cn=Carla Fuentes,ou=people,dc=siroe,dc=com` 的 DN。因此， Carla Fuentes 的条目是一个模板项，它反过来提供 `318842` 作为 `departmentNumber` 属性的属性值。

通过间接说明符，间接 CoS 可使用目录中的任意条目作为其模板。出于安全和性能的原因，应小心使用这种类型的 CoS。在许多情况下，利用典型 CoS 或使用不太灵活的指针 CoS 机制限制目标项的位置，也可以取得相同的结果。

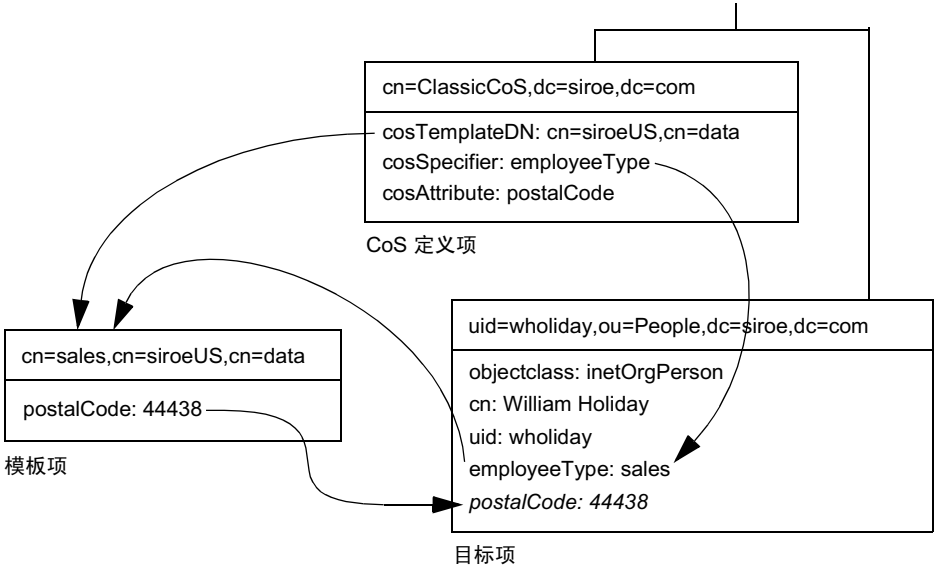
典型 CoS 示例

典型 CoS 机制根据定义项中给定的基本 DN 以及目标项中的说明符确定模板 DN。说明符属性的属性值将当作模板 DN 中的 `cn` 值。因此，典型 CoS 的模板 DN 必须具有以下结构：

`cn=`*specifierValue*`,`*baseDN*

下图中的示例显示一个生成邮政编码属性值的典型 CoS (classic CoS) 定义：

图 5-3 典型 CoS 定义和模板示例



本例中， Cos 定义项的 `cosSpecifier` 属性指定了 `employeeType` 属性。该属性与模板 DN 一起，将模板项 (template entry) 标识为 `cn=sales,cn=siroeUS,cn=data`。模板项随即将 `postalCode` 属性的值提供给目标项。

CoS 限制

CoS 功能是一个复杂的机制，它在性能和安全方面受以下条件的约束。

受限制的子树。不可以在 `cn=config` 或 `cn=schema` 子树上创建 CoS 定义。因此，这些条目不可以包含虚拟属性。

受限制的属性类型。下列属性类型不应通过 CoS 机制生成，因为它们的行为与同名实际属性的行为不相同：

- `userPassword` — CoS 生成的口令值不可用于绑定到目录服务器。
- `aci` — 目录服务器将不会基于由 CoS 定义的虚拟 ACI 值的内容而应用任何访问控制。
- `objectclass` — 目录服务器将不会基于由 CoS 定义的虚拟对象类的值而执行模式检查。
- `nsRoleDN` — CoS 生成的 `nsRoleDN` 值将不会被服务器用来生成角色。

所有模板必须是本地模板 — 模板项 DN，无论是在 CoS 定义中还是在目标项的说明符中，都必须指向目录服务器中的本地条目。它们所包含的模板和值无法通过目录链接或引荐进行检索。

CoS 虚拟值不能和实际值组合。 CoS 属性值永远都不能和条目中的实际值及模板中的虚拟值进行组合。当 CoS 覆盖实际属性值时，它将用模板中的虚拟值替换所有实际值 （请参阅第 170 页上的 “覆盖实际属性值”）。但是， CoS 机制可以组合几个 CoS 定义项的虚拟值，有关信息见第 170 页上的 “多值的 CoS 属性”。

已过滤的角色不可使用 CoS 的生成属性。已过滤的角色的过滤器字符串不可基于 CoS 虚拟属性的值。但是， Cos 定义中的说明符属性可以引用由角色定义生成的 `nsRole` 属性 （请参阅第 174 页上的 “创建基于角色的属性”）。

访问控制指令 (ACI)。对于由 CoS 生成的属性，服务器的访问控制方式与常规存储属性的完全相同。但是，如果访问控制规则与由 CoS 生成的属性值有关，则访问控制规则受第 164 页上的 “CoS 限制”中所述的条件的约束。

CoS 高速缓存等待时间。 CoS 高速缓存是一个内部的 Directory Server 结构，它将所有 CoS 数据保存在内存中以提高性能。该高速缓存已被优化，以便检索用于计算虚拟属性的 CoS 数据，甚至是在 CoS 定义项和模板项被更新时。因此，一旦添加或修改定义项和模板项，在它们生效前可能会有少许的延迟。该延迟取决于 CoS 定义的数量和复杂程度以及当前的服务器负载，但通常为几秒钟。

使用控制台管理 CoS

本部分介绍如何通过 iPlanet Directory Server Console 创建和编辑 CoS 定义。其中包含以下部分：

创建新 CoS

在指针 CoS 和典型 CoS 情况下，必须在创建定义项之前先创建模板项：

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 浏览目录树，然后选择要在其下存储模板项的父项。
3. 从 “对象”菜单或单击右键出现的上下文关联菜单中，选择 “新建” > “其他”，然后从 “新对象”对话框中的列表中选择 `costemplate`。

   “属性编辑器”对话框打开，并显示新模板中某些属性的默认值。

4. 按以下方式编辑新模板对象：

   a. 将 LDAPsubentry 和 extensibleobject 值添加到 objectclass 属性中。

   b. 添加 cn 属性并赋予其一个用于标识模板的值，例如 cosTemplateForHeadquartersFax。

   c. 将命名属性更改为新的 cn 属性。

      您可以添加任何其它属性并将其用作命名属性，但使用 cn 是惯例。

   d. 通过将 cosPriority 属性设置为一个整数值对其进行修改，或者，如果不需要该优先属性，可将其删除。

   e. 添加希望在目标项上由 CoS 机制生成的属性及其属性值。

5. 在 “属性编辑器”对话框中，单击 “确定”以创建该模板项。

6. 如果要定义该模板的指针 CoS，则在目录树中选择这个新的模板项，然后从菜单中选择 “编辑” > “复制 DN”。

对于所有类型的 CoS 而言，创建定义项的步骤都是相同的：

1. 浏览目录树，然后选择要使新服务类在其下生效的父项。

2. 从 “对象”菜单或单击右键出现的上下文关联菜单中，选择 “新建” > “服务类”。

   显示 “创建新服务类”对话框。

3. 选择左侧窗口中的 “常规”。在右侧窗口的 “类名”字段中，输入新服务类的名称。该名称将出现在 CoS 定义项的 cn 命名属性中。在 “说明”字段中输入类的说明。

4. 单击左侧窗口中的 “属性”。右侧窗口将显示由 CoS 机制在目标项上生成的属性列表。

   单击 “添加”可浏览可用属性列表并将属性添加到列表中。

5. 将属性添加到列表后，服务类行为栏中将显示一个下拉列表。单击以下单元格将选择相应的覆盖行为：

   - **不覆盖目标项属性 —** 只有在没有相应的属性值存储于目标项的同一属性中时才会生成 CoS 属性值。
   - **覆盖目标项属性 —** 由 CoS 生成的属性值将覆盖目标项中该属性的任何值。
   - **覆盖目标项属性并且是可操作的 —** 该属性将覆盖任何目标值，并且是可操作属性。这样，除非有明确请求，否则该属性对于客户机应用程序而言不可见。

注意 只有模式中也将属性定义为可操作的情况下，才能将该属性设置为可操作。

6. 单击左侧窗口中的 “模板”。在右侧窗口中，选择模板项的标识方式，然后填写相应的字段。这将确定要定义的 CoS 类型。
   - **按照其 DN —** 该选项将定义指针 CoS：在 “模板 DN”字段中输入一个模板 DN。单击 “浏览”从目录中选择模板 DN，或者按住 Ctrl-V 以粘贴在创建模板项后复制的模板 DN。
   - **使用其中一个目标项的属性值 —** 该选项将定义间接 CoS：在 “属性名”字段中输入说明符属性的名称。请务必选择包含 DN 值的属性。单击 “更改”，从列表中选择属性。
   - **使用一个 DN 以及其中一个目标项的属性值 —** 该选项将定义典型 CoS：输入模板的基本 DN 和属性名。单击 “浏览”选择潜在目标项的父项，然后单击 “更改”从列表中选择属性。
7. 单击 “确定”以创建 CoS 定义项。

编辑现有 CoS

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 浏览目录树然后选择包含 CoS 定义的父项。 CoS 条目作为该父项的子项出现。
3. 双击 CoS。

   此时显示 “编辑项目”对话框。
4. 单击左侧窗口中的 “常规”以更改 CoS 名称和说明。
5. 在左侧窗口中单击 “属性”以添加或删除由 CoS 机制生成的虚拟属性。
6. 在左侧窗口中单击 “模板”，重新定义模板说明符属性的名称或模板项 DN。该对话框也可用于重新定义 CoS 定义的类型。
7. 单击 “确定”按钮，保存修改结果。

删除 CoS

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 浏览目录树然后选择包含 CoS 定义的父项。 CoS 条目作为该父项的子项出现。
3. 右键单击 CoS，然后选择 “删除”。此时显示的对话框将要求对删除予以确认。单击 “是”。
4. 此时显示 “已删除的项目”对话框，通知用户已成功删除 CoS。单击 “确定”。

从命令行管理 CoS

由于所有配置信息和模板数据均作为目录项进行存储，因此可以使用标准 LDAP 工具进行 CoS 的配置和管理。该部分包含下列主题：

从命令行创建 CoS 定义项

所有 CoS 定义项都继承 `LDAPsubentry` 和 `cosSuperDefinition` 对象类。此外，每种 CoS 都继承特定的对象类并包含相应的属性。下表列出了与每种 CoS 定义项 (CoS definition entry) 相关联的对象类和属性：

表 5-1 CoS 定义项

CoS 类型	CoS 定义项

指针 CoS	`objectclass: top`
	`objectclass: LDAPsubentry`
	`objectclass: cosSuperDefinition`
	`objectclass: cosPointerDefinition`
	`cosTemplateDN:` *DN_string*
	`cosAttribute:` *list_of_attributes qualifier*

间接 CoS	`objectclass: top`
	`objectclass: LDAPsubentry`
	`objectclass: cosSuperDefinition`
	`objectclass: cosIndirectDefinition`
	`cosIndirectSpecifier:` *attribute_name*
	`cosAttribute:` *list_of_attributes qualifier*

典型 CoS	`objectclass: top`
	`bbjectclass: LDAPsubentry`
	`objectclass: cosSuperDefinition`
	`objectclass: cosClassicDefinition`
	`cosTemplateDn:` *DN_string*
	`cosSpecifier:` *attribute_name*
	`cosAttribute:` *list_of_attributes qualifier*

CoS 定义项中可以使用下列属性 （有关这些属性的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*）：

表 5-2 CoS 定义项属性

属性	在 CoS 定义项内的用途
`cosAttribute:` *attribute_name override merge*	定义要为其生成值的虚拟属性的名称。该属性为多值属性，每个值为将从模板生成其值的属性指定名称。限定符指定在特殊情况下计算 CoS 属性值的方式。
`cosIndirectSpecifier:` *attribute_name*	定义目标项中其值被间接 CoS 用来标识模板项 (template entry) 的属性的名称。已命名的属性被称为说明符，它在每个目标项中必须包含一个完整的 DN 字符串。该属性为单值属性，但说明符属性可以是多值属性以指定多个模板。
`cosSpecifier:` *attribute_name*	定义目标项中其值被典型 CoS 用来标识模板项 (template entry) 的属性的名称。已命名的属性被称为说明符，它必须包含一个可在目标项的 RDN 中找到的字符串。该属性为单值属性，但说明符属性可以是多值属性以指定多个模板。
`cosTemplateDn:` *DN_string*	为指针 CoS 定义提供模板项 (template entry) 的完整 DN，或为典型 CoS 提供模板项 (template entry) 的基本 DN。

`cosAttribute` 属性允许在 CoS 属性的名称后使用两个限定符。 *override* 限定符可以是下列值之一：

- `default` （或没有限定符）— 指示在实际属性与虚拟属性同属一个类型时，服务器将不覆盖存储在条目中的实际属性值。
- `override` — 指示即使该条目存储有属性值，服务器仍会一直返回由 CoS 生成的值。
- `operational` — 指示只有在搜索中明确请求某个属性时，才会返回该属性。要返回可操作属性时，无须通过模式检查。它还具有与 `override` 限定符相同的特性。

  只有模式中也将属性定义为可操作的情况下，才能将该属性设置为可操作。例如，如果 CoS 为 `description` 属性生成了值，则不可以使用 `operational` 限定符，因为该属性在模式中未被标记为可操作。

merge 限定符可以空缺，或者被赋予下列值：

- `merge-schemes` — 允许通过多个模板或多个 CoS 定义使虚拟 CoS 属性成为多值属性。有关详细信息，请参阅第 170 页上的 “多值的 CoS 属性”。

覆盖实际属性值

可以创建包含 `override` 限定符的指针 CoS 定义项，如下所示：

```
dn: cn=pointerCoS,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: cosSuperDefinition
objectclass: cosPointerDefinition
cosTemplateDn: cn=siroeUS, cn=data
cosAttribute: postalCode override
```

该指针 CoS 定义项指示它与生成 `postalCode` 属性值的模板项 `cn=siroeUS, cn=data` 相关联。 **override** 限定符指示该值将优先于 `postalCode` 的属性值 （如果该属性在目标项中存在）。

注意 如果用 operational 或 override 限定符定义 CoS 属性，则无法手动在 CoS 范围内的任何条目中更新该属性，因为它作为常规属性出现。

多值的 CoS 属性

指定 `merge-schemes` 限定符时，生成的 CoS 属性可以是多值属性。使 CoS 属性成为多值属性有两种方法：

- 在间接或典型 CoS 中，目标项中的说明符属性可以是多值的。在这种情况下，每个值确定一个模板，并且每个模板的值都是生成的值的一部分。
- 在其 `cosAttribute` 属性中包含相同属性名的任何类型 CoS 都可以有多个 CoS 定义项。在这种情况下，如果所有定义都包含 `merge-schemes` 限定符，则生成的属性将包含用每个定义计算得到的所有值。

这两种情况可能会一起出现，从而需要定义更多的值。然而，在所有情况下，重复值在生成的属性中将只返回一次。

如果没有 `merge-schemes` 限定符，则模板项的 `cosPriority` 属性将被用于在所有模板间为生成属性确定单个值，有关说明见下一部分。

`merge-schemes` 限定符永远不会将在目标中定义的 “实际”值与从模板生成的值合并。 *merge* 限定符与 *override* 限定符彼此独立，因此各种配对都有可能，并且每种配对所暗示的行为都是可取的。另外，在属性名称后可以任意顺序指定限定符。

注意 当同一属性有多个 CoS 定义时，它们必须有相同的 *override* 和 *merge* 限定符。如果 CoS 定义中出现不同的限定符配对，则在所有定义之间随意选择其中一种组合。

Cos 属性优先级

如果有多个 CoS 定义或多值说明符但没有 `merge-schemes` 限定符，则 Directory Server 使用优先的属性选择单个模板，用于定义虚拟属性的单个值。

`cosPriority` 属性表示特定模板在所有备选模板中的全局优先性。优先级 0 代表最高的优先级。不含 `cosPriority` 属性的模板被视为优先级最低。当两个或更多模板都提供属性值，但它们有相同的优先级 （或没有优先级设置），则值的选择具有任意性。

使用 merge-schemes 限定符时，将不考虑模板优先级。合并情况下，所有相关模板共同定义属性值，而不管它们的定义优先级。 `cosPriority` 属性在 CoS 模板项中定义，有关说明见以下部分。

从命令行创建 CoS 模板项

使用指针 CoS 或典型 CoS 时，模板项将继承 `LDAPsubentry` 对象类，并且也是 `cosTemplate` 对象类的实例。模板项必须是为 CoS 定义专门创建的。将 CoS 模板项作为 `LDAPsubentry` 对象类的实例，即可进行正常搜索而避免受到配置项的干扰。

间接 CoS 机制指向目录中任意的现有模板项。它无需事先标识，也不需要提供给 `LDAPsubentry` 对象类。只有在评估 CoS 以生成虚拟属性及其属性值时才访问间接 CoS 模板。

在任何情况下， CoS 模板项都必须包含由 CoS 在目标项中生成的属性和属性值。属性名在 CoS 定义项的 `cosAttribute` 属性中指定。

下例显示一个生成 `postalCode` 属性的指针 CoS 的最高优先级模板项：

```
dn:cn=siroeUS,cn=data,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
postalCode: 44438
cosPriority: 0
```

下列部分提供模板项的示例及每种 CoS 定义项的示例。

指针 CoS 的示例

要创建与 dc=siroe,dc=com 目录树中的所有条目共用同一邮政编码的指针 CoS (pointer CoS)，请运行以下 ldapmodify 命令：

```
ldapmodify -a -D "cn=directory manager" -w secret -h host -p 389
```

```
dn: cn=pointerCoS,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: cosSuperDefinition
objectclass: cosPointerDefinition
cosTemplateDn: cn=siroeUS,cn=data,dc=siroe,dc=com
cosAttribute: postalCode
```

```
dn:cn=siroeUS,cn=data,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
postalCode: 44438
```

CoS 模板项 (cn=siroeUS,dn=cata,dc=siroe,dc=com) 将其 postalCode 属性中存储的值提供给 dc=siroe,dc=com 后缀下的所有条目。

间接 CoS 的示例

该间接 CoS (indirect CoS) 使用目标项 (target entry) 的 team 属性来标识 CoS 模板项。要将新的间接 CoS 定义项添加到 dc=siroe,dc=com 后缀，请运行以下 ldapmodify 命令：

```
ldapmodify -a -D "cn=directory manager" -w secret -h host -p 389
```

```
dn: cn=indirectCoS,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: cosSuperDefinition
objectclass: cosIndirectDefinition
cosIndirectSpecifier: manager
cosAttribute: departmentNumber
```

然后，按如下所示为经理 Carla Fuentes 创建模板项：

```
dn:cn=Carla Fuentes,cn=data,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
departmentNumber: 318842
```

按如下所示为经理 Sue Jacobs 创建另一个模板项：

```
dn:cn=Sue Jacobs,cn=data,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
departmentNumber: 71776
```

对于包含 `manager` 属性的条目而言，定义项似乎位于目标项中（`dc=siroe,dc=com` 下的条目），因为该属性是在定义项的 `cosIndirectSpecifier` 属性中指定的。模板项的 `manager` 属性可以指向下列两个模板之一：`cn=Carla Fuentes,cn=data,dc=siroe,dc=com` 和 `cn=Sue Jacobs,cn=data,dc=siroe,dc=com`。根据经理的不同，部门号也将有所区别。

典型 CoS 的示例

在下例中，典型 CoS (classic CoS) 利用模板 DN 及 `cosSpecifier` 属性中指定的属性自动生成邮政编码。要创建典型 CoS 定义项，请运行以下 `ldapmodify` 命令：

```
ldapmodify -a -D "cn=directory manager" -w secret -h host -p 389
```

```
dn: cn=classicCoS,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: cn=siroeUS,cn=data,dc=siroe,dc=com
cosSpecifier: employeeType
cosAttribute: postalCode override
```

然后，按如下所示为销售部和营销部创建模板项：

```
dn: cn=sales,cn=siroeUS,cn=data,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
postalCode: 44438
```

```
dn: cn=marketing,cn=siroeUS,cn=data,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
postalCode: 99111
```

典型 CoS 定义项将应用于 dc=siroe,dc=com 后缀下的所有条目。根据条目中找到的 employeeType 属性与 cosTemplate DN 的组合，它将到达两个模板之一：一个是 sales 模板：为销售部的员工提供特定的邮政编码；另一个是 marketing 模板：为营销部的员工提供特定的邮政编码。

创建基于角色的属性

您可以创建典型 CoS 模式，从而基于条目所拥有的角色为该条目生成属性值。例如，使用基于角色的属性 (role-based attributes) 可以逐条设置服务器的审核限制。

要创建基于角色的属性，请在典型 CoS 的 CoS 定义项中将 nsRole 属性用作 cosSpecifier。由于 nsRole 属性可以为多值属性，因此可以定义有多个可用模板项的 CoS 模式。为解决使用哪个模板项 (template entry) 的不定性问题，可以在 CoS 模板项 (CoS template entry) 中包含 cosPriority 属性。

例如，可以创建允许 manager 角色的成员超出标准邮箱限额的 CoS。manager 角色如下所示：

```
dn: cn=ManagerRole,ou=people,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: nsRoleDefinition
objectclass: nsComplexRoleDefinition
objectclass: nsFilteredRoleDefinition
cn: ManagerRole
nsRoleFilter: o=managers
Description: filtered role for managers
```

典型 CoS 定义项 (CoS definition entry) 如下所示：

```
dn: cn=managerCOS,dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: cosSuperDefinition
objectlass: cosClassicDefinition
cosTemplateDn: cn=managerCOS,dc=siroe,dc=com
cosSpecifier: nsRole
cosAttribute: mailboxquota override
```

cosTemplateDn 属性所提供的值与 cosSpecifier 属性中指定的属性 （例中就是目标项的 nsRole 属性）一起，可以标识该 CoS 模板项 (CoS template entry)。 CoS 模板项为 mailboxquota 属性提供值。附加的限定符 override 告知 CoS 应覆盖目标项中任何现有的 mailboxquota 属性值。

按如下所示定义相应的 CoS 模板项：

```
dn:cn="cn=ManagerRole,ou=people,dc=siroe,dc=com",cn=managerCOS,
 dc=siroe,dc=com
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectlass: cosTemplate
mailboxquota: 1000000
```

该模板为 `mailboxquota` 属性提供属性值 1000000。

注意 角色 (role) 条目和 CoS 定义项应位于目录树中的同一位置，这样在其范围内有相同的目标项。 CoS 目标项也应位于同一位置，这样方便查找和维护。

保护 CoS 安全

读取操作的访问控制对条目的实际属性和虚拟属性都适用。由服务类 (CoS) 机制生成的虚拟属性的读取方式与常规属性相同，并且应给予相同的读取保护。

但是，为保护 CoS 值的安全，必须保护它所使用的所有信息源：定义项、模板项和目标项。对更新操作也有相同的要求：应控制对各个信息源的写入权限，以保护从这些信息源生成的值。

下列部分介绍在各种 CoS 条目中进行数据读写保护的一般原则。有关定义各个访问控制指令 (ACI) 的详细步骤，请参见第 6 章 “管理访问控制”。

保护 CoS 定义项

虽然 CoS 定义项不包含生成属性的值，但它提供查找该值的信息。读取 CoS 定义项可以从中发现查找包含属性值的模板项的方法，而编写该定义项可以修改生成虚拟属性的方式。

因此，应在 CoS 定义项中定义读取和写入访问控制。

保护 CoS 模板项

CoS 模板项包含 CoS 生成属性的属性值。因此，至少应保护对模板中 CoS 属性的读取和更新。

在指针 CoS 情况下，应有一个模板项不允许重命名。在大多数情况下，最简单的方法是保护整个模板项。

在典型 CoS 中，所有模板项都有一个在定义项中指定的公共父项。如果只有模板存储在父项中，则对父项的访问控制将可以保护模板。否则，如果需要访问父项下的其它条目，则各模板项需要单独保护。

在间接 CoS 中，模板可以是目录中的任何条目，包括可能还需要访问的用户条目。根据需要，可以在整个目录范围内控制对 CoS 属性的访问，或者必须保证 CoS 属性在用作模板的各个条目中的安全。

保护 CoS 目标项

在 CoS 定义范围内的所有条目，除为其生成虚拟 CoS 属性外，还用于计算属性值。

当目标项中已经存在 CoS 属性时，默认情况下， CoS 机制将不覆盖该值。如果需要覆盖，则应定义 CoS 以覆盖目标项 （参见第 170 页上的 “覆盖实际属性值”），或者应在所有潜在目标项中保护 CoS 属性。

间接 CoS 和典型 CoS 也都与目标项中的说明符属性相关。该属性提供要使用的模板项 DN 或 RDN。应在整个 CoS 范围内全局保护该属性，或者在需要的每个目标项中进行个别保护。

保护其它相关项

最后，可能需要用其它生成的 CoS 属性和角色来定义虚拟 CoS 属性。为确保对虚拟 CoS 属性的保护，需要理解并保护这些相关项。

例如，目标项中的 CoS 说明符属性可能是 `nsRole`，因此角色定义也需要进行保护。有关详细信息，请参阅第 158 页上的 “安全使用角色”。

总之，计算虚拟属性值所涉及的任何属性或条目应具有读写访问控制。为此，应对复杂的相关项进行精心规划，或者进行简化以减少复杂性。保持与其它虚拟属性的最低限度的相关性还有利于提高目录性能和降低维护难度。

第 6 章

管理访问控制

iPlanet Directory Server 可用于控制对目录的访问。本章将介绍访问控制机制。

该部分包含下列主题：

- 访问控制原则
- 默认 ACI
- 手动创建 ACI
- 绑定规则
- 从控制台创建 ACI
- 访问控制用法示例
- 查看条目的 ACI
- 高级访问控制：使用宏 ACI
- 访问控制和复制
- 记录访问控制信息
- 与早期版本的兼容性

为充分利用访问控制机制的功能和灵活性，在目录部署的规划阶段就应将访问控制战略作为整体安全策略不可分割的一部分进行定义。有关规划访问控制战略的技巧，请参阅 *iPlanet Directory Server 部署指南*。

访问控制原则

定义访问权的机制称为*访问控制*。服务器接收到请求时，会利用用户在绑定操作中提供的身份验证信息及服务器中定义的访问控制指令 (ACI) 来允许或拒绝访问目录信息。服务器可以允许或拒绝诸如读、写、搜索及比较等权限。授予用户的权限级别可能与所提供的身份验证信息有关。

使用访问控制可以控制对整个目录、目录的子树、目录中特定条目 （包括定义配置任务的条目）或特定条目属性集的访问权。您可以设置特定用户、所有属于特定组或角色的用户或所有目录用户的权限。最后，您还可以定义对特定位置 （例如 IP 地址或 DNS 名称）的访问权。

ACI 结构

与条目属性一样，访问控制指令存储在目录中。`aci` 属性是一种操作性属性。它可以用于目录的各个条目上，而不管是否为该条目的对象类所定义。当接收到来自客户机的 LDAP 请求时，目录服务器将使用该属性来评估应允许或拒绝的权限。如果有特别请求，则在 `ldapsearch` 操作中返回 `aci` 属性。

ACI 语句的三个主要部分是：

- 目标
- 权限
- 绑定规则

ACI 的权限和绑定规则部分是成对设置的，也称为 “访问控制规则” (ACR)。是否授予指定的权限与伴随的规则是否被评估为真有关。

ACI 布置

如果包含 ACI 的条目中没有任何子条目，则 ACI 将仅适用于该条目。如果该条目中有子条目，则 ACI 将同时适用于该条目及下属的所有条目。因此，在评估对给定条目的访问权时，服务器将在所请求的访问权、目录后缀及条目自身的 ACI 之间核查各个条目的 ACI。

`aci` 属性为多值属性，即可以为同一条目或子树定义多个 ACI。

您可以在某个条目上创建不直接适用于该条目，但适用于其下属子树中部分或全部条目的 ACI。这样做的好处在于：可以在目录树的上层放置通用的 ACI，从而能有效应用于很可能位于目录树下层的条目。例如，可以在 `organizationalUnit` 条目或 `locality` 条目级别上创建以包含 `inetorgperson` 对象类的条目为目标的 ACI。

利用该功能可以在上层分支点上放置通用的规则，从而使目录树中的 ACI 数达到最小。要限制更为具体的规则范围，应将其尽可能放到接近叶条目的位置处。

注意 放在根 DSE 条目上的 ACI 仅适用于该条目。

ACI 评估

评估特定条目的访问权时，服务器会编辑该条目自身及至目录服务器中所存顶级条目内的 ACI 列表。在评估期间，服务器以这种顺序处理 ACI。 ACI 的评估是跨特定目录服务器的所有数据库而进行的，但并不跨目录服务器。

在 ACI 之间适用的优先规则为：拒绝访问的 ACI 优先于允许访问的 ACI。允许访问的 ACI 之间遵循组合原则，因此即使服务器首先处理最靠近目标条目的 ACI，操作过程中也没有优先性。

例如，如果目录的根级别上拒绝写权限，则所有用户都将无法向目录中写入，而不管授予这些用户的具体权限如何。要向目录授予特定用户的写入权限，则必须限定原拒绝写入权限的范围，从而使其不包括该用户。

ACI 限制

创建目录服务的访问控制策略时，需要注意下列限制条件：

- 如果目录树是利用链接功能在多个服务器上分布的，则有些限制将适用于访问控制语句中所用的关键字：
  - 与组条目 （`groupdn` 关键字）有关的 ACI 必须位于该组条目所在的同一服务器上。如果组为动态组，则所有组成员也须在服务器上具有相应的条目。如果组为静态组，则成员条目可以位于远程服务器上。
  - 与角色定义 （`roledn` 关键字）有关的 ACI 必须位于该角色定义条目所在的同一服务器上。计划分配该角色的每个条目也必须位于同一服务器上。

  但可以将目标条目中存储的值与绑定用户条目中存储的值进行值的匹配 （例如使用 `userattr` 关键字）。即使绑定用户在持有 ACI 的服务器上没有条目，系统也会进行正常的权限评估。

  有关如何链接访问控制评估的详细信息，请参阅第 104 页上的 “数据库链接和访问控制评估”。

- 由 CoS 生成的属性无法用于所有 ACI 关键字。尤其不应将 CoS 生成的属性用于 `userattr` 关键字，因为访问控制规则不起作用。有关该关键字的详细信息，请参阅第 200 页上的 “使用 userattr 关键字”。有关 CoS 的详细信息，请参阅第 5 章 “高级条目管理”。
- 访问控制规则的评估始终在本地服务器上进行。因此，没有必要在 ACI 关键字中所用的 LDAP URL 中指定服务器的主机名或端口号。即使指定，系统也不会考虑 LDAP URL。有关 LDAP URL 的详细信息，请参阅附录 C “LDAP URL”。
- 授予代理权限时，不能授予用户对代理的目录管理员权限，也不能将代理权限授予目录管理员。

默认 ACI

安装目录服务器时，下列默认 ACI 将应用于 `userRoot` 数据库中所存储的目录信息：

- 用户可以修改自己在目录中的条目，但不能删除。其中的 `aci` 和 `nsroledn` 属性是不能修改的。
- 用户可以匿名访问目录，以执行搜索、比较和读取操作。
- 管理员（默认为 `uid=admin,ou=Administrators,ou=TopologyManagement,o=NetscapeRoot`）具有除代理权限之外的所有权限。
- “配置管理员”组的所有成员都具有除代理权限之外的所有权限。
- “目录管理员”组的所有成员都具有除代理权限之外的所有权限。
- SIE 组。

在目录中创建新数据库时，顶级条目具有上面所列的默认 ACI。

NetscapeRoot 子树具有自己的缺省 ACI 集：

- 在 NetscapeRoot 子树上，“配置管理员”组的所有成员都具有除代理权限之外的所有权限。
- 用户可以匿名访问 NetscapeRoot 子树，以执行搜索和读取操作。
- 组扩充。
- 所有通过身份验证的用户都对标识管理服务器的配置属性具有搜索、比较和读取的权限。

以下各部分将介绍如何修改这些缺省设置以满足您所在机构的实际需要。

手动创建 ACI

利用 LDIF 语句可以手动创建访问控制指令，然后可使用 `ldapmodify` 实用程序将其添加到目录树中。下列部分将详细介绍如何创建 LDIF 语句。

提示 LDIF ACI 语句可能非常复杂。但如果是为大量目录项设置访问控制，则使用 LDIF 相对于使用控制台在节省时间方面更具有优越性。

但为了熟悉 LDIF ACI 语句的用法，您可以使用 iPlanet Directory Server Console 来设置 ACI，然后单击访问控制编辑器上的 “手动编辑”按钮。这样会显示正确的 LDIF 语法。如果操作系统允许，您甚至可以将 LDIF 从访问控制编辑器复制并粘贴到 LDIF 文件中。

ACI 语法

`aci` 属性使用下列语法：

```
aci: (target)(version 3.0;acl "name";permission  bind_rules;)
```

其中

- *target* 指定要对其访问权加以控制的目标，包括条目、属性或条目和属性的集合等。目标可以是特异名称、一个或多个属性，或者是单个 LDAP 过滤器。目标是 ACI 的可选组件。
- `version 3.0` 为必需的字符串，用于标识 ACI 的版本。
- `"`*name*`"` 是 ACI 的名称。该名称可以是任何标识 ACI 的字符串。 ACI 名称为必需项。
- *permission* 专门说明所允许或拒绝的权限 （例如读取或搜索权）。
- *bind_rules* 指定用户为获取访问权而必须提供的凭证和绑定参数。绑定规则也可以用于专门拒绝某些用户或用户组的访问权。

每个目标可以有多个 “权限 – 绑定”规则对。这样可以有效地为给定的目标设置多种访问控制。例如：

target (*permission bind_rule*) (*permission bind_rule*) . . .

如果一个 ACI 语句中有多个 ACR，则语法格式为：

```
aci: (target)(version 3.0;acl "name"; permission  bind_rule;
 permission  bind_rule;  . . .  permission  bind_rule;)
```

ACI 示例

下面是一个完整的 LDIF ACI 示例：

```
aci: (target="ldap:///uid=bjensen,dc=siroe,dc=com")(targetattr=*)
 (version 3.0;acl "aci1";allow (write) userdn="ldap:///self";)
```

本例中， ACI 声明用户 bjensen 有权修改其目录项中的所有属性。

下列部分将详细介绍 ACI 各部分的语法。

定义目标

目标标识 ACI 所应用的对象。如果未指定目标， ACI 将应用于包含该 `aci` 属性的条目及其下面的条目。

目标可以是：

- 目录项或子树中的所有条目，如第 184 页上的 “以目录项为目标”中所述。
- 条目的属性，如第 186 页上的 “以属性为目标”中所述。
- 一组与指定的 LDAP 过滤器相匹配的条目或属性，如第 187 页上的 “使用 LDAP 过滤器确定条目或属性目标”中所述。
- 与指定的 LDAP 过滤器相匹配的属性值或值的组合，如第 188 页上的 “使用 LDAP 过滤器确定属性值目标”中所述。

目标的常规语法是：

(*keyword* = "*expression*")

(*keyword* != "*expression*")

其中：

- *keyword* 指示目标的类型
- 等于号 (=) 指示目标是 *expression* 中指定的对象，而不等于号 (!=) 则指示目标不是 *expression* 中指定的对象。
- *expression* 是用于确定目标的表达式

expression 上的引号 ("") 为必需项。 *expression* 的形式与提供的 *keyword* 有关。

下表列出了各个关键字及相关的表达式：

表 6-1 LDIF 目标关键字

关键字	有效的表达式	是否允许使用通配符?
`target`	ldap:///*distinguished_name*	是
`targetattr`	*attribute*	是
`targetfilter`	*LDAP_filter*	是
`targattrfilters`	*LDAP_operation:LDAP_filter*	是

无论何时都必须记住：在条目中放置 ACI 时，如果该条目不是叶条目，则 ACI 也将适用于其下面的所有条目。例如，如果条目的目标为 `ou=accounting,dc=siroe,dc=com`，则所设置的权限将应用于 siroe.com 目录树 accounting 分支中的所有条目。然而，它们不适用于 `uid=sarette,ou=people,dc=siroe,dc=com` 条目，因为该条目并不位于 accounting 目录树的下面。

以目录项为目标

要将目录项 （及下面的条目）作为目标，则必须使用 `target` 关键字。

`target` 关键字可以接受下列格式的值：

```
target="ldap:///distinguished_name"
```

它标识访问控制规则所应用的条目的特异名称。例如：

```
(target = "ldap:///uid=bjensen,dc=siroe,dc=com")
```

注意 如果应用访问控制的条目的 DN 中包含逗号，则必须用单个反斜杠 (\) 对逗号进行转义。例如：

```
(target="ldap:///uid=lfuentes,o=siroe.com
Bolivia\, S.A.")
```

利用 `target` 关键字将特异名称作为目标时，也可以使用通配符。通配符表示所有字符、字符串或子串都与该通配符匹配。模式匹配的基础是用通配符指定其它所有字符串。

以下是合法的通配符用法示例：

- `(target="ldap:///uid=*,dc=siroe,dc=com")`

  匹配整个 siroe.com 目录树中条目的 RDN 中有 `uid` 属性的所有条目。

- `(target="ldap:///uid=*Anderson,dc=siroe,dc=com")`

  匹配 siroe.com 节点下 `uid` 属性以 Anderson 结尾的所有条目。

- `(target="ldap:///*Anderson,dc=siroe,dc=com")`

  匹配 siroe.com 节点下条目中的 RDN 以 Anderson 结尾的所有条目。

因此，通配符可替代部分 DN。例如， `uid=andy*,dc=siroe,dc=com` 以整个 siroe.com 目录树下具有相应的 uid 属性的所有目录项为目标，而非只是 `dc=siroe, dc=com` 节点下的项。该目标将与以下两个表达式都匹配：

```
uid=andy,ou=eng,dc=siroe,dc=com
uid=andy,ou=marketing,dc=siroe,dc=com
```

允许使用多个通配符，例如 `uid=*,ou=*,dc=siroe,dc=com`。该示例匹配 siroe.com 目录树下其特异名称中包含 `uid` 和 `ou` 属性的所有条目。因此，以下条目匹配：

```
uid=fchen,ou=Engineering,dc=siroe,dc=com
uid=claire,ou=Engineering,ou=people,dc=siroe,dc=com
```

但以下条目不匹配：

```
uid=bjensen,dc=siroe,dc=com
ou=Engineering,dc=siroe,dc=com
```

注意 特异名称的后缀部分不能使用通配符。即：如果目录使用后缀 `c=US` 和 `c=GB`，则*不能*使用下列目标来指代这两个后缀：

`(target="ldap:///dc=siroe,c=*")`

也不能使用诸如 `uid=bjensen,dc=*.com` 等目标。

以属性为目标

除了以目录项为目标外，还能以目标条目中所含的一个或多个属性为目标。这在您想拒绝或允许对条目部分信息的访问的情况下非常有用。例如，可以允许仅能访问给定条目的通用名、姓氏和电话号码等属性。或者，也可拒绝访问敏感信息（例如口令）。

您可以指定目标等于或不等于特定的属性。所提供的属性无需在模式中定义。如果不进行模式检查，就可能会在初次设置目录服务时即实施访问控制策略，即使所创建的 ACL 并不在当前目录内容中应用。

要以属性为目标，请使用 `targetattr` 关键字并指定属性名称。`targetattr` 关键字使用下列语法：

```
(targetattr = "attribute")
```

利用下列语法结构下的 `targetattr` 关键字可以将目标指向多个属性：

```
(targetattr = "attribute1 || attribute2 ... || attributen")
```

其中：*attribute* 是要作为目标的属性的名称。

例如，要以条目的通用名、姓氏及 uid 属性为目标，则使用：

```
(targetattr = "cn || sn || uid")
```

在 `targetattr` 关键字中指定的属性将应用于作为 ACI 目标的条目及其下面的所有条目。即：如果以条目 `uid=bjensen,ou=Marketing,dc=siroe,dc=com` 上的口令属性为目标，则受 ACI 影响的将只有 `bjensen` 条目上的口令属性，因为该条目为叶条目。

但如果以目录树的分支点 `ou=Marketing,dc=siroe,dc=com` 为目标，则 ACI 将影响该分支点下包含口令属性的所有条目。

作为目标的属性包括已命名属性的所有子类型。例如，`(targetattr = "locality")` 还将以 `locality;fr` 为目标。也可以专门以子类型为目标，例如 `(targetattr = "locality;fr;quebec")`。

同时以条目和属性为目标

默认情况下，作为包含 targetattr 关键字的 ACI 目标的条目就是放置 ACI 的条目。即：如果将 ACI

```
aci: (targetattr = "uid")(access_control_rules;)
```

放到 ou=Marketing,dc=siroe,dc=com 条目上，则 ACI 将应用于整个 Marketing 子树。但也可以利用 target 关键字明确指定目标，如下所示：

```
aci: (target="ldap:///ou=Marketing, dc=siroe,dc=com")
 (targetattr="uid") (access_control_rules;)
```

指定 target 和 targetattr 关键字的顺序并不重要。

使用 LDAP 过滤器确定条目或属性目标

使用 LDAP 过滤器可以将符合一定标准的一组条目作为目标。为此，需将 targetfilter 关键字与 LDAP 过滤器配合使用。

targetfilter 关键字的语法是：

```
(targetfilter = "LDAP_filter")
```

其中：*LDAP_filter* 是一个标准 LDAP 搜索过滤器。有关过滤器语法的详细信息，请参阅第 462 页上的 “LDAP 搜索过滤器”。

例如，假设代表员工或承包人的所有条目都有一个薪酬状况和一个表示工作小时数的属性 （全职岗位工作小时数的百分比）。要以代表承包人或兼职员工的所有条目为目标，可以使用下列过滤器：

```
(targetfilter = "(|(employment=contractor)(fulltime<=99))")
```

注意 描述国际化值匹配规则的过滤器语法在 ACI 中不被支持。例如，以下目标过滤器无效：

```
(targetfilter = "(locality:fr:=<= Quebec)")
```

目标过滤器将选择全部条目作为 ACI 的目标。通过建立 targetfilter 和 targetattr 关键字的关联关系，可以创建应用于目标条目属性子集的 ACI。

下列 LDIF 示例允许工程管理组的成员修改 Engineering 业务类别中所有条目的 departmentNumber 和 manager 属性。本例中使用 LDAP 过滤器选择将具有 businessCategory 属性的所有条目均设置为 Engineering。

```
dn: dc=siroe,dc=com
objectClass: top
objectClass: organization
aci: (targetattr="departmentNumber || manager")
 (targetfilter="(businessCategory=Engineering)")
 (version 3.0; acl "eng-admins-write"; allow (write)
 groupdn ="ldap:///cn=Engineering Admins, dc=siroe,dc=com";)
```

提示 尽管在以目录中分布的条目和属性为目标时使用 LDAP 过滤器较为有用，但有时结果却不可预测，这是因为过滤器不会直接命名访问权受到监管的对象。添加或删除属性时，作为过滤后的 ACI 的目标条目组可能会发生变化。因此，如果在 ACI 中使用 LDAP 过滤器，则应核查作为目标的条目和属性是否正确，方法是在 `ldapsearch` 操作中使用同一过滤器。

使用 LDAP 过滤器确定属性值目标

可以使用访问控制来指定以特定的属性值为目标。这意味着如果某个属性的值满足 ACI 中定义的标准，则可以对该属性授予或拒绝权限。以属性值为基础而授予或拒绝访问权的 ACI 称为基于值的 ACI。

例如，可以向您所属机构中的所有用户授予修改各自条目中 `nsRoleDN` 属性的权限。但是，您可能还希望确保他们不会授予自身诸如 “顶级管理员” 等主要角色。此时，可以使用 LDAP 过滤器来检查属性值上的条件是否得到满足。

要创建基于值的 ACI，则必须使用 `targattrfilters` 关键字，其语法为：

```
(targattrfilters="add=attr1:F1 && attr2:F2... && attrn:Fn,
                  del=attr1:F1 && attr2:F2 ... && attrn:Fn")
```

其中：

- `add` 代表创建属性的操作。
- `del` 代表删除属性的操作。
- *attrx* 代表目标属性。
- *Fn* 代表仅应用于关联属性的过滤器。

创建条目时，如果过滤器应用于此新条目的属性，则该属性的各个实例都必须满足该过滤器。删除条目时，如果过滤器应用于此条目的属性，则该属性的各个实例也都必须满足该过滤器。

修改条目时，如果是添加属性，则必须满足应用于该属性的添加过滤器；如果是删除属性，则必须满足应用于该属性的删除过滤器。如果属性的各个值已在要替换的条目中存在，则必须同时满足添加和删除过滤器。

例如，请考虑下列属性过滤器：

```
(targattrfilters="add=nsroleDN:(!(nsRoleDN=cn=superAdmin)) &&
telephoneNumber:(telephoneNumber=123*)")
```

该过滤器可用于允许用户向自己的条目中添加任何角色 （nsRoleDN 属性），但 superAdmin 角色除外。它同时还允许用户添加带有 123 前缀的电话号码。

注意 从服务器控制台无法创建基于值的 ACI。

以单个目录项为目标

以单个目录项为目标并非简单明了，因为它违反了访问控制机制的设计原则。但可以通过下列方式来实现：

- 创建绑定规则，使绑定请求中的用户输入与目标条目中所存储的属性值相匹配。有关详细信息，请参阅第 200 页上的 “基于值匹配定义访问权限”。
- 使用 targetattr 和 targetfilter 关键字

可以使用 targetattr 关键字来指定仅在目标条目中存在，而不在目标条目下面任何条目中存在的属性。例如，如果想将目标确定为 ou=people,dc=siroe,dc=com，但该节点下未定义任何组织单元 (ou)，则可以指定 ACI 包含：

```
targetattr=ou
```

更为安全的方法是使用 targetfilter 关键字并明确指定只在该条目中存在的属性值。例如，目录服务器安装期间将创建以下 ACI：

```
aci: (targetattr="*")(targetfilter=(o=NetscapeRoot))(version 3.0;
 acl "Default anonymous access"; allow (read, search)
 userdn="ldap:///anyone";)
```

该 ACI 只能应用于 o=NetscapeRoot 条目。

这种方法的危险在于：目录树在将来可能会发生更改，因此必须要记住修改此 ACI。

定义权限

权限指定允许或拒绝的访问类型。您可以允许在目录中执行特定的操作，也可以拒绝特定的操作权限。可分配的各种操作称为*权限*。

设置权限涉及两部分：

- 允许或拒绝访问
- 分配权限

允许或拒绝访问

您可以明确地允许或拒绝对目录树的访问权限。有关何时允许、何时拒绝访问的详细指导，请参阅 *iPlanet Directory Server 部署指南*。

注意 从服务器控制台无法明确拒绝访问，而只能授予权限。

分配权限

权限详细说明了用户所能对目录数据执行的具体操作。可以允许或拒绝所有权限，也可以分配下列一个或多个权限：

Read。 读取权限，指示用户是否能读取目录数据。该权限仅适用于搜索操作。

Write。 写入权限，指示用户是否能修改条目，即添加、修改或删除*属性*。该权限适用于修改和 modrdn 操作。

Add。 添加权限，指示用户是否能创建*条目*。该权限仅适用于添加操作。

Delete。 删除权限，指示用户是否能删除*条目*。该权限仅适用于删除操作。

Search。 搜索权限，指示用户是否能搜索目录数据。用户必须具有搜索和读取权限才能查看搜索结果所返回的数据。该权限仅适用于搜索操作。

Compare。 比较权限，指示用户是否能将所提供的数据与目录中所存储的数据进行比较。通过比较权限，目录可以返回有关查询响应成功或失败的信息，但用户却无法看到条目或属性的值。该权限仅适用于比较操作。

Selfwrite。 自写权限，指示用户是否能针对组添加或删除自己的 DN。该权限仅适用于组管理。自写权限用于代理验证：它授予向组条目中添加或删除代理 DN （不是绑定用户的 DN）的权限。

Proxy。 代理权限，指示指定的 DN 是否能用另一个条目的权限来访问目标。您可以用目录中除目录管理员 DN 以外的任何用户 DN 来授予代理访问权限。但是，不能向目录管理员授予代理权限。在第 232 页上的 “代理验证 ACI 示例”中提供一个示例。有关代理访问权限的概述，请参阅 *iPlanet Directory Server 部署指南*。

All。 全部权限，指示指定的 DN 对目标条目具有全部权限 （读取、写入、搜索、删除、比较和自写），但代理权限*除外*。

权限的授予彼此独立。这意味着 （举例而言）被授予添加权限的用户可以创建条目，但如果未专门授予删除权限，则不能删除条目。因此，规划目录的访问控制策略时，必须确保所授予的权限对用户而言比较合理。例如，授予写入权限而不授予读取和搜索权限一般而言没有意义。

LDAP 操作所需的权限

本部分介绍根据要授权用户可以执行的 LDAP 操作类型而需要授予用户哪些权限。

添加条目：

- 对所要添加的条目授予添加权限。
- 对条目中各个属性的值授予写入权限。默认情况下将授予该权限，但可以利用 `targattrfilters` 关键字加以限制。

删除条目：

- 对所要删除的条目授予删除权限。
- 对条目中各个属性的值授予写入权限。默认情况下将授予该权限，但可以利用 `targattrfilters` 关键字加以限制。

修改条目中的属性：

- 对该属性类型授予写入权限。
- 对各个属性类型的值授予写入权限。默认情况下将授予该权限，但可以利用 `targattrfilters` 关键字加以限制。

修改条目的 RDN：

- 对该条目授予写入权限。
- 对新 RDN 中所用的属性类型授予写入权限。
- 如果希望对旧 RDN 授予删除权限，则对旧 RDN 中所用的属性类型授予写入权限。
- 对新 RDN 中所用属性类型的值授予写入权限。默认情况下将授予该权限，但可以利用 `targattrfilters` 关键字加以限制。

比较属性的值：

- 对属性类型授予比较权限。

搜索条目：

- 对搜索过滤器中所用的各个属性类型授予搜索权限。
- 对条目中所用的属性类型授予读取权限。

设置用于允许用户搜索目录的权限可通过示例进行更好的说明。请考虑下列 `ldapsearch` 操作：

```
% ldapsearch -L -h host -s suffix -b "uid=bjensen,dc=siroe,dc=com" \
             objectclass=* mail
```

下列 ACI 用于确定用户 `bkolics` 是否能被授予访问权：

```
aci: (targetattr = "mail")(version 3.0; acl "self access to mail";
 allow (read, search) userdn = "ldap:///self";)
```

由于该 ACI 未授予对 objectclass 属性的访问权，因此搜索结果列表为空。如果希望上述搜索操作成功，则必须将 ACI 修改为如下所示：

```
aci: (targetattr = "mail || objectclass")(version 3.0; acl "self
 access to mail"; allow (read, search) userdn = "ldap:///self";)
```

权限语法

在 ACI 语句中，权限的语法结构为：

`allow|deny (`*rights*`)`

其中 *rights* 是 1 至 8 个由逗号分隔、用括号括起的关键字列表。有效的关键字为 **`read`**、**`write`**、**`add`**、**`delete`**、**`search`**、**`compare`**、**`selfwrite`**、**`proxy`** 或 **`all`**。

在下例中，只要绑定规则被评估为 “真”，即允许进行读取、搜索及比较访问。

```
aci:  (target="ldap:///dc=siroe,dc=com") (version 3.0;acl "example";
 allow (read, search, compare) bind_rule;)
```

绑定规则

根据为目录所定义的 ACI，某些操作需要*绑定*到目录中。*绑定*的意思是：通过提供绑定 DN 和口令 （如果使用的是 SSL，则用证书）来向目录进行登录或验证。绑定操作中所提供的凭证及绑定环境决定了是否允许访问该目录。

ACI 中设置的各个权限都有对应的绑定规则，其中详细说明了所需的凭证和绑定参数。

绑定规则可以较为简单。例如，绑定规则可以只是声明：访问目录的人必须属于特定的组。绑定规则也可以较为复杂。例如，绑定规则可以声明：某个人必须属于特定的组，且必须在上午 8 点和下午 5 点之间从某台具有特定 IP 地址的计算机上登录。

绑定规则定义了能访问目录的人及访问的时间和地点。具体而言，绑定规则可以指定：

- 被授予访问权的用户、组和角色
- 实体必须从中进行绑定的位置
- 绑定的时间或日期
- 绑定期间必须使用的验证类型

此外，绑定规则还可以是这样一种复杂结构：利用布尔运算符将上述标准组合起来。有关详细信息，请参阅第 208 页上的 “使用布尔绑定规则”。

服务器根据与用于评估 LDAP 过滤器的逻辑相似的三值逻辑评估在 ACI 中使用的逻辑表达式，如 RFC 2251 *轻型目录访问协议 (v3)* 中说明。总而言之，这意味着如果表达式中的任何组件评估为 Undefined （例如，如果由于资源限制中止表达式的评估），则服务器可以正确处理这种情况：它不会因为在复杂布尔表达式中出现 Undefined 值而错误地授予访问权限。

绑定规则语法

是否允许访问与 ACI 绑定规则是否被评估为真有关。绑定规则使用下列两种格式之一：

keyword = "*expression*";

keyword != "*expression*";

其中等号 (=) 指示 *keyword* 和 *expression* 必须匹配时，绑定规则才为真；而不等号 (!=) 则指示 *keyword* 和 *expression* 必须不匹配时，绑定规则才为真。

注意 timeofday 关键字也支持不等于表达式 （<、 <=、 >、 >=）。这是唯一支持这些表达式的关键字。

在 *expression* 上的引号 ("") 及定界用的分号 (;) 为必需项。可以使用的表达式与关联的 *keyword* 有关。

下表列出了各个关键字及相关的表达式。它同时还指示表达式中是否允许使用通配符。

表 6-2 LDIF 绑定规则关键字

关键字	有效的表达式	使用允许使用通配符?
userdn	ldap:///*distinguished_name* ldap:///all ldap:///anyone ldap:///self ldap:///parent ldap:///*suffix*??sub?(*filter*)	是，仅限于 DN
groupdn	ldap:///*DN* || *DN*	否
roledn	ldap:///*DN* || *DN*	否
userattr	*attribute*#*bindType* 或 *attribute*#*value*	否
ip	*IP_address*	是
dns	*DNS_host_name*	是
dayofweek	sun mon tue wed thu fri sat	否
timeofday	0 - 2359	否
authmethod	none simple ssl sasl *authentication_method*	否

下列部分将详细说明各个关键字的绑定规则语法。

定义用户访问权 — userdn 关键字

用户访问权可用 userdn 关键字定义。 userdn 关键字要求下列格式的一个或多个有效特异名称：

```
userdn = "ldap:///dn [|| ldap:///dn]...[||ldap:///dn]"
```

其中 *dn* 可以是 DN 或是表达式 anyone、 all、 self 或 parent 之一：

`userdn = "ldap:///anyone"` — 定义匿名访问

`userdn = "ldap:///all"` — 定义常规访问

`userdn = "ldap:///self"` — 定义自访问

`userdn = "ldap:///parent"` — 定义父项访问

userdn 关键字也可表示为 LDAP 过滤器的形式：

`ldap:///`*suffix*`??sub?(`*filter*`)`

注意 如果 DN 中包含逗号，则必须在逗号的前面加一个反斜杠 (\) 进行字符转义。

匿名访问 （anyone 关键字）

授予对目录进行匿名访问的权利，意味着任何人都能在不提供绑定 DN 或口令的情况下访问该目录，而不管绑定环境如何。可以将匿名访问限制为特定类型的访问（例如读取或搜索），或者限定到目录中特定的子树或个别条目上。

在服务器控制台上，将通过访问控制编辑器定义匿名访问。请参阅第 209 页上的“从控制台创建 ACI”。

常规访问 （all 关键字）

可以使用绑定规则来指示某个权限适用于已成功绑定到目录上的任何人，即所有经过验证的用户。这样即允许进行常规访问，而同时又可防止匿名访问。

在服务器控制台上，将通过访问控制编辑器定义常规访问。有关详细信息，请参阅第 209 页上的 “从控制台创建 ACI”。

自访问 （self 关键字）

指定用户被授予或被拒绝对自己条目的访问权。这种情况下，如果绑定 DN 与目标条目的 DN 相匹配，就会授予或拒绝访问。

在服务器控制台上，将通过访问控制编辑器设置自访问。有关详细信息，请参阅第 209 页上的 “从控制台创建 ACI”。

父项访问 （parent 关键字）

指定只有当用户的绑定 DN 是目标条目的父项时，才会授予或拒绝访问权。

使用服务器控制台无法设置父项访问控制。

LDAP URL

将 URL 与过滤器一起使用可以在 ACI 中动态确定用户目标，如下所示：

```
userdn = "ldap:///<suffix>??sub?(filter)"
```

例如，基于下列 URL， siroe.com 目录树财务和工程分支中的所有用户都将被动态地授予或拒绝对目标资源的访问权：

```
userdn = "ldap:///dc=siroe,dc=com??sub?(|(ou=engineering)(ou=accounting))"
```

注意 请勿在 LDAP URL 中指定主机名或端口号。LDAP URL 始终应用于本地服务器。

有关 LDAP URL 的详细信息，请参阅附录 C “LDAP URL”。

通配符

也可以利用通配符 (*) 来指定一组用户。例如，指定用户 DN 为 `uid=u*,dc=siroe,dc=com` 时，将指示只有具有以字母 `u` 开头的绑定 DN 的用户才会根据所设权限而被允许或拒绝访问。

在服务器控制台上，将通过访问控制编辑器来设置用户访问权。有关详细信息，请参阅第 209 页上的 “从控制台创建 ACI”。

示例

本部分中包含 `userdn` 语法的示例。

包含 LDAP URL 的 Userdn 关键字：

```
userdn = "ldap:///uid=*,dc=siroe,dc=com";
```

如果用户利用指定模式的任何特异名称而绑定到目录上，绑定规则即被评估为“真”。例如，下列两种绑定 DN 均被评估为真：

```
uid=ssarette,dc=siroe,dc=com
uid=tjaz,ou=Accounting,dc=siroe,dc=com
```

而下列绑定 DN 则评估为 “假”：

```
cn=Babs Jensen,dc=siroe,dc=com
```

包含 LDAP URL 逻辑 “或”的 Userdn 关键字：

```
userdn="ldap:///uid=bj,c=siroe.com ||
ldap:///uid=kc,dc=siroe,dc=com";
```

如果客户机绑定为所提供的两个特异名称之一，绑定规则即被评估为真。

不包括特定 LDAP URL 的 Userdn 关键字：

```
userdn != "ldap:///uid=*,ou=Accounting,dc=siroe,dc=com";
```

如果客户机未绑定为 accounting 子树中基于 UID 的特异名称，绑定规则即被评估为真。只有当目标条目未在目录树 accounting 分支的下面时，该绑定规则才有意义。

包含 self 关键字的 Userdn 关键字：

```
userdn = "ldap:///self";
```

如果用户所访问的条目是由用户绑定到目录所用的 DN 表示的，则绑定规则被评估为真。即：如果用户已绑定为 `uid=ssarette,dc=siroe,dc=com`，而该用户又试图对 `uid=ssarette,dc=siroe,dc=com` 条目执行操作，则绑定规则被评估为真。

例如，如果希望授予 siroe.com 目录树中所有用户对其 `userPassword` 属性的写入权限，则可在 `dc=siroe,dc=com` 节点上创建以下 ACI：

```
aci: (targetattr = "userPassword") (version 3.0;
 acl "write-self"; allow (write) userdn = "ldap:///self";)
```

包含 all 关键字的 Userdn 关键字：

```
userdn = "ldap:///all";
```

对于任何有效的绑定 DN，绑定规则均被评估为真。为评估为真，用户必须在绑定操作期间提供有效的特异名称和口令。

例如，如果希望将对整个目录树的读取权限授予所有经过验证的用户，则在 `dc=siroe,dc=com` 节点上创建以下 ACI：

```
aci: (version 3.0; acl "all-read"; allow (read)
 userdn="ldap:///all";)
```

包含 anyone 关键字的 Userdn 关键字：

```
userdn = "ldap:///anyone";
```

对于任何人而言，绑定规则都将被评估为真；使用该关键字可以提供对目录的匿名访问。

例如，如果想允许匿名读取和搜索整个 siroe.com 目录树，请在 dc=siroe,dc=com 节点上创建以下 ACI：

```
aci: (version 3.0; acl "anonymous-read-search";
 allow (read, search) userdn = "ldap:///anyone";)
```

包含 parent 关键字的 Userdn 关键字：

```
userdn = "ldap:///parent";
```

如果绑定 DN 是目标条目的父项，则绑定规则将被评估为真。

例如，如果想授予对各个用户子条目的写入权限，请在 dc=siroe,dc=com 节点上创建以下 ACI：

```
aci: (version 3.0; acl "parent access";
 allow (write) userdn="ldap:///parent";)
```

如果用户属于 engineering 或 sales 子树，则绑定规则将被评估为真。

定义组访问权 — groupdn 关键字

有些特定组的成员可以访问目标资源。这称为*组访问*。组访问权是用 groupdn 关键字定义的，可指定在用户利用属于特定组的 DN 进行绑定时，是授予还是拒绝对目标条目的访问。

groupdn 关键字要求一个或多个具有下列格式的有效特异名称：

```
groupdn="ldap:///dn [|| ldap:///dn]...[|| ldap:///dn]"
```

如果绑定 DN 属于已命名的组，则绑定规则将被评估为真。

注意 如果 DN 中包含逗号，则必须用反斜杠 (\) 对逗号进行转义。

在服务器控制台上，可以通过访问控制编辑器来定义特定的组。有关详细信息，请参阅第 209 页上的 “从控制台创建 ACI”。

示例

本部分中包含 groupdn 语法的示例。

包含 LDAP URL 的 Groupdn 关键字：

```
groupdn = "ldap:///cn=Administrators,dc=siroe,dc=com";
```

如果绑定 DN 属于管理员组，则绑定规则将被评估为真。如果想授予管理员组对整个目录树的写入权限，请在 dc=siroe,dc=com 节点上创建以下 ACI：

```
aci: (version 3.0; acl "Administrators-write"; allow (write)
 groupdn="ldap:///cn=Administrators,dc=siroe,dc=com";)
```

包含 LDAP URL 逻辑 “或” 的 Groupdn 关键字：

```
groupdn = "ldap:///cn=Administrators,dc=siroe,dc=com" ||
"ldap:///cn=Mail Administrators,dc=siroe,dc=com";
```

如果绑定 DN 属于管理员组或邮件管理员组，则绑定规则将被评估为真。

定义角色访问权 — roledn 关键字

某些特定角色的成员可以访问目标资源。这称为*角色访问*。角色访问权是用 roledn 关键字定义的，可指定在用户利用属于特定组的 DN 进行绑定时，是授予还是拒绝对目标条目的访问。

roledn 关键字要求下列格式的一个或多个有效特异名称：

```
roledn = "ldap:///dn [|| ldap:///dn]... [|| ldap:///dn]"
```

如果绑定 DN 属于指定的角色，则绑定规则将被评估为真。

注意 如果 DN 中包含逗号，则必须用反斜杠 (\) 对逗号进行转义。

roledn 关键字的语法结构及用法与 groupdn 关键字的相同。

基于值匹配定义访问权限

您可以设置绑定规则，指定用于绑定到目录的条目的属性值必须匹配目标条目的属性值。

例如，可以指定绑定 DN 必须匹配用户条目 `manager` 属性的 DN 时，才会应用 ACI。这种情况下，只有用户的管理者才能访问该条目。

本例即基于 DN 匹配。但是，可以将绑定中所用条目的任何属性与目标条目相匹配。例如，可以创建这样的 ACI：它允许任何属性 `favoriteDrink` 为 “beer” 的用户读取其它具有相同 `favoriteDrink` 值的用户的所有条目。

使用 userattr 关键字

`userattr` 关键字可指定用于绑定的条目与目标条目之间哪些属性值必须匹配。

您可以指定：

- 用户 DN
- 组 DN
- 角色 DN
- LDAP 过滤器 （位于 LDAP URL 中）
- 任何属性类型

`userattr` 关键字的 LDIF 语法如下所示：

```
userattr = "attrName#bindType"
```

或者，如果使用的属性类型要求用户 DN、组 DN、角色 DN 或 LDAP 过滤器以外的值：

```
userattr = "attrName#attrValue"
```

其中：

- *attrName* 是用于值匹配的属性的名称
- *bindType* 是 `USERDN`、 `GROUPDN`、 `LDAPURL` 之一
- *attrValue* 是任何代表属性值的字符串

注意 千万不要将服务类 (CoS) 定义生成的属性与 `userattr` 关键字一起使用。如果 ACI 包含与 CoS 生成的属性值有关的绑定规则，则 ACI 将不起作用。

下列部分将提供各种绑定类型下 userattr 关键字的示例：

USERDN 绑定类型的示例

下面示例中的 userattr 关键字与基于用户 DN 的绑定相关联：

```
userattr = "manager#USERDN"
```

如果绑定 DN 与 manager 属性的值相匹配，则绑定规则将被评估为真。您可以用此允许用户的管理者修改员工的属性。该机制只有在目标条目中的 manager 属性表示为完整的 DN 时才有效。

下例授予管理者对其员工条目完全的访问权：

```
aci: (target="ldap:///dc=siroe,dc=com")(targetattr=*)(version 3.0;
 acl "manager-write"; allow (all) userattr = "manager#USERDN";)
```

GROUPDN 绑定类型的示例

下面示例中的 userattr 关键字与基于组 DN 的绑定相关联：

```
userattr = "owner#GROUPDN"
```

如果绑定 DN 是目标条目 owner 属性中所指定的组的成员，则绑定规则将被评估为真。例如，使用该机制可以允许组来管理员工的状态信息。只要所用的属性中包含组条目的 DN，即可使用 owner 以外的其它属性。

所指向的组可以是动态组，而组的 DN 则可以位于数据库的任何后缀下。但是，服务器对此类 ACI 的评估需要占用大量的资源。

如果使用的是静态组且与目标条目位于同一后缀下，则可以使用下列表达式：

```
userattr = "ldap:///dc=siroe,dc=com?owner#GROUPDN"
```

本例中，组条目位于 dc=siroe,dc=com 后缀下。相对上例而言，服务器可以更为快速地处理此类语法。

ROLEDN 绑定类型的示例

下面示例中的 `userattr` 关键字与基于角色 DN 的绑定相关联：

```
userattr = "siroeEmployeeReportsTo#ROLEDN"
```

如果绑定 DN 属于目标条目 `siroeEmployeeReportsTo` 属性中指定的角色，则绑定规则将被评估为真。例如，如果您为公司中的所有管理者都创建了嵌套角色，即可使用该机制向各级管理者授予对比自己级别低的员工进行信息访问的权限。

注意 本例假定已向模式中添加了 `siroeEmployeeReportsTo` 属性，且所有员工条目中都包含该属性。同时，它还假定该属性的值为角色条目的 DN。

有关设计模式的信息，请参阅 *iPlanet Directory Server 部署指南*。有关向模式中添加属性的信息，请参阅第 314 页上的 “创建属性”。

角色的 DN 可以位于数据库中任何后缀的下面。此外，如果使用的是已过滤的角色，则此类 ACI 的评估会占用大量的服务器资源。

如果使用的是静态角色定义，且角色条目位于与目标条目相同的后缀下，则可以使用下列表达式：

```
userattr = "ldap:///dc=siroe,dc=com?employeeReportsTo#ROLEDN"
```

本例中，角色条目位于 `dc=siroe,dc=com` 后缀下。相对上例而言，服务器可以更为快速地处理此类语法。

LDAPURL 绑定类型的示例

下面示例中的 `userattr` 关键字与基于 LDAP 过滤器的绑定相关联：

```
userattr = "myfilter#LDAPURL"
```

如果绑定 DN 与目标条目 *myfilter* 属性中指定的过滤器相匹配，则绑定规则将被评估为真。 *myfilter* 属性可被替换为任何包含 LDAP 过滤器的属性。

任意属性值的示例

下面示例中的 `userattr` 关键字与基于任意属性值的绑定相关联：

```
userattr = "favoriteDrink#Beer"
```

如果绑定 DN 和目标 DN 都包含值为 **`Beer`** 的 `favoriteDrink` 属性，则绑定规则将被评估为真。

继承性与 userattr 关键字的配合使用

使用 `userattr` 关键字来建立用于绑定的条目与目标条目之间的关联时， ACI 将仅应用于所指定的目标，而不适用于其下面的条目。有些情况下，您可能希望将 ACI 的应用范围向目标条目以下扩展几级。使用 parent 关键字并指定目标向下继承 ACI 的级别数，即有可能实现上述要求。

将 `userattr` 关键字与 `parent` 关键字配合使用时，语法结构如下所示：

```
userattr = "parent[inheritance_level].attrName#bindType"
```

或者，如果使用的属性类型要求用户 DN、组 DN、角色 DN 或 LDAP 过滤器以外的值：

```
userattr = "parent[inheritance_level].attrName#attrValue"
```

其中：

- *inheritance_level* 是一个由逗号分隔的列表，指示目标下继承 ACI 的级别数。目标条目下可以包含五级 `[0,1,2,3,4]`； 0 指示目标条目。
- *attribute* 是作为 `userattr` 或 `groupattr` 关键字目标的属性。
- *bindType* 可以是 `USERDN`、 `GROUPDN`、 `LDAPURL` 之一。

例如：

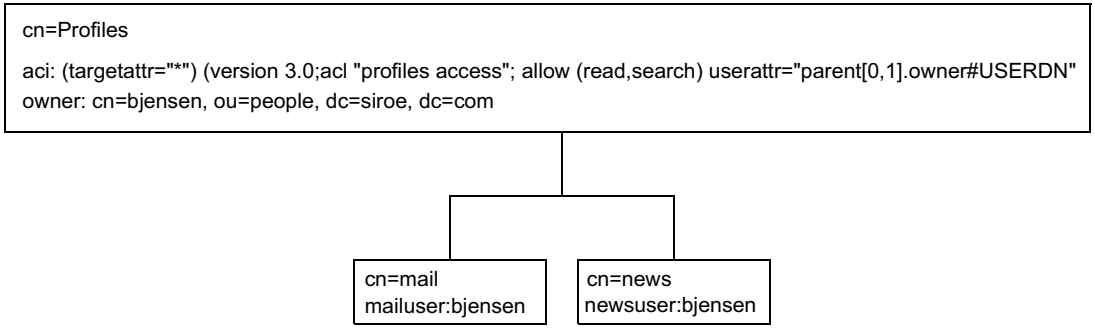
```
userattr = "parent[0,1].manager#USERDN"
```

如果绑定 DN 与目标条目的 manager 属性相匹配，则绑定规则将被评估为真。当绑定规则评估为真时，所授予的权限将应用于目标条目*及*所有紧跟在它下面的条目。

userattr 继承性的示例

下图中的示例指示允许用户 `bjensen` 读取和搜索 `cn=Profiles` 条目及包含 `cn=mail` 和 `cn=news` 的第一级子项，从而允许该用户搜索自己的邮件和新 ID。

图 6-1 将继承性与 `userattr` 关键字配合使用



如果本例中未使用继承性，则必须使用下列方法之一来获得相同的结果：

- 为用户 bjensen 明确设置对目录的 cn=Profiles、 cn=mail 和 cn=news 条目的读取和搜索权。
- 将值为 bjensen 的 owner 属性添加到 cn=mail 和 cn=news 条目中，然后将以下 ACI 添加到 cn=mail 和 cn=news 条目中。

```
aci: (targetattr="*") (version 3.0; acl "profiles access"; allow
 (read,search) userattr="owner#USERDN";)
```

使用 userattr 关键字授予添加权限

如果是将 userattr 关键字与 **all** 或 **add** 权限配合使用，就可能会发现服务器的行为并非自己所期望的那样。典型的表现是：当在目录中创建新条目时， iPlanet Directory Server 将对所创建的条目进行访问权的评估，但并不对父项进行评估。但是，对于使用 userattr 关键字的 ACI 而言，这种行为会造成安全漏洞。为避免这种情况，需要改变服务器的常规行为。

请考虑下例：

```
aci: (target="ldap:///dc=siroe,dc=com")(targetattr=*) (version 3.0;
 acl "manager-write"; allow (all) userattr = "manager#USERDN";)
```

该 ACI 授予管理者对向其报告的员工条目的全部权限。但是，由于访问权是在所创建的条目上进行评估的，因此此类 ACI 也会允许任意员工创建其 manager 属性被设为自身 DN 的条目。例如，心怀不满的员工 Joe (cn=Joe,ou=eng,dc=siroe,dc=com) 可能想在目录树的人力资源分支中创建一个条目，从而使用 （或滥用）授予人力资源部门员工的特权。

他可以创建下列条目：

```
dn: cn= Trojan Horse,ou=Human Resources,dc=siroe,dc=com
objectclass: top
...
cn: Trojan Horse
manager: cn=Joe,ou=eng,dc=siroe,dc=com
```

为避免出现此类危险， ACI 评估过程不会在级别 0 上 （即向条目自身）授予添加权限。但是，可以使用 parent 关键字在现有条目的下面授予添加权限。此时必须为添加权限指定父项以下的级别数。例如，下列 ACI 允许将子项添加到具有与绑定 DN 相匹配的 manager 属性的 dc=siroe,dc=com 中的任何条目中：

```
aci: (target="ldap:///dc=siroe,dc=com")(targetattr=*)
 (version 3.0; acl "parent-access"; allow (add)
 userattr = "parent[0,1].manager#USERDN";)
```

该 ACI 可确保仅将添加权限授予其绑定 DN 与父项的 manager 属性相匹配的用户。

定义从特定 IP 地址进行访问时的访问权限

利用绑定规则，可以指示绑定操作必须从特定的 IP 地址发出。这通常用于强制所有目录更新都从给定的计算机或网络域中发生。

设置基于 IP 地址的绑定规则时所用的 LDIF 语法结构如下所示：

```
ip = "IP_address" 或 ip != "IP_address"
```

IP 地址必须为点表示法。可以使用通配符 (*) 来包含多台计算机。例如，下列字符串有效：

```
ip = "12.123.1.*";
```

如果访问目录的客户机位于已命名的 IP 地址处，则绑定规则将被评估为真。这一点对于只允许从特定的子网或计算机来访问某类目录的情况较为有用。

例如，可以使用通配符 IP 地址 12.3.45.* 来指定特定的子网，或者使用 123.45.6.*+255.255.255.115 来指定子网掩码。

在服务器控制台上，通过访问控制编辑器可以指定 ACI 所应用的特定计算机。有关详细信息，请参阅第 209 页上的 “从控制台创建 ACI”。

定义从特定域进行访问时的访问权限

绑定规则可以指定绑定操作必须从特定的域或主机发出。这通常用于强制所有目录更新都从给定的计算机或网络域中发生。

设置基于 DNS 主机名的绑定规则时所用的 LDIF 语法结构如下所示：

```
dns = "DNS_Hostname" 或 dns != "DNS_Hostname"
```

警告 dns 关键字要求计算机上使用的命名服务为 DNS。如果命名服务不是 DNS，则应使用 ip 关键字。

dns 关键字要求完全限定的 DNS 域名。如果授予主机访问权而不指定域，就会造成潜在的安全隐患。例如，系统允许使用下列表达式，但我们建议不要使用：

```
dns = "legend.eng";
```

应使用完全限定的名称，例如：

```
dns = "legend.eng.siroe.com";
```

dns 关键字允许使用通配符。例如：

```
dns = "*.siroe.com";
```

如果访问目录的客户机位于已命名的域中，则绑定规则将被评估为真。这一点对于仅允许从特定的域进行访问而言较为有用。注意：如果系统使用的命名服务不是 DNS，则通配符将无法正常工作。这种情况下，如果想限制对特定域的访问权，请使用 ip 关键字，说明见第 205 页上的 “定义从特定 IP 地址进行访问时的访问权限”。

定义特定时间或日期的访问权限

使用绑定规则可以指定只能在特定时间或特定日期进行绑定。例如，可以设置仅允许在周一至周五上午 8 点到下午 5 点之间进行访问的规则。赋予访问权限的时间是目录服务器上的时间，而非客户机上的时间。

设置基于时间的绑定规则的 LDIF 语法如下所示：

```
timeofday operator "time"
```

其中 *operator* 可以是下列符号之一：等于号 (=)、不等于号 (!=)、大于号 (>)、大于等于号 (>=)、小于号 (<) 或小于等于号 (<=)。

关键字 `timeofday` 要求时间以小时和分钟表示，采用 24 小时制时钟 （0 至 2359）。

注意 赋值的时间是服务器上的时间，而非客户机上的时间。

设置基于日期的 LDIF 绑定规则的语法如下所示：

```
dayofweek = "day1, day2 ..."
```

dayofweek 关键字的可能取值是英文的三字母星期缩写：sun、 mon、 tue、 wed、 thu、 fri、 sat。

示例

下面是 `timeofday` 及 `dayofweek` 语法的示例：

```
timeofday = "1200";
```

如果客户机恰好在中午访问目录，则绑定规则将被评估为真。

```
timeofday != "0100";
```

如果客户机在除上午 1 点之外的时间访问目录，则绑定规则将被评估为真。

```
timeofday > "0800";
```

如果客户机在上午 8 点以后的任何时间访问目录，则绑定规则将被评估为真。

```
timeofday < "1800";
```

如果客户机在下午 6 点之前的任何时间访问目录，则绑定规则将被评估为真。

```
timeofday >= "0800";
```

如果客户机是在上午 8 点或以后的任何时间访问目录，则绑定规则将被评估为真。

```
timeofday <= "1800";
```

如果客户机在下午 6 点或之前的任何时间访问目录，则绑定规则将被评估为真。

```
dayofweek = "Sun, Mon, Tue";
```

如果客户机在周日、周一或周二访问目录，则绑定规则将被评估为真。

基于验证方法定义访问权限

您可以设置这样的绑定规则：声明客户机必须使用特定的验证方法绑定到目录上。可用的身份验证方法包括：

- **None** — 无须身份验证。此为缺省值。它代表匿名访问。
- **Simple** — 客户机必须提供用户名和口令才能绑定到目录。
- **SSL** — 客户机必须通过 “安全套接层” (SSL) 或 “传输层安全协议” (TLS) 连接来绑定到目录上。

  对于 SSL 的情况，连接是在第二个 LDAPS 端口上建立的；对于 TLS 而言，连接则是通过 “启动 TLS” 操作来建立的。两种情况下都必须提供证书。有关设置 SSL 的信息，请参阅第 11 章 “管理 SSL”。
- **SASL** — 客户机必须通过 “简单验证和安全层” (SASL) 连接来绑定到目录上。注意：iPlanet Directory Server 不提供 SASL 模块。

通过访问控制编辑器无法设置基于验证的绑定规则。

设置基于验证方法的绑定规则的 LDIF 语法结构如下所示：

```
authmethod = "authentication_method"
```

其中 *authentication_method* 取 **`none`**、 **`simple`**、 **`ssl`** 或 "**`sasl`** *sasl_mechanism*"。

示例

下面是 authmethod 关键字的示例：

```
authmethod = "none";
```

绑定规则评估期间将不进行验证。

```
authmethod = "simple";
```

如果客户机使用用户名和口令访问目录，则绑定规则将被评估为真。

```
authmethod = "ssl";
```

如果客户机使用证书通过 LDAPS 进行目录验证，则绑定规则将被评估为真。如果客户机使用简单 (simple) 验证方法 （绑定 DN 和口令）通过 ldaps 进行验证，则绑定规则将不会被评估为真。

```
authmethod = "sasl DIGEST-MD5";
```

如果客户机使用 SASL DIGEST-MD5 机制访问目录，则绑定规则将被评估为真。另一种受支持的 SASL 机制为 EXTERNAL。

使用布尔绑定规则

绑定规则可以是使用布尔表达式 AND （和）、 OR （或）和 NOT （非）来精确设置访问规则的复杂表达式。使用服务器控制台无法创建布尔绑定规则。您必须创建 LDIF 语句。

布尔绑定规则的 LDIF 语法结构如下所示：

bind_rule [*boolean*] [*bind_rule*] [*boolean*] [*bind_rule*] . . . ;)

例如，如果绑定 DN 是管理员组或邮件管理员组的成员，且客户机是在 siroe.com 域中运行的，则下列绑定规则将被评估为真：

```
(groupdn = "ldap:///cn=administrators,dc=siroe,dc=com" or
groupdn = "ldap:///cn=mail administrators,dc=siroe,dc=com" and
dns = "*.siroe.com";)
```

尾随的分号 (;) 为必需的定界符，必须位于最后一条绑定规则的后面。

布尔表达式的评估顺序如下所示：

- 首先是最内侧的括号表达式，并依次向外
- 所有表达式都按从左向右的顺序进行
- NOT 先于 AND 或 OR 运算符

布尔表达式 OR 和 AND 运算符之间优先性相同。

请考虑下列布尔绑定规则：

（*bind_rule_A*） OR （*bind_rule_B*）

（*bind_rule_B*） OR （*bind_rule_A*）

由于布尔表达式是从左向右评估的，因此，在第一种情况下，绑定规则 A 将先于绑定规则 B 被评估；而在第二种情况下，绑定规则 B 则先于绑定规则 A 被评估。

但是，布尔运算符 NOT *先于*布尔运算符 OR 和 AND 的评估。因此，在下例中：

（*bind_rule_A*） AND NOT （*bind_rule_B*）

绑定规则 B 将先于绑定规则 A 被评估，而此时将忽略自左向右的评估规则。

从控制台创建 ACI

您可以使用 iPlanet Directory Server Console 来查看、创建、编辑和删除目录的访问控制指令。本部分提供下列内容的一般说明：

- 显示访问控制编辑器
- 查看当前 ACI
- 创建新 ACI
- 编辑 ACI
- 删除 ACI

有关 iPlanet Directory Server Console 安全策略中常用访问控制规则的集合及利用 iPlanet Directory Server 创建访问控制规则的循序渐进式说明，请参阅第 214 页上的 “访问控制用法示例”。

在可视编辑模式下，访问控制编辑器不允许创建某些较复杂的 ACI。尤其要说明的是，在访问控制编辑器中无法执行下列操作：

- 拒绝访问 （请参阅第 192 页上的 “权限语法”）
- 创建基于值的 ACI （请参阅第 188 页上的 “使用 LDAP 过滤器确定属性值目标”）
- 定义父项访问 （请参阅第 196 页上的 “父项访问 （parent 关键字）”）
- 创建包含布尔绑定规则的 ACI （请参阅第 208 页上的 “使用布尔绑定规则”）
- 创建使用下列关键字的 ACI：roledn、userattr、authmethod （一般而言）

提示 在访问控制编辑器中，可以随时单击 “手动编辑”按钮来检查 LDIF 上所示的、通过图形界面完成的更改。

显示访问控制编辑器

1. 启动 iPlanet Directory Server Console。利用诸如目录管理员 （具有为目录所配置的 ACI 的写入权限）等特权用户的绑定 DN 和口令进行登录。

   有关说明，请参阅第 26 页上的 “使用 iPlanet Directory Server Console”。

2. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。

3. 右键单击导航树中要设置访问控制的条目，然后从弹出菜单中选择 “设置访问权限”。或者，也可以突出显示该条目，然后从 “对象”菜单中选择 “设置访问权限”。

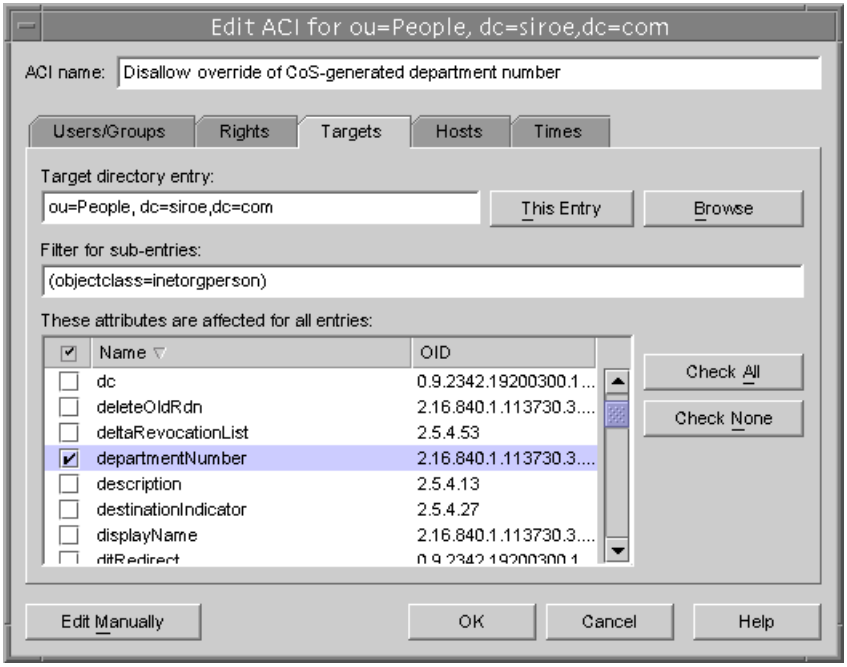
   下图显示屏幕上的访问控制管理对话框。该对话框列出在所选条目上定义的所有 ACI 的说明，并允许编辑或删除这些 ACI 和创建新的 ACI。

   选中 “显示继承的 ACI”复选框还将列出由所选条目的父项定义并适用于该条目的所有 ACI。但是，不能编辑或删除继承的 ACI，必须在定义它们的条目中对它们进行管理。

图 6-2 访问控制管理对话框

4. 单击“新建”以在所选对象及其整个子目录树上定义新的访问权限。此时显示如下图所示的访问控制编辑器。

图 6-3 访问控制编辑器对话框



该对话框顶部的 ACI 名称是对在访问控制管理对话框中出现的 ACI 的说明。指定一个描述性的 ACI 名称将大大方便管理整个目录范围的 ACI，尤其是在查看叶条目上的继承的 ACI 的时候。

访问控制编辑器的选项卡用于指定被授予或被拒绝访问权限的用户、将被访问或限制的目标，以及诸如所允许使用的主机名和操作次数等高级参数。有关访问控制选项卡各个字段的详细信息，请参阅在线帮助。

查看当前 ACI

如果想查看应用于目录中特定子树的 ACI，请执行下列步骤：

1. 在 “目录”选项卡上，右键单击子树中的顶级条目，然后从弹出菜单中选择 “设置访问权限”。

   此时显示访问控制管理器窗口。其中包含属于该条目的 ACI 列表。

2. 如果想显示应用于该条目的 ACI 完整列表，请选中 “显示继承的 ACI”。

创建新 ACI

要创建新 ACI：

1. 显示访问控制编辑器。

   有关该任务的说明，请参见第 210 页上的 “显示访问控制编辑器”。

   如果显示的视图与第 211 页的图 6-3 的不同，则单击 “可视编辑”按钮。

2. 在 “ACI 名称”文本框中键入名称，从而命名 ACI。

   名称可以是任何用于唯一标识该 ACI 的字符串。如果不输入名称，服务器好就会使用**未命名的 ACI**。

3. 在 “用户/组”选项卡上，选择要授予其访问权的用户，方法是突出显示 “所有用户”，或者单击 “添加”按钮以搜索要为用户添加的目录。

   在 “添加用户和组”窗口中：

   a. 从下拉列表中选择搜索区域，在 “搜索”字段中输入搜索字符串，然后单击 “搜索”按钮。

      搜索结果将显示在窗口下方的列表中。

   b. 在搜索结果列表中突出显示所需的条目，然后单击 “添加”按钮，从而将其添加到具有访问权限的条目列表中。

   c. 单击 “确定”以关闭 “添加用户和组”窗口。

      所选的条目现在将在 ACI 编辑器的 “用户/组”选项卡中列出。

4. 在访问控制编辑器中，单击 “权限”选项卡，然后使用复选框选择要授予的权限。

5. 单击 “目标”选项卡，然后单击 “此条目”，从而显示作为 ACI 目标的节点。

   您可以更改目标 DN 的值，但新的 DN 必须为所选条目的直接或间接子项。

   如果不希望该节点下子树中的每个条目都成为 ACI 的目标，则必须在 “子项的过滤器”字段中输入过滤器。

   此外，通过在属性列表中选择要作为目标的属性，还可以将 ACI 的范围仅限于某些属性。

6. 单击 “主机”选项卡，然后单击 “添加”按钮，从而显示添加主机过滤器对话框。

   您可以指定主机名或 IP 地址。如果指定的是 IP 地址，则可以使用通配符 (*)。

7. 单击 “次数”选项卡，显示允许访问的时间表。

   缺省情况下随时都可进行访问。单击并在表格上拖动光标，即可更改访问时间。但您不能选择不连续的时间块。

8. 编辑完 ACI 后，单击 “确定”。

   此时将关闭 ACI 编辑器，而新的 ACI 将在 ACI 管理器窗口中列出。

注意 创建 ACI 的过程中，可以随时单击 “手动编辑”按钮来显示对应于输入的 LDIF 语句。您可以修改该语句，但更改结果却不一定在图形界面上显示出来。

编辑 ACI

要编辑 ACI：

1. 在 “目录”选项卡上，右键单击子树中的顶级条目，然后从弹出菜单中选择 “设置访问权限”。

   此时显示访问控制管理器窗口。其中包含属于该条目的 ACI 列表。

2. 在访问控制管理器窗口中，突出显示要编辑的 ACI，然后单击 “编辑”。

   此时显示访问控制编辑器。有关可利用该对话框进行编辑的信息的详情，请参阅在线帮助。

3. 在访问控制编辑器的各个选项卡上，按自己的需要进行相应的更改。

4. 编辑完 ACI 后，单击 “确定”。

   此时将关闭 ACI 编辑器，而修改后的 ACI 将在 ACI 管理器中列出。

删除 ACI

要删除 ACI：

1. 在 “目录”选项卡上，右键单击子树中的顶级条目，然后从弹出菜单中选择 “设置访问权限”。

   此时显示访问控制管理器窗口。其中包含属于该条目的 ACI 列表。

2. 在访问控制管理器窗口中，选择要删除的 ACI。

3. 单击 “删除”。

   该 ACI 将不会在访问控制管理器中再次列出。

访问控制用法示例

本部分所提供的示例展示了虚构中的 ISP 公司 siroe.com 如何实施其访问控制策略。所有示例都旨在介绍如何通过控制台或 LDIF 文件执行给定的任务。

siroe.com 的业务主要是提供 web 主机服务及 internet 接入服务。siroe.com 的部分 web 主机服务涉及为客户公司提供目录管理服务。 siroe.com 实际为两个中等规模的公司 Company333 和 Company999 提供目录的主机服务，且部分管理着这些目录。它同时还为许多个人用户提供 internet 接入服务。

下面是 siroe.com 要投入使用的访问控制规则：

- 授予 siroe.com 员工对整个 siroe.com 目录树的匿名读取、搜索和比较权 （请参阅第 215 页上的 “授予匿名访问权”）。
- 授予 siroe.com 员工对个人信息 （例如 `homeTelephoneNumber`、`homeAddress`）的写入权 （请参阅第 217 页上的 “向个人条目授予写入权限”）。
- 授予 siroe.com 员工向自己的条目中添加任何角色的权利，但某些关键角色除外 （请参阅第 220 页上的 “限制对重要角色的访问”）。
- 授予 siroe.com 人力资源组对人员分支条目的全部权限 （请参阅第 222 页上的 “向后缀授予组完全访问权限”）。
- 授予所有 siroe.com 员工在目录的 Social Committee 分支下创建组条目的权限，并可删除其中的组条目 （请参阅第 223 页上的 “授予添加和删除组条目的权限”）。
- 授予所有 siroe.com 员工将自身添加到目录的 Social Committee 分支下组条目中的权限 （请参阅第 230 页上的 “允许用户向组中添加自身或从组中删除自身”）。

- 授予 Company333 和 Company999 目录管理员 （角色）对目录树上各自分支的访问权，但要具备某些条件，例如 SSL 验证、时间和日期限制及指定的位置等 （请参阅第 225 页上的 “授予对组或角色的条件访问权限”）。
- 授予个人用户访问自己条目的权限 （请参阅第 217 页上的 “向个人条目授予写入权限”）。
- 拒绝个人用户访问自己条目中的计费信息 （请参阅第 227 页上的 “拒绝访问”）。
- 授予任何人对个人用户子树的匿名访问权，但特别要求不予以列出的用户除外。（这部分目录可能位于防火墙外的从属服务器，且需每天更新一次）。请参阅第 215 页上的 “授予匿名访问权”和第 230 页上的 “使用过滤功能设置目标”。

授予匿名访问权

多数目录在运行时都允许至少匿名访问一个后缀，从而进行读取、搜索或比较操作。例如，如果运行的是公司人事目录，且希望员工能进行搜索 （如对电话簿），就可能需要设置这些权限。 siroe.com 的内部情况就是这样，其说明参见 ACI “Anonymous siroe.com”示例。

作为 ISP， siroe.com 还希望创建全球均可访问的公共电话簿，从而公告其所有用户的联系信息。说明详见 ACI “Anonymous World”示例。

ACI “Anonymous siroe.com”

在 LDIF 中，要向 siroe.com 员工授予对整个 siroe.com 目录树的读取、搜索和比较权限，则应编写下列语句：

```
aci: (targetattr !="userPassword")(version 3.0; acl "Anonymous
 Siroe"; allow (read, search, compare) userdn= "ldap:///anyone" and
 dns="*.siroe.com";)
```

本例假设将 `aci` 添加到 `dc=siroe,dc=com` 条目中。注意：`userPassword` 属性已被排除在 ACI 的范围以外。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中的 siroe.com 节点，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。
3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 Anonymous siroe.com。检查在被授予访问权限的用户列表中是否已显示 “所有用户”。
4. 在 “权限”选项卡上，选中代表读取、比较和搜索权限的复选框。务必清除其它复选框。

5. 在 “目标”选项卡上，单击 “此条目”，从而在目标目录项字段中显示 dc=siroe,dc=com 后缀。在属性表中，找到 userPassword 属性并清除对应的复选框。

   应选中其它所有复选框。如果单击 “名称”标题以按字母顺序组织列表，即可使上述任务更为简单。

6. 在 “主机”选项卡上，单击 “添加”，然后在 “DNS 主机过滤器”字段中键入 *.siroe.com。单击 “确定”以关闭该对话框。

7. 单击访问控制编辑器窗口中的 “确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

ACI “Anonymous World”

在 LDIF 中，要向全球所有人授予对个人用户子树的读取和搜索权限，而同时拒绝访问未列出用户的信息，则应编写下列语句：

```
aci: (targetfilter= "(!(unlistedSubscriber=yes))")
 (targetattr="homePostalAddress || homePhone || mail") (version 3.0;
 acl "Anonymous World"; allow (read, search) userdn=
 "ldap:///anyone";)
```

本例假设将 ACI 添加到 ou=subscribers,dc=siroe,dc=com 条目中。它同时还假设每个用户条目都有属性 unlistedSubscriber，且设置为 yes 或 no。目标定义会根据该属性的值将未列出的用户过滤掉。有关过滤器定义的详细信息，请参阅第 230 页上的 “使用过滤功能设置目标”。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中 siroe.com 节点下的 Subscribers 条目，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。

2. 单击 “新建”以显示访问控制编辑器。

3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 Anonymous World。检查在被授予访问权限的用户列表中是否已显示 “所有用户”。

4. 在 “权限”选项卡上，选中代表读取和比较权限的复选框。务必清除其它复选框。

5. 在“目标”选项卡上，单击“此条目”，从而在目标目录项字段中显示 dc=subscribers,dc=siroe,dc=com 后缀。

   a. 在“子项的过滤器”字段中，键入下列过滤器：

      (!(unlistedSubscriber=yes))

   b. 在属性表中，选中分别代表 homePhone、 homePostalAddress 和 mail 属性的复选框。

      应清除其它所有复选框。如果单击“不检查”按钮以清除表中所有对应于属性的复选框，并单击“名称”标题以按字母顺序组织列表，然后再选择相应的项，即可使上述任务更为简单。

6. 单击“确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

向个人条目授予写入权限

许多目录管理员希望允许内部用户更改自己条目的某些但并非全部属性。siroe.com 的目录管理员想允许用户更改自己的口令、家庭电话号码及家庭地址，但不能更改其它内容。说明详见 ACI “Write siroe.com”示例。

siroe.com 的另一项策略是：只要用户建立到目录的 SSL 连接，即允许其更新自己在 siroe.com 目录树中的个人信息。说明详见 ACI “Write Subscribers”示例。

ACI “Write siroe.com”

注意 设置该权限后，将同时授予删除属性值的权限。

在 LDIF 中，为授予 siroe.com 员工更新自己口令、家庭电话号码和家庭地址的权限，需要编写下列语句：

```
aci: (targetattr="userPassword || homePhone || homePostalAddress")
 (version 3.0; acl "Write siroe.com"; allow (write) userdn=
 "ldap:///self" and dns="*.siroe.com";)
```

本例假定将 ACI 添加到 ou=siroe-people,dc=siroe,dc=com 条目中。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中的 siroe.com 节点，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。
3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 Write siroe.com。在被授予访问权限的用户列表中，执行下列操作：

   a. 选择并删除 “所有用户”，然后单击 “添加”。

      此时显示 “添加用户和组”对话框。

   b. 将搜索区域设置为 “特殊权限”，然后从搜索结果列表中选择 “自身”。

   c. 单击 “添加”按钮，从而在被授予了访问权限的用户列表中列出 “自身”。

   d. 单击 “确定”以关闭 “添加用户和组”对话框。
4. 在 “权限”选项卡上，选中代表写入权限的复选框。务必清除其它复选框。
5. 在 “目标”选项卡上，单击 “此条目”，从而在目标目录项字段中显示 `dc=siroe,dc=com` 后缀。在属性表中，选中分别代表 `homePhone`、`homePostalAddress` 和 `userPassword` 属性的复选框。

   应清除其它所有复选框。如果单击 “不检查”按钮以清除表中所有对应于属性的复选框，并单击 “名称”标题以按字母顺序组织列表，然后再选择相应的项，即可使上述任务更为简单。
6. 在 “主机”选项卡上，单击 “添加”以显示 “添加主机过滤器”对话框。在 “DNS 主机过滤器”字段中，键入 `*.siroe.com`。单击 “确定”以关闭该对话框。
7. 单击访问控制编辑器窗口中的 “确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

ACI “Write Subscribers”

注意 设置该权限后，将同时授予删除属性值的权限。

在 LDIF 中，为授予 siroe.com 用户更新自己口令和家庭电话号码的权限，需要编写下列语句：

```
aci: (targetattr="userPassword || homePhone") (version 3.0; acl
 "Write Subscribers"; allow (write) userdn= "ldap://self" and
 authmethod="ssl";)
```

本例假定将 `aci` 添加到 `ou=subscribers,dc=siroe,dc=com` 条目中。

注意：siroe.com 用户不具有家庭地址的写入权限，因为他们可能会删除该属性，而 siroe.com 则需要该信息来索要付款。因此，家庭地址是对业务而言十分重要的信息。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中 siroe.com 节点下的 Subscribers 条目，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。
3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 Write Subscribers。在被授予访问权限的用户列表中，执行下列操作：
   a. 选择并删除 “所有用户”，然后单击 “添加”。

      此时显示 “添加用户和组”对话框。
   b. 将搜索区域设置为 “特殊权限”，然后从搜索结果列表中选择 “自身”。
   c. 单击 “添加”按钮，从而在被授予了访问权限的用户列表中列出 “自身”。
   d. 单击 “确定”以关闭 “添加用户和组”对话框。
4. 在 “权限”选项卡上，选中代表写入权限的复选框。务必清除其它复选框。

5. 在“目标”选项卡上，单击“此条目”，从而在目标目录项字段中显示 dc=subscribers,dc=siroe,dc=com 后缀。

   a. 在“子项的过滤器”字段中，键入下列过滤器：

      (!(unlistedSubscriber=yes))

   b. 在属性表中，选中分别代表 homePhone、 homePostalAddress 和 mail 属性的复选框。

      应清除其它所有复选框。如果单击“不检查”按钮以清除表中所有对应于属性的复选框，并单击“名称”标题以按字母顺序组织列表，然后再选择相应的项，即可使上述任务更为简单。

6. 如果希望用户使用 SSL 进行验证，请单击“手动编辑”按钮以切换到手动编辑状态，然后向 LDIF 语句中添加 authmethod=ssl，从而使其为：

   ```
   (targetattr="homePostalAddress || homePhone || mail") (version
   3.0; acl "Write Subscribers"; allow (write) (userdn=
   "ldap:///self") and authmethod="ssl";)
   ```

7. 单击“确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

限制对重要角色的访问

在目录中使用角色定义可以标识对公司业务至关重要的功能、标识网络和目录的管理，或者是用于其它用途。

例如，通过标识公司范围内在特定时间、特定日期可用的系统管理员子集，可以创建 superAdmin 角色。也可以创建 First Aid 角色，其中包括特定地点所有受过紧急救助训练的员工。有关创建角色定义的信息，请参阅第 150 页上的“分配角色”。

当角色对公司或业务关键功能可提供任何类型的用户特权时，应考虑对该角色的访问加以限制。例如， siroe.com 的员工可向自己的条目中添加除 superAdmin 角色之外的任何角色。说明详见 ACI “Roles”示例。

ACI “Roles”

在 LDIF 中，要授予 siroe.com 员工向自己的条目中添加任何角色的权限（superAdmin 角色除外），应编写下列语句：

```
aci: (targetattr="*") (targattrfilters="add=nsRoleDN:(nsRoleDN !=
 "cn=superAdmin, dc=siroe, dc=com")") (version 3.0; acl "Roles";
 allow (write) userdn= "ldap:///self" and dns="*.siroe.com";)
```

本例假定将 ACI 添加到 ou=siroe-people,dc=siroe,dc=com 条目中。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中的 siroe.com 节点，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。
3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 Roles。在被授予访问权限的用户列表中，执行下列操作：
   a. 选择并删除 “所有用户”，然后单击 “添加”。

      此时显示 “添加用户和组”对话框。

   b. 将 “添加用户和组”对话框中的搜索区域设置为 “特殊权限”，然后从搜索结果列表中选择 “自身”。
   c. 单击 “添加”按钮，从而在被授予了访问权限的用户列表中列出 “自身”。
   d. 单击 “确定”以关闭 “添加用户和组”对话框。
4. 在 “权限”选项卡上，选中代表写入权限的复选框。务必清除其它复选框。
5. 在 “主机”选项卡上，单击 “添加”以显示 “添加主机过滤器”对话框。在 “DNS 主机过滤器”字段中，键入 `*.siroe.com`。单击 “确定”以关闭该对话框。
6. 要为角色创建基于值的过滤器，请单击 “手动编辑”按钮以切换到手动编辑状态。向 LDIF 语句的开头添加以下内容：

   ```
   (targattrfilters="add=nsRoleDN:(nsRoleDN != "cn=superAdmin,
   dc=siroe,dc=com")")
   ```

   LDIF 语句应该为：

   ```
   (targetattr="*") (targattrfilters="add=nsRoleDN:(nsRoleDN !=
   "cn=superAdmin, dc=siroe,dc=com")") (target =
   "ldap:///dc=siroe,dc=com") (version 3.0; acl "Roles"; allow
   (write) (userdn = "ldap:///self") and (dns="*.siroe.com");)
   ```

7. 单击 “确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

向后缀授予组完全访问权限

多数目录都有用于标识某些公司功能的组。可授予这些组对所有或部分目录的完全访问权限。通过对组应用访问权限，可以避免向各个成员分别设置访问权限。只要将用户添加到组中，即授予其相应的访问权限。

对 siroe.com 来说，其人力资源 (HR) 组对目录的 `ou=siroe-people` 分支具有完全访问权限，因此可以更新员工数据库。说明详见 ACI “HR”示例。

ACI “HR”

在 LDIF 中，要授予 HR 组对目录中员工分支的全部权限，应编写下列语句：

```
aci: (targetattr="*") (version 3.0; acl "HR"; allow (all) userdn=
 "ldap:///cn=HRgroup,ou=siroe-people,dc=siroe,dc=com";)
```

本例假定将 ACI 添加到 `ou=siroe-people,dc=siroe,dc=com` 条目中。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中的 siroe.com 节点下的 siroe.com-people 条目，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。
3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 HR。在被授予访问权限的用户列表中，执行下列操作：
   a. 选择并删除 “所有用户”，然后单击 “添加”。

      此时显示 “添加用户和组”对话框。

   b. 将搜索区域设置为 “用户和组”，然后在 “搜索”字段中键入 “HRgroup”。

      本例假定已创建 HR 组或角色。有关组和角色的详细信息，请参阅第 5 章 “高级条目管理”。

   c. 单击 “添加”按钮，从而在被授予了访问权限的用户列表中列出该 HR 组。
   d. 单击 “确定”以关闭 “添加用户和组”对话框。
4. 在 “权限”选项卡上，单击 “全部检查”按钮。

   这将选中除 “代理”权限以外的所有复选框。
5. 单击 “确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

授予添加和删除组条目的权限

有些机构希望只要能提高工作效率或对公司有所帮助，即允许员工在目录树中创建条目。

例如， siroe.com 中有一个活跃的社会活动委员会，下属有几个俱乐部：网球、游泳、滑雪、戏剧等。任何 siroe.com 员工都可以创建代表新俱乐部的组条目。说明详见 ACI “Create Group”示例。任何 siroe.com 员工都能成为上述某个组的成员。说明详见第 230 页上的 “允许用户向组中添加自身或从组中删除自身”中的 ACI “Group Members”示例。只有组的所有者才能修改或删除组条目。说明详见 ACI “Delete Group”示例。

ACI “Create Group”

在 LDIF 中，要授予 siroe.com 员工在 `ou=Social Committee` 分支下创建组条目的权限，应编写下列语句：

```
aci: (target="ldap:///ou=social committee,dc=siroe,dc=com)
 (targetattr="*") (targattrfilters="add=objectClass:
 (objectClass=groupOfNames)") (version 3.0; acl "Create Group";
 allow (read,search,add) (userdn= "ldap:///uid=*,ou=siroe-people,
 dc=siroe,dc=com") and dns="*.siroe.com";)
```

注意 该 ACI 并不授予写入权限，即条目的创建者无法修改该条目。

本例假定将 ACI 添加到 `ou=social committee,dc=siroe,dc=com` 条目中。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中 siroe.com 节点下的 Social Committee 条目，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。
3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 Create Group。在被授予访问权限的用户列表中，执行下列操作：
   a. 选择并删除 “所有用户”，然后单击 “添加”。

      此时显示 “添加用户和组”对话框。
   b. 将搜索区域设置为 “特殊权限”，然后从搜索结果列表中选择 “所有已鉴定的用户”。
   c. 单击 “添加”按钮，从而在被授予了访问权限的用户列表中列出 “所有已鉴定的用户”。
   d. 单击 “确定”以关闭 “添加用户和组”对话框。
4. 在 “权限”选项卡上，选中分别代表读取、搜索和添加权限的复选框。务必清除其它复选框。
5. 在 “目标”选项卡上，单击 “此条目”，从而在目标目录项中显示 `ou=social committee,dc=siroe,dc=com` 后缀。
6. 在 “主机”选项卡上，单击 “添加”以显示 “添加主机过滤器”对话框。在 “DNS 主机过滤器”字段中，键入 `*.siroe.com`。单击 “确定”以关闭该对话框。
7. 要创建基于值的过滤器，且仅允许员工向该子树中添加组条目，请单击 “手动编辑”按钮，从而切换到手动编辑状态。向 LDIF 语句的开头添加以下内容：

   ```
   (targattrfilters="add=objectClass:(objectClass=groupOfNames)")
   ```

   LDIF 语句应该为：

   ```
   (targetattr = "*")
   (targattrfilters="add=objectClass:(objectClass=groupOfNames)")
   (target="ldap:///ou=social committee,dc=siroe,dc=com) (version
   3.0; acl "Create Group"; allow (read,search,add) (userdn=
   "ldap:///all") and (dns="*.siroe.com"); )
   ```
8. 单击 “确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

ACI “Delete Group”

在 LDIF 中，要授予 siroe.com 员工对其在 `ou=Social Committee` 分支下的组条目的修改或删除权限，应编写下列语句：

```
aci: (target="ou=social committee,dc=siroe,dc=com)(targetattr = "*")
 (targattrfilters="del=objectClass:(objectClass=groupOfNames)")
 (version 3.0; acl "Delete Group"; allow (write,delete) userattr=
 "owner#GROUPDN";)
```

本例假定将 `aci` 添加到 `ou=social committee,dc=siroe,dc=com` 条目中。

使用控制台创建该 ACI 并非十分有效，因为此时必须使用手动编辑模式来创建目标过滤器及检查组的所有权关系。

授予对组或角色的条件访问权限

许多情况下，当授予组或角色对目录的特权时，会希望确保这些特权不会受到试图扮演特权用户的入侵者的利用。因此，授予组或角色关键访问权的访问控制规则常常关联着许多条件。

例如， siroe.com 为使用其主机服务的公司 Company333 和 Company999 创建了目录管理员角色。它希望这些公司能管理自己的数据并实施自己的访问控制规则，而同时确保不受入侵者侵害。因此， Company333 和 Company999 对目录树的各个分支都具有完全权限，但前提是满足下列条件：

- 使用 SSL 进行连接验证；
- 访问请求发生在周一至周四上午 8 点和下午 6 点之间；
- 访问请求来自为各个公司指定的 IP 地址。

这些条件的示例见各个公司相应的 ACI，即 ACI “Company333”和 ACI “Company999”。由于这些 ACI 的内容相同，因此下面中仅给出 “Company333”的 ACI 示例。

ACI “Company333”

在 LDIF 中，要在上述条件下授予 Company333 对自己目录分支的完全访问权限，则应编写下列语句：

```
aci: (target="ou=Company333,ou=corporate-clients,dc=siroe,dc=com")
 (targetattr = "*") (version 3.0; acl "Company333"; allow (all)
 (roledn= "ldap:///cn=DirectoryAdmin,ou=Company333,
 ou=corporate-clients, dc=siroe,dc=com") and (authmethod="ssl") and
 (dayofweek="Mon,Tues,Wed,Thu") and (timeofday >= "0800" and
 timeofday <= "1800") and (ip="255.255.123.234"); )
```

本例假定将 ACI 添加到 `ou=Company333,ou=corporate-clients,dc=siroe,dc=com` 条目中。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中 siroe.com 节点下的 Company333 条目，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。
3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 Company333。在被授予访问权限的用户列表中，执行下列操作：
   a. 选择并删除 “所有用户”，然后单击 “添加”。

      此时显示 “添加用户和组”对话框。

   b. 将搜索区域设置为 “用户和组”，然后在 “搜索”字段中键入 DirectoryAdmin。

      本例假定已创建管理员角色，其 `cn` 为 `DirectoryAdmin`。

   c. 单击 “添加”按钮，从而在被授予了访问权限的用户列表中列出该管理员角色。

   d. 单击 “确定”以关闭 “添加用户和组”对话框。
4. 在 “权限”选项卡上，单击 “全部检查”按钮。
5. 在 “目标”选项卡上，单击 “此条目”，从而在目标目录项字段中显示 `ou=Company333,ou=corporate-clients,dc=siroe,dc=com` 后缀。

6. 在 “主机” 选项卡上，单击 “添加” 以显示 “添加主机过滤器” 对话框。在 “IP 地址主机过滤器” 字段中，键入 `255.255.123.234`。单击 “确定” 以关闭该对话框。

   该 IP 地址必须是 Company333 管理员连接 siroe.com 目录时所用的有效主机 IP 地址。

7. 在 “次数” 选项卡上，选择对应于周一至周四上午 8 点至下午 6 点的时间块。

   表的下面将显示一条信息，指示所选时间块。

8. 要从 Company333 管理员处强制进行 SSL 验证，请单击 “手动编辑” 按钮，从而切换到手动编辑状态。在 LDIF 语句的末尾添加下列内容：

   and (authmethod="ssl")

   LDIF 语句应类似于：

```
aci: (targetattr = "*") (target="ou=Company333,
 ou=corporate-clients,dc=siroe,dc=com") (version 3.0; acl
 "Company333"; allow (all) (roledn="ldap:///cn=DirectoryAdmin,
 ou=Company333,ou=corporate-clients, dc=siroe,dc=com") and
 (dayofweek="Mon,Tues,Wed,Thu") and (timeofday >= "0800" and
 timeofday <= "1800") and (ip="255.255.123.234") and
 (authmethod="ssl"); )
```

9. 单击 “确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

拒绝访问

如果目录中装有对业务至关重要的信息，就可能想拒绝对该目录的访问。

例如，siroe.com 希望所有用户都能读取自己条目下诸如连接时间或帐户结余等记费信息，但又明确希望拒绝向该信息中写入任何内容。示例分别参见 ACI “Billing Info Read” 和 ACI “Billing Info Deny”。

ACI “Billing Info Read”

在 LDIF 中，要授予用户读取自己条目中记费信息的权限，则应编写下列语句：

```
aci: (targetattr="connectionTime || accountBalance") (version 3.0;
 acl "Billing Info Read"; allow (search,read)
 userdn="ldap:///self";)
```

本例假定模式中已创建相关的属性，且将 ACI 添加到 `ou=subscribers,dc=siroe, dc=com` 条目中。

从控制台上，执行下列操作可设置该权限：

1. 在 “目录”选项卡上，右键单击左侧导航树中 siroe.com 节点下的 subscribers 条目，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。
3. 在 “用户/组”选项卡上，在 “ACI 名称”字段中键入 Billing Info Read。在被授予访问权限的用户列表中，执行下列操作：
   a. 选择并删除 “所有用户”，然后单击 “添加”。

      此时显示 “添加用户和组”对话框。

   b. 将 “添加用户和组”对话框中的搜索区域设置为 “特殊权限”，然后从搜索结果列表中选择 “自身”。
   c. 单击 “添加”按钮，从而在被授予了访问权限的用户列表中列出 “自身”。
   d. 单击 “确定”以关闭 “添加用户和组”对话框。
4. 在 “权限”选项卡上，选中分别代表搜索和读取权限的复选框。务必清除其它复选框。
5. 在 “目标”选项卡上，单击 “此条目”，从而在目标目录项字段中显示 `ou=subscribers,dc=siroe,dc=com` 后缀。在属性表中，选中分别代表 `connectionTime` 和 `accountBalance` 属性的复选框。

   应清除其它所有复选框。如果单击 “不检查”按钮以清除表中所有对应于属性的复选框，并单击 “名称”标题以按字母顺序组织列表，然后再选择相应的项，即可使上述任务更为简单。

   本例假定已将 `connectionTime` 和 `accountBalance` 属性添加到模式中。
6. 单击 “确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

ACI “Billing Info Deny”

在 LDIF 中，要拒绝用户修改自己条目中记费信息的权限，则应编写下列语句：

```
aci: (targetattr="connectionTime || accountBalance") (version 3.0;
 acl "Billing Info Deny"; deny (write) userdn= "ldap:///self";)
```

本例假定模式中已创建相关的属性，且将 ACI 添加到 `ou=subscribers,dc=siroe,dc=com` 条目中。

从控制台上，执行下列操作可设置该权限：

1. 在“目录”选项卡上，右键单击左侧导航树中 siroe.com 节点下的 subscribers 条目，然后从弹出菜单中选择“设置访问权限”，从而显示访问控制管理器。
2. 单击“新建”以显示访问控制编辑器。
3. 在“用户/组”选项卡上，在“ACI 名称”字段中键入 Billing Info Deny。在被授予访问权限的用户列表中，执行下列操作：
   a. 选择并删除“所有用户”，然后单击“添加”。

      此时显示“添加用户和组”对话框。

   b. 将“添加用户和组”对话框中的搜索区域设置为“特殊权限”，然后从搜索结果列表中选择“自身”。
   c. 单击“添加”按钮，从而在被授予了访问权限的用户列表中列出“自身”。
   d. 单击“确定”以关闭“添加用户和组”对话框。
4. 在“权限”选项卡上，选中代表写入权限的复选框。务必清除其它复选框。
5. 单击“手动编辑”按钮，然后在所显示的 LDIF 语句中将 **`allow`** 更改为 **`deny`**。
6. 在“目标”选项卡上，单击“此条目”，从而在目标目录项字段中显示 `ou=subscribers,dc=siroe,dc=com` 后缀。在属性表中，选中分别代表 `connectionTime` 和 `accountBalance` 属性的复选框。

   应清除其它所有复选框。如果单击“不检查”按钮以清除表中所有对应于属性的复选框，并单击“名称”标题以按字母顺序组织列表，然后再选择相应的项，即可使上述任务更为简单。

   本例假定已将 `connectionTime` 和 `accountBalance` 属性添加到模式中。
7. 单击“确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

使用过滤功能设置目标

如果想设置访问控制以允许访问跨多个目录分布的大量条目，则可以使用过滤器来设置目标。记住：由于搜索过滤器不直接命名要进行访问权限管理的对象，因此容易无意间向错误的对象授予或拒绝访问权限，尤其是在目录结构较为复杂的情况下。此外，过滤器可导致难于对目录中的访问控制故障进行故障查找。

下列过程展示如何授予用户 `bjensen` 对财务机构所有成员的部门号、家庭电话号码、家庭邮政地址、 JPEG 图像及管理者等属性的写入权限。

设置这些权限前，必须首先创建 accounting 分支点 (`ou=accounting,dc=siroe,dc=com`)。使用 iPlanet Directory Server Console 上的目录选项卡可以创建组织单元分支点。

允许用户向组中添加自身或从组中删除自身

许多目录将 ACI 设置为允许用户向组中添加自身或从组中删除自身。这一点较为有用。例如，可以允许用户向邮寄列表中添加自身，或者从中删除自身。

siroe.com 的员工可以将自身添加到 `ou=social committee` 子树下的任何组条目中。说明详见 ACI “Group Members”示例。

ACI “Group Members”

在 LDIF 中，要授予 siroe.com 员工向组中添加自身或从组中删除自身的权限，则应编写下列语句：

```
aci: (targettattr="member")(version 3.0; acl "Group Members";
 allow (selfwrite)
 (userdn= "ldap:///uid=*,ou=siroe-people,dc=siroe,dc=com") ;)
```

本例假定将 ACI 添加到 `ou=social committee,dc=siroe,dc=com` 条目中。

从控制台上，执行下列操作可设置该权限：

1. 在“目录”选项卡上，右键单击左侧导航树中 siroe.com 节点下的 siroe-people 条目，然后从弹出菜单中选择 “设置访问权限”，从而显示访问控制管理器。
2. 单击 “新建”以显示访问控制编辑器。

3. 在“用户/组”选项卡上，在“ACI 名称”字段中键入 Group Members。在被授予访问权限的用户列表中，执行下列操作：

   a. 选择并删除 “所有用户”，然后单击 “添加”。

      此时显示 “添加用户和组”对话框。

   b. 将 “添加用户和组”对话框中的搜索区域设置为 “特殊权限”，然后从搜索结果列表中选择 “所有已鉴定的用户”。

   c. 单击 “添加”按钮，从而在被授予了访问权限的用户列表中列出 “所有已鉴定的用户”。

   d. 单击 “确定”以关闭 “添加用户和组”对话框。

4. 在 “权限”选项卡上，选中代表自写权限的复选框。务必清除其它复选框。

5. 在 “目标”选项卡上，在目标目录项字段中键入 `dc=siroe,dc=com` 后缀。在属性表中，选中代表 `member` 属性的复选框。

   应清除其它所有复选框。如果单击 “不检查”按钮以清除表中所有对应于属性的复选框，并单击 “名称”标题以按字母顺序组织列表，然后再选择相应的项，即可使上述任务更为简单。

6. 单击 “确定”。

   新的 ACI 随即添加到访问控制管理器窗口的 ACI 列表中。

定义包含逗号的 DN 的权限

包含逗号的 DN 要求在 LDIF ACI 语句中进行特殊的处理。在 ACI 语句的目标和绑定规则部分，逗号必须用单个反斜杠 (\) 进行转义。下面给出该语法的示例：

```
dn: dc=siroe.com Bolivia\, S.A.,dc=com
objectClass: top
objectClass: organization
aci: (target="ldap:///dc=siroe.com Bolivia\,
 S.A.,dc=com")(targetattr="*") (version 3.0; acl "aci 2"; allow
 (all)groupdn = "ldap:///cn=Directory Administrators,dc=siroe.com
 Bolivia\, S.A.,dc=com";)
```

代理验证 ACI 示例

代理验证 (proxy authorization) 方法是一种特殊的验证方式：使用自身的标识绑定到目录的用户通过代理验证被授予其他用户的权限。

本例中，假设：

- 客户机应用程序的绑定 DN 为 "uid=MoneyWizAcctSoftware, ou=Applications,dc=siroe,dc=com"。
- 客户机应用程序所请求访问的目标子树为 ou=Accounting,dc=siroe,dc=com。
- 目录中存在对 ou=Accounting,dc=siroe,dc=com 子树具有访问权限的财务管理员。

为使客户机应用程序获得对 Accounting 子树的访问权 （具有与财务管理员相同的访问权限）：

- 财务管理员必须具有对 ou=Accounting,dc=siroe,dc=com 子树的访问权限。例如，下列 ACI 授予财务管理员条目全部权限：

```
aci: (target="ldap:///ou=Accounting,dc=siroe,dc=com")
 (targetattr="*") (version 3.0; acl "allowAll-AcctAdmin";
 allow (all) userdn="uid=AcctAdministrator,ou=Administrators,
 dc=siroe,dc=com")
```

- 目录中必须存在向客户机应用程序授予代理权限的以下 ACI：

```
aci: (target="ldap:///ou=Accounting,dc=siroe,dc=com")
 (targetattr="*") (version 3.0; acl "allowproxy-
 accountingsoftware"; allow (proxy) userdn=
 "uid=MoneyWizAcctSoftware,ou=Applications,dc=siroe,dc=com")
```

利用该 ACI， MoneyWizAcctSoftware 客户机应用程序即可绑定到目录上并发送申请对代理 DN 访问权限的 LDAP 命令 （例如 ldapsearch 或 ldapmodify）。

注意 目录管理员 DN 不能用作代理 DN。也不能将代理权限授予目录管理员。另外，如果 iPlanet Directory Server 在同一绑定操作中接收到一个以上的代理验证控制，就会向客户机应用程序返回错误，并且绑定企图将失败。

查看条目的 ACI

通过运行以下 `ldapsearch` 命令，可以查看目录中单个后缀下的所有 ACI：

```
ldapsearch -h host -p port -b baseDN -D rootDN -w rootPassword (aci=*) aci
```

在控制台上，利用访问控制管理器可以查看应用于特定条目的所有 ACI。

1. 在 Directory Console 上，进入“目录”选项卡，右键单击导航树中的条目，然后选择“设置访问权限”。

   此时显示访问控制管理器。其中包含属于所选条目的 ACI 列表。

2. 选中“显示继承的 ACI”复选框，从而显示也同时应用的、在所选条目之上的条目中创建的所有 ACI。

高级访问控制：使用宏 ACI

在使用重复目录树结构的机构中，有时可以使用宏对目录中所用的 ACI 数量进行优化。减少目录树中的 ACI 数可以简化对访问控制策略的管理，同时提高 ACI 内存使用的效率。

宏是 ACI 中用于代表 DN （或部分 DN）的占位符。使用宏可以表示 ACI 中目标部分或绑定规则部分的 DN，或者是上述两部分的 DN。实际上，当 iPlanet Directory Server 获取到进入的 LDAP 操作时，ACI 宏将按 LDAP 操作所确定的目标进行资源匹配。如果存在匹配，宏就会替换为目标资源的 DN 值。 iPlanet Directory Server 随即对 ACI 进行正常评估。

宏 ACI 示例

最好用示例来解释宏 ACI 及其工作原理。第 235 页的图 6-4 给出的目录树中就使用了宏 ACI 来有效地减少 ACI 的整体数量。

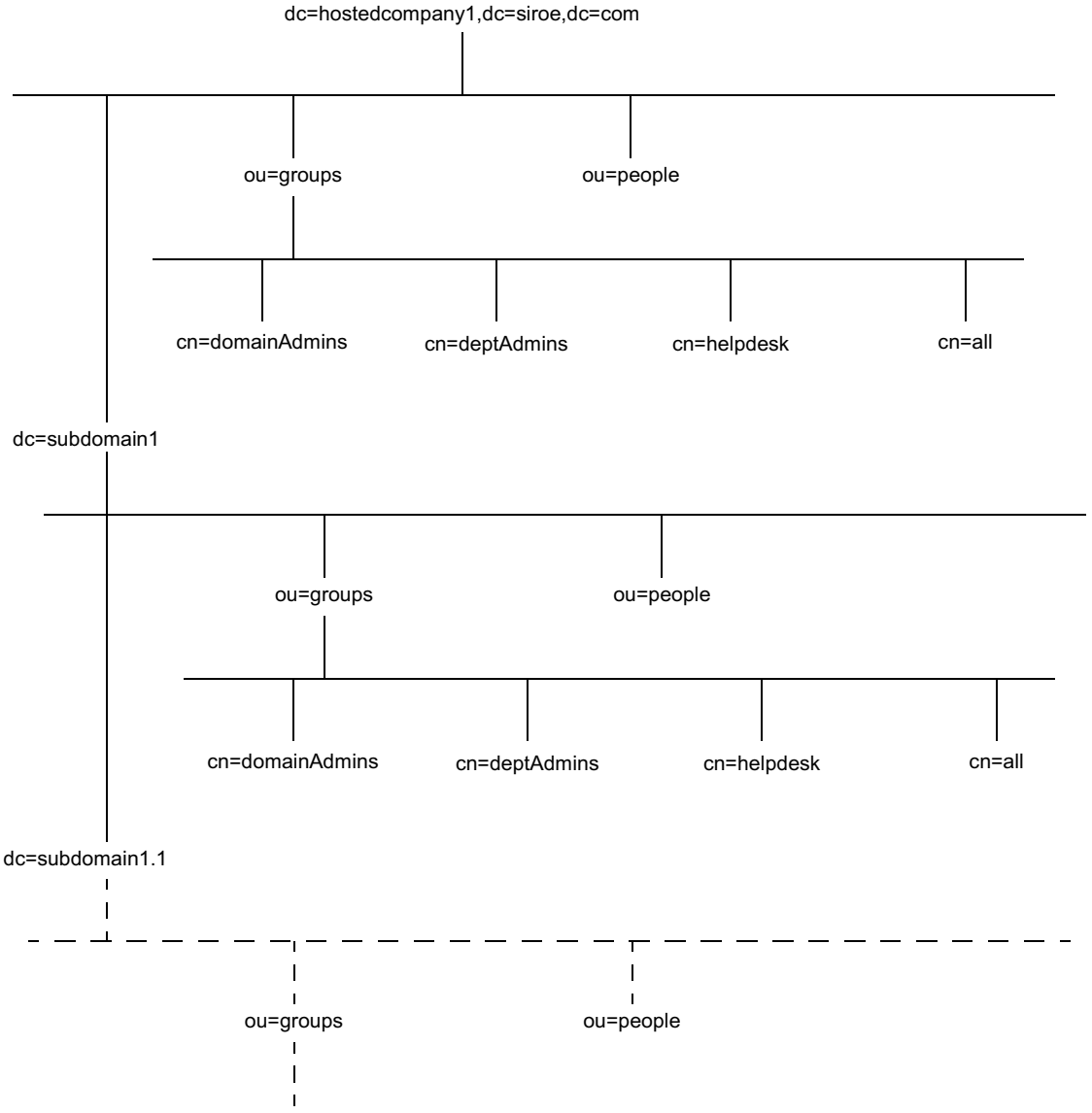
本例中，请注意同一目录树结构 (ou=groups, ou=people) 的子域重复形式。这种形式也跨目录树重复，因为 siroe.com 目录树存储有下列后缀：`dc=hostedCompany2, dc=siroe,dc=com` 和 `dc=hostedCompany3,dc=siroe,dc=com`。

目录树中应用的 ACI 也具有重复形式。例如，下列 ACI 位于 `dc=hostedCompany1, dc=siroe,dc=com` 节点上：

```
aci: (targetattr="*")(targetfilter=(objectClass=nsManagedDomain))
 (version 3.0; acl "Domain access"; allow (read,search) groupdn=
 "ldap:///cn=DomainAdmins,ou=Groups,dc=hostedCompany1,dc=siroe,
 dc=com";)
```

该 ACI 向 DomainAdmins 组授予对 `dc=hostedCompany1,dc=siroe,dc=com` 目录树中所有条目的读取和搜索权限。

图 6-4 宏 ACI 的目录树示例



下列 ACI 位于 dc=hostedCompany1,dc=siroe,dc=com 节点上：

```
aci: (targetattr="*")(targetfilter=(objectClass=nsManagedDomain))
 (version 3.0; acl "Domain access"; allow (read,search)
 groupdn="ldap:///cn=DomainAdmins,ou=Groups,dc=hostedCompany1,
 dc=siroe,dc=com";)
```

下列 ACI 位于 dc=subdomain1,dc=hostedCompany1,dc=siroe,dc=com 节点上：

```
aci: (targetattr="*")(targetfilter=(objectClass=nsManagedDomain))
 (version 3.0; acl "Domain access"; allow (read,search)
 groupdn="ldap:///cn=DomainAdmins,ou=Groups,dc=subdomain1,
 dc=hostedCompany1,dc=siroe,dc=com";)
```

下列 ACI 位于 dc=hostedCompany2,dc=siroe,dc=com 节点上：

```
aci: (targetattr="*")(targetfilter=(objectClass=nsManagedDomain))
 (version 3.0; acl "Domain access"; allow (read,search)
 groupdn="ldap:///cn=DomainAdmins,ou=Groups,dc=hostedCompany2,
 dc=siroe,dc=com";)
```

下列 ACI 位于 dc=subdomain1,dc=hostedCompany2,dc=siroe,dc=com 节点上：

```
aci: (targetattr="*")(targetfilter=(objectClass=nsManagedDomain))
 (version 3.0; acl "Domain access"; allow (read,search)
 groupdn="ldap:///cn=DomainAdmins,ou=Groups, dc=subdomain1,
 dc=hostedCompany2,dc=siroe,dc=com";)
```

在上述四个 ACI 中，唯一的区别是 groupdn 关键字中指定的 DN。通过使用宏来表示 DN，即可用目录树根位置 （dc=siroe,dc=com 节点）上的单个 ACI 来替换这些 ACI。该 ACI 表示为：

```
aci: (target="ldap:///ou=Groups,($dn),dc=siroe,dc=com")
 (targetattr="*")(targetfilter=(objectClass=nsManagedDomain))
 (version 3.0; acl "Domain access"; allow (read,search) groupdn=
 "ldap:///cn=DomainAdmins,ou=Groups,[$dn],dc=siroe,dc=com";)
```

注意：此处引入了以前从未使用过的目标关键字 target。

在上例中， ACI 的数量从四个减少到一个。但真正的益处在于跨目录树及沿目录树向下所具有的重复形式数。

宏 ACI 语法

宏 ACI 提供下列类型的表达式来替换 DN 或 DN 的一部分：

- `($dn)`
- `[$dn]`
- `($attr.attrName)`，其中 *attrName* 代表目标条目所含的属性

简言之，用于提供绑定凭证 （例如 `userdn`、 `roledn`、 `groupdn` 和 `userattr`）的 ACI 关键字统称为*主体*，以对比 ACI 的目标。宏 ACI 可用于 ACI 的目标部分或主体部分。

表 6-3 给出可使用 DN 宏的 ACI 部分：

表 6-3 ACI 关键字中的宏

宏	ACI 关键字
`($dn)`	target, targetfilter, userdn, roledn,groupdn, userattr
`[$dn]`	targetfilter, userdn, roledn, groupdn, userattr
`($attr.attrName)`	userdn, roledn, groupdn, userattr

下列限制条件适用：

- 如果在 `targetfilter`、 `userdn`、 `roledn`、 `groupdn`、 `userattr` 中使用 `($dn)`，则*必须*定义包含 `($dn)` 的目标。
- 如果在 `targetfilter`、 `userdn`、 `roledn`、 `groupdn`、 `userattr` 中使用 `[$dn]`，则*必须*定义包含 `($dn)` 的目标。

简言之，使用任何宏时将*始终*需要包含 `($dn)` 宏的目标定义。

可以将 `($dn)` 宏和 `($attr.attrName)` 宏组合到一起。

($dn) 宏匹配

($dn) 宏将被 LDAP 请求中作为目标的资源匹配部分所替换。例如，假定 LDAP 请求的目标为 cn=all,ou=groups,dc=subdomain1,dc=hostedCompany1,dc=siroe,dc=com 条目，而定义目标的 ACI 如下所示：

```
(target="ldap:///ou=Groups,($dn),dc=siroe,dc=com")
```

($dn) 宏匹配 "dc=subdomain1, dc=hostedCompany1"。

当 ACI 的主体也使用 ($dn) 时，与目标相匹配的子串将用于扩展主体：例如：

```
aci: (targetattr="*") (target="ldap:///ou=*,($dn),dc=siroe,dc=com")
 (version 3.0; acl "Domain access"; allow (read,search) groupdn=
 "ldap:///cn=DomainAdmins,ou=Groups,($dn),dc=siroe,dc=com";)
```

这种情况下，如果目标中匹配 ($dn) 的字符串为 dc=subdomain1, dc=hostedCompany1，则同一字符串也将用于主体中。上述 ACI 的扩展如下所示：

```
aci: (targetattr="*") (target="ldap:///ou=Groups,dc=subdomain1,
 dc=hostedCompany1, dc=siroe,dc=com") (version 3.0; acl "Domain
 access"; allow (read,search)
 groupdn="ldap:///cn=DomainAdmins,ou=Groups,
 dc=subdomain1,dc=hostedCompany1,dc=siroe,dc=com";)
```

扩展宏后， iPlanet Directory Server 将按常规过程评估 ACI，从而确定是否授予访问权限。

[$dn] 宏匹配

[$dn] 的匹配机制与 ($dn) 的略有不同。目标资源的 DN 将被检查数次，每次都会丢弃最左侧的 RDN 组件，直到找到匹配的对象。

例如，假设 LDAP 请求的目标为 cn=all,ou=groups,dc=subdomain1, dc=hostedCompany1,dc=siroe,dc=com 子树，且具有下列 ACI：

```
aci: (targetattr="*") (target="ldap:///ou=Groups,($dn),dc=siroe,
 dc=com") (version 3.0; acl "Domain access"; allow (read,search)
 groupdn="ldap:///cn=DomainAdmins,ou=Groups,[$dn],dc=siroe,dc=com";
 )
```

扩展该 ACI 的步骤如下所示：

1. 对象中的 ($dn) 匹配 dc=subdomain1,dc=hostedCompany1。
2. 将主体中的 [$dn] 替换为 dc=subdomain1,dc=hostedCompany1。

   结果为 groupdn="ldap:///cn=DomainAdmins,ou=Groups,dc=subdomain1, dc=hostedCompany1,dc=siroe,dc=com"。如果绑定 DN 是该组的成员，则终止匹配过程，然后对 ACI 进行评估。如果不匹配，过程将继续。

3. 将主体中的 [$dn] 替换为 dc=hostedCompany1。

   结果为 groupdn="ldap:///cn=DomainAdmins,ou=Groups, dc=hostedCompany1,dc=siroe,dc=com"。这种情况下，如果绑定 DN 不是该组的成员，则不对 ACI 进行评估。如果是成员，则对 ACI 进行评估。

[$dn] 宏的优势在于：它提供了一种灵活地向域级别上的管理员授予对目录树*所有*子域访问权的方式。因此，它对于表示域之间的层次关系较为有用。

例如，请考虑下列 ACI：

```
aci: (target="ldap:///ou=*, ($dn),dc=siroe,dc=com")
 (targetattr="*")(targetfilter=(objectClass=nsManagedDomain))
 (version 3.0; acl "Domain access"; allow (read,search) groupdn=
 "ldap:///cn=DomainAdmins,ou=Groups,[$dn],dc=siroe,dc=com";)
```

它授予 cn=DomainAdmins,ou=Groups,dc=hostedCompany1,dc=siroe,dc=com 成员对 dc=hostedCompany1 下所有子域的访问权，从而使属于该组的管理员可以访问诸如 ou=people,dc=subdomain1.1,dc=subdomain1 的子树。

但同时， cn=DomainAdmins,ou=Groups,dc=subdomain1.1 的成员也将被拒绝访问 ou=people,dc=hostedCompany1 和 ou=people,dc=hostedCompany1 节点。

($attr.*attrName*) 宏匹配

($attr.attrname) 宏始终用于 DN 的主体部分。例如，可以定义下列 roledn：

```
roledn = "ldap:///cn=DomainAdmins,($attr.ou)"
```

假设服务器接收到以下列条目为目标的 LDAP 操作：

```
dn: cn=Heather Blue, ou=People, dc=HostedCompany1, dc=siroe, dc=com
cn: Heather Blue
sn: Blue
ou: Engineering, dc=HostedCompany1, dc=siroe, dc=com
...
```

为评估 ACI 的 roledn 部分，服务器将查找目标条目中存储的 ou 属性，然后使用该属性的值来扩展宏。因此，本例中的 roledn 将扩展为：

```
roledn = "ldap:///cn=DomainAdmins,ou=Engineering,dc=HostedCompany1,
dc=siroe,dc=com"
```

iPlanet Directory Server 随即按照正常的 ACI 评估算法对 ACI 进行评估。

如果属性为多值属性，就会使用各个值来扩展宏，然后使用第一个成功匹配的值。

请考虑下例：

```
dn: cn=Heather Blue, ou=People, dc=HostedCompany1, dc=siroe, dc=com
cn: Heather Blue
sn: Blue
ou: Engineering, dc=HostedCompany1, dc=siroe, dc=com
ou: People, dc=HostedCompany1,dc=siroe, dc=com
...
```

这种情况下，当 iPlanet Directory Server 评估 ACI 时，它将对下列扩展表达式执行逻辑 OR 操作：

```
roledn = "ldap:///cn=DomainAdmins,ou=Engineering,dc=HostedCompany1,
dc=siroe,dc=com"
```

```
roledn = "ldap:///cn=DomainAdmins,ou=People,dc=HostedCompany1,
dc=siroe,dc=com"
```

访问控制和复制

ACI 是作为条目属性存储的。因此，如果包含 ACI 的条目是所复制的数据库的一部分，则 ACI 将象其它任何属性那样被复制。

ACI 的评估始终在为进入的 LDAP 请求提供服务的 iPlanet Directory Server 上进行。这就意味着当客户服务器接收到更新请求时，在评估原版服务器是否能向该请求提供服务之前，它会向原版服务器返回引荐。

记录访问控制信息

要获取错误日志中有关访问控制的信息，则必须设置相应的日志级别。

要从控制台设置错误日志的级别：

1. 在控制台上，单击 “目录”选项卡，右键单击 config 节点，然后从弹出菜单中选择 “属性”。

   这样将为 `cn=config` 条目显示属性编辑器。

2. 向下滚动属性值对的列表，查找 `nsslapd-errorlog-level` 属性。

3. 为 `nsslapd-errorlog-level` 值字段中已显示的值加 128。

   例如，如果已显示的值为 8192 （复制调试），则应将值更改为 8320。有关错误日志级别的完整信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

4. 单击 “确定”以关闭属性编辑器。

与早期版本的兼容性

有些在早期版本的 iPlanet Directory Server 中使用的 ACI 关键字在 iPlanet Directory Server 5.1 中已不赞成使用。但出于向后兼容性的原因，目前系统仍支持这些关键字。这些关键字是：

- `userdnattr`
- `groupdnattr`

因此，如果已在传统供给服务器和客户 iPlanet Directory Server 5.1 之间建立复制协议，则在复制 ACI 时就不应遇到任何问题。

第 7 章

用户帐户管理

当用户连接到目录服务器时，首先将对用户进行身份验证。然后，目录可根据验证 (authentication) 过程中建立的标识向用户授予访问权限 (access rights) 和资源限制。

本章介绍用于用户帐户管理的任务，包括为目录配置口令和帐户锁定策略、拒绝用户组对目录的访问、根据其绑定 DN 限制用户可用的系统资源等。

本章包含以下几部分：

- 管理口令策略
- 去活用户和角色
- 基于绑定 DN 设置资源限制

管理口令策略

口令策略 (password policy) 通过执行以下操作，可以将使用口令的风险降到最低：

- 用户必须按计划更改口令
- 用户必须提供非一般性的口令

为目录建立口令策略后，即可通过配置帐户锁定策略来保护用户口令，避免它被潜在线程所使用。对于通过反复猜测用户口令而试图侵入目录的黑客而言，帐户锁定可保护帐户免遭这些黑客的攻击。

本部分提供有关配置口令和帐户锁定策略的信息。它包括以下过程：

- 第 244 页上的 “配置口令策略”
- 第 248 页上的 “设置用户口令”
- 第 249 页上的 “配置帐户锁定策略”
- 第 251 页上的 “在复制环境中管理口令策略”

配置口令策略

配置的口令策略将应用于目录中除目录管理员 (Directory Manager) 以外的所有用户。口令策略由以下信息组成：

口令添加和修改信息。口令信息包括口令语法和口令历史详细信息。

绑定信息。绑定信息包括跟踪绑定失败和口令时效属性。

本部分介绍下列有关配置口令策略的过程：

- 第 244 页上的 “使用控制台配置口令策略”
- 第 245 页上的 “使用命令行配置口令策略”

配置完口令策略之后，建议配置帐户锁定策略。有关配置帐户锁定策略的详细信息，请参阅第 249 页上的 “配置帐户锁定策略”。

使用控制台配置口令策略

要设置或修改 iPlanet Directory Server 的口令策略：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择数据节点。
2. 在右侧窗口中，选择 “口令”选项卡。

   该选项卡中包含 iPlanet Directory Server 的口令策略。
3. 通过选中 “复位后用户必须更改口令”复选框，可指定用户必须在首次登录时更改自己的口令。

   如果选中该复选框，则仅有目录管理员被授予重置用户口令的权限（使用步骤 9 中所述的字段）。普通管理用户不能强制用户更新其口令。

4. 要指定用户可更改自己的口令，请选中 “用户可更改口令”复选框。
5. 通过在 “允许更改：X 天”文本框中输入天数，可以指定用户在规定时间内不能更改其口令。
6. 要配置服务器以维护每个用户所用口令的历史记录列表，请选中 “保留口令的历史”复选框。在 “记住 X 个口令”文本框中，指定希望服务器为每个用户保留的口令个数。
7. 如果不想让用户口令过期，则选择 “口令没有过期”单选按钮。
8. 如果要强制用户必须定期更改口令，则选择 “口令到期之前 X 天”单选按钮，然后输入用户口令的有效天数。
9. 如果已选择 “口令到期之前 X 天”单选按钮，则需指定在口令到期之前多长时间向用户发送警告。在 “发送警告：X 口令到期之前的天数”文本框中输入口令到期之前多少天开始发送警告。
10. 如果要让服务器检查用户口令的语法以确保其满足在口令策略中设置的最低要求，则选中 “检查口令语法”复选框。然后，在 “口令最小长度”文本框中指定可以接受的最短口令长度。
11. 从 “口令加密”下拉菜单中指定存储口令时想让服务器使用的加密方法。

    有关加密方法的详细信息，请参见第 246 页的表 7-1 中的 `passwordStorageScheme` 属性。

    由于目录可根据其在目录中找到的现有加密方法动态地创建菜单，因此 “口令加密”菜单中可能还包含其它加密方法。
12. 更改完口令策略之后，单击 “保存”。

使用命令行配置口令策略

该部分介绍设置用于创建服务器口令策略的属性。使用 `ldapmodify` 更改 `cn=config` 条目中的这些属性。

下表描述了可用于配置口令策略的属性：

表 7-1 口令策略属性

属性名	定义
passwordMustChange	当为 on 时，该属性会要求用户在首次登录目录时或目录管理员重置口令后更改其口令。当为 on 时，即使已禁用用户定义的口令，还是会要求用户更改其口令。 如果选择将该属性设置为 off，则由目录管理员分配的口令就不会遵循任何明确的约定，且会比较难于查找。 该属性默认情况下为 off。
passwordChange	当为 on 时，该属性指示用户可更改自己的口令。如果选择让用户设置自己的口令，则要冒用户选择容易记忆的口令的风险。 但是，为用户设置好的口令需要管理员做出极大努力。而向用户提供对其无意义的口令时，用户就可能会将口令写在易于发现的地方，这是比较危险的。 该属性默认情况下为 on。
passwordExp	当为 on 时，该属性指示用户口令将在 passwordMaxAge 属性给定的时间间隔之后到期。使口令到期有助于保护目录数据，因为口令使用的时间越长，则它越有可能被发现。 该属性默认情况下为 off。
passwordMaxAge	该属性指示用户密码在多少秒后过期。要使用该属性，必须使用 passwordExp 属性启用口令到期功能。 常用策略是使口令每 30 至 90 天到期。默认情况下，口令的最长有效期设置为 8640000 秒 （100 天）。
passwordWarning	指示在用户口令即将到期以前多少秒向该用户发送警告消息。 根据 LDAP 客户机 (LDAP client) 应用程序，系统在发送警告时可能会提示用户更改其口令。 iPlanet Directory Express 和 Directory Server Gateway 都提供该功能。 默认情况下，目录将在口令到期前 86400 秒 （1 天）发送警告。但在设置警告消息之前，口令永不会过期。因此，如果用户绑定到 iPlanet Directory Server 的时间不超过 passwordMaxAge，则用户仍会及时获得更改口令的警告消息。

表 7-1 口令策略属性 *（续）*

属性名	定义
passwordCheckSyntax	当为 on 时，该属性指示在保存口令以前将由服务器检查该口令的语法。 检查口令语法可确认口令字符串达到或超过了最短口令长度要求，且该字符串中不包含任何一般性的词。一般性的词是指存储在用户条目的 uid、 cn、 sn、 givenName、 ou 或 mail 属性中的属性值。 该属性默认情况下为 off。
passwordMinLength	该属性指定口令中必须使用的最少字符数。口令越短，则破译起来就越容易。 您可以要求口令长度为 2 至 512 个字符。一般而言，长度为 6 至 8 个字符的口令已难于破译，而同时也短到使用户不用写下来即可记住。 该属性默认情况下为 6 个字符。
passwordMinAge	该属性指示用户在可更改自己的口令之前必须等待多少秒。该属性与 passwordInHistory 属性配合使用时，可防止用户重新使用旧口令。 例如，将口令的最短有效期设置为 2 天来防止用户在一个会话过程中反复更改自己的口令，从而避免用户轮换口令历史记录并重新使用已从历史记录中删除的旧口令。 可指定为 0 至 2147472000 秒（24,855 天）。值为 0 指示用户可立即更改口令。 该属性默认值为 0。
passwordHistory	该属性指示目录是否存储口令历史记录。设置为 on 时，目录将在历史记录的 passwordInHistory 属性中存储指定的口令数。如果用户试图重新使用某个口令，则将拒绝用户使用该口令。 将该属性设置为 off 时，则保留历史记录中存储的任何口令。如果该属性重新设置为 on，则在禁用该属性之前，用户将不能重新使用记录在历史记录中的口令。 该属性默认情况下为 off，意味着用户可重新使用旧口令。
passwordInHistory	该属性指示目录在历史记录中存储的口令数。可在历史记录中存储 2 至 24 个口令。除非将 passwordHistory 属性设置为 on，否则将无法启用该功能。 该属性默认情况下为 6 个字符。

表 7-1 口令策略属性 *（续）*

属性名	定义
passwordStorageScheme	该属性指定用于存储 Directory Server 口令的加密类型。iPlanet Directory Server 支持以下加密类型： • SSHA（经验安全散列算法）。推荐该方法的原因是因为它最安全。这是默认的方法。 • SHA（安全散列算法）。一种单向散列算法，是 Directory Server 4.x 中的默认加密模式。 • CRYPT。为实现与 UNIX 口令的兼容而提供的 UNIX 加密算法。 • clear。此加密类型指示口令将显示为纯文本。 注意：使用 CRYPT、 SHA 或 SSHA 格式存储的口令将无法用于通过 SASL Digest MD5 进行安全登录。 如果要提供自定义存储模式，请咨询 iPlanet 专业服务。

设置用户口令

对于任何条目而言，仅当该条目有 userpassword 属性且还未去活时，才可用于目录绑定。因为用户口令存储在目录中，所以可使用通常用于更新目录的 LDAP 操作来设置或重置用户口令。

有关创建和修改目录条目的信息，请参阅第 2 章 “创建目录项”。有关去活用户帐户的信息，请参阅第 252 页上的 “去活用户和角色”。

也可使用 Administration Server 或 iPlanet Directory Server Console 的 “用户和组”区域来设置或重置用户口令。有关如何使用 “用户和组”区域的信息，请参阅 Administration Server 中的可用在线帮助。有关如何使用 Gateway 创建或修改目录条目的信息，请参阅 Gateway 中的可用在线帮助。

配置帐户锁定策略

锁定策略与口令策略配合工作时，可以提高安全性。对于通过反复猜测用户口令而试图侵入目录的黑客而言，帐户锁定可保护帐户免遭这些黑客的攻击。帐户锁定计数器是目录服务器的局部功能。该功能未被设计为目录服务的全局锁定，也就是说，即使在复制环境中，帐户锁定计数器也不被复制。有关详细信息，请参阅第 251 页上的 “在复制环境中管理口令策略”。

可设置口令策略，从而使特定用户在绑定尝试失败次数达到给定数目后被锁定在目录外。

下列部分介绍如何配置帐户锁定策略：

- 第 249 页上的 “使用控制台配置帐户锁定策略”
- 第 250 页上的 “使用命令行配置帐户锁定策略”

使用控制台配置帐户锁定策略

要安装或修改 iPlanet Directory Server 的帐户锁定策略：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择数据节点。
2. 在右侧窗口中，选择 “帐户锁定”选项卡。
3. 要启用帐户锁定，则选中 “帐户可能被锁定”复选框。
4. 在 “锁定帐户之前 X 次登录失败”文本框中，输入允许绑定失败的最大次数。对于超出此处所指定限制值的用户，服务器将予以锁定。
5. 在 “复位失败，计数上限为 X 分钟”文本框中，输入将绑定失败计数器重置为 0 之前服务器要等待的分钟数。
6. 设置要将用户锁定在目录之外的时间间隔。

   选择 “永远锁定”单选按钮，从而在管理员重置用户口令之前一直锁定用户。

   通过选择 “锁定持续时间”单选按钮并在文本框中输入时间 （以分钟计），可以设置具体的锁定时间。
7. 更改完帐户锁定策略后，单击 “保存”。

使用命令行配置帐户锁定策略

本部分介绍的属性可用于创建帐户锁定策略，从而保护存储在服务器中的口令。使用 ldapmodify 更改 cn=config 条目中的这些属性。

下表列出了可用于配置帐户锁定策略的属性：

表 7-2 帐户锁定策略属性

属性名	定义
passwordLockout	该属性指示在绑定尝试失败次数达到给定数目后是否将用户锁定在目录之外。利用 passwordMaxFailure 属性，可以设定在锁定用户以前绑定尝试失败的次数。 您可以将用户锁定一段时间，也可以设定在管理员重置口令以前一直锁定用户。 该属性默认情况下设置为 off，意味着用户将不会被锁定在目录之外。
passwordMaxFailure	该属性指示在将用户锁定于目录之外以前绑定尝试失败的次数。 该属性仅当 passwordLockout 属性设置为 on 时生效。 该属性默认情况下为 3 次绑定失败。
passwordLockoutDuration	该属性指示用户被锁定在目录之外的时间 （以秒计）。使用 passwordUnlock 属性，也可指定在管理员重置用户的口令之前一直锁定用户。 默认情况下将锁定用户 3600 秒。
passwordResetFailureCount	该属性指定重置口令故障计数器之前等待的时间 （以秒计）。 每次从用户帐户发送无效的口令时，口令故障计数器都会增加 1。如果将 passwordLockout 属性设置为 on，则当计数器达到 passwordMaxFailure 属性指定的失败数时，用户将被锁定在目录之外。帐户将在 passwordLockoutDuration 属性指定的时间间隔内被锁定，之后故障计数器将被重置为 0。 因为计数器的用途是当黑客设法获得访问系统的权限时进行测量，所以计数器必须保持较长的时间，以便检测到黑客。但是，如果计数器在一段时间内无限增加，则可能会在无意中锁定有效用户。 重置口令故障计数器属性默认情况下被设置为 600 秒。

在复制环境中管理口令策略

如下所示，复制环境中将强制使用口令和帐户锁定策略：

- 在原版数据服务器 (data master) 上强制使用口令策略。
- 在所有参与复制 (replication) 环境的服务器上强制使用帐户锁定。

目录中的部分口令策略信息将会被复制。所复制的属性包括：

- `passwordMinAge` 和 `passwordMaxAge`
- `passwordExp`
- `passwordWarning`

但是，配置信息将保存在本地且不被复制。这些信息包括口令语法和口令修改的历史记录。帐户锁定计数器也不会被复制。

在复制环境中配置口令策略时，需要考虑以下几点：

- 来自口令即将到期的服务器的警告将由所有副本发布。该信息将在每台服务器上本地保存到。因此，如果用户依次绑定到几个副本上，就会多次接收到相同的警告信息。此外，如果用户更改口令，则将该信息过滤给副本时可能会占用较长的时间。如果用户更改口令并立即重新绑定，则在副本注册这些更改之前，用户可能会发现绑定失败。
- 您可能希望所有的服务器发生相同的绑定行为 （包括原版和副本）。请确保每台服务器上创建的口令策略配置信息相同。
- 在多原版环境中，帐户锁定计数器的工作方式可能与预计的有所不同。
- 为复制所创建的条目 （例如服务器身份）需要有永不到期的口令。要确保这些特殊用户拥有永不到期的口令，请向条目中添加 `passwordExpirationTime` 属性，同时为其赋值 `20380119031407Z` （有效范围的上限）。

去活用户和角色

您可以临时去活单个用户帐户或一组帐户。去活后，用户即无法绑定到目录。身份验证操作将失败。

使用操作属性 nsAccountLock 可以去活用户和角色。条目中包含值为 true 的 nsAccountLock 属性时，服务器将拒绝绑定。

去活用户和角色的操作过程相同。但在去活角色时，所去活的是角色 (role) 成员，而不是角色条目本身。有关角色的详细信息，尤其是角色与访问控制之间相互作用的详细信息，请参阅第 5 章 “高级条目管理”。

本部分的后续内容将介绍以下过程：

- 第 252 页上的 “使用控制台去活用户和角色”
- 第 253 页上的 “使用命令行去活用户和角色”
- 第 253 页上的 “使用控制台激活用户和角色”
- 第 254 页上的 “使用命令行激活用户和角色”

警告 数据库上的根条目 （对应于根后缀或子后缀的条目）是无法去活的。

有关创建根后缀或子后缀条目的详细信息，请参阅第 2 章 “创建目录项”。有关创建根后缀或子后缀的详细信息，请参阅第 3 章 “配置目录数据库”。

使用控制台去活用户和角色

下列过程说明如何使用控制台来去活用户或角色：

1. 在 Directory Server Console 中，选择 “目录”选项卡。
2. 浏览左侧导航窗口中的导航树，然后双击要去活的用户或角色。

   此时显示 “编辑项目”对话框。

   作为一种捷径，也可从 “对象”菜单中选择 “去活”。
3. 单击左侧窗口中的 “帐户”。右侧窗口指示角色或用户已处于去活状态。单击 “激活”可激活用户或角色。
4. 单击 “确定”以关闭对话框并保存更改。

   去活后，通过从 “视图”菜单中选择 “去活状态”，即可查看对象的状态。对象的图标随即出现在控制台的右侧窗口中，上面有一道红色斜杠。

使用命令行去活用户和角色

要去活用户帐户，请使用 `/usr/sbin/directoryserver account-inactivate` 命令：

```
# /usr/sbin/directoryserver account-inactivate
```

下例说明如何使用此命令来去活 Joe Frasier 的用户帐户：

```
/usr/sbin/directoryserver account-inactivate -h server.siroe.com \
                -p 389 -D "cn=Directory Manager" -w password \
                 -I "uid=jfrasier,ou=people,dc=siroe,dc=com"
```

下表列出示例中所用的选项：

表 7-3 示例中所用的 account-inactivate 选项说明

选项	说明
`-h`	命名目录服务器的主机。
`-p`	指定目录服务器使用的端口。
`-D`	指定目录管理员的 DN。
`-w`	指定目录管理员的口令。
`-I`	指定要去活的用户帐户或角色的 DN。

使用控制台激活用户和角色

下列过程说明如何使用控制台来激活用户或角色：

1. 在 Directory Server Console 中，选择 “目录”选项卡。
2. 浏览左侧导航窗口中的导航树，然后双击所要激活的用户或角色。

   此时显示 “编辑项目”对话框。

   作为一种捷径，也可从 “对象”菜单中选择 “激活”。
3. 单击左侧窗口中的 “帐户”。右侧窗口指示角色或用户已处于激活状态。单击 “激活”可激活用户或角色。

4. 如果用户或角色是另一个去活角色的成员，则控制台中将显示用于查看去活角色的选项。单击 “显示不活动的角色”以查看用户或角色所属的角色列表。

5. 完成后，单击 “确定”。

   重新激活后，可通过从 “视图”菜单中选择 “去活状态”来查看对象的状态。控制台右侧窗口中的角色或用户图标将正常显示。贯穿图标、表明处于非活动状态的红色斜杠将消失。

使用命令行激活用户和角色

要激活用户帐户，请使用 `/usr/sbin/directoryserver account-activate` 命令：

```
# /usr/sbin/directoryserver account-activate
```

下例说明如何使用 `account-activate` 命令来激活 Joe Frasier 的用户帐户：

```
/usr/sbin/directoryserver account-activate -h server.siroe.com \
               -p 389 -D "cn=Directory Manager" -w password \
               -I "uid=jfrasier,ou=people,dc=siroe,dc=com"
```

下表说明示例中所用的选项：

表 7-4 示例中所用的 account-activate 选项说明

选项	说明
`-h`	命名目录服务器的主机。
`-p`	指定目录服务器使用的端口。
`-D`	指定目录管理员的 DN。
`-w`	指定目录管理员的口令。
`-I`	指定要激活的用户帐户或角色的 DN。

基于绑定 DN 设置资源限制

对于绑定到目录上的客户机应用程序而言，可以使用特殊的操作属性值来控制服务器对搜索操作的限制。可设置以下搜索操作限制：

审核限制。指定搜索操作中所检查的条目数。

大小限制。指定服务器响应搜索操作而返回给客户机应用程序的最大条目数。

时间限制。指定处理搜索操作时服务器所用的最长时间。

空闲超时。指定断开连接前，服务器的连接处于空闲状态的时间。

注意 默认情况下，目录管理员可接收无限的资源。

为客户机应用程序设置的资源限制优先于在全局服务器配置中设置的缺省资源限制。

本部分说明以下过程：

- 第 255 页上的 “使用控制台设置资源限制”
- 第 256 页上的 “使用命令行设置资源限制”

使用控制台设置资源限制

下列过程说明如何使用控制台来设置用户或角色的资源限制：

1. 在 Directory Server Console 中，选择 “目录”选项卡。
2. 浏览左侧导航窗口中的导航树，然后双击要为其设置资源限制的用户或角色。

   此时显示 “编辑项目”对话框。
3. 单击左侧窗口中的 “帐户”。右侧窗口中包含可在 “资源限制”部分设置的四种限制。

   输入值为 -1 时表示无限制。
4. 完成后，单击 “确定”。

使用命令行设置资源限制

使用命令行可以为每个条目设置下列操作属性。使用 ldapmodify 可以将以下属性添加到条目中：

属性	说明
nsLookThroughLimit	指定搜索操作中所检查的条目数。指定为条目数。赋予该属性值为 -1 时表示无限制。
nsSizeLimit	指定服务器响应搜索操作而返回给客户机应用程序的最大条目数。赋予该属性值为 -1 时表示无限制。
nsTimeLimit	指定处理搜索操作时服务器所用的最长时间。赋予该属性值为 -1 时表示无时间限制。
nsIdleTimeout	指定断开连接前服务器连接处于空闲状态的时间。值的单位为秒。赋予该属性值为 -1 时表示无限制。

例如，通过执行 ldapmodify 可以设置大小限制，如下所示：

```
ldapmodify -h myserver -p 389 -D "cn=directory manager" -w secretpwd
dn: uid=bjensen,ou=people,dc=siroe,dc=com
changetype: modify
add:nsSizeLimit
nsSizeLimit: 500
```

ldapmodify 语句将 nsSizeLimit 属性添加到 Babs Jensen 条目中，并将其搜索返回的大小限制为 500 个条目。

第 8 章

管理复制

复制是一种拓展目录服务，从而使之超过单服务器配置限制的重要机制。本章介绍为设置单原版复制、多原版复制和级联复制而在供给服务器和客户服务器上执行的任务。本章包含下列主题：

- 复制概述
- 复制环境
- 复杂复制配置的步骤概要
- 详细的复制任务
- 配置单原版复制
- 配置多原版复制
- 配置级联复制
- 删除更改日志
- 初始化客户
- 保持副本同步
- SSL 环境下的复制
- 早期版本的复制
- 使用回退更改日志插件
- 监控复制状态
- 解决常见复制冲突

有关如何在目录部署中使用复制的概念性信息，请参阅 *iPlanet Directory Server 部署指南*。

复制概述

复制就是在 iPlanet Directory Server 之间自动复制目录数据的机制。任何类型的更新—条目添加、修改甚至删除 — 都将被自动镜像到其它使用复制的 iPlanet Directory Server 上。本部分包含有关下列复制概念的信息：

副本

我们将参与复制过程的数据库定义为*副本*。副本有下面几种类型：

- 原版副本：包含原版目录数据的读写数据库。原版副本可以处理来自目录客户机的更新请求。
- 客户副本：包含在原版副本中保留的信息的只读数据库。客户副本可以处理来自目录客户机的搜索请求，但是将更新请求指向原版副本。
- 中枢副本：与客户副本非常相似的只读数据库。区别在于，该数据库存储的信息供 iPlanet Directory Server 使用，而后者同时充当复制信息的客户服务器和供给服务器 （中枢）。

您可以配置 iPlanet Directory Server 以管理多个数据库。每个数据库可以充当不同的复制角色。例如，可以让 iPlanet Directory Server 将 `dc=engineering,dc=siroe,dc=com` 后缀储存在原版副本中，将 `dc=sales,dc=siroe,dc=com` 后缀储存在客户副本中。

供给器/客户

如果一个服务器管理被复制到其它服务器的原版副本，则该服务器称为*供给器 (supplier) 服务器*或*原版服务器*。如果一个服务器管理被其它服务器更新的客户副本，则该服务器称为*客户 (consumer) 服务器*。

尽管由于服务器可以同时充当供给器和客户这一功能使得将服务器角色分别称作供给器和客户欠缺准确，但它仍不失为一个方便的方法。在下列情况下使用这种称呼很方便：

- 当 iPlanet Directory Server 管理原版副本和客户副本的组合时；
- 当 iPlanet Directory Server 用作*中枢供给器 (hub supplier)* 时，也就是它从原版服务器接收更新，然后将更改复制到客户服务器。有关详细信息，请参阅第 267 页上的 “级联复制”。
- 在多原版复制中，当原版副本位于两个不同的 iPlanet Directory Server 上时：每个 iPlanet Directory Server 同时充当另一个 iPlanet Directory Server 的供给器和客户。有关详细信息，请参阅第 264 页上的 “多原版复制”。

在 iPlanet Directory Server 5.1 中，复制始终由供给服务器启动，而非由客户服务器启动。这种操作称为供给器启动的复制 (supplier-initiated replication)。它允许对供给服务器进行配置，从而将数据推送到一个或多个客户服务器。

iPlanet Directory Server 的早期版本允许使用客户启动的复制 (consumer-initiated replication)，这种方法可以配置客户服务器，以便从供给服务器中获取数据。在 iPlanet Directory Server 5.1 中，这种方法已被客户服务器提示供给器发送更新的过程取代。有关该功能的详细信息，请参阅第 297 页上的 “保持副本同步”。

更改日志

每个供给服务器都维护一份*更改日志 (change log)*。更改日志是描述原版副本修改内容的记录。供给服务器随即会将这些修改内容在存储于客户服务器的副本上进行重现。在多原版复制的情况下，则在其它原版上进行重现。

修改条目后，描述所执行的 LDAP 操作的更改记录将被记录到更改日志中。

配置更改日志的方法与配置常规 LDBM 数据库的方法相同。

在 iPlanet Directory Server 5.0 中，更改日志的格式已有所修改。在 iPlanet Directory Server 的早期版本中，更改日志通过 LDAP 访问。但是，现在该功能仅面向服务器内部使用。如果应用程序需要阅读更改日志，则可使用回退更改日志插件以实现向后兼容。有关详细信息，请参阅第 302 页上的 “使用回退更改日志插件”。

复制单位

在 iPlanet Directory Server 5.1 中，复制的最小单位是数据库。这就意味着您可以复制整个数据库，但不能复制数据库内的子树。因此，当创建目录树时，必须考虑复制计划。有关如何设置目录树的详细信息，请参阅 *iPlanet Directory Server 部署指南*。

复制机制还要求一个数据库对应一个后缀。这就是说，无法使用自定义分布逻辑复制在两个或多个数据库上分布的后缀 （或名称空间）。有关该主题的详细信息，请参阅第 80 页上的 “创建和维护数据库”。

复制标识

当两个服务器之间进行复制操作时，客户服务器在供给器绑定以便发送复制更新时对其进行验证。以便发送复制更新。该验证过程要求供给器用来绑定到客户的条目需储存在客户服务器上。该条目被称为复制管理器条目或供给器绑定 DN。

复制管理器条目或创建用于执行该角色的任何条目都必须符合以下标准：

- 在每个管理客户副本 （或中枢副本）的服务器上必须至少有一个条目。
- 由于安全原因，该条目不得为所复制的数据库的组成部分。

注意 该条目有一个忽略客户服务器上所定义的所有访问控制规则的特殊用户配置文件。

在两个服务器之间配置复制时，必须在两个服务器上都标识复制管理器 （供给器绑定 DN）：

- 在客户服务器或中枢供给器上配置客户副本或中枢副本时，必须指定一个或多个供给器绑定 DN，并且这些 DN 与授权执行复制更新的条目对应。
- 在供给服务器上，当配置复制协议时，必须在协议中指定该条目的 DN。

注意 在 iPlanet Directory Server Console 中，这个复制管理器条目被称为 *供给器绑定 DN*，这可能会带来某些误导，因为该条目并不位于供给服务器上。该条目之所以被称为供给器绑定 DN 是因为它必须出现在客户服务器上，这样它可以在供给器绑定时对其进行验证，从而向客户服务器提供复制更新。

有关创建复制管理器条目的详细信息，请参阅第 270 页上的 “创建供给器绑定 DN 条目”。

复制协议

iPlanet Directory Server 使用复制协议定义它们的复制配置。复制协议 (replication agreement) 描述仅在一个供给器和一个客户之间进行的复制。该协议在供给服务器中配置。它指定：

- 所要复制的数据库
- 要将数据发送到的客户服务器
- 复制发生的时间
- 供给服务器必须用于绑定的 DN 和凭证 （称为复制管理器条目或供给器绑定 DN）
- 确保连接安全的方式 （SSL、客户机验证）

与 iPlanet Directory Server 早期版本的兼容性

iPlanet Directory Server 5.0 和 5.1 中的复制机制不同于 iPlanet Directory Server 早期版本中所用的机制。兼容性的实现方式：

- 旧复制插件
- 回退更改日志插件

旧复制插件使 iPlanet Directory Server 5.1 在客户角色中充当 4.x 目录服务器。有关如何使用该插件执行旧复制任务的信息，请参阅第 300 页上的 “早期版本的复制”。

当希望 iPlanet Directory Server 5.1 供给器维护 4.x 类型的更改日志时，可以使用回退更改日志插件。对于依赖 iPlanet Directory Server 4.x 更改日志格式的应用程序（比如 iPlanet Meta Directory）而言，有时则是必须的，因为它们从更改日志中读取信息。有关回退更改日志插件的详细信息，请参阅第 302 页上的 “使用回退更改日志插件”。

复制环境

本部分介绍最常用的复制环境：

- 第 262 页上的 “单原版复制”
- 第 264 页上的 “多原版复制”
- 第 267 页上的 “级联复制”

可以对这些基本环境进行组合，从而建立最适合自己需要的复制环境。

注意 无论选择实现何种复制环境，都应考虑模式复制。有关详细信息，请参阅 *iPlanet Directory Server 部署指南*。

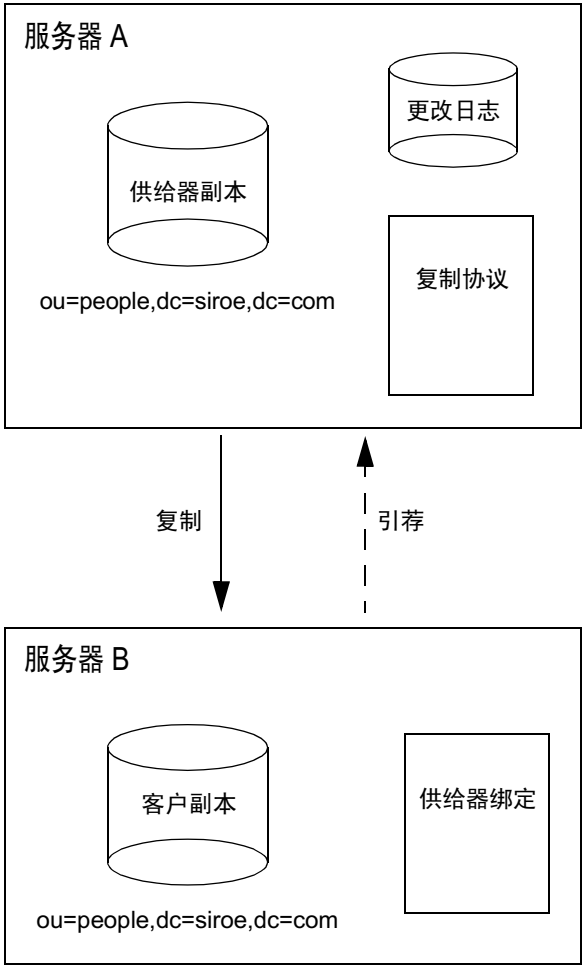
单原版复制

在最简单的复制环境中，原版的目录数据将保存在一台服务器 （称为供给服务器）的单个原版副本上。该服务器向储存在客户服务器上的客户副本提供更新。

供给服务器维护记录对原版副本的所有更改的更改日志。供给服务器还储存复制协议。

客户服务器储存与供给器绑定 DN 对应的条目，这样它可以在供给器绑定以发送复制更新时对其进行验证。

图 8-1 单原版复制



在图 8-1 说明的示例中， `ou=people,dc=siroe,dc=com` 后缀接收大量来自客户机的搜索和更新请求。因此，为分担负荷，原版位于服务器 A 上的后缀将被复制到位于服务器 B 的客户副本中。

服务器 B 可以处理和响应来自客户机的搜索请求，但无法处理修改目录条目的请求。服务器 B 通过向客户机返回到服务器 A 的引荐，处理从客户机接收的修改请求。

注意 在复制中，客户服务器储存关于供给服务器的引荐信息，但不把来自客户机的修改请求转发给供给服务器。客户机必须遵循客户服务器返回的引荐。

有关如何配置单原版复制环境的信息，请参阅第 277 页上的 “配置单原版复制”。

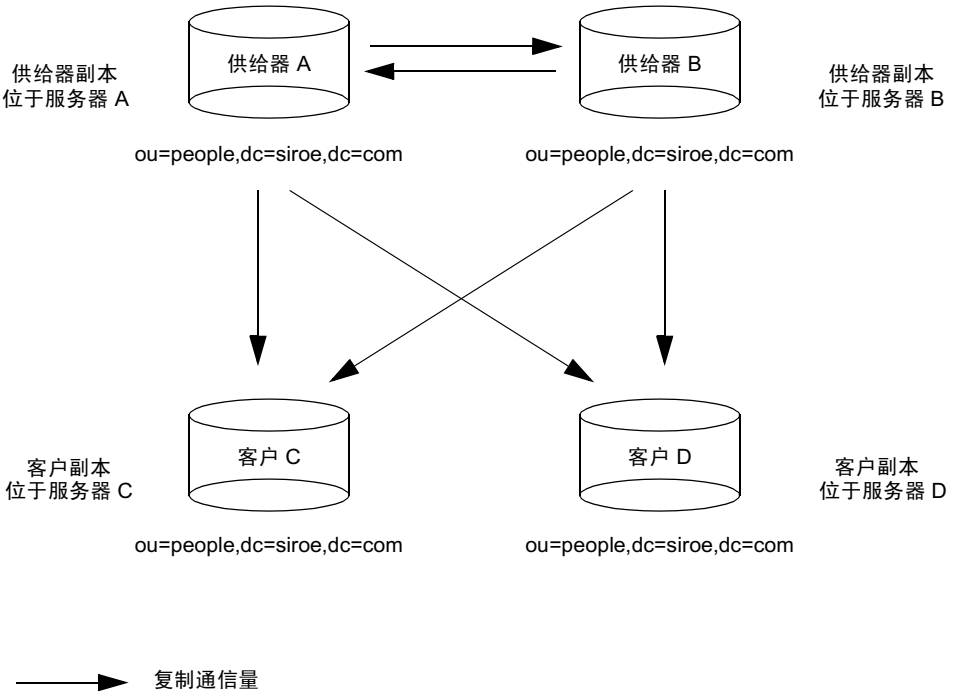
多原版复制

iPlanet Directory Server 5.1 也支持复杂的复制环境，其中同一信息的原版可位于两台服务器上。该信息被保存在每台服务器的原版副本上。这就是说，每台服务器都会维护副本的更改日志。

此类配置可与任意数量的客户服务器一起协同工作。客户可以接收来自两个供给器的更新。客户还为这两个供给器定义引荐。这样的环境称为多原版配置。

图 8-2 显示了多原版复制环境的一个示例。有关需要为多原版复制设置的复制协议、更改日志和供给器绑定 DN 的详细视图，请参阅图 8-3。

图 8-2 多原版复制

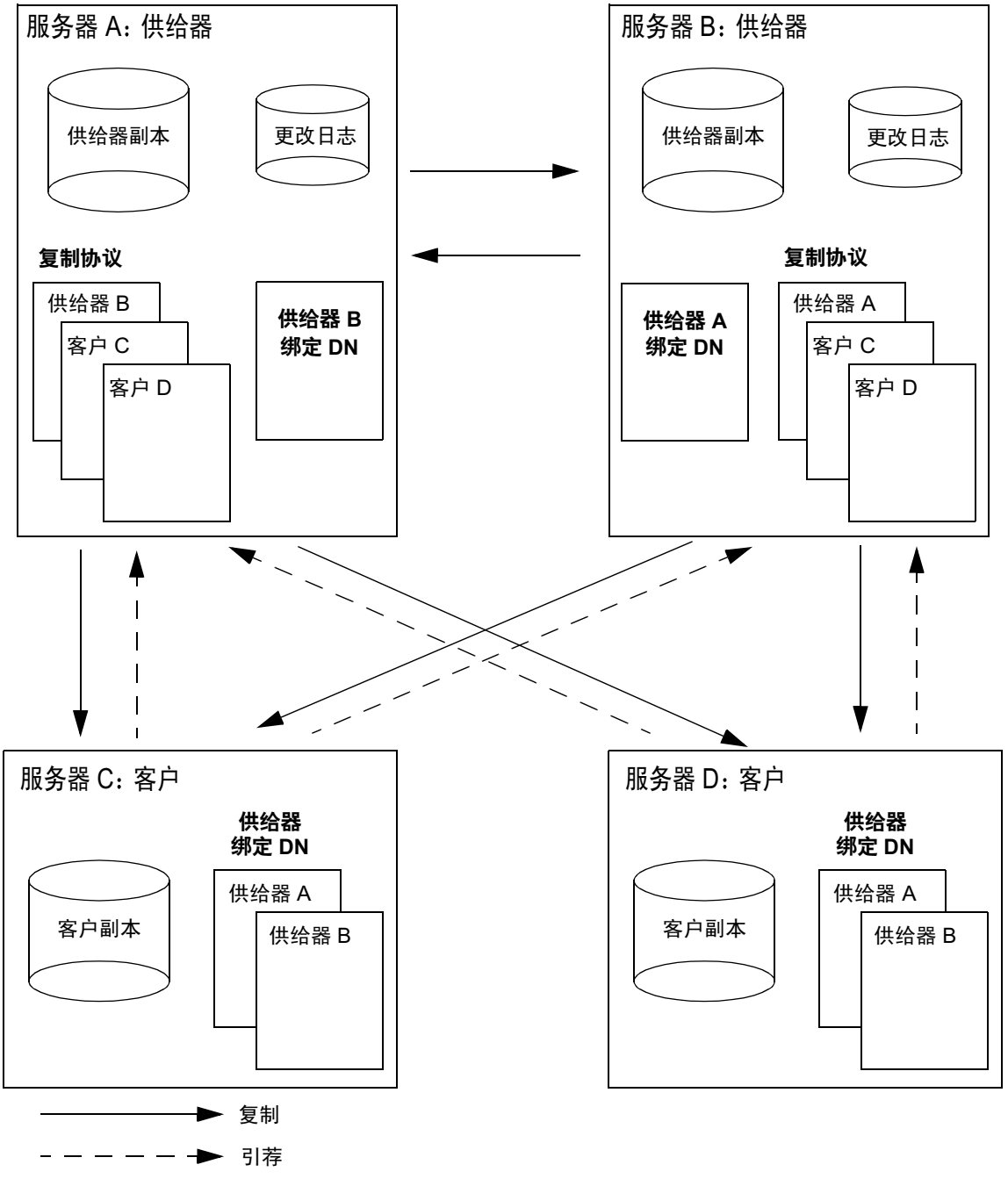


多原版配置具有以下优势：

- 当一个供给器不可访问时，系统就会自动进行写入故障替换
- 在按地理位置分布的环境中，更新将在本地供给器上进行

注意 复制，特别是多原版复制，在高速链路上比在低速链路 （例如在按地理位置分布的环境中使用 WAN）上工作得更好。

图 8-3 多原版复制的详细视图



在图 8-3 说明的示例中，要确保 ou=people,dc=siroe,dc=com 后缀始终可用于修改操作，其原版必须位于两台供给服务器上。每台供给服务器维护自己的更改日志。当其中一个原版处理来自客户机的修改请求时，它在自己的更改日志中记录下操作，然后将复制更新发送给其它供给服务器和客户。

这就是供给服务器互相之间以及与客户服务器之间需要有复制协议的原因。每台供给服务器还储存一个绑定 DN，它允许其它原版绑定以提供复制更新。

在该示例中，每台客户服务器储存两个对应于供给器绑定 DN 的条目，这样，它可以在供给器绑定以发送复制更新时对其进行验证。每个客户都可能只有一个对应于供给器绑定 DN 的条目。在这种情况下，两个供给器使用相同的供给器绑定 DN 进行绑定。

在多原版复制环境中，当客户服务器接收到来自客户机的修改请求时，它将向客户机返回到两台供给器的引荐。

注意 客户服务器储存有关供给服务器的引荐信息。客户服务器并不将来自客户机的修改请求转发给供给器。客户机必须遵循客户服务器返回的引荐。

有关设置具有两台供给服务器和两台客户服务器的多原版复制的信息，请参阅第 281 页上的 “配置多原版复制”。

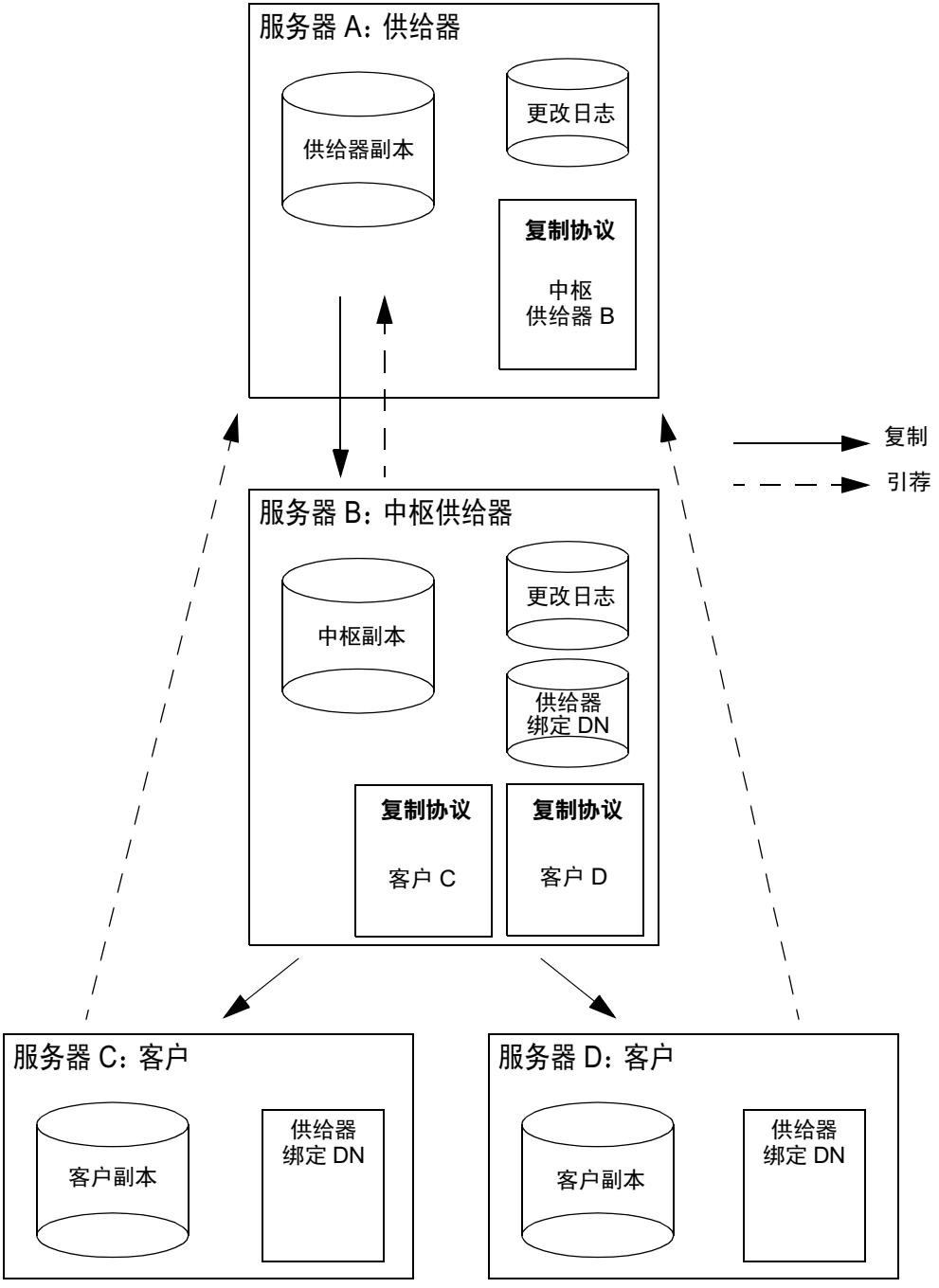
级联复制

在级联复制环境中，一个通常称为*中枢供给器*的服务器将对于特定的副本既充当客户又充当供给器。对于持有数据原版副本的供给服务器而言，它将接收来自该供给服务器的更新，并继而将这些更新提供给客户。级联复制在下列情况中非常有用：

- 当需要平衡巨大的通信量负载时：例如，因为供给服务器需要处理所有的更新通信量，所以它在承担沉重负载的情况下还需要支持到客户服务器的所有复制通信量。您可以将复制通信量卸载到中枢服务器，它可以为大量客户提供复制更新服务。
- 通过在按地理位置分布的环境中使用本地中枢供给器，降低连接开销。
- 提高目录服务的性能：如果将所有执行读取操作和所有执行更新操作的客户机应用程序分别定向到客户和供给器，则可以从中枢服务器删除所有索引（除系统索引外）。这将显著提高供给器和中枢服务器之间的复制速度。

图 8-4 显示了级联复制的一个示例。本例显示的是一个简单的级联复制环境。您可以用若干个中枢供给器和数量众多的客户创建更复杂的环境。

图 8-4 级联复制


服务器 A：供给器
供给器副本
更改日志
复制协议
中枢
供给器 B
复制
引荐
服务器 B：中枢供给器
中枢副本
更改日志
供给器
绑定 DN
复制协议
客户 C
复制协议
客户 D
服务器 C：客户
客户副本
供给器
绑定 DN
服务器 D：客户
客户副本
供给器
绑定 DN

在图 8-4 说明的示例中，中枢供给器用于平衡复制更新的负载，方法是在供给服务器和中枢供给器之间共享负载。

供给服务器和中枢供给服务器共同维护一个更改日志。但是，只有供给服务器才可以处理来自客户机的目录修改请求。

客户服务器和中枢供给器可以处理从客户机接收的搜索请求，但对于修改请求，它们将向客户机返回到供给服务器的引荐。图 8-4 说明客户服务器 C 和 D 含有到供给服务器 A 的引荐。这些引荐是在配置客户副本期间指定供给服务器时创建的自动引荐。

注意 客户服务器和中枢供给器存储有关供给服务器的引荐信息。它们并不将来自客户机的修改请求转发给供给器。客户机必须遵循客户服务器返回的引荐。

有关设置级联复制的信息，请参阅第 286 页上的 “配置级联复制”。

注意 可以将多原版复制和级联复制组合使用。例如，在第 266 页的图 8-3 中所示的多原版环境中，服务器 C 和服务器 D 可以是被复制到任意数目的客户服务器中的中枢供给器。

复杂复制配置的步骤概要

如果是在为大量服务器配置复制功能，且配置工作相对复杂，则为了提高效率，应按以下顺序进行：

1. 在所有客户服务器上：
   - 创建副本数据库
   - 创建至少一个复制管理器或供给器绑定 DN 条目
   - 指定客户副本的设置
2. 在所有中枢供给器上：
   - 创建副本数据库
   - 创建复制管理器或供给器绑定 DN 条目
   - 为复制指定供给器设置 （包括更改日志配置）
   - 指定中枢副本的设置

3. 在所有供给器上：
   - 创建副本数据库
   - 为复制指定供给器设置 （包括更改日志配置）
   - 指定供给器副本的设置
4. 配置所有供给器上的复制协议：
   - 在多原版集的供给器之间
   - 在供给器和客户之间
   - 在供给器和中枢供给器之间

   也可选择在该阶段初始化客户服务器和中枢供给器上的副本。在多原版复制中，从一个供给器副本初始化另一个供给器副本。千万不可试图一起初始化所有供给器副本。
5. 在所有中枢供给器上，配置中枢供给器和指定客户之间的复制协议。

   也可选择在该阶段初始化客户服务器上的副本。

注意 在试图创建复制协议前，创建和配置所有副本非常重要。这也意味着在创建复制协议时，可以选择立即初始化客户副本。客户初始化始终是设置复制的最后一个阶段。

详细的复制任务

本部分介绍配置复制时需要执行的任务。

创建供给器绑定 DN 条目

设置复制时，关键的部分是创建供给器用于绑定到客户服务器以执行复制更新所需的条目 （称为复制管理器条目或供给器绑定 DN 条目）。

供给器绑定 DN 必须满足的标准和特性在第 260 页上的 “复制标识”中说明。

要创建供给器绑定条目：

1. 在每台充当复制协议中客户的服务器上，创建供给器用于绑定时所需的特殊条目。

   该条目不得为所复制的数据库的组成部分。例如，可以使用 `cn=Replication Manager,cn=config`。确保所建条目中包含复制协议中指定的验证方法所需的属性。

2. 指定 `userPassword` 属性 – 值对。

3. 如果已启用口令到期策略 （或打算将来如此），则必须牢记要禁用该策略，避免复制因口令到期而失败。要禁用 `userPassword` 属性的口令到期策略，需要添加值为 20380119031407Z 的 `passwordExpirationTime` 属性，这就使得口令永远不会过期。

配置客户副本时，必须使用该条目的 DN 来定义供给器绑定 DN。

注意 该供给器绑定 DN 对应于有权限的用户，因为他们不受访问控制的限制。

配置供给器设置

在持有供给器副本或中枢副本的任何服务器上，必须指定供给器设置，即更改日志参数。

要配置供给器设置：

1. 在 Directory Server Console 中，单击 “配置”选项卡。

   有关启动 Directory Server Console 的详细信息，请参阅第 26 页上的 “使用 iPlanet Directory Server Console”。

2. 在左侧的导航树中，突出显示 “复制”节点。

3. 在右侧导航窗口中，单击 “供给器设置”选项卡。

4. 选中 “启用更改日志”复选框。

   这将激活下面窗口中以前不可用的所有字段。

5. 单击 “使用默认值”按钮以指定更改日志，或者单击 “浏览”以显示文件选择器。

6. 设置更改日志的数目和存在周期参数。

   要指定不同的值，则必须清除 “不限制”复选框。

7. 单击 “保存”以保存目录服务器的供给器设置。

配置供给器副本

对于每一个供给器副本而言，必须指定适当的复制设置。

要配置供给器副本：

1. 在 Directory Server Console 中，单击 “配置”选项卡。

   有关启动 Directory Server Console 的信息，请参阅第 26 页上的“使用 iPlanet Directory Server Console”。

2. 在左侧的导航树中，展开 “复制”文件夹，然后突出显示所要复制的数据库。

   “副本设置”选项卡将显示在右侧的导航窗口中。

3. 选中 “启用副本”复选框。

4. 在 “副本角色”部分中，选择 “单原版”或 “多原版”单选按钮。

5. 在 “通用设置”部分中，指定副本 ID （1 到 65534 之间的整数，包括首尾数字）。

   对于每个供给器副本而言，副本 ID 必须唯一。确保指定的 ID 不同于该服务器及其它服务器上其它供给器副本所用的 ID。

6. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

   该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

7. 单击 “保存”以保存数据库的复制设置。

配置客户副本

对于每一个客户副本而言，必须指定适当的复制设置。

1. 在 Directory Server Console 中，单击 “配置”选项卡。

   有关启动 Directory Server Console 的信息，请参阅第 26 页上的 “使用 iPlanet Directory Server Console”。

2. 在左侧的导航树中，展开 “复制”文件夹，然后突出显示副本数据库。

   “副本设置”选项卡将显示在右侧的导航窗口中。

3. 选中 “启用副本”复选框。

4. 在 “副本角色”部分中，选择 “指定客户”单选按钮。

5. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

   该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

   由于无需指定客户的复制 ID （对于所有客户副本而言，它被自动设置为 65535），因此 “复制 ID”字段一直处于不可用状态。

6. 在 “更新设置”部分中，指定供给器用于绑定到副本所需的供给器绑定 DN （复制管理器 DN）。

   第一次配置副本时，“当前的供给器 DN”列表为空。每个副本可以有多个供给器绑定 DN，但每个复制协议只有一个供给器 DN。

   要指定新的供给器绑定 DN：

   a. 在相应的字段中输入新的供给器绑定 DN。

      所输入的 DN 必须对应于在客户服务器上创建的条目 （例如， `cn=Replication Manager,cn=config`）。

   b. 单击 “添加”。

      新的供给器绑定 DN 将直接出现在上述 “当前的供给器 DN”列表中。

   c. 为每个要添加到列表中的供给器绑定 DN 重复以上操作。

7. 指定更新所要指向的服务器的 LDAP URL。

   第一次配置副本时，“当前的引荐 URL”列表为空。默认情况下，该列表不列出存放数据副本的原版的服务器的 URL （客户服务器自动创建这个引荐）。

   自动引荐假定客户机将通过常规连接来进行绑定，因此格式为 `ldap://servername:port`。如果想使用 SSL 将客户机绑定到供给器，则可使用
   该字段来指定以下格式的引荐：`ldaps://servername:port` （其中， `ldaps` 中的 `s` 表示安全连接）。

   如果为引荐指定 LDAP URL，则目录服务器首先将修改请求指向所指定的 URL。如果不指定，则修改请求将指向包含当前副本的供给器。

   要为引荐指定一个新的 URL：

   a. 在相应的字段中输入新的 LDAP URL，或者单击 “构造”以显示一个帮助构建 LDAP URL 的对话框。

   b. 单击 “添加”。

      新的 LDAP URL 将直接出现在上述 “当前的引荐 URL”列表中。

   c. 为每个要添加到列表中的引荐重复以上操作。

8. 单击 “保存”以保存副本的复制设置。

配置中枢副本

在级联复制环境中，按如下所述配置中枢供给器：

1. 在 Directory Server Console 中，单击 “配置”选项卡。

   有关启动 Directory Server Console 的信息，请参阅第 26 页上的“使用 iPlanet Directory Server Console”。

2. 在左侧的导航树中，展开 “复制”文件夹，然后突出显示所要复制的数据库。

   “副本设置”选项卡将显示在右侧的导航窗口中。

3. 选中 “启用副本”复选框。

4. 在 “副本角色”部分中，选择 “中枢”单选按钮。

5. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

   该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

   由于无需指定中枢供给器的复制 ID （与客户副本相似，它被自动设置为 65535），因此 “复制 ID”字段一直处于不可用状态。

6. 在 “更新设置”部分中，指定供给器用于绑定到中枢副本所需的供给器绑定 DN （复制管理器 DN）。

   第一次配置副本时，“当前的供给器 DN”列表为空。每个副本可以有多个供给器绑定 DN，但每个复制协议只有一个供给器 DN。

   要指定新的供给器绑定 DN：

   a. 在相应的字段中输入新的供给器绑定 DN。

      所输入的 DN 必须对应于在客户服务器上创建的条目（例如， `cn=Replication Manager,cn=config`）。

   b. 单击 “添加”。

      新的供给器绑定 DN 将直接出现在上述 “当前的供给器 DN”列表中。

   c. 为每个要添加到列表中的供给器绑定 DN 重复以上操作。

7. 指定更新所要指向的服务器的 LDAP URL。

   第一次配置副本时，“当前的引荐 URL”列表为空。默认情况下，该列表不显示存放数据副本的原版的服务器的 URL （中枢服务器自动创建这个引荐）。

   自动引荐假定客户机将通过常规连接来进行绑定，因此格式为 `ldap://`*servername*`:`*port*。如果想使用 SSL 将客户机绑定到供给器，则可使用该字段来指定以下格式的引荐：`ldaps://servername:port` （其中， `ldaps` 中的 `s` 表示安全连接）。

   如果为引荐指定 LDAP URL，则目录服务器首先将修改请求指向所指定的 URL。如果不指定，则修改请求将指向包含当前副本的供给器。

要为引荐指定一个新的 URL：

a. 在相应的字段中输入新的 LDAP URL，或者单击 “构造”以显示一个帮助构建 LDAP URL 的对话框。

b. 单击 “添加”。

新的 LDAP URL 将直接出现在上述 “当前的引荐 URL”列表中。

c. 为每个要添加到列表中的引荐重复以上操作。

8. 单击 “保存”以保存数据库的复制设置。

创建复制协议

本部分说明如何创建复制协议。必须在供给服务器上为提供给客户服务器或中枢供给器的每个供给器副本创建复制协议。

创建复制协议之前，您必须：

- 在服务器上配置供给器设置，如第 271 页上的 “配置供给器设置”中所述。
- 为供给器配置复制设置，如第 272 页上的 “配置供给器副本”中所述。
- 为中枢供给器 （如果有）和客户配置复制设置，如第 274 页上的 “配置中枢副本”和第 273 页上的 “配置客户副本”中所述。

要创建复制协议：

1. 在 Directory Server Console 中，单击 “配置”选项卡。

2. 在导航树中，展开 “复制”文件夹，右键单击所要复制的数据库，然后选择 “新复制协议”。

也可以突出显示数据库，然后从 “对象”菜单中选择 “新复制协议”。这将启动 “复制协议向导”。

3. 单击 “下一步”以继续执行后面的步骤，从而完成复制向导中的步骤。

有关每个字段要输入的内容的详细说明，请参考在线帮助。

完成后，在数据库图标的下面显示一个代表复制协议的图标。该复制协议图标表示复制协议已经完成设置。

注意 对于 SSL 上的复制协议，客户服务器主机名必须被指定为一个完全限定的域名（例如 `server.remote.siroe.com`）。请不要输入别名、IP 地址或仅仅是一个域名的本地部分，因为这样将不允许进行 SSL 复制，并且可能受到“man-in-the-middle”攻击。

默认情况下，供给服务器将确认客户服务器证书的证书路径。供给服务器的信任 CA 根库只能包含那些正在用于 SSL 复制或客户机验证的 CA 的证书。要保护 SSL 复制免受 man-in-the-middle 攻击，`nsSslServerAuth` 配置属性的值必须为 `cncheck`，（如果已知客户服务器的证书包含带 CN 属性的特异名称，或者包含与其完全限定的域名相匹配的扩展名）。

配置单原版复制

本部分提供如何配置单原版复制的分步说明。要按第 263 页的图 8-1 中所示的配置在持有供给器副本的供给服务器 A 和持有客户副本的客户服务器 B 之间设置单原版复制，必须执行以下任务：

1. 配置客户服务器（供给器绑定 DN 以及可选的修改请求引荐）和客户副本。

   有关该过程的说明见第 277 页上的“配置客户服务器和副本”。

2. 配置供给服务器（更改日志和复制 ID）和供给器副本。

   有关该过程的说明见第 279 页上的“配置供给服务器和副本”。

3. 初始化客户服务器上的副本。

   有关该过程的说明见第 280 页上的“初始化单原版复制中的副本”。

配置客户服务器和副本

1. 创建副本数据库（如果不存在）。

   有关说明，请参阅第 72 页上的“创建后缀”。

2. 创建对应于客户服务器上的供给器绑定 DN 的条目（如果不存在）。这是供给器用于绑定时的特殊条目。

   a. 在 Directory Server Console 中，单击“目录”选项卡，然后创建条目。例如，可以使用 `cn=Replication Manager,cn=config`。

   b. 指定 `userPassword` 属性 – 值对。

c. 如果已启用口令到期策略 （或打算将来如此），则必须牢记要禁用该策略，避免复制因口令到期而失败。要禁用 `userPassword` 属性的口令到期策略，需要添加值为 20380119031407Z 的 `passwordExpirationTime` 属性，这就使得口令永远不会过期。

注意 该供给器绑定 DN 对应于有权限的用户，因为他们不受访问控制的限制。该条目不得为所复制的数据库的组成部分。

---

3. 指定客户副本所需的复制设置。

a. 在 Directory Server Console 中，单击 “配置”选项卡。

b. 在导航树中，展开 “复制”文件夹，然后突出显示副本数据库。

“副本设置”选项卡将显示在窗口的右侧。

c. 选中 “启用副本”复选框。

d. 在 “副本角色”部分中，选择 “指定客户”单选按钮。

e. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

由于无需指定客户的复制 ID （对于所有客户副本而言，它被自动设置为 65535），因此 “复制 ID”字段一直处于不可用状态。

f. 在 “更新设置”部分中，指定供给器用于绑定到副本所需的供给器绑定 DN （复制管理器 DN）。

第一次配置副本时，“当前的供给器 DN”列表为空。每个副本可以有多个供给器绑定 DN，但每个复制协议只有一个供给器 DN。

要指定新的供给器绑定 DN：

- 在相应的字段中输入新的供给器绑定 DN。所输入的 DN 必须对应于在步骤 2 上创建的条目 （例如， `cn=Replication Manager,cn=config`）。
- 单击 “添加”。新的供给器绑定 DN 将直接出现在上述的 “当前的供给器 DN”列表中。
- 为每个要添加到列表中的供给器绑定 DN 重复以上操作。

g. 另外，指定更新所要指向的服务器的 LDAP URL。

第一次配置副本时，“当前的引荐 URL”列表为空。默认情况下，该列表不显示存放数据副本的原版的服务器的 URL （该服务器自动创建这个引荐）。

自动引荐假定客户机将通过常规连接来进行绑定，因此格式为 `ldap://servername:port`。如果想使用 SSL 将客户机绑定到供给器，则可使用该字段来指定以下格式的引荐：`ldaps://servername:port` （其中，`ldaps` 中的 `s` 表示安全连接）。

如果为引荐指定 LDAP URL，则目录服务器首先将更新请求指向所指定的 URL。如果不指定，则更新请求将指向包含当前副本的供给器。

要为引荐指定一个新的 URL：

- 在相应的字段中输入新的 LDAP URL，或者单击 “构造”以显示一个帮助构建 LDAP URL 的对话框。
- 单击 “添加”。新的 LDAP URL 将直接出现在上述 “当前的引荐 URL”列表中。
- 为每个要添加到列表中的引荐重复以上操作。

4. 单击 “保存”以保存副本的复制设置。

配置供给服务器和副本

1. 指定服务器的供给器设置。

a. 在 Directory Server Console 中，单击 “配置”选项卡。

b. 在导航树中，突出显示 “复制”节点。

c. 在窗口右侧，在 “供给器设置”选项卡中选中 “启用更改日志”复选框。

这将激活下面窗口中以前不可用的所有字段。

d. 单击 “使用默认值”按钮以指定更改日志，或单击 “浏览”按钮以显示文件选择器。

e. 设置更改日志参数 （数目和存在周期）

如果要指定不同的值，则必须清除 “不限制”复选框。

f. 单击 “保存”以保存供给器设置。

2. 指定供给器副本所需的复制设置。

   a. 在导航树的 “配置” 选项卡中，展开 “复制” 节点并突出显示所要复制的数据库。

      “副本设置” 选项卡将显示在窗口的右侧。

   b. 选中 “启用副本” 复选框。

   c. 在 “副本角色” 部分中，选择 “单原版” 单选按钮。

   d. 在 “通用设置” 部分中，指定副本 ID （1 到 65534 之间的整数，包括首尾数字）。

      对于每个供给器副本而言，副本 ID 必须唯一。确保指定的 ID 不同于该服务器及其它服务器上其它供给器副本所用的 ID。

   e. 在 “通用设置” 部分中，指定 “清除延迟” 字段中的一个清除延迟。

      该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

   f. 单击 “保存” 以保存数据库的复制设置。

3. 创建该供给器与客户之间的复制协议。

   a. 在导航树的 “配置” 选项卡中，右键单击所要复制的数据库，然后选择 “新复制协议”。

      也可以突出显示数据库，然后从 “对象” 菜单中选择 “新复制协议”。这将启动 “复制协议向导”。

   b. 单击 “下一步” 以继续执行后面的步骤，从而完成复制向导中的步骤。

      有关每个字段要输入的内容的详细说明，请参考在线帮助。

   c. 完成操作后，即设置好复制协议。

初始化单原版复制中的副本

您可以从 “复制协议向导” 初始化客户副本，也可以后随时初始化客户副本。有关初始化客户副本的信息，请参阅第 294 页上的 “初始化客户”。

配置多原版复制

本部分说明如何配置多原版复制。在多原版复制中，两个供给器可以接受更新、实现彼此同步及更新所有客户。客户将更新请求指向两个原版。本部分提供如何配置多原版复制的分步说明。

要按第 265 页的图 8-2 中所示的配置在两个供给器之间 （例如两个都持有供给器副本的服务器 A 和服务器 B）和两个客户之间 （两个都持有客户副本的服务器 C 和服务器 D）设置多原版复制，必须执行以下任务：

1. 配置客户服务器 （供给器绑定 DN 以及可选的修改请求引荐）和客户副本。

   有关该过程的说明见第 281 页上的 “配置客户服务器和副本”。

2. 配置供给服务器 （更改日志和复制 ID）以及供给器副本。

   有关该过程的说明见第 283 页上的 “配置供给服务器和副本”。

3. 初始化客户服务器上的客户副本。

   有关该过程的说明见第 286 页上的 “初始化多原版复制中的副本”。

配置客户服务器和副本

在每台客户服务器上执行以下步骤：

1. 创建副本数据库 （如果不存在）。

   有关说明，请参阅第 72 页上的 “创建后缀”。

2. 创建对应于供给器绑定 DN 的条目 （如果不存在）。这是供给器用于绑定时的特殊条目。

   a. 在 Directory Server Console 中，单击 “目录”选项卡，然后创建条目。例如，可以使用 `cn=Replication Manager,cn=config`。

   b. 指定 `userPassword` 属性 – 值对。

   c. 如果已启用口令到期策略 （或打算将来如此），则必须牢记要禁用该策略，避免复制因口令到期而失败。要禁用 `userPassword` 属性的口令到期策略，需要添加值为 20380119031407Z 的 `passwordExpirationTime` 属性，这就使得口令永远不会过期。

注意 该供给器绑定 DN 对应于有权限的用户，因为他们不受访问控制的限制。该条目不得为所复制的数据库的组成部分。

3. 指定客户副本所需的复制设置。

   a. 在 Directory Server Console 中，单击 “配置”选项卡。

   b. 在导航树中，展开 “复制”文件夹，然后突出显示副本数据库。

      “副本设置”选项卡将显示在窗口的右侧。

   c. 选中 “启用副本”复选框。

   d. 在 “副本角色”部分中，选择 “指定客户”单选按钮。

   e. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

      该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

      由于无需指定客户的复制 ID （对于所有客户副本而言，它被自动设置为 65535），因此 “复制 ID”字段一直处于不可用状态。

   f. 在 “更新设置”部分中，指定供给器用于绑定到副本所需的供给器绑定 DN （复制管理器 DN）。

      第一次配置副本时，“当前的供给器 DN”列表为空。每个副本可以有多个供给器绑定 DN，但每个复制协议只有一个供给器 DN。

      要指定新的供给器绑定 DN：

      - 在相应的字段中输入新的供给器绑定 DN。所输入的 DN 必须对应于在步骤 2 上创建的条目 （例如， `cn=Replication Manager,cn=config`）。
      - 单击 “添加”。新的供给器绑定 DN 将直接出现在上述的 “当前的供给器 DN”列表中。
      - 为每个要添加到列表中的供给器绑定 DN 重复以上操作。

g. 另外，指定更新所要指向的服务器的 LDAP URL。

第一次配置副本时，“当前的引荐 URL”列表为空。默认情况下，该列表不列出存放数据副本的原版的服务器的 URL （客户服务器自动创建这个引荐）。

自动引荐假定客户机将通过常规连接来进行绑定，因此格式为 `ldap://`*servername*`:`*port*。如果想使用 SSL 将客户机绑定到供给器，则可使用该字段来指定以下格式的引荐：`ldaps://`*servername*`:`*port* （其中，`ldaps` 中的 `s` 表示安全连接）。

如果为引荐指定 LDAP URL，则目录服务器首先将更新请求指向所指定的 URL。如果不指定，则更新请求将指向包含当前副本的供给器。

要为引荐指定一个新的 URL：

- 在相应的字段中输入新的 LDAP URL，或者单击 “构造”以显示一个帮助构建 LDAP URL 的对话框。
- 单击 “添加”。新的 LDAP URL 将直接出现在上述 “当前的引荐 URL”列表中。
- 为每个要添加到列表中的引荐重复以上操作。

4. 单击 “保存”以保存副本的复制设置。

在复制配置中，为每个客户服务器重复上述步骤。

配置供给服务器和副本

在每台供给服务器上执行以下步骤：

1. 在服务器 A 和服务器 B 上，为每台服务器指定供给器设置。

a. 在 Directory Server Console 中，单击 “配置”选项卡。

b. 在导航树中，突出显示 “复制”节点。

c. 在窗口右侧，单击 “供给器设置”选项卡。

d. 选中 “启用更改日志”复选框。

这将激活下面窗口中以前不可用的所有字段。

e. 单击 “使用默认值”按钮以指定更改日志，或单击 “浏览”按钮以显示文件选择器。

f. 设置更改日志参数 （数目和存在周期）

如果要指定不同的值，则必须清除 “不限制”复选框。

g. 单击 “保存”以保存供给器设置。

2. 创建对应于供给器绑定 DN 的条目 （如果不存在）。对于多原版复制而言，必须在供给服务器 （以及客户）上创建此供给器绑定 DN，因为这些服务器充当其它供给服务器的客户和供给器。

a. 在 Directory Server Console 中，单击 “目录”选项卡，然后创建条目。例如，可以使用 `cn=Replication Manager,cn=config`。

b. 指定 `userPassword` 属性 – 值对。

c. 如果已启用口令到期策略 （或打算将来如此），则必须牢记要禁用该策略，避免复制因口令到期而失败。要禁用 `userPassword` 属性的口令到期策略，需要添加值为 20380119031407Z 的 `passwordExpirationTime` 属性，这就使得口令永远不会过期。

注意 该供给器绑定 DN 对应于有权限的用户，因为他们不受访问控制的限制。该条目不得为所复制的数据库的组成部分。

---

3. 在服务器 A 和服务器 B 上，为多原版供给器副本指定复制设置。

a. 在导航树的 “配置”选项卡中，展开 “复制”节点并突出显示所要复制的数据库。

“副本设置”选项卡将显示在窗口的右侧。

b. 选中 “启用副本”复选框。

c. 在 “副本角色”部分中，选择 “多原版”单选按钮。

d. 在 “通用设置”部分中，指定副本 ID （1 到 65534 之间的整数，包括首尾数字）。

对于每个供给器副本而言，副本 ID 必须唯一。确保指定的 ID 不同于该服务器及其它服务器上其它供给器副本所用的 ID。

e. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

f. 在 “更新设置”部分中，指定供给器用于绑定到副本所需的供给器绑定 DN （复制管理器 DN）。

第一次配置副本时，“当前的供给器 DN”列表为空。每个副本可以有多个供给器绑定 DN，但每个复制协议只有一个供给器 DN。

要指定新的供给器绑定 DN：

- 在相应的字段中输入新的供给器绑定 DN。所输入的 DN 必须对应于在步骤 2 上创建的条目 （例如， `cn=Replication Manager,cn=config`）。
- 单击 “添加”。新的供给器绑定 DN 将直接出现在上述 “当前的供给器 DN”列表中。
- 为每个要添加到列表中的供给器绑定 DN 重复以上操作。

注意 在供给服务器上，无需为引荐指定 LDAP URL。

---

g. 单击 “保存”以保存数据库的复制设置。

4. 在服务器 A 上，设置下列复制协议：

- 一个用于供给服务器 B，此处 B 被配置为副本的 *客户*。
- 一个用于每个客户：服务器 C 和服务器 D。

a. 在导航树的 “配置”选项卡中，右键单击所要复制的数据库，然后选择 “新复制协议”。

也可以突出显示数据库，然后从 “对象”菜单中选择 “新复制协议”。这将启动 “复制协议向导”。

b. 单击 “下一步”以继续执行后面的步骤，从而完成复制向导中的步骤。

有关每个字段要输入的内容的详细说明，请参考在线帮助。

您可以从 “复制协议向导”中初始化服务器 B 上的客户副本和供给器副本，也可以后随时初始化上述副本。有关初始化客户副本时顺序和步骤的信息，请参阅第 286 页上的 “初始化多原版复制中的副本”和第 294 页上的 “初始化客户”。

完成操作后，即设置好复制协议。

5. 在服务器 B 上，设置下列复制协议：
   - 一个用于供给服务器 A，此处 A 被声明为副本的*客户*。如果已按照步骤 4 中所述从服务器 A 中初始化服务器 B，则在该操作过程中不要从服务器 B 初始化服务器 A。
   - 一个用于每个客户：服务器 C 和服务器 D。

注意 完成这些步骤后，服务器 A 和服务器 B 有共同的复制协议，因此它们可以接受彼此的更新。

如果已配置好持有供给器副本的服务器、必要的复制协议及持有客户副本的服务器，即可初始化复制过程。在供给服务器上创建复制协议时可以执行该任务，也可在以后随时执行该任务。

初始化多原版复制中的副本

如果是多原版复制，则应按下列顺序初始化副本：

1. 确保一个原版有完整的待复制数据。使用该原版初始化多原版复制集中另一个原版上的供给器副本。
2. 从两个原版中的任何一个来初始化客户服务器上的客户副本。

有关初始化副本的信息，请参阅第 294 页上的 “初始化客户”。

配置级联复制

本部分说明如何设置级联复制。在级联复制环境中，供给服务器更新中间服务器（称为中枢服务器），中间服务器继而更新一个或几个客户服务器。本部分提供如何设置级联复制的分步说明。

要按第 268 页的图 8-4 中所示的配置在服务器 A 上的供给器、中枢服务器 B 和客户服务器 C 之间设置级联复制，必须执行以下任务：

1. 配置客户服务器 （供给器绑定 DN 以及可选的修改请求引荐）和客户副本。

   有关该过程的说明见第 287 页上的 “配置客户服务器和副本”。

2. 配置中枢供给器 （更改日志、供给器绑定 DN 以及可选的修改请求的引荐）和中枢副本。

   有关该过程的说明见第 289 页上的 “配置中枢服务器和副本”。

3. 配置供给服务器 （更改日志和复制 ID）和供给器副本。

   有关该过程的说明见第 291 页上的 “配置供给服务器和副本”。

4. 在供给服务器和中枢供给器上配置复制协议。

   有关该过程的说明见第 292 页上的 “配置复制协议”。

5. 在中枢供给器和客户服务器上初始化副本。

   有关该过程的说明见第 293 页上的 “初始化级联复制中的副本”。

配置客户服务器和副本

1. 在客户服务器上创建副本数据库 （如果不存在）。

   有关说明，请参阅第 72 页上的 “创建后缀”。

2. 在客户服务器上创建对应于供给器绑定 DN 的条目 （如果不存在）。这是供给器用于绑定时的特殊条目。

   a. 在 Directory Server Console 中，单击 “目录”选项卡，然后创建条目。例如，可以使用 `cn=Replication Manager,cn=config`。

   b. 指定 `userPassword` 属性 – 值对。

   c. 如果已启用口令到期策略 （或打算将来如此），则必须牢记要禁用该策略，避免复制因口令到期而失败。要禁用 `userPassword` 属性的口令到期策略，需要添加值为 20380119031407Z 的 `passwordExpirationTime` 属性，这就使得口令永远不会过期。

注意 该供给器绑定 DN 对应于有权限的用户，因为他们不受访问控制的限制。该条目不得为所复制的数据库的组成部分。

---

3. 在客户服务器上为客户副本指定复制设置。

   a. 在 Directory Server Console 中，单击 “配置”选项卡。

   b. 在导航树中，展开 “复制”文件夹，然后突出显示副本数据库。

      “副本设置”选项卡将显示在窗口的右侧。

   c. 选中 “启用副本”复选框。

   d. 在 “副本角色”部分中，选择 “指定客户”单选按钮。

e. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

由于无需指定客户的复制 ID （对于所有客户副本而言，它被自动设置为 65535），因此 “复制 ID”字段一直处于不可用状态。

f. 在 “更新设置”部分中，指定供给器用于绑定到副本所需的供给器绑定 DN （复制管理器 DN）。

第一次配置副本时，“当前的供给器 DN”列表为空。每个副本可以有多个供给器绑定 DN，但每个复制协议只有一个供给器 DN。

要指定新的供给器绑定 DN：

- 在相应的字段中输入新的供给器绑定 DN。所输入的 DN 必须对应于在步骤 2 上创建的条目 （例如， `cn=Replication Manager,cn=config`）。
- 单击 “添加”。新的供给器绑定 DN 将直接出现在上述 “当前的供给器 DN”列表中。
- 为每个要添加到列表中的供给器绑定 DN 重复以上操作。

g. 另外，指定更新所要指向的服务器的 LDAP URL。

第一次配置副本时，“当前的引荐 URL”列表为空。默认情况下，该列表不列出存放数据副本的原版的服务器的 URL （客户服务器自动创建这个引荐）。

自动引荐假定客户机将通过常规连接来进行绑定，因此格式为 `ldap://`*servername*`:`*port*。如果想使用 SSL 将客户机绑定到供给器，则可使用该字段来指定以下格式的引荐：`ldaps://`*servername*`:`*port* （其中，`ldaps` 中的 `s` 表示安全连接）。

如果为引荐指定 LDAP URL，则目录服务器首先将更新请求指向所指定的 URL。如果不指定，则更新请求将指向包含当前副本的供给器。

要为引荐指定一个新的 URL：

- 在相应的字段中输入新的 LDAP URL，或者单击 “构造”以显示一个帮助构建 LDAP URL 的对话框。
- 单击 “添加”。新的 LDAP URL 将直接出现在上述 “当前的引荐 URL”列表中。
- 为每个要添加到列表中的引荐重复以上操作。

4. 单击 “保存”以保存副本的复制设置。

配置中枢服务器和副本

在从原版接收复制更新并将其传播给客户的中枢供给器上，请执行以下步骤：

1. 创建副本数据库 （如果不存在）。

   有关说明，请参阅第 72 页上的 “创建后缀”。

2. 创建对应于供给器绑定 DN 的条目 （如果不存在）。这是供给器用于绑定时的特殊条目。

   a. 在 Directory Server Console 中，单击 “目录”选项卡，然后创建条目。例如，可以使用 `cn=Replication Manager,cn=config`。

   b. 指定 `userPassword` 属性 – 值对。

   c. 如果已启用口令到期策略 （或打算将来如此），则必须牢记要禁用该策略，避免复制因口令到期而失败。要禁用 `userPassword` 属性的口令到期策略，需要添加值为 20380119031407Z 的 `passwordExpirationTime` 属性，这就使得口令永远不会过期。

注意 该供给器绑定 DN 对应于有权限的用户，因为他们不受访问控制的限制。该条目不得为所复制的数据库的组成部分。

---

3. 指定中枢副本的复制设置。

   a. 在 Directory Server Console 中，单击 “配置”选项卡。

   b. 在导航树中，展开 “复制”文件夹，然后突出显示副本数据库。

      “副本设置”选项卡将显示在窗口的右侧。

   c. 选中 “启用副本”复选框。

   d. 在 “副本角色”部分中，选择 “中枢”单选按钮。

   e. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

      该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。

      由于无需指定中枢供给器的复制 ID （与客户副本非常相似，它被自动设置为 65535），因此 “复制 ID”字段一直处于不可用状态。

f. 在 “更新设置”部分中，指定供给器用于绑定到副本所需的供给器绑定 DN （复制管理器 DN）。

第一次配置副本时，“当前的供给器 DN”列表为空。每个副本可以有多个供给器绑定 DN，但每个复制协议只有一个供给器 DN。

要指定新的供给器绑定 DN：

- 在相应的字段中输入新的供给器绑定 DN。所输入的 DN 必须对应于在步骤 2 上创建的条目 （例如， `cn=Replication Manager,cn=config`）。
- 单击 “添加”。新的供给器绑定 DN 将直接出现在上述 “当前的供给器 DN”列表中。
- 为每个要添加到列表中的供给器绑定 DN 重复以上操作。

g. 另外，指定更新所要指向的服务器的 LDAP URL。

第一次配置副本时，“当前的引荐 URL”列表为空。默认情况下，该列表不显示存放数据副本的原版的服务器的 URL （中枢服务器自动创建这个引荐）。

自动引荐假定客户机将通过常规连接来进行绑定，因此格式为 `ldap://`*servername*`:`*port*。如果想使用 SSL 将客户机绑定到供给器，则可使用该字段来指定以下格式的引荐：`ldaps://`*servername*`:`*port* （其中，`ldaps` 中的 `s` 表示安全连接）。

如果为引荐指定 LDAP URL，则目录服务器首先将修改请求指向所指定的 URL。如果不指定，则修改请求将指向包含当前副本的供给器。

要为引荐指定一个新的 URL：

- 在相应的字段中输入新的 LDAP URL，或者单击 “构造”以显示一个帮助构建 LDAP URL 的对话框。
- 单击 “添加”。新的 LDAP URL 将直接出现在上述的 “当前的引荐 URL”列表中。
- 为每个要添加到列表中的引荐重复以上操作。

4. 单击 “保存”以保存副本的复制设置。

配置供给服务器和副本

在持有数据库原始副本的供给服务器上执行以下步骤：

1. 指定服务器的供给器设置。
   a. 在 Directory Server Console 中，单击 “配置”选项卡。
   b. 在导航树中，突出显示 “复制”节点。
   c. 在窗口右侧，单击 “供给器设置”选项卡。
   d. 选中 “启用更改日志”复选框。

      这将激活下面窗口中以前不可用的所有字段。
   e. 单击 “使用默认值”按钮以指定更改日志，或单击 “浏览”按钮以显示文件选择器。
   f. 设置更改日志参数 （数目和存在周期）。

      如果要指定不同的值，则必须清除 “不限制”复选框。
   g. 单击 “保存”以保存供给器设置。
2. 指定所需的复制设置。
   a. 在导航树的 “配置”选项卡中，展开 “复制”节点并突出显示所要复制的数据库。

      “副本设置”选项卡将显示在窗口的右侧。
   b. 选中 “启用副本”复选框。
   c. 在 “副本角色”部分中，选择 “单原版”单选按钮。
   d. 在 “通用设置”部分中，指定副本 ID （1 到 65534 之间的整数，包括首尾数字）。

      对于每个供给器副本而言，副本 ID 必须唯一。确保指定的 ID 不同于该服务器及其它服务器上其它供给器副本所用的 ID。
   e. 在 “通用设置”部分中，指定 “清除延迟”字段中的一个清除延迟。

      该选项指示状态信息在复制条目中存储时间的长短。清除前的延迟时间不仅必须足够长以允许复制关闭或在出错后恢复，而且必须适当短以避免在条目中保留过多的数据。缺省设置为 1 个星期。
   f. 单击 “保存”以保存数据库的复制设置。

配置复制协议

当配置级联复制环境时，必须先按下列顺序创建复制协议：

- 首先，在供给服务器上定义该供给器与中枢供给器之间的复制；
- 其次，在中枢供给器上定义中枢供给器与客户服务器之间的复制。

通过执行上述顺序的操作，在创建复制协议时还可以在中枢供给器和客户服务器上初始化副本。

1. 在供给服务器上，设置该服务器与中枢供给器之间的复制协议。

   a. 在导航树的“配置”选项卡中，右键单击所要复制的数据库，然后选择“新复制协议”。

      也可以突出显示数据库，然后从“对象”菜单中选择“新复制协议”。这将启动“复制协议向导”。

   b. 单击“下一步”以继续执行后面的步骤，从而完成复制向导中的步骤。

      有关每个字段要输入的内容的详细说明，请参考在线帮助。

      您此时可以初始化中枢供给器上的副本，也可在以后随时初始化该副本。有关在以后阶段初始化副本的信息，请参阅第 294 页上的“初始化客户”。

2. 在中枢供给器上，设置该服务器与客户之间的复制协议。

   执行在步骤 1 说明的相同步骤。可以通过复制向导来初始化客户服务器上的副本。如果选择在以后初始化客户服务器，请参阅第 294 页上的“初始化客户”以获得有关指导说明。

注意 对于 SSL 上的复制协议，客户服务器主机名必须被指定为一个完全限定的域名（例如 `server.remote.siroe.com`）。请不要输入别名、IP 地址或仅仅是一个域名的本地部分，因为这样将不允许进行 SSL 复制，并且可能受到“man-in-the-middle”攻击。

默认情况下，供给服务器将确认客户服务器证书的证书路径。供给服务器的信任 CA 根库只能包含那些正在用于 SSL 复制或客户机验证的 CA 的证书。要保护 SSL 复制免受 man-in-the-middle 攻击，`nsSslServerAuth` 配置属性的值必须为 `cncheck`，（如果已知客户服务器的证书包含带 CN 属性的特异名称，或者包含与其完全限定的域名相匹配的扩展名）。

初始化级联复制中的副本

如果选择不在配置复制协议时初始化副本，则可以随时执行此初始化操作，有关的操作说明见第 294 页上的 “初始化客户”。但是，在级联复制中，请记住始终应按下列顺序初始化副本：

1. 从供给服务器中初始化中枢供给器上的副本。
2. 从中枢供给器中初始化客户上的副本。

删除更改日志

更改日志记录对给定副本的全部修改，供给器使用它将这些修改内容在存储于客户服务器的副本上进行重现 （或者在多原版复制的情况下，则在其它原版上进行重现）。如果供给服务器离线，则是否能够删除更改日志很重要，因为它不再保留所有修改的真实记录，所以不应作为复制的基础。删除更改日志后，可以初始化客户并重新启动复制过程。要删除更改日志，可以将其移除或移到新的位置。

本部分包含下列步骤的信息：

- 第 293 页上的 “删除更改日志”
- 第 294 页上的 “将更改日志移到新位置”

删除更改日志

可以使用删除 iPlanet Directory Server Console 更改日志。要从供给服务器删除更改日志：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧导航树中选择 “复制”文件夹，然后选择右侧窗口中的 “供给服务器设置”选项卡。
3. 清除 “启用更改日志”复选框。

   这将删除更改日志。
4. 单击 “保存”。
5. 重新启动 iPlanet Directory Server。
6. 重新初始化客户。

   有关信息，请参阅第 294 页上的 “初始化客户”。

注意 如果要删除更改日志，则需要重新初始化客户服务器。

将更改日志移到新位置

要在服务器正在运行并且继续记录更改时删除更改日志，将该更改日志移到一个新位置即可。通过移动更改日志，在所指定的目录中就创建了一个新的更改日志，而旧的更改日志就被删除了。改变更改日志的位置相当于对其重新初始化，这继而要求客户重新初始化。

例如，可以将更改日志从默认位置的 `/var/ds5/slapd-`*serverID*`/changelogdb` 移到 `/var/ds5/slapd-`*serverID*`/newchangelog`。该操作应通过 iPlanet Directory Server Console 执行，并且不使用文件系统的 `rename` 或 `mv` 命令。

初始化客户

创建复制协议后，必须初始化客户，也就是说，必须真正将数据从供给服务器复制到客户服务器上。本部分首先详细介绍客户初始化，然后说明两种不同的初始化客户的方法。本部分分为下列几个组成部分：

- 第 294 页上的 “初始化客户的时间”
- 第 295 页上的 “通过控制台进行的在线客户初始化”
- 第 295 页上的 “使用命令行进行的手动客户初始化”

初始化客户的时间

客户初始化涉及将数据从供给服务器复制到客户服务器的过程。子树实际位于客户上后，供给服务器即可开始在客户服务器上重现更新操作。

正常操作条件下，客户无须重新进行初始化。但如果由于某种原因而需要从备份中恢复供给服务器数据，则应重新初始化所有由其提供更新的客户。

您可以使用控制台在线初始化客户，也可以使用命令行手动初始化客户。通过控制台进行的在线客户初始化是一种初始化少量客户的有效方法。但是，由于它按顺序对每个副本进行初始化，因此，这种方法不适合初始化大量的副本。配置供给服务器上的复制协议时，如果想同时初始化客户，即可使用 “在线客户初始化”方法。

使用命令行进行的手动客户初始化是一种从单个 LDIF 文件初始化大量客户的有效方法。

通过控制台进行的在线客户初始化

通过控制台进行的在线客户初始化是初始化或重新初始化客户的最容易方法。但是，如果是通过慢速链路进行复制，则该过程可能会非常耗时，而您可能会发现使用命令行进行的手动客户初始化是更为高效的方法 （有关详细信息，请参阅第 295 页上的 “使用命令行进行的手动客户初始化”）。

注意 当通过在线客户创建方式对客户服务器进行初始化时，对副本的所有操作（包括搜索）都将被指向供给服务器，直到完成初始化过程。

执行在线客户初始化

要在线初始化或重新初始化客户：

1. 在供给服务器上，在 iPlanet Directory Server Console 中，选择 “配置”选项卡。
2. 展开 “复制”文件夹，然后展开所复制的数据库。右键单击复制协议，然后从下拉菜单中选择 “初始化客户”。

   此时将显示一条消息，警告客户的副本中所存储的所有信息都将予以删除。
3. 单击确认框中的 “是”。

在线客户初始化过程随即开始。可以在复制协议中检查在线客户初始化的状态。在线客户初始化过程中，状态信息将显示正在初始化副本。

要更新该窗口，请右键单击导航树中的复制协议图标，然后选择 “刷新”。完成在线客户初始化后，状态的改变会反映出这一情况。

有关监控复制和初始化状态的详细信息，请参阅第 305 页上的 “监控复制状态”。

使用命令行进行的手动客户初始化

在复制大量条目的情况下，使用命令行进行的手动客户初始化是最快的客户初始化方法。但与在线客户初始化过程相比，手动客户初始化过程更为复杂。如果由于性能方面的原因而导致在线过程不适用时，建议使用手动过程。

本部分分为下列几个组成部分：

- 第 296 页上的 “手动客户初始化概述”
- 第 296 页上的 “将副本导出到 LDIF”
- 第 296 页上的 “将 LDIF 文件导入客户服务器”

手动客户初始化概述

要手动初始化或重新初始化服务器：

1. 将供给服务器上的副本导出到 LDIF 文件。

   请参阅第 296 页上的 “将副本导出到 LDIF”。

2. 将包含供给器副本内容的 LDIF 文件导入客户服务器。

   有关信息，请参阅第 296 页上的 “将 LDIF 文件导入客户服务器”。

注意 在级联复制环境中，可以使用从供给服务器导出的 LDIF 文件初始化中枢服务器和中枢服务器的客户。

将副本导出到 LDIF

可以使用下列三种方法之一将副本转换为 LDIF：

1. 通过选择复制向导 “初始化客户”对话框中的 “创建客户初始化文件”来创建复制协议时。
2. 在 iPlanet Directory Server Console 中，随时右键单击 “复制”文件夹下的复制协议，并从弹出菜单中选择 “导出副本”。
3. 从命令行中，使用第 139 页上的 “从命令行导出到 LDIF”中所述的导出命令。

将 LDIF 文件导入客户服务器

使用 iPlanet Directory Server Console 中导入功能，或者使用 `directoryserver ldif2db` 命令或 `directoryserver ldif2db-task`，可以将包含供给器副本内容的 LDIF 文件导出到客户服务器。有关两种导入方法的信息，请参阅第 134 页上的 “从命令行导入”。

如果使用 `ldif2db-task`，请记住使用客户服务器上所配置的供给器绑定 DN 来进行绑定。

注意 如果使用 `ldif2db-task`，则 LDIF 文件导入操作不需要事先关闭服务器。

保持副本同步

如果由于正常维护工作而停止复制所涉及的目录服务器，则在该目录服务器返回在线状态时，需要确保能通过复制过程立即获得更新。如果是多原版环境中的原版服务器，则需要由多原版集内的其它原版来更新目录信息。在其它情况下，如果中枢供给器或指定客户由于维护工作而离线，则在它们返回在线状态时，需要由供给服务器对其进行更新。

本部分介绍复制重试算法，以及如何不需要等待下次重试就强制进行复制更新。

注意 本部分所介绍的步骤仅在已设置复制功能*且*已完成客户的初始化后方可使用。

复制重试算法

当供给服务器在企图复制到客户失败时，它会以增量时间间隔定期重试。重试模式如下所示：10 秒、 20 秒、 40 秒、 80 秒，直到间隔达到 5 分钟。以后它将每 5 分钟重试一次。

请注意：如果已将复制协议配置为始终保持供给服务器和客户服务器同步，这不足以确保离线已超过五分钟的服务器实现更新。

要确保目录信息在服务器返回在线状态后立即实现同步，可以使用持有目录信息参考副本的供给服务器上的 iPlanet Directory Server Console 或者自定义脚本。

从控制台强制进行复制更新

当客户或多原版复制配置内的供给器在经过一段时间的离线后重返在线状态时，为确保能立即发送复制更新，可在持有目录信息当前版本的供给服务器上执行下列步骤：

1. 在 iPlanet Directory Server Console 上，单击 “配置” 选项卡，展开 “复制” 文件夹和数据库节点，直到选中与必须更新的副本相对应的复制协议。
2. 右键单击复制协议，然后从下拉列表中选择 “立即发送更新”。

   这将初始化持有需更新信息的服务器的复制。

SSL 环境下的复制

您可以配置复制中所涉及的 iPlanet Directory Server，以便使所有复制操作都通过 SSL 连接进行。

要在 SSL 中使用复制，则必须首先执行以下操作：

- 将供给服务器和客户服务器配置为使用 SSL。
- 对客户服务器进行配置，使之可将供给服务器的证书识别为供给器的 DN。只有在想使用 SSL 客户机验证而非简单验证的情况下，才能执行上述操作。

有关这些步骤的说明见第 11 章 “管理 SSL”。

注意 下列情况下，通过 SSL 进行复制时将会失败：

- 供给器的证书是自签名证书
- 供给器的证书是仅针对 SSL 服务器的证书，即它无法在 SSL 握手期间充当客户机。

将服务器配置为使用 SSL 时，可以使用下列方法来确保通过 SSL 连接实施复制操作：

- 在两个 iPlanet Directory Server 之间设置复制协议时，使用复制向导。
- 在配置完初始复制协议后的任何时候，使用 iPlanet Directory Server Console。

使用复制向导来配置 SSL 环境下的复制

1. 在供给服务器的 iPlanet Directory Server Console 上，单击 “配置”选项卡，展开 “复制”文件夹，然后选择所要复制的数据库。
2. 右键单击数据库，然后从下拉菜单中选择 “新复制协议”。

   此时会显示 “复制协议向导”。
3. 完成 “复制协议向导”中的各个步骤，直到显示 “源和目标”窗口。
4. 在 “连接”部分，选中 “使用加密 SSL 连接”。

5. 选择 “SSL 客户机验证”或 “简单验证”。

   如果选择 “SSL 客户机验证”，则供给服务器和客户服务器将使用证书来彼此进行验证。

   如果选择 “简单验证”，则供给服务器和客户服务器将使用绑定 DN 和口令来彼此进行验证。必须在所提供的文本字段中指定该信息。指定该选项后，简单验证将在安全信道上进行，但没有证书。

6. 单击 “下一步”，继续复制的设置过程。

注意 对于 SSL 上的复制协议，客户服务器主机名必须被指定为一个完全限定的域名（例如 `server.remote.siroe.com`）。请不要输入别名、 IP 地址或仅仅是一个域名的本地部分，因为这样将不允许进行 SSL 复制，并且可能受到 “man-in-the-middle”攻击。

默认情况下，供给服务器将确认客户服务器证书的证书路径。供给服务器的信任 CA 根库只能包含那些正在用于 SSL 复制或客户机验证的 CA 的证书。要保护 SSL 复制免受 man-in-the-middle 攻击， `nsSslServerAuth` 配置属性的值必须为 `cncheck`，（如果已知客户服务器的证书包含带 CN 属性的特异名称，或者包含与其完全限定的域名相匹配的扩展名）。

使用控制台来配置 SSL 环境下的复制

1. 在供给服务器的 iPlanet Directory Server Console 上，单击 “配置”选项卡，展开 “复制”文件夹，然后选择要进行修改以启用 SSL 复制的复制协议。

2. 单击右侧导航窗口中的 “连接”选项卡。

   这将显示复制连接设置。

3. 在 “连接”部分，选中 “使用加密 SSL 连接”。

4. 选择 “SSL 客户机验证”或 “简单验证”。

   如果选择 “SSL 客户机验证”，则供给服务器和客户服务器将使用证书来彼此进行验证。

   如果选择 “简单验证”，则供给服务器和客户服务器将使用绑定 DN 和口令来彼此进行验证。必须在所提供的文本字段中指定该信息。指定该选项后，简单验证将在安全信道上进行，但没有证书。

5. 单击 “保存”。

早期版本的复制

本部分说明如何使用 iPlanet Directory Server 的早期版本来优化复制。 iPlanet Directory Server 5.1 可涉及 iPlanet Directory Server 早期版本的复制环境，前提是满足以下条件：

- 在复制协议中， iPlanet Directory Server 5.1 被定义为客户。
- 传统供给器可以是 iPlanet Directory Server 4.0 或 4.1x。

下列限制条件适用：

- 传统 iPlanet Directory Server 和 5.1 iPlanet Directory Server 不能更新同一副本。然而， 5.1 iPlanet Directory Server 却可有不同的副本：一个由传统 iPlanet Directory Server 提供，另一个则由 5.1 iPlanet Directory Server 提供。
- iPlanet Directory Server 5.1 不能作为其它副本的供给器。

能够将 iPlanet Directory Server 5.1 用作传统 iPlanet Directory Server 客户的主要优势在于：更便于复制环境的移植。

将 iPlanet Directory Server 5.1 配置为传统目录服务器的客户

如果打算将 iPlanet Directory Server 5.1 用作 iPlanet Directory Server 早期版本的客户，则必须按如下所示进行配置：

1. 在 Directory Server Console 中，单击 “配置”选项卡。
2. 在 “配置”选项卡中，选择 “复制”节点，然后单击右侧窗口中的 “传统客户设置”选项卡
3. 选中 “启用传统客户”复选框。

   这将激活 “验证”框中的字段。
4. 指定供传统供给服务器用于绑定的供给器 DN。

   另外，可以为供给器指定一个口令。口令必须至少含 8 个字符。
5. 单击 “保存”。

   对于从传统供给器接收更新的每个副本而言，现在必须配置其客户设置。
6. 在导航树中，展开 “复制”节点，然后选择将从传统供给器接收更新的副本。
7. 在右侧窗口中的 “副本设置”选项卡中，选中 “通用设置”框内的 “启用副本”和 “用 4.x 副本更新”两个复选框。

   使用复制功能时，只需选中上述选项即可。但不必指定供给器 DN，因为系统将使用步骤 4 中所指定的供给器 DN。
8. 单击 “保存”。

   对于从传统供给器接收更新的每个客户副本而言，重复步骤 7 和步骤 8。

   要完成传统复制的设置过程，则现在必须对传统供给器进行配置，从而复制到 5.1 iPlanet Directory Server。有关在 4.x iPlanet Directory Server 上配置复制协议的说明，请参阅传统 iPlanet Directory Server 文档。

注意 iPlanet Directory Server Console 不会阻止用户将数据库配置为供给器副本及启用传统客户设置。这使得移植过程更为容易，因为移植后可以按照自己的需要对 5.1 iPlanet Directory Server 进行配置，且将仅在移植期间激活传统客户设置。

使用回退更改日志插件

回退更改日志插件允许对 iPlanet Directory Server 5.1 进行配置，从而维护与 iPlanet Directory Server 4.x 中所用更改日志相兼容的更改日志。在 iPlanet Directory Server 5.1 与 iPlanet Meta Directory 共存的部署中，维护回退更改日志至关重要。如果目录客户机依赖于 iPlanet Directory Server 4.x 类型的更改日志，则也可能需要维护回退更改日志。

要使用回退更改日志插件，则必须在单原版复制环境中将 iPlanet Directory Server 5.1 配置为供给服务器。

将 iPlanet Directory Server 5.1 配置为维护回退更改日志后，该更改日志将存储在特殊后缀 `cn=changelog` 下的单独数据库中。

回退更改日志由单级条目组成。更改日志中的每个条目都有对象类 `changeLogEntry`，且可包含表 8-1 中所列的属性。

表 8-1 回退更改日志条目的属性

属性	定义
`changeNumber`	该单值属性始终存在。它包含一个唯一标识各更改结果的整数。此数值与发生更改的顺序有关。数值越大，更改时间越晚。
`targetDN`	该属性包含受 LDAP 操作影响的条目的 DN。如果是 `modrdn` 操作，则 `targetDN` 属性中将包含修改或移动操作前条目的 DN。
`changeTime`	该属性指定执行更改操作的时间。
`changeType`	指定 LDAP 操作的类型。该属性的值可以是下列之一：**`add`**、**`delete`**、 **`modify`** 或 **`modrdn`**。
`changes`	对于添加和修改操作而言，它包含条目所做的更改 （LDIF 格式）。
`newRDN`	如果是 `modrdn` 操作，则指定条目的新 RDN。
`deleteOldRdn`	如果是 `modrdn` 操作，则指定条目的原 RDN。
`newSuperior`	如果是 `modrdn` 操作，则指定条目的 `newSuperior` 属性。

本部分包含有关下列回退更改日志项的信息：

启用回退更改日志插件

回退更改日志插件配置信息位于 `dse.ldif` 的 `cn=Retro Changelog Plugin, cn=plugins,cn=config` 条目中。

从 iPlanet Directory Server Console 中启用回退更改日志插件的步骤与其它所有 iPlanet Directory Server 插件的相同。有关信息，请参阅第 418 页上的 “从服务器控制台启用和禁用插件”。

要从命令行中启用回退更改日志插件：

1. 创建包含下列 LDIF 更新语句的 LDIF 文件：

```
dn: cn=Retro Changelog Plugin,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginenabled
nsslapd-pluginenabled: on
```

2. 使用 `ldapmodify` 命令可以将 LDIF 文件导入到目录中。
3. 重新启动服务器。

   有关重新启动服务器的信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

回退更改日志将在目录树的特殊后缀 `cn=changelog` 中予以创建。

修整回退更改日志

在指定的时间后，可自动删除更改日志中的条目。要配置将条目从更改日志中自动予以删除时所经历的时间，则必须设置 `cn=Retro Changelog Plugin, cn=plugins,cn=config` 条目中的 `nsslapd-changelogmaxage` 配置属性。

`nsslapd-changelogmaxage` 属性是以下格式的单值属性：

`nsslapd-changelogmaxage: Integer timeUnit`

其中 *integer* 代表一个数字，而 *timeUnit* 为下列之一：**s** 表示秒，**m** 表示分钟，**h** 表示小时，**d** 表示天，**w** 表示星期。

注意 *Integer* 和 *timeUnit* 变量之间不应有空格。上述语法结构中的空格旨在显示属性值是由两个变量组件组成的，而非一个变量。

`nsslapd-changelogmaxage` 值的示例：

```
nsslapd-changelogmaxage: 2d
```

搜索和修改回退更改日志

更改日志支持搜索操作。对于包含下列格式过滤器的搜索过程而言，它可以起到优化作用：

```
(&(changeNumber>=X)(changeNumber<=Y))
```

作为一般规则，不应在回退更改日志条目上执行添加或修改操作，但可以删除条目以减少更改日志的大小。只有在修改默认的访问控制策略时才需要在回退更改日志上执行修改操作。

回退更改日志和访问控制策略

创建回退更改日志时，将默认应用下列访问控制规则：

- 授予所有经过验证的用户 （`userdn=anyone`，不要与 `userdn=all` 匿名访问相混淆）对回退更改日志顶级条目 `cn=changelog` 的读取、搜索和比较权限。
- 不授予写入和删除权限，但隐含授予目录管理员的除外。

不应向匿名用户授予读取权，因为更改日志条目中可能包含对敏感信息 （例如口令）的修改。只有经过验证的应用程序和用户才允许访问该信息。

如果要修改适用于回退更改日志的默认访问控制策略，可修改 `cn=changelog` 条目的 `aci` 属性。

监控复制状态

可以使用 iPlanet Directory Server Console 来监控复制状态。

要查看复制状态的概要：

1. 在 iPlanet Directory Server Console 中，选择 “状态”选项卡，然后在左侧导航树中选择 “复制状态”。

   此时将在右侧窗口中出现一个表，其中包含有关该服务器所配置的每个复制协议的信息。

2. 单击 “刷新”以更新选项卡的内容。

所显示的状态信息的说明见表 8-2。

表 8-2 iPlanet Directory Server Console — “状态”选项卡

表格标题	说明
协议	包含设置复制协议时所提供的名称。
副本后缀	包含所复制的后缀。
供给器	指定协议中的供给服务器。
客户	指定协议中的客户服务器。
更改数量	指示自服务器启动以来发送给该副本的更改数。
上一次副本更新已开始	指示最新复制更新的开始时间。
上一次副本更新已结束	指示最新复制更新的结束时间。
上一次更新消息	提供最新复制更新的状态。

表 8-2 iPlanet Directory Server Console — “状态”选项卡 *（续）*

表格标题	说明
客户初始化	提供客户初始化的当前状态（是否在进行中）。
上一次客户初始化更新消息	提供客户上一次初始化的状态。
上一次客户初始化已开始	指示客户副本开始初始化的时间。
上一次客户初始化已结束	指示客户副本结束初始化的时间。

解决常见复制冲突

多原版复制使用松散一致性复制模型。这就意味着可在不同服务器上更改同一条目。当两个服务器之间进行复制时，需要解决有冲突的更改内容。大多数情况下，根据与每台服务器上的更改相关联的时间戳，系统可以自动解决有冲突的更改。最近发生的更改具有优先性。

但有些情况下则需要人为干预来解决更改冲突问题。更改冲突无法由复制过程自动解决的条目中包含一个冲突标记属性 `nsds5ReplConflict`。`nsdsReplConflict` 属性是操作属性 (operational attribute)。因此，它可以简化对包含该属性的条目的搜索过程。

例如，可以使用如下 `ldapmodify` 命令：

```
% ldapsearch -D adminDN -w passwd \
-b "dc=siroe,dc=com" "nsds5ReplConflict=*"
```

请注意，默认情况下创建 `nsds5ReplConflict` 属性的索引。

本部分介绍下列冲突解决过程的步骤：

- 第 307 页上的 “解决命名冲突”
- 第 309 页上的 “解决孤项冲突”
- 第 309 页上的 “解决潜在的互操作性问题”

解决命名冲突

当在不同服务器上创建具有相同 DN 的两个条目时，自动解决冲突过程将在复制期间重命名后一条目，方法是在 DN 中包含条目的唯一标识符。每个目录项都包括由操作属性 nsuniqueid 所给的唯一标识符。出现命名冲突时，该唯一性 ID 将被添加到非唯一性 ID 中。

例如，条目 uid=adamss,ou=people,dc=siroe,dc=com 于时间 t1 在服务器 A 上创建，于时间 t2 在服务器 B 上创建，这里 t2 大于 （迟于） t1。复制后，服务器 A 和服务器 B 都拥有以下条目：

- uid=adamss,ou=people,dc=siroe,dc=com （t1 时创建）
- nsuniqueid=66446001-1dd211b2+uid=adamss,dc=siroe,dc=com （t2 时创建）

第二个条目需要以具有唯一性 DN 为原则而进行重命名。重命名过程与命名属性是单值还是多值有关。每种过程均说明如下。

重命名具有多值命名属性的条目

要重命名具有多值命名属性的条目：

1. 使用命名属性的新值重命名条目，同时保持原 RDN。例如：

```
prompt% ldapmodify -D adminDN -w passwd

>dn: nsuniqueid=66446001-1dd211b2+uid=adamss,dc=siroe,dc=com
>changetype: modrdn
>newrdn: uid=NewValue
>deleteoldrdn: 0
```

2. 删除命名属性的原 RDN 值及冲突标记属性。例如：

```
prompt% ldapmodify -D adminDN -w passwd

>dn: uid=NewValue,dc=siroe,dc=com
>changetype: modify
>delete: uid
>uid: adamss
>-
>delete: nsds5ReplConflict
>-
```

注意 这是一个包含两个步骤的过程，因为无法删除唯一标识符属性 nsuniqueid。

有关 ldapmodify 命令的详细信息，请参阅第 51 页上的 “从命令行管理条目”和 *iPlanet Directory Server 配置、命令和文件参考指南*。

重命名具有单值命名属性的条目

要重命名具有单值命名属性的条目：

1. 使用不同的命名属性重命名条目，同时保持原 RDN。例如：

```
prompt% ldapmodify -D adminDN -w passwd

>dn: nsuniqueid=66446001-1dd211b2+dc=pubs,dc=siroe,dc=com
>changetype: modrdn
>newrdn: cn=TempValue
>deleteoldrdn: 0
```

2. 删除命名属性的原 RDN 值及冲突标记属性。例如：

```
prompt% ldapmodify -D adminDN -w passwd

>dn: cn=TempValue,dc=siroe,dc=com
>changetype: modify
>delete: dc
>dc: pubs
>-
>delete: nsds5ReplConflict
>-
```

注意 这是一个包含两个步骤的过程，因为无法删除唯一标识符属性 nsuniqueid。

3. 用自己希望的属性 – 值对来重命名条目。例如：

```
prompt% ldapmodify -D adminDN -w passwd

dn: cn=TempValue,dc=siroe,dc=com
changetype: modrdn
newrdn: dc=NewValue
deleteoldrdn: 1
```

通过将 deleteoldrdn 属性设置为 **1**，即可删除临时属性 – 值对 cn=TempValue。如果想保持该属性，可将 deleteoldrdn 属性的值设置为 **0**。

有关 ldapmodify 命令的详细信息，请参阅第 51 页上的 “从命令行管理条目”。

解决孤项冲突

复制删除操作时，如果客户服务器发现要删除的条目有子项，解决冲突的过程就会创建一个*紧附*项，以避免目录中出现孤项。

同样，复制添加操作时，如果客户服务器找不到父项，解决冲突的过程就会创建一个代表父项的紧附条目，以便使新条目不是孤项。

紧附条目是包含对象类 `glue` 和 `extensibleObject` 的临时条目。创建紧附条目的方式有以下几种：

- 如果解决冲突的过程发现删除的条目具有匹配的唯一标识符，则紧附条目就是该条目的再生条目，外加 `glue` 对象类和 `nsds5ReplConflict` 属性。

  这种情况下，您可以修改紧附条目以删除 `glue` 对象类和 `nsds5ReplConflict` 属性，从而将条目保持为常规条目，也可以删除紧附条目及其子项。

- 服务器将创建具有 `glue` 和 `extensibleObject` 对象类的最小条目。

  这种情况下，您必须修改条目以使其具有一定的意义，或者删除该条目及其所有子项。

解决潜在的互操作性问题

对于依赖属性唯一性的应用程序 （例如邮件服务器）而言，为实现互操作性，您最好对包含 `nsds5ReplConflict` 属性的条目进行访问限制。如果没有限制对这些条目的访问，则需要一个属性的应用程序将同时选择原始条目和包含 `nsds5ReplConflict` 的冲突解决条目，并导致操作失败。

要限制访问，需要使用下列命令修改授予匿名读取访问权限的默认 ACI：

```
ldapmodify -h hostname -D "cn=Directory Manager" -w passwd
```

```
> dn: dc=siroe,dc=com
> changetype: modify
> delete: aci
> aci: (target ="ldap:///dc=siroe,dc=com")(targetattr
!="userPassword")(version 3.0;acl "Anonymous read-search
access";allow (read, search, compare)(userdn = "ldap:///anyone");)
> -
> add: aci
```

```
> aci:
(target="ldap:///dc=siroe,dc=com")(targetattr!="userPassword")
(targetfilter="(!(nsds5ReplConflict=*))")(version 3.0;acl "Anonymous
read-search access";allow (read, search, compare)
(userdn="ldap:///anyone");)
```

```
> -
```

新的 ACI 从搜索结果中过滤掉所有包含 `nsds5ReplConflict` 属性的条目。

第 9 章

扩展目录模式

iPlanet Directory Server 带有一个包含数百个对象类和属性的标准*模式 (schema)*。虽然标准对象类和属性应能够满足大部分要求，但可能还需要创造新的对象类和属性以对模式进行扩展。

本章在以下部分中将介绍如何对模式进行扩展：

- 扩展模式概述
- 打开和关闭模式检查
- 管理对象类
- 管理属性

扩展模式概述

向模式中添加新属性时，必须创建包含该属性的新对象类。如果只将所需的属性添加到包含大部分所需属性的现有对象类中，这样尽管看似简便，实际上却会危及 LDAP 客户机的互操作性。

iPlanet Directory Server 与现有 LDAP 客户机的互操作性依赖于标准 LDAP 模式。如果更改标准模式，则在升级服务器时就会遇到困难。出于同样原因，也不能删除标准模式的元素。

有关对象类、属性、目录模式以及模式扩展准则的详细信息，请参阅 *iPlanet Directory Server 部署指南*。有关标准属性和对象类的详细信息，请参阅 *iPlanet Directory Server 模式参考指南*。

要扩展目录模式，应按以下顺序进行：

1. 创建新属性。有关信息，请参阅第 314 页上的 “创建属性”。
2. 创建包含新属性的对象类并将属性添加至对象类中。有关信息，请参阅第 317 页上的 “创建对象类”。

管理属性

通过 iPlanet Directory Server Console，可以查看模式中的所有属性，并可创建、编辑和删除模式中的属性扩展。下列部分介绍如何管理属性：

- 第 312 页上的 “查看属性”
- 第 314 页上的 “创建属性”
- 第 314 页上的 “编辑属性”
- 第 315 页上的 “删除属性”

有关管理对象类的信息，请参阅第 315 页上的 “管理对象类”。

查看属性

要查看目录模式中当前存在的所有属性的信息：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 在左侧导航树中，选择 “模式”文件夹，然后在右侧窗口中选择 “属性”选项卡。

   此选项卡包含的表格列出了该模式中所有的标准 （只读）和用户定义属性。将鼠标停留在表中的某一行上将显示相应属性的说明文本。

   下表说明属性表中的各个字段。

表 9-1 属性选项卡中表格的各列

列标题	说明
名称	属性的名称 （有时称为类型）。
OID	属性的对象标识符。 OID 是唯一标识对象 （例如对象类或属性）的字符串，通常为十进制小数。如果未指定 OID， iPlanet Directory Server 就会自动使用 `attribute_name-oid`。例如，如果创建了属性 `birthdate` 而没有提供 OID, iPlanet Directory Server 会自动将 `birthdate-oid` 用作 OID。 有关 OID 的详细信息，或者想为企业申请前缀，请向 IANA (Internet Assigned Number Authority) 发邮件，地址是 `iana@iana.org`；也可访问 IANA 网址：`http://www.iana.org/iana/`。
语法	语法描述允许该属性使用的属性值格式，在第 313 页的表 9-2 中列出了可以使用的语法。

表 9-2 属性语法定义

语法与 OID	定义
二进制 （Binary，以前为 `bin`）	表示该属性的值是二进制的。
布尔 (Boolean)	表示该属性有以下两个值之一：`True` 或 `False`。
国家字符串 (Country String)	表示该属性的值被严格限制为两个可打印的字符串，例如 `fr`。
DN （以前为 `dn`）	表示该属性的值为 DN （特异名称）。
DirectoryString （以前为 `cis`）	表示该属性的值不区分大小写。
GeneralizedTime	表示该属性的值编码为可打印的字符串。必须指定时区。强烈推荐使用 GMT。
IA5String （以前为 `ces`）	表示该属性的值区分大小写。
整数 （Integer，以前为 `int`）	表示该属性的值是整数。
OctetString	性质与二进制相同。
邮政地址 (Postal Address)	表示该属性的值编码为 *dstring*[`$` *dstring*]* 其中，每个 *dstring* 组件均被编码为 DirectoryString 语法的值。*dstring* 内的反斜杠和美元字符必须被括起来，以免误认为行分隔符。许多服务器都限制邮政地址为 6 行且不超过 30 个字符。例如： `1234 Main St.$Anytown, TX 12345$USA`
电话号码 （TelephoneNumber， 以前为 `tel`）	表示该属性的值使用电话号码的格式。推荐使用国际化格式的电话号码。
URI	表示该属性的值使用 URL 的格式，用诸如 `http://`、`https://`、 `ftp`、 `LDAP` 的字符串引导。 URI 的性质与 IA5String 相同。请参见 RFC 2396。

创建属性

可以使用 iPlanet Directory Server Console 创建新的属性。向模式中添加新属性后，必须创建包含这些属性的新对象类。有关信息，请参阅第 317 页上的 “创建对象类”。

要创建新属性：

1. 显示 “属性”选项卡。

   本步骤的说明见第 312 页上的 “查看属性”。

2. 单击 “创建”。

   此时显示 “创建属性”对话框。

3. 在 “属性名”文本框中，输入属性具有唯一性的名称。

4. 在 “属性 OID （可选）”文本框中，输入属性的对象标识符。

   OID 在第 312 页的表 9-1 中做了描述。

5. 在 “语法”下拉菜单中，选择描述属性所持数据的语法。

   有关可用语法的说明见第 312 页的表 9-1。

6. 如果希望属性为多值，则选择 “多值”复选框。

   iPlanet Directory Server 允许每个条目有多个多值属性的实例。

7. 单击 “确定”。

编辑属性

您只能对自己创建的属性进行编辑。不能编辑标准属性。

要编辑属性：

1. 显示 “属性”选项卡。

   本步骤的说明见第 312 页上的 “查看属性”。

2. 在 “用户定义的属性”表中，选择要编辑的属性，然后单击 “编辑”。

   此时显示 “编辑属性”对话框。

3. 要更改属性的名称，请在 “属性名”文本框中输入新的属性名。

4. 要更改属性的对象标识符，请在 “属性 OID （可选）” 文本框中输入新的对象标识符。

   OID 在第 312 页的表 9-1 中做了描述。

5. 要更改描述属性所持数据的语法，请在 “语法” 下拉菜单中选择新的语法。
6. 有关可用语法的说明见第 312 页的表 9-1。
7. 要使属性为多值，请选中 “多值” 复选框。

   iPlanet Directory Server 允许每个条目有多个多值属性的实例。

8. 完成属性编辑后，单击 “确定”。

删除属性

您只能删除自己创建的属性。不能删除标准属性。

要删除属性：

1. 显示 “属性” 选项卡。

   本步骤的说明见第 312 页上的 “查看属性”。

2. 在 “用户定义的属性” 表中，选择属性并单击 “删除”。
3. 如果出现提示，请确认删除。

   服务器将立即删除属性。该操作无法撤消。

管理对象类

可以使用 iPlanet Directory Server Console 来管理模式的对象类。通过控制台，您可以查看所有模式的对象类，也可以创建、编辑和删除模式中的对象类扩展。下列部分介绍如何管理对象类：

- 第 316 页上的 “查看对象类”
- 第 317 页上的 “创建对象类”
- 第 318 页上的 “编辑对象类”
- 第 319 页上的 “删除对象类”

有关管理属性的信息，请参阅第 312 页上的 “管理属性”。

查看对象类

要查看目录模式中当前存在的所有对象类的信息：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 在导航树中，选择 “模式”文件夹，然后在右侧窗口中选择 “对象类”选项卡。
3. 在 “对象类”列表中，选择要查看的对象类。

   选项卡的其它字段显示有关所选标准或用户定义对象类的信息。

下表说明 “对象类”选项卡的字段。

表 9-3 “对象类”选项卡各字段

字段	说明
父项	父项标识该对象类从中继承属性和结构的对象类。例如，`inetOrgPerson` 对象类的父对象是 `organizationalPerson` 对象。这意味着包含 `inetOrgPerson` 对象类的条目自动继承 `organizationalPerson` 对象类的必需和允许的属性。 一般而言，如果要为用户条目添加新属性，则父项就是 `inetOrgPerson` 对象类。如果要为公司条目添加新属性，则父项通常是 `organization` 或 `organizationalUnit`。如果要为组条目添加新属性，则父项通常是 `groupOfNames` 或 `groupOfUniqueNames`。
OID	对象类的对象标识符。 OID 是唯一标识对象 （例如对象类或属性）的字符串，通常为十进制小数。如果未指定 OID， iPlanet Directory Server 就会自动使用 `ObjectClass_name-oid`。例如，如果创建对象类 `division` 时未提供 OID， iPlanet Directory Server 即自动将 `division-oid` 用作 OID。 有关 OID 的详细信息，或者想为企业申请前缀，请向 IANA (Internet Assigned Number Authority) 发邮件，地址是 `iana@iana.org`；也可访问 IANA 网址：`http://www.iana.org/iana/`。
对象类	此列表中包含 iPlanet Directory Server 模式中的所有标准和用户定义对象类。
必需的属性	包含必须出现在使用此对象类的条目中的属性列表。该列表包括继承的属性。
允许的属性	包含可能出现在使用此对象类的条目中的属性列表。该列表包括继承的属性。

创建对象类

提供对象类一个具有唯一性的名称，为此新的对象类选择父对象，然后添加必需的属性和可选的属性，这样即可创建对象类。

要创建对象类：

1. 显示 “对象类”选项卡。

   本步骤的说明见第 316 页上的 “查看对象类”。

2. 在 “对象类”选项卡中，单击 “创建”。

   此时显示 “创建对象类”对话框。

3. 在 “名称”文本框中，为对象类输入唯一名。

4. 在 “OID （可选）”文本框中，为新的对象类输入对象标识符。

   OID 在第 316 页的表 9-3 中做了描述。

5. 在 “父项”下拉菜单中，为对象类选择父对象。

   可以选择任何现有的对象类。有关父对象类的信息，请参阅第 316 页的表 9-3。

6. 要添加*必须*在使用新对象类的条目中出现的属性，请突出显示 “可用属性”列表中的属性，然后单击 “必需的属性”框左侧的 “添加”按钮。

   可以使用标准属性或创建新属性。有关信息，请参阅第 312 页上的 “管理属性”。

7. 要添加可能在使用新对象类的条目中出现的属性，请突出显示 “可用属性”列表中的属性，然后单击 “允许的属性”框左侧的 “添加”按钮。

8. 要删除以前添加的属性，请突出显示 “必需的属性”列表或 “允许的属性”列表中的属性，然后单击相应的 “删除”按钮。

   继承自父对象类的允许的或必需的属性都是不能删除的。

9. 对象类的定义完成后，单击 “确定”清除对话框。

编辑对象类

可以使用 iPlanet Directory Server Console 编辑所创建的对象类。不能编辑标准对象类。

编辑对象类：

1. 显示 “对象类”选项卡。

   本步骤的说明见第 316 页上的 “查看对象类”。

2. 从 “对象类”列表中选择要编辑的对象类，然后单击 “编辑”。

   “编辑对象类”对话框会显示出来。

3. 要改变对象类的名字，请在 “名字”文本框中输入新名字。

4. 要改变对象类的对象标识符，请在 “OID （可选）”文本框中输入新的 OID。

   OID 在第 316 页的表 9-3 中做了描述。

5. 要改变对象类的父对象，请在父项目下拉菜单中选择新的父项目。

6. 要添加必须在使用新对象类的条目中出现的属性，请突出显示 “可用属性”列表中的属性，然后单击 “必需的属性”框左侧的 “添加”按钮。

   或者使用标准属性或者创建新的属性。有关信息，请参阅第 312 页上的 “管理属性”。

7. 要添加可能在使用新对象类的条目中出现的属性，请突出显示 “可用属性”列表中的属性，然后单击 “允许的属性”框左侧的 “添加”按钮。

8. 要删除以前添加的属性，请突出显示 “必需的属性”列表或 “允许的属性”列表中的属性，然后单击相应的 “删除”按钮。

   允许的或必需的继承属性都是不能删除的。

9. 对对象类的定义感到满意后，单击 “确定”以关闭对话框。

删除对象类

您只能删除自己创建的对象类。不能删除标准对象类。

要删除对象类：

1. 显示 “对象类”选项卡。

   本步骤的说明见第 316 页上的 “查看对象类”。

2. 选择要删除的对象类，然后单击 “删除”。

3. 如果出现提示，请确认删除。

   服务器将立即删除此对象类。该操作无法撤消。

打开和关闭模式检查

当模式检查处于打开状态时， iPlanet Directory Server 可以确保：

- 所用的对象类和属性是在目录模式中定义的。
- 对象类必需的属性包含于此条目中。
- 只有对象类允许的属性包含于此条目中。

在 iPlanet Directory Server 中，模式检查默认为打开。应始终在模式检查处于打开的状态下运行 iPlanet Directory Server。惟有要加速 LDAP 导入操作时，才可能需要关闭模式检查。但是，仍有可能出现导入条目不符合模式的情况。所以，搜索这些条目是不可能的。

要打开和关闭模式检查：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。

2. 突出显示导航树顶部的服务器图标，然后选择右侧窗口中的 “设置”选项卡。

3. 要启用模式检查，请选中 “启用模式检查”复选框。清除该复选框将关闭模式检查。

4. 单击 “保存”。

也可以使用 `nsslapd-schemacheck` 属性打开和关闭模式检查。有关信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

第 10 章

管理索引

*iPlanet Directory Server 部署指南*引入了索引的概念，并说明随 iPlanet Directory Server 一起提供的索引的利弊和不同类型。本章首先介绍搜索算法，以便说明引入索引机制的环境，然后介绍如何创建、删除和管理索引。本章包含以下几部分：

- 关于索引
- 创建索引
- 删除索引
- 管理索引
- 属性名称快速参考表

关于索引

本部分概述 Directory Server 中的索引。其中包含下列主题：

关于索引类型

索引存储在目录数据库的文件中。文件的名称以索引属性为基础，而非文件中所包含的索引类型。如果为特定的属性维护有多个索引，则每个索引文件均可包含多种类型的索引。例如，所有为通用名属性保持的索引都包含在 `cn.db3` 文件中。

Directory Server 支持以下类型的索引：

- 存在索引 (`pres`)

  存在索引 (presence index) 中包含具有特定属性的条目列表。在诸如检查包含访问控制信息的条目等情况下，该索引较为有用。生成包括存在索引的 `aci.db3` 文件可以有效地进行 `ACI=*` 搜索，从而为服务器生成访问控制列表。

  存在索引不用于基本对象的搜索。

- 等同索引 (`eq`)

  等同索引 (equality index) 可以有效地搜索含有特定属性值的条目。例如，`cn` 属性的等同索引允许用户更为有效地搜索 `cn=Babs Jensen`。

- 近似索引 (`approx`)

  近似索引 (approximate index) 可以执行有效的近似或 “大致” 搜索。例如，一个条目包含属性值 cn=Robert E Lee。针对 cn~=Robert Lee、 cn~=Robert 或 cn~=Lee 的近似搜索将会返回该值。同样，针对 l~=San Fransisco （注意拼写有误）进行搜索将返回包含 l=San Francisco 的条目。

- 子字符串索引 (`sub`)

子字符串索引 (substring index) 是一种维护代价较高的索引，但它可以有效地搜索条目内的子字符串。

例如，下列形式的搜索：

```
cn=*derson
```

将匹配包含诸如以下字符串的通用名：

```
Bill Anderson
Jill Anderson
Steve Sanderson
```

同样，搜索

```
telephonenumber= *555*
```

将返回目录中包含电话号码 555 的所有条目。

注意 对于每个条目而言，子字符串索引被限制为最少两个字符。

- 国际索引

  国际索引 (international index) 可以加速国际目录中信息的搜索。除了通过将区域设置 (locale) (OID) 与要所索引的属性进行关联而应用匹配规则 (matching rule) 外，创建国际索引的过程与创建常规索引的过程类似。

  有关受支持的区域设置及其相关 OID 的列表，请参阅附录 D “国际化”。如果要将目录服务器配置为接受其它匹配规则 (matching rule)，请与 iPlanet 专业服务联系。

- 浏览 （虚拟列表视图）索引

  浏览索引 (browsing index) （或称虚拟列表视图索引 (virtual list view index)）可以加速 iPlanet Directory Server Console 中条目的显示。该索引在目录分支中包含大量条目的情况下特别有用 （例如 `ou=people` 分支）。可以在目录树的任何分支点上创建浏览索引，从而改善显示性能。这通过 iPlanet Directory Server Console 或使用 `/usr/sbin/directoryserver vlvindex` 命令来实现。

关于缺省索引、系统索引及标准索引

安装 iPlanet Directory Server 时，每个数据库实例都将创建一组缺省索引和系统索引 (system index)。为维护这些索引，目录使用标准索引 (standard index)。

缺省索引概述

尽管应在删除缺省索引前确保企业内没有服务器插件或其它服务器依赖于该索引，但仍可根据索引的实际需要而修改缺省索引 (default index)。

下表列出随目录一起安装的缺省索引：

表 10-1 缺省索引

属性	Eq	Pres	Sub	用途
cn	X	X	X	改善最常见的用户目录搜索性能。
givenName	X	X	X	改善最常见的用户目录搜索性能。
mail	X	X	X	改善最常见的用户目录搜索性能。
mailHost	X			用于 iPlanet Messaging Server。
member	X			改善 iPlanet 服务器性能。该索引也用于参照完整性插件。有关详细信息，请参阅第 65 页上的 “保持参照完整性”。
owner	X			改善 iPlanet 服务器性能。该索引也用于参照完整性插件。有关详细信息，请参阅 *iPlanet Directory Server 管理员指南*。
seeAlso	X			改善 iPlanet 服务器性能。该索引也用于参照完整性插件。有关详细信息，请参阅第 65 页上的 “保持参照完整性”。
sn	X	X	X	改善最常见的用户目录搜索性能。
telephoneNumber	X	X	X	改善最常见的用户目录搜索性能。
uid	X			改善 iPlanet 服务器性能。
uniquemember	X			改善 iPlanet 服务器性能。该索引也用于参照完整性插件。有关详细信息，请参阅第 65 页上的 “保持参照完整性”。

系统索引概述

系统索引为无法删除或修改的索引。它们是目录正常工作所必需的。下表列出随目录提供的系统索引：

表 10-2 系统索引

属性	Eq	Pres	用途
aci		X	允许目录服务器快速获得数据库中所维护的访问控制信息。
dnComp	X		用于帮助加速目录中的子树搜索。
objectClass	X		用于帮助加速目录中的子树搜索。
entryDN	X		基于 DN 搜索而加快条目检索。
parentID	X		在一级搜索过程中提高目录性能。
numSubordinates		X	供 iPlanet Directory Server Console 用于提高 “目录”选项卡上的显示性能。
nsUniqueID	X		用于搜索特定条目。

标准索引概述

由于需要维护缺省索引和其它内部索引机制，因此 iPlanet Directory Server 还维护某些标准索引文件。默认情况下存在下列标准索引，您无需创建这些索引：

- id2entry.db3 — 包含实际的目录数据库条目。可以从该索引重建其它所有数据库文件。
- id2children.db3 — 限制一级搜索 （即检查条目直接子项的搜索）的范围。
- dn.db3 — 控制子树搜索的范围；也就是检查条目及其下属子树中所有条目的搜索。
- dn2id.db3 — 通过将条目的特异名称映射为其 ID 号而有效地启动所有搜索。

搜索算法概述

索引用于加速搜索。了解目录中索引的用法将有助于了解搜索算法。每个索引中都包含一个属性列表 （例如 cn、通用名、属性）及对应于每个值的条目指针。iPlanet Directory Server 按如下所示处理搜索请求：

1. LDAP 客户机应用程序 （例如 Netscape Communicator 或 iPlanet Directory Server Console）向目录发送搜索请求。
2. 目录将检查进入的请求，以确保指定的基本 DN 匹配其一个或多个数据库或数据库链接中所含的后缀。
   - 如果确实匹配，目录就会处理该请求。
   - 如果不匹配，则目录将向客户机返回一条错误信息，指示后缀不匹配。如果 `cn=config` 下的 `nsslapd-referral` 属性中指定了引荐，则目录还返回 LDAP URL，客户机可利用它来尝试追踪请求。
3. 如果单个索引即可满足每个数据库属性的搜索请求，则服务器将读取该索引以生成潜在的匹配项列表。

   如果属性无索引，则目录将生成一个包含数据库中所有条目的候选列表。这会使搜索速度变得相当慢。（对于服务器正在使用的索引关键字 (index key) 而言，如果已设置 “所有 ID 令牌”，则目录也会执行上述操作。有关 “所有 ID”的信息，请参阅第 342 页上的 “管理索引”。）

   如果搜索请求包含多个属性，则目录将查询多个索引，然后将候选条目的结果列表组合起来。
4. 如果属性有索引，则目录将从索引文件中以条目 ID 号序列的格式提取出候选匹配项。
5. 目录将使用返回的条目 ID 号读取 `id2entry.db3` 文件中相应的条目。目录服务器随即检查每个候选条目，以查看是否有条目与搜索标准相匹配。每找到一个匹配条目，目录就将该条目返回给客户机。

   目录将继续进行操作，直到检查了所有候选条目，或者达到下列属性中所设的限制：
   - `nsSizeLimit` — 指定搜索操作所返回的最大条目数。如果达到该限制，目录将返回已找到的、与搜索请求相匹配的所有条目，同时返回超过该条目限制的错误消息。
   - `nsTimeLimit` — 指定分配给搜索请求的秒数最大值。如果达到该限制，目录将返回已找到的、与搜索请求相匹配的所有条目，同时返回超过时间限制的错误消息。
   - `nsLookthroughLimit` — 指定为响应搜索请求而检查候选条目时，目录将检查的最大条目数。

有关这些属性的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

此外，目录还使用变音位语音算法的变体来执行近似索引搜索。每个值都被视为单词的序列，同时还为每个单词生成音标代码。

注意 iPlanet Directory Server 中的变音位语音算法仅支持 US-ASCII 字母。因此，近似索引仅能与英文值一起使用。

近似搜索中输入的值将被近似地转换为音标代码序列。如果下列两个条件都为真，则认为条目与查询匹配：

- 所有查询字符串代码均与条目字符串中生成的代码相匹配。
- 所有查询字符串代码的顺序与条目字符串代码的顺序相同。

例如，下表说明几个查询如何匹配条目名称 `Alice B. Sarette`，其语音代码为 `ALS B SRT`。

表 10-3 用音标代码近似搜索

查询字符串	音标代码	匹配注释
Alice Sarette	ALS SRT	匹配。代码顺序正确。
Alice Sarrette	ALS SRT	匹配。尽管 Sarette 拼写有误，但代码顺序正确。
Surette	SRT	匹配。尽管 Sarette 拼写有误，但生成的代码包含于原名称中。
Bertha Sarette	BR0 SRT	不匹配。代码 BR0 在原名称中不存在。
Sarette, Alice	SRT ALS	不匹配。代码顺序不正确。

权衡索引的利弊

创建新索引以前，请权衡维护索引的利弊得失。请记住：

- 近似索引对于通常包含数字 （例如电话号码）的属性而言效率不高。
- 子字符串索引对于二进制属性无效。如果值很大 （例如属性中计划包含照片或带加密数据的口令），则应避免使用等同索引。
- 如果为搜索中不常用的属性维护索引，则管理费用会增大，而全局搜索性能并不会得到改善。
- 根据搜索类型的不同，尽管搜索性能可能会显著降低，但仍然可在搜索请求中指定未索引的属性。
- 请记住：维护的索引越多，需要的磁盘空间就越多。

下例详尽描述了索引对时间的消耗程度。考虑创建特定属性的过程：

1. 目录服务器收到添加或修改操作。
2. 目录服务器检查索引属性，以确定是否为属性值维护索引。
3. 如果创建的属性值已有索引，则目录服务器将生成新的索引条目。
4. 服务器完成索引后，将根据客户机的请求而创建实际的属性值。

例如，假设系统要求目录服务器添加条目

```
dn: cn=Bill Pumice, ou=People, o=siroe.com
objectclass: top
objectClass: person
objectClass: orgperson
objectClass: inetorgperson
cn: Bill Pumice
cn: Bill
sn: Pumice
ou: Manufacturing
ou: people
telephonenumber: 408 555 8834
description: Manufacturing lead for the Z238 line.
```

不妨进一步假设目录服务器在维护下列索引：

- 通用名和姓氏属性的等同、近似和子字符串索引
- 电话号码属性的等同和子字符串索引
- 说明属性的子字符串索引

为将该条目添加到目录中，目录服务器必须执行以下步骤：

1. 为“Bill”和“Bill Pumice”创建通用名等同索引条目。
2. 为“Bill”和“Bill Pumice”创建合适的通用名近似索引条目。
3. 为“Bill”和“Bill Pumice”创建合适的通用名子字符串索引条目。
4. 为“Pumice”创建姓氏等同索引条目。
5. 为“Pumice”创建合适的姓氏近似索引条目。
6. 为“Pumice”创建合适的姓氏子字符串索引条目。
7. 为“408 555 8834”创建电话号码等同索引条目。
8. 为“408 555 8834”创建合适的电话号码子字符串索引条目。
9. 为“Manufacturing lead for the Z238 line of widgets.”创建合适的说明子字符串索引条目。此时将为该字符串生成大量的子字符串条目。

本例表明索引的成本比较高。

创建索引

本部分说明如何使用 iPlanet Directory Server Console 和命令行而为特定的属性创建存在、等同、近似、子字符串和国际索引。它还介绍创建浏览索引的分步过程。

注意 鉴于 iPlanet Directory Server 5.1 可以在单数据库或多数据库环境中运行，因此需要记住应在*每个*数据库实例中创建新索引，因为新建的索引不会在其它数据库中被自动创建。

但是，缺省索引会在后续数据库实例中自动出现并被维护，而*不被添加*到现有的数据库实例中。换句话说，目录会使用后续数据库中最近创建的缺省索引集。这意味着：如果将缺省索引添加到第二个数据库实例中，则该索引不会在第一个数据库实例中被维护，而是在所有后续实例中被维护。

本部分包含下列步骤：

- 从服务器控制台创建索引
- 从命令行创建索引
- 从服务器控制台创建浏览索引
- 从命令行创建浏览索引

从服务器控制台创建索引

使用 iPlanet Directory Server Console 可以为特定的属性创建存在、等同、近似、子字符串和国际索引。

要创建索引：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 展开数据节点，再展开要创建索引的数据库的后缀，然后选择该数据库。
3. 在右侧窗口中，选择 “索引”选项卡。

注意 不要单击 “数据库设置”节点，因为这样做将转到缺省索引设置窗口，而不是转到为每个数据库配置索引的窗口。

4. 如果要创建索引的属性已在 “附加索引”表中列出，请跳到步骤 6。否则，单击 “添加属性”。

   此时出现的对话框中包含服务器模式中所有的可用属性列表。

5. 选择要创建索引的属性，然后单击 “确定”。

   服务器将把属性添加到 “附加索引”表中。

6. 选中要为每个属性维护所有索引类型对应的复选框。
7. 如果要为非英语语言创建索引，则输入要在 “匹配规则”字段中使用的对照顺序 (collation order) 之 OID。

   通过列出由逗号分隔 （无空格）的多个 OID，可以使用多种语言来为属性创建索引。有关语言列表、其相关 OID 及关于对照顺序的详细信息，请参阅附录 D “国际化”。”

8. 单击 “保存”。

   此时出现索引对话框，其中显示索引创建状态并通知创建索引的时间。可单击 “状态日志”框以查看所创建索引的状态。创建完索引后，单击 “关闭”以退出该索引对话框。

对于目录中新添的数据和现有数据而言，新索引将立即处于活动状态。此时无须重新启动服务器。

从命令行创建索引

从命令行可以创建特定属性的存在、等同、近似、子字符串和国际索引。

从命令行创建索引包括两个步骤：

- 使用 `ldapmodify` 命令行实用程序添加新的索引条目或编辑现有的索引条目。
- 运行 `/usr/sbin/directoryserver db2index-task` 命令，生成由服务器维护的新的索引集。

注意 由于 iPlanet Directory Server 中的系统索引属于硬编码，因此无法创建新的系统索引。

下列部分介绍创建索引的步骤。

添加索引条目

使用 `ldapmodify` 将新的索引属性添加到目录中。如果要创建将成为缺省索引的新索引，请将新索引属性添加到 `cn=default indexes,cn=config,cn=ldbm database,cn=plugins,cn=config` 条目中。

要为特定的数据库创建新索引，请将其添加到 `cn=index,cn=instanceName, cn=ldbm database,cn=plugins,cn=config` 条目中，其中 `cn=`*instanceName* 对应于数据库的名称。

注意 应避免在 `dse.ldif` 文件中的 `cn=config` 条目下创建条目。 `cn=config` 条目存储在简单的、平面化的 `dse.ldif` 配置文件中，而不象一般条目那样存储在同一个、具有高度伸缩性的数据库中。因此，如果有许多条目，尤其是可能要经常更新的条目储存在 `cn=config` 下面，则性能将会受到严重影响。

然而，虽然由于性能原因不推荐在 `cn=config` 下存储简单的用户条目，但是将诸如目录管理员条目或复制管理器 （供给器绑定 DN）等特殊的用户条目储存在 `cn=config` 下很有用，因为这可以将配置信息集中起来。

有关添加条目时所需的 LDIF 更新语句的信息，请参阅第 56 页上的 “LDIF 更新语句”。

例如，假设要为 Siroe1 数据库中的 `sn` （姓氏）属性创建存在、等同和子字符串索引。

如下所示，运行 ldapmodify 命令行实用程序：

```
ldapmodify -a -h server.siroe.com -p 389 \
           -D "cn=Directory Manager" -w password
```

ldapmodify 实用程序将绑定到服务器并准备向配置文件中添加条目。有关 ldapmodify 命令行实用程序的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

接着，为新索引添加以下条目：

```
dn: cn=sn,cn=index,cn=Siroe1,cn=ldbm database,cn=plugins,cn=config
objectClass: top
objectClass:nsIndex
cn:sn
nsSystemIndex:false
nsIndexType:pres
nsIndexType:eq
nsIndexType:sub
nsMatchingRule:2.16.840.1.113730.3.3.2.3.1
```

cn 属性包含要创建索引的属性的名称，本例中为 sn 属性。该条目是 nsIndex 对象类的成员。 nsSystemIndex 属性为 false，指示该索引不是 Directory Server 操作所必需的。多值 nsIndexType 属性指定存在 (pres)、等同 (eq) 和子字符串 (sub) 索引。请注意：每个关键字都必须在单独的行上输入。 nsMatchingRule 属性指定保加利亚语对照顺序的 OID。

在 nsIndexType 属性中指定无值的索引条目时，将维护特定属性的所有索引 （国际索引除外）。例如，假设为新的 sn 索引指定以下条目：

```
dn: cn=sn,cn=index,cn=instance,cn=ldbm database,cn=plugins,cn=config
objectClass: top
objectClass:nsIndex
cn:sn
nsSystemIndex:false
nsIndexType:
```

该新条目会建立 sn （姓氏）属性的所有索引。

在 nsIndexType 属性中使用关键字 none 可指定不维护该属性的任何索引。例如，假设要临时禁用刚在 Siroe1 数据库中创建的 sn 索引，按如下所示将 nsIndexType 变换为 none：

```
dn: cn=sn,cn=index,cn=Siroe1,cn=ldbm database,cn=plugins,cn=config
objectClass: top
objectClass:nsIndex
cn:sn
nsSystemIndex:false
nsIndexType:none
```

有关对照顺序及其 OID 的完整列表，请参阅附录 D “国际化”。

有关索引配置属性的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。

注意 创建索引时，应始终使用属性的主要名称 （而非属性的别名）。属性的主要名称就是模式中为属性列出的第一个名称，例如 `uid` 是 `userid` 属性的主要名称。有关所有属性的主要名称及别名的列表，请参阅第 348 页的表 10-6。

运行 db2index-task 命令

创建完索引条目或将附加索引类型添加到现有的索引条目中后，即可运行 `/usr/sbin/directoryserver db2index-task` 命令以生成由 iPlanet Directory Server 维护的新索引集。一旦运行该命令，对于目录中新添的数据及现有的数据而言，新的索引集即处于活动状态。

命令如下：

```
# /usr/sbin/directoryserver db2index-task
```

下面的例子将创建一个索引。

```
#!/bin/sh
/usr/sbin/directoryserver db2index-task \
  -D "cn=Directory Manager" -w password -n Database1 -t sn
```

表 10-4 示例中所用的 db2index-task 选项说明

选项	说明
`-D`	指定目录管理员的 DN。
`-w`	指定目录管理员的口令。
`-n`	指定数据库名称，该数据库包含要创建索引的条目。
`-t`	指定要建立索引的数据库的属性名称。

从服务器控制台创建浏览索引

要使用 iPlanet Directory Server Console 创建浏览索引：

1. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。
2. 在左侧导航树中选择要为其创建索引的条目 （例如 `People`），然后从 “对象”菜单中选择 “创建浏览索引”。

   也可在导航树中选择并右键单击要为其创建索引的条目，然后从弹出菜单中选择 “创建浏览索引”。
3. 此时出现 “创建浏览索引”对话框，其中显示索引创建状态。可单击 “状态日志”框以查看所创建索引的状态。
4. 单击 “关闭”以退出 “创建浏览索引”对话框。

   对于添加到目录中的所有新数据而言，新索引将立即处于活动状态。此时无须重新启动服务器。

从命令行创建浏览索引

从命令行创建浏览索引 （或称虚拟列表视图 (VLV) 索引）涉及下面两个步骤：

- 使用 `ldapmodify` 添加新的浏览索引条目或编辑现有的浏览索引条目。
- 运行 `/usr/sbin/directoryserver vlvindex` 命令，生成由服务器维护的新的浏览索引集。

下列部分介绍创建浏览索引所涉及的步骤。

添加浏览索引条目

所要创建的浏览索引条目的类型与要加速的 `ldapsearch` 属性排序类型有关。考虑以下几点至关重要：

- 搜索范围 （base、 one、 sub）。
- 搜索的基础 （用作搜索起点的条目）。
- 所要排序的属性。
- 搜索过滤器。有关指定搜索过滤器的详细信息，请参阅附录 B “查找目录条目”。
- 构成搜索基的条目所属的 ldbm 数据库。

注意 您只能在 ldbm 数据库中创建浏览索引。

例如，假设您想在 Siroe1 数据库保存的 "dc=siroe,dc=com" 条目上创建浏览索引，从而加速 ldapsearch。其中，搜索基是 "dc=siroe,dc=com"，搜索过滤器是 (|(objectclass=*)(objectclass=ldapsubentry))，范围是 one，而返回属性的排序顺序是 cn、 givenname、 o、 ou 和 sn。

如下所示，运行 ldapmodify 命令行实用程序：

```
ldapmodify -a -h server -p 389 -D "cn=directory manager" -w password
```

ldapmodify 实用程序将绑定到服务器并准备向配置文件中添加条目。

接着，您需要添加两个定义浏览索引的浏览索引条目。

添加的第一个条目指定浏览索引的基础、范围和过滤器：

```
dn: cn="dc=siroe,dc=com",cn=Siroe1,cn=ldbm
database,cn=plugins,cn=config
objectClass: top
objectClass:vlvSearch
cn:"dc=siroe,dc=com"
vlvbase:"dc=siroe,dc=com"
vlvscope:one
vlvfilter:(|(objectclass=*)(objectclass=ldapsubentry))
```

cn 包含浏览索引标识符，用于指定要创建浏览索引的条目。本例中为 "dc=siroe,dc=com" 条目。建议将条目的 dn 用作浏览索引标识符 （即 iPlanet Directory Server Console 所采用的方法），以防创建相同的浏览索引。该条目是 vlvSearch 对象类的成员。 vlvbase 属性值指定创建浏览索引的条目。本例中为 "dc=siroe,dc=com" 条目 （即浏览索引标识符）。 vlvscope 属性为 one，指示所要加速的搜索的基础是 one。将 one 作为搜索基意味着仅搜索 cn 属性中所指定条目的直接子项，而并不搜索条目本身。 vlvfilter 指定搜索使用的过滤器。本例中为 (|(objectclass=*)(objectclass=ldapsubentry))。

第二个条目为返回属性指定需要的排序顺序：

```
dn:cn=sort_cn_givenname_o_ou_sn,cn="dc=siroe,dc=com",cn=Siroe1,
cn=ldbm database,cn=plugins,cn=config
objectClass: top
objectClass:vlvIndex
cn:cn=sort_cn_givenname_o_ou_sn
vlvsort:cn givenname o ou sn
```

cn 包含浏览索引排序标识符。建议使用能为所创建的浏览索引清楚地标识出搜索排序顺序的排序标识符，例如本例中的显式排序标识符 cn=sort_cn_givenname_o_ou_sn。该条目是 vlvIndex 对象类的成员。 vlvsort 属性值指定属性的排序顺序。本例中是：cn、 givenname、 o、 ou，然后是 sn。

注意 第一个浏览索引条目*必须*添加到 cn=instanceName,cn=ldbm database,cn=plugins,cn=config 目录树节点，第二个条目*必须*是第一个条目的子项。

运行 vlvindex 命令

创建完两个浏览索引条目或将附加属性类型添加到现有的索引浏览条目中后，即可运行 /usr/sbin/directoryserver vlvindex 命令以生成由 iPlanet Directory Server 维护的新浏览索引集。一旦运行该命令，对于目录中新添的数据和现有的数据而言，新索引集即处于活动状态。

要生成浏览索引，请使用以下命令：

```
# /usr/sbin/directoryserver vlvindex
```

下例使用 vlvindex 命令生成浏览索引：

```
# /usr/sbin/directoryserver vlvindex -n Database1 -T \
  "dc=siroe,dc=com"
```

表 10-5 示例中所用的 vlvindex 选项说明

选项	说明
-n	指定数据库名称，该数据库包含要创建索引的条目。
-t	指定在创建浏览索引时使用的浏览索引标识符。

删除索引

本部分介绍如何删除特定属性的存在、等同、近似、子字符串、国际和浏览索引。

注意 由于 iPlanet Directory Server 5.1 可在单数据库或多数据库环境中运行，因此必须从*每个*数据库实例中删除任何不必要的索引。

删除的任何缺省索引都*不会*从现有数据库实例的原有索引集中被删除。

由于删除浏览索引的步骤有所不同，因此该步骤将另行说明。本部分包含下列步骤：

- 从服务器控制台删除索引
- 从命令行删除索引
- 从服务器控制台删除浏览索引
- 从命令行删除浏览索引

警告 请勿删除系统索引，因为删除系统索引会严重影响 Directory Server 的性能。系统索引位于以下条目中：`cn=index,cn=instance, cn=ldbm database,cn=plugins,cn=config` 和 `cn=default indexes,cn=config,cn=ldbm database, cn=plugins,cn=config`。

删除缺省索引时要小心，因为这样也会影响 iPlanet Directory Server 的工作。

有关系统索引和缺省索引的详细信息，请参阅 *iPlanet Directory Server 部署指南*。

从服务器控制台删除索引

使用 iPlanet Directory Server Console 可以删除所创建的索引、其它 iPlanet 服务器软件 （例如 Messaging Server 或 Calendar Server）所用的索引及缺省索引。不能删除系统索引。

要使用 iPlanet Directory Server Console 删除索引：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 展开数据节点，然后展开与包含该索引的数据库相关联的后缀。选择要从中删除索引的数据库。
3. 找到包含所要删除的索引的属性。清除该索引下的复选框。

   如果要删除为特定属性所维护的所有索引，则在 “属性名称”下选择该属性的单元格，然后单击 “删除属性”。
4. 单击 “保存”。

   此时出现删除索引警告对话框，要求确认是否要删除该索引。单击 “是”以删除索引。
5. 此时出现 “删除浏览索引”对话框，显示索引删除的状态。可单击 “状态日志”按钮以查看所删除索引的状态。创建完索引后，单击 “关闭”以退出 “删除浏览索引”对话框。

从命令行删除索引

使用 `ldapdelete` 命令行实用程序可以删除索引，如下所示：

- 使用 `ldapdelete` 命令行实用程序从现有索引条目中删除整个索引条目或删除不必要的索引类型。
- 使用 `/usr/sbin/directoryserver db2index-task` 命令，重新生成由服务器维护的其余索引。

下列部分介绍删除索引所涉及的步骤。

删除索引条目

使用 `ldapdelete` 命令行实用程序从现有条目中删除整个索引条目或不必要的索引类型。

如果要删除特定数据库的索引，则从 `cn=index,cn=instanceName, cn=ldbm database,cn=plugins,cn=config` 条目中删除索引条目，其中 `cn=instanceName` 对应于数据库的名称。

要删除缺省索引，则从 cn=default indexes,cn=config,cn=ldbm database, cn=plugins,cn=config 条目中删除。

例如，您可能想删除 Siroe1 数据库中 sn 属性的存在、等同和子字符串索引。

可以删除以下条目：

```
dn: cn=sn,cn=index,cn=Siroe1,cn=ldbm database,cn=plugins,cn=config
objectClass: top
objectClass:nsIndex
cn:sn
nsSystemIndex:false
nsIndexType:pres
nsIndexType:eq
nsIndexType:sub
nsMatchingRule:2.16.840.1.113730.3.3.2.3.1
```

如下所示，执行 ldapdelete：

```
ldapdelete -h server.siroe.com -p 389 \
  -D "cn=Directory Manager" -w password \
  "cn=sn,cn=index,cn=Siroe1,dn=ldbm database,cn=plugins,dn=config"
```

删除该条目后， Siroe1 数据库即不再维护 sn 属性的存在、等同和子字符串索引。

重新生成其余索引

删除索引条目或删除索引条目的某些索引类型后，需要重新生成由 iPlanet Directory Server 维护的其余索引集。

要重新生成索引，请执行第 334 页上的 “运行 db2index-task 命令”中介绍的步骤。一旦运行该命令，对于目录中新添的数据及现有的数据而言，新的索引集即处于活动状态。

从服务器控制台删除浏览索引

要使用 iPlanet Directory Server Console 删除浏览索引：

1. 在 iPlanet Directory Server Console 上，选择 “数据库”选项卡。
2. 在导航树中选择要从中删除索引的条目 （例如 People），然后从 “对象”菜单中选择 “删除浏览索引”。也可在导航树中选择并右键单击该条目，然后从下拉菜单中选择 “删除浏览索引”。
3. 此时出现 “删除浏览索引”对话框，要求确认是否删除该索引。单击 “是”以删除索引。
4. 此时出现 “删除浏览索引”对话框，显示索引删除的状态。

从命令行删除浏览索引

从命令行删除浏览索引（或称虚拟列表视图 (VLV) 索引）涉及以下两个步骤：

- 使用 ldapmodify 删除浏览索引条目或编辑现有的浏览索引条目。
- 运行 /usr/sbin/directoryserver vlvindex 命令，重新生成其余的索引。

下列部分介绍删除浏览索引所涉及的步骤。

删除浏览索引条目

使用 ldapdelete 命令行实用程序删除浏览索引条目或编辑现有的浏览索引条目。

如果要删除特定数据库的索引，则从 cn=index,cn=instanceName,cn=ldbm database,cn=plugins,cn=config 条目中删除浏览索引条目，其中 cn=instanceName 对应于数据库的名称。

例如，您可能想删除 Siroe1 数据库中 "dc=siroe,dc=com" 条目上用于加速 ldapsearch 操作的浏览索引，其中搜索基是 "dc=siroe,dc=com"，搜索过滤器是 (|(objectclass=*)(objectclass=ldapsubentry))，范围是 one，返回属性的排序顺序是 cn、 givenname、 o、 ou 和 sn。

要删除该浏览索引，需要删除以下两个相应的浏览索引条目：

```
dn: cn="dc=siroe,dc=com",cn=Siroe1,cn=ldbm
database,cn=plugins,cn=config
objectClass: top
objectClass:vlvSearch
cn:"dc=siroe,dc=com"
vlvbase:"dc=siroe,dc=com
vlvscope:one
vlvfilter:(|(objectclass=*)(objectclass=ldapsubentry))
```

和

```
dn:cn=sort_cn_givenname_o_ou_sn,cn="dc=siroe,dc=com",cn=Siroe1,
cn=ldbm database,cn=plugins,cn=config
objectClass: top
objectClass:vlvIndex
cn:cn=sort_cn_givenname_o_ou_sn
vlvsort:cn givenname o ou sn
```

如下所示，执行 ldapdelete：

```
ldapdelete -h siroe.server.com -p 389 -D "cn=Directory Manager" -w password \
  "cn="dc=siroe,dc=com",cn=Siroe1,cn=ldbm database,cn=plugins,cn=config" \
  "cn=sort_cn_givenname_o_ou_sn,cn="dc=siroe,dc=com",cn=Siroe1, \
  cn=ldbm database,cn=plugins,cn=config"
```

删除这两个浏览索引条目后，则 Siroe1 数据库中所持有的 "dc=siroe,dc=com" 条目上用于加速 ldapsearch 操作的浏览索引（其中搜索基是 "dc=siroe,dc=com"，搜索过滤器是 (|(objectclass=*)(objectclass=ldapsubentry))，搜索范围是 one，返回属性的排序顺序是 cn、 givenname、 o、 ou 和 sn）将不再由 Siroe1 数据库来维护。

重新生成其余索引

删除浏览索引条目或删除现有浏览索引条目中不必要的属性类型后，需要重新生成由 iPlanet Directory Server 维护的其余索引集。

要重新生成浏览索引，请执行第 337 页上的 “运行 vlvindex 命令”中介绍的步骤。一旦运行该命令，对于目录中新添的数据及现有的数据而言，新的索引集即处于活动状态。

管理索引

目录所用的每个索引都由索引关键字表和匹配的条目 ID 列表 (entry ID list) 组成。目录使用条目 ID 列表来建立可能与客户机应用程序的搜索请求相匹配的候选条目列表 （详细信息，请参阅第 321 页上的 “关于索引”）。

对于每个条目 ID 列表，都会在 nsslapd-allidsthreshold 属性中为其指定一个大小限制。该大小限制普遍适用于由服务器管理的所有索引关键字，因而在逻辑上称为所有 ID 阈值 (All IDs Threshold)。当单个 ID 列表大小达到该限制值时，服务器将用一个 “所有 ID 令牌”来替换该条目 ID 列表。

“所有 ID 令牌”将使服务器假定所有目录项都与索引关键字匹配。事实上，所有 ID 令牌 (All IDs token) 会使服务器认为无任何索引可用于该搜索类型。目录假定搜索请求的其它某些方面将使服务器在处理请求前缩小其候选列表。

下例部分分析了 “所有 ID”机制的利弊。同时还给出了调整 “所有 ID 阈值”的建议。

“所有 ID”机制的优点

对于搜索结果将是大多数或所有目录项的情况 （例如 cn=* 搜索）而言，“所有 ID”机制可以提高搜索的性能。假定所有条目 ID 均由 iPlanet Directory Server 返回：

- 不会无限地增加条目 ID 列表，这样会减小 iPlanet Directory Server 的可用磁盘空间。
- 不会为响应搜索请求而无谓地将大的条目 ID 列表加载到内存中以得到所有目录项，这样可通过减少大量的磁盘读取操作来提高搜索的性能。
- 不会要求大量的 RAM 以将不必要的大条目 ID 列表保存在内存中。

“所有 ID”机制的缺点

如果 “所有 ID 阈值”对于目录大小而言设置得过低 （这是最常见的问题）或过高，就可能出现性能问题。

当 “所有 ID 阈值”设置得过低时

当 “所有 ID 阈值”设置得过低时，将会有很多索引关键字包含 “所有 ID 令牌”。这会导致过多的目录搜索过程检查目录中的每个条目。搜索性能受到很大的影响。

例如，假设您在管理通用名 (cn) 属性上的等同索引。存储在 cn 索引中的索引关键字之一是 cn=James。相应的条目 ID 列表中包含属性设置为 James 的每个条目的 ID 号。

cn 属性的等同索引易于维护，因为目录中只有很小一部分条目包括 cn=James。对于使用 cn=James 过滤器的搜索而言，性能将得到提高，因为在响应搜索请求时只需检查很小部分的条目 ID。

但是，目录可能会随时间而持续增大。这样就可能会添加越来越多的 James，但它们仍然只占总目录条目的一小部分。最后， cn=James 条目 ID 列表可能会变得相当大，但它对于搜索性能而言仍是必需的列表。如果目录变得足够大，从而导致添加的 cn=James 条目达到 “所有 ID 阈值”的限制，则 cn=James 条目 ID 列表将被替换为 “所有 ID 令牌”。每次搜索 cn=James 时，目录服务器都为响应搜索请求而检查目录中的各个条目。

当数据库变得非常大时，“所有 ID 阈值”将被设为所有索引关键字的较大比例，从而导致搜索性能显著降低。

当 “所有 ID 阈值”设置得过高时

“所有 ID 阈值”设置得过高时同样会产生性能问题。如果 “所有 ID 阈值”非常高，则不得不维护一个较大的条目 ID 列表，且该列表将在响应搜索请求时被加载到内存中。如果 “所有 ID 阈值”非常高，有可能抵消 “所有 ID”机制的全部优点 （详细信息，请参阅第 343 页上的 ““所有 ID”机制的优点”）。

单个企业目录的 “所有 ID 阈值”调整建议

更改服务器的 “所有 ID 阈值”缺省值时，请务必小心。如果将阈值更改为不恰当的值，则可能会降低而非提高服务器的性能。该调整建议主要适用于 80,000 个条目以下的单个企业目录。

如果目录大小较为稳定，则将 “所有 ID 阈值”设置为目录中总存储条目数的 5%。也就是说，如果目录中有 50,000 个条目，则将 “所有 ID 阈值”设为 2,500。

如果近期计划在目录中添加大量的条目，则应仔细考虑 “所有 ID 阈值”的值。应考虑以下情况：

- 更改 “所有 ID 阈值”意味着必须重建数据库。该操作的代价可能比较高，特别是对于包含数百万个条目的目录而言。
- 虽然我们建议将 “所有 ID 阈值”设为当前数据库大小的 5%，但即使 “所有 ID 阈值”的设置低至当前数据库大小的 0.5% 或高达 50%，也不应出现严重的性能问题。但是，我们仍然建议将该值尽可能地保持在 5% 左右。

您应该根据当前的目录需要和将来的扩展计划进行权衡，以避免以后更改 “所有 ID 阈值”（需要重建数据库）。

例如，假设当前数据库的大小为 50,000 个条目。但预计目录会在几年内增长到 1,000,000 个条目。如果将 “所有 ID 阈值”设置为 50,000 的 5%(2,500)，则当目录增长为 1,000,000 个条目时，就会出现性能问题。对于包含 1,000,000 个条目的数据库来说， 2,500 太低，因为 1,000,000 个条目的数据库的下限是 1,000,000 的 0.5%，即 5,000 个条目。

如果预计目录会在将来增长到非常大，则可执行如下操作之一：

- 将 “所有 ID 阈值”设置为当前的最佳值 (2,500)，同时当目录变得足够大时，可规划重建数据库以保证其仍然可靠。数据库的重建意味着在重建期间将一直关闭目录，或至少将目录置于只读模式。这也意味着将重新初始化从目录服务器接收复制条目的所有客户服务器。
- 确定一个略大于当前需要但能很好地满足未来需要的值。例如，如果当前目录包含 50,000 个条目，则尝试将 “所有 ID 阈值”设置为 20,000，即 50,000 的 40％ （满足当前目录需要的范围之内）或 1,000,000 的 2％ （满足未来目录需要的范围之内）。

选择何种战略取决于目录部署的需要。当 “所有 ID 阈值”的值偏离 5％ 这一理想设置时，请考虑重建数据库 （及所有相关客户服务器）的代价与潜在性能影响之间的关系。

注意 在客户服务器上设置不同的 “所有 ID 阈值”也是有一定意义的，因为这样可以调整该阈值以满足不同搜索的需要。

同时，还应考虑目录增长的速度以及增加目录大小要花费的时间。如果目录增长需要若干年的时间，则可以计划重建数据库。如果几个月内目录大小即以数量级或更快速度增长，则考虑设置 “所有 ID 阈值”的方法以使重建数据库的时间间隔降到最低。

对于服务供应商和 Extranet 的 “所有 ID 阈值”调整建议

对于主机服务供应商和 Extranet 的目录或超过 80,000 个条目的目录而言，若需调整建议，可以联系 *iPlanet 专业服务*。

“所有 ID 阈值”的缺省值

默认情况下，目录服务器的 “所有 ID 阈值”设为 4000。该值适用于最多 80,000 个条目的数据库。如果预计数据库大于 80,000 个条目，则建议在填充数据库前将 “所有 ID 阈值”更改为较大的值。

“所有 ID 阈值”不合适的征兆

当“所有 ID 阈值”的设置不正确时，搜索性能将降低。但其它原因也可能造成搜索性能降低。例如：

- 用户正在对未维护其索引的对象执行大量搜索操作。
- 数据库缓存大小和条目缓存大小可能设置不正确。有关详细信息，请参阅第 391 页上的 “调整 Directory Server 的性能”。

在更改 “所有 ID 阈值”前，应先仔细检查这些可能性。

如果您认为服务器的 “所有 ID 阈值”过低，可查看访问日志。请参阅第 12 章 “监控服务器和数据库活动”。任何导致返回所有条目 ID 的搜索均包含 `notes=U` 标记。以下搜索将返回 `notes=U` 标记：

- 未为其维护索引的搜索
- 未为其维护 ID 列表的搜索 （因为已达到索引关键字的 “所有 ID 阈值”）

要确定搜索结果是否属于本应创建索引的搜索，则必须使访问日志中的 RESULT 行与其上面 SRCH 行中的 `conn` 和 `op` 值相匹配。 SRCH 行将显示用于搜索请求的搜索过滤器。如果指定的搜索过滤器有索引，则达到索引关键字的 “所有 ID 阈值”时将返回 `notes=U` 标记。例如，访问日志如下所示：

```
[24/July/1998:15:12:20 -0800] conn=2 op=1 SRCH base="o=siroe.com"
scope=0 filter="(cn=James)"

[24/July/1998:15:12:20 -0800] conn=2 op=1 RESULT err=0 tag=101
nentries=10000 notes=U
```

`notes=U` 标记的存在指示已达到 `cn` 属性索引的 “所有 ID 阈值”。

更改 “所有 ID 阈值”的值

要更改服务器的 “所有 ID 阈值”：

1. 关闭 iPlanet Directory Server。
2. 使用命令行将所有目录数据库导入 LDIF。

   有关详细信息，请参阅第 4 章 “填充目录数据库”。
3. 使用 `ldapmodify` 实用程序编辑 `nsslapd-allidsthreshold` 条目，或编辑以下文件：

   `/var/ds5/slapd-`*serverID*`/config/dse.ldif`
4. 找到 `nsslapd-allidsthreshold` 属性，然后将其值更改为所希望的设置。
5. 使用 `ldif2db` 初始化所有数据库。

   请参阅第 4 章 “填充目录数据库”。
6. 重新启动 iPlanet Directory Server。

增大 “所有 ID 阈值”的值后，检查数据库缓存的大小。

增大 “所有 ID 阈值”会因条目 ID 列表变大而导致内存需求增大。根据所维护索引的数量和类型，内存需求增加量也会有所不同，但内存需求始终不会大于 `nsslapd-allidsthreshold` 属性值的增大系数。也就是说，如果 `nsslapd-allidsthreshold` 属性值加倍，则数据库缓存大小的增长不应超过其当前值的一倍。

最为极端的情况是数据库缓存大小的增长与 “所有 ID 阈值”的增大系数持平。如果有可用的物理内存，则尝试按照 `nsslapd-allidsthreshold` 值增加量的 25% 来增加数据库的缓存大小。例如，如果将 “所有 ID 阈值”翻倍，则将数据库缓存大小按 50% 增长。必要时，可缓慢增加缓存大小，直到服务器性能令人满意为止。

使用 `nsslapd-dbcachesize` 属性设置数据库缓存的大小。有关详细信息，请参 *iPlanet Directory Server 配置、命令和文件参考指南*中的 `nsslapd-dbcachesize` 属性。

属性名称快速参考表

下表列出了有主要名称 （或真实名称）及别名的所有属性。创建索引时，请务必使用主要名称。

表 10-6 属性主要名称及其别名

属性主要名称	属性别名
dn	distinguishedName
cn	commonName
sn	surName
c	countryName
l	localityName
st	stateOrProvinceName
street	streetAddress
o	organization
ou	organizationalUnitName
facsimileTelephoneNumber	fax
uid	userId
mail	rfc822mailbox
mobile	mobileTelephoneNumber
pager	pagerTelephoneNumber
co	friendlyCountryName
labeledUri	labeledUri
ttl	timeToLive
dc	domainComponent
authorCn	documentAuthorCommonName
authorSn	documentAuthorSurname
drink	favoriteDrink

第 11 章

管理 SSL

为了提供网络上的安全通讯， iPlanet Directory Server 中提供了 LDAPS 通讯协议。LDAPS 是标准 LDAP 协议，但它在安全套接层 (SSL) 的最上层运行。

本章以下部分介绍如何将 SSL 用于 iPlanet Directory Server：

- iPlanet Directory Server 中 SSL 简介
- 获取和安装服务器证书
- 激活 SSL
- 设置安全性首选项
- 使用基于证书的验证
- 配置 LDAP 客户机以使用 SSL

iPlanet Directory Server 中 SSL 简介

使用 SSL，您可在 LDAP 客户端和 iPlanet Directory Server 之间，或通过复制协议绑定的 iPlanet Directory Server 相互之间，或数据库链接和远程数据库之间提供安全通讯。 SSL 可与简单验证 （绑定 DN 和口令）配合使用，或与基于证书的验证配合使用。

将 SSL 和简单验证配合使用可确保私密性和数据完整性。使用证书而非绑定 DN 和口令进行 iPlanet Directory Server 验证的优势包括：

- 提高效率

  如果所用的应用程序提示您输入自己的证书数据库口令，并随即将该证书用于所有后续绑定和验证操作，则相对于不断提供绑定 DN 和口令而言效率会更高。

- 提高安全性

  使用基于证书的验证比执行无证书绑定操作更安全。这是因为基于证书的验证使用公开密钥密码术。因此将不能跨网络截取绑定凭证。

iPlanet Directory Server 能同时进行 SSL 和非 SSL 通讯。这意味着您不必为 iPlanet Directory Server 在 SSL 和非 SSL 通讯之间作出选择；您可同时使用这两种通讯。

注意 如果 iPlanet Directory Server 在 UNIX 平台上运行，则启用 SSL 也将同时支持 StartTLS 扩展操作。 StartTLS 扩展操作可提供正常 LDAP 连接上的安全性。

启用 SSL：步骤摘要

要使用 LDAPS，则必须执行以下操作：

1. 获取并安装 iPlanet Directory Server 证书，然后将 iPlanet Directory Server 配置为信任证书授权机构的证书。

   有关信息，请参阅第 351 页上的 “获取和安装服务器证书”。

2. 在目录中打开 SSL。

   有关信息，请参阅第 355 页上的 “激活 SSL”。

3. 配置管理服务器以连接到启用 SSL 的 iPlanet Directory Server。

   有关信息，请参阅*通过 iPlanet Console 管理服务器*。

4. 对于要进行 SSL 验证的所有客户机而言，也可确保 iPlanet Directory Server 的每位用户都获取并安装该客户机的个人证书。

   有关信息，请参阅第 359 页上的 “配置 LDAP 客户机以使用 SSL”。

有关 SSL、 Internet 安全和证书的完整说明，请参阅*通过 iPlanet Console 管理服务器*。

获取和安装服务器证书

本部分介绍以下过程：创建证书数据库、获取并安装 iPlanet Directory Server 所用的证书，以及将 iPlanet Directory Server 配置为信任证书授权机构 (CA) 的证书。

该过程是您在目录中打开 SSL 以前必须执行的第一步。如果已完成这些任务，请参阅第 355 页上的 “激活 SSL”。

获取并安装证书包括以下步骤：

- 步骤 1：生成证书请求
- 步骤 2：发送证书请求到证书授权机构
- 步骤 3：安装证书
- 步骤 4：信任证书授权机构
- 步骤 5：确认已安装新的证书

您需要使用 “证书请求向导”生成证书请求 （步骤 1），然后将该请求发送到证书授权机构 （步骤 2）。之后，请使用 “证书安装向导”来安装证书 （步骤 3），同时信任证书授权机构的证书 （步骤 4）。

这些向导将自动执行创建证书数据库和安装密钥对的过程。

步骤 1：生成证书请求

要生成证书请求并将该请求发送到 CA：

1. 在 iPlanet Directory Server Console 上，选择 “任务”选项卡，然后单击 “管理证书”。

   此时出现 “管理证书”窗口。

2. 选择 “服务器证书”选项卡，然后单击 “请求”按钮。

   此时出现 “证书请求向导”。

3. 单击 “下一步”。

4. 在空白文本字段中输入 “请求方信息”，然后单击 “下一步”。

   输入以下信息：

   服务器名。输入 iPlanet Directory Server 在 DNS 查找中所用的全限定主机名（例如：`dir.siroe.com`）。

组织。输入公司或机构的法律名称。大多数 CA 会要求用营业执照副本等法律文件来证实该信息。

组织单元。（可选）。为公司内的组织输入具有说明性的名称。

地区。（可选）。输入公司所在城市的名称。

州或省。输入公司所在州或省的全称 （不要缩写）。

国家。为国家名称选择两个字符的缩写 （ISO 格式）。美国的国家代码为 US。在 *iPlanet Directory Server 模式参考指南*中包含 ISO 国家代码的完整列表。

5. 输入要用于保护专用密钥的口令，然后单击 “下一步”。

   提供口令之前，“下一步”字段将一直变灰。单击 “下一步”，出现 “请求提交”对话框。

6. 选择 “复制到剪贴板”或 “保存到文件”，从而保存必须发送到证书授权机构的证书请求信息。

7. 单击 “完成”以关闭 “证书请求向导”。

生成请求后，即可将该请求发送给 CA。

步骤 2：发送证书请求

执行下列步骤可将证书信息发送给 CA：

1. 使用电子邮件程序创建新的电子邮件。

2. 将证书请求信息从剪贴板或保存的文件中复制到邮件正文内。

   其内容类似于下例：

```
-----BEGIN NEW CERTIFICATE REQUEST-----
MIIBrjCCARcCAQAwbjELMAkGA1UEBhMCVXMxEzARBgNVBAgTCkNBTElGT1JOSUExLD
AqBgVBAoTI25ldHNjYXBlIGNvbW11bmljYXRpb25zIGNvcnBvcmF0aW9uMRwwGgYDV
QQDExNtZWxsb24ubmV0c2NhcGUuY29tMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQK
BgQCwAbskGh6SKYOgHy+UCSLnm3ok3X3u83Us7ug0EfgSLR0f+K41eNqqWRftGR83e
mqPLDOf0ZLTLjVGJaH4Jn4l1gG+JDf/n/zMyahxtV7+mT8GOFFigFfuxJaxMjr2j7I
vELlxQ4IfZgWwqCm4qQecv3G+N9YdbjveMVXW0v4XwIDAQABoAAwDQYJKoZIhvcNAQ
EEBQADgYEAZyZAm8UmP9PQYwNy4Pmypk79t2nvzKbwKVb97G+MT/gw1pLRsI1uBoKi
nMfLgKp1Q38K5Py2VGW1E47K7/rhm3yVQrIiwV+Z8Lcc=
-----END NEW CERTIFICATE REQUEST-----
```

3. 向 CA 发送电子邮件。

通过电子邮件发出请求后，必须等待 CA 对证书做出响应。请求的响应时间会各有不同。例如，如果 CA 在您公司内部，则可能只需一两天时间便会对请求做出响应。如果所选的 CA 在公司外部，则可能需要数周才会对请求做出响应。

当 CA 做出响应时，请务必将信息保存到文本文件中。安装证书时将需要该数据。

同时，应将证书数据备份到安全的地方。如果系统一旦丢失证书数据，则可使用备份文件来重新安装证书。

收到证书后，即可在服务器的证书数据库中安装该证书。

步骤 3：安装证书

要安装服务器证书：

1. 在 iPlanet Directory Server Console 上，选择 “任务”选项卡，然后单击 “管理证书”。

   此时出现 “管理证书”窗口。

2. 选择 “服务器证书”选项卡，然后单击 “安装”。

   此时出现 “证书安装向导”。

3. 选择以下证书位置选项之一，然后单击 “下一步”。

   本文件中。在该字段中输入证书的绝对路径。

   在下列编码文本块中。将文本从 CA 电子邮件或所创建的文本文件复制并粘贴到该字段中。例如：

```
-----BEGIN CERTIFICATE-----
MIICMjCCAZugAwIBAgICCEEwDQYJKoZIhvcNAQEFBQAwfDELMAkGA1UEBhMCVVMx
IzAhBgNVBAoTGlBhbG9va2FWaWxsZSBXaWRnZXRzLCBJbmMuMR0wGwYDVQQLExRX
aWRnZXQgTWFrZXJzICdSJyBVczEpMCcGA1UEAxMgVGVzdCBUZXN0IFRlc3QgVGVz
dCBUZXN0IFRlc3QgQ0EwHhcNOTgwMzEyMDIzMzU3WhcNOTgwMzI2MDIzMzU3WjBP
MQswCQYDVQQGEwJVUzEoMCYGA1UEChMfTmV0c2NhcGUgRGlyZWN0b3J5IFB1Ymxp
Y2F0aW9uczEWMBQGA1UEAxMNZHVgh49dq2itLmNvbTBaMA0GCSqGSIb3DQEBAQUA
A0kAMEYCQQCksMR/aLGdfp4m0OiGcgijG5KgOsyRNvwGYW7kfW+8mmijDtZRjYNj
jcgpF3VnlsbxbclX9LVjjNLC57u37XZdAgEDozYwNDARBglghkgBhvhCAQEEBAMC
APAwHwYDVR0jBBgwFoAU67URjwCaGqZuUpSpdLxlzweJKiMwDQYJKoZIhvcNAQEF
BQADgYEAJ+BVem3vBOP/BveNdLGfjlb9hucgmaMcQa98A/db8qimKT/ue9UGOJqL
bwbMKBBopsD56p2yV3PLJIsBgrcuSoBCuFFnxBnqSiTS/7YiYgCWqWaUAExJFmD6
6hBLseqkSWulk+hXHN7L/NrViO+7zNtKcaZLlFPf7d7j2MgX4Bo=
-----END CERTIFICATE-----
```

4. 检查显示的证书信息是否正确，然后单击 “下一步”。
5. 指定证书的名称，然后单击 “下一步”。
6. 提供用于保护私用密钥的口令以核查证书。

   该口令与第 351 页上的 “步骤 1：生成证书请求”中所提供的口令相同。

安装完证书后，需要将服务器配置为信任颁发服务器证书的证书授权机构。

步骤 4：信任证书授权机构

将 iPlanet Directory Server 配置为信任证书授权机构的过程由两部分组成：获取 CA 证书；将证书安装到服务器的证书数据库中。该过程因使用的证书授权机构而异。某些商业性 CA 提供网站，供自动下载证书。其它 CA 则会在收到请求后以电子邮件的方式将证书发送出去。

获得 CA 证书后，即可使用 “证书安装向导”将 iPlanet Directory Server 配置为信任证书授权机构。

1. 在 iPlanet Directory Server Console 上，选择 “任务”选项卡，然后单击 “管理证书”。

   此时出现 “管理证书”窗口。

2. 转到 “CA 证书”选项卡，然后单击 “安装”。

   此时出现 “证书安装向导”。

3. 如果已将 CA 证书保存到文件中，则在所提供的字段中输入路径。如果是通过电子邮件收到的 CA，则将证书连同标题一起复制并粘贴到所提供的文本字段中。单击 “下一步”。
4. 检查显示的证书信息是否正确，然后单击 “下一步”。
5. 指定证书的名称，然后单击 “下一步”。
6. 选择信任该证书授权机构的目的 （可两个全选）：

   接受来自客户机的连接 （客户机验证）。服务器将检查信任的证书授权机构是否已向客户机颁发证书。

   接受到其它服务器的连接 （服务器验证）。该服务器将检查受托的证书授权机构是否已向正在连接的目录 （例如：用于复制更新）颁发证书。

7. 单击 “完成”以关闭向导。

安装完证书并信任 CA 证书后，即可激活 SSL。但是，首先应确保已正确安装证书。

步骤 5：确认已安装新的证书

1. 在 iPlanet Directory Server Console 上，选择 “任务”选项卡，然后单击 “管理证书”。

   此时出现 “管理证书”窗口。

2. 选择 “服务器证书”选项卡。

   此时将显示服务器上所有已安装证书的列表。

3. 滚动列表。找到自己安装的证书。

   现在，服务器上已可以激活 SSL。

激活 SSL

大多数情况下，您会希望服务器在运行时启用 SSL。如果临时禁用 SSL，则在处理需要保密、验证或数据完整性的事务以前，务必重新启用 SSL。

激活 SSL 之前，必须如第 351 页上的 “获取和安装服务器证书”中所述创建证书数据库，获取并安装服务器证书并信任 CA 证书。

要激活 SSL 通信：

1. 设置希望服务器用于 SSL 通信的安全端口。有关信息，请参阅第 36 页上的 “更改目录服务器端口号”。

   指定的加密端口号不得与常规 LDAP 通信所用的端口号相同。默认情况下，标准端口号为 389，而安全端口号为 636。

2. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树最上端的条目。

3. 在右侧窗口中，选择 “加密”选项卡。

   选项卡上将显示当前服务器的加密设置。

4. 选中 “启用该服务器的 SSL”复选框，即指示想启用加密功能。

5. 选中 “使用该加密系列”复选框。

6. 从下拉菜单中选择所要使用的证书。

7. 单击 “加密设置”。

   此时出现 “加密首选项”对话框。

8. 选中所要使用的密码旁边的复选框，然后单击 “确定”以关闭 “加密首选项”对话框。

   有关特定密码的详细信息，请参阅第 356 页上的 “设置安全性首选项”。

9. 设置客户机验证的首选项。

   不允许客户机验证。选中该选项时，服务器将忽略客户机的证书。这并不意味着绑定失败。

   允许客户机验证。此为默认设置。选中该选项时，将在客户机发出请求时执行验证。有关基于证书的验证的详细信息，请参阅第 358 页上的 “使用基于证书的验证”。

   要求客户机验证。选中该选项时，服务器将请求进行客户机验证。

注意 如果是将基于证书的验证与复制一起使用，则必须将客户服务器配置为允许或要求进行客户机验证。

---

10. 如果在与 iPlanet Directory Server 通讯时希望 iPlanet Console 使用 SSL，则在 iPlanet Console 中选择 “使用 SSL”。
11. 单击 “保存”。
12. 重新启动 iPlanet Directory Server。

    有关详细信息，请参阅第 39 页上的 “在启用 SSL 的情况下启动服务器”。

设置安全性首选项

选择要用于 SSL 通讯的密码类型。*密码*是一种加密算法。有的密码比其它密码更安全或*功能更强大*。一般而言，密码在加密时使用的位数越多，就越难进行密钥解密。有关算法及其功能强大性的完整说明，请参阅*通过 iPlanet Console 管理服务器*。

当客户机启动与服务器的 SSL 连接时，客户机会告诉服务器它优先选用什么密码对信息加密。在双向加密过程中，双方必须使用相同的密码。可用的密码有许多。服务器需要能使用客户机应用程序连接服务器时所用的密码。

iPlanet Directory Server 提供下列 SSL 3.0 密码：

- 带 40 位加密和 MD5 消息验证的 RC4 密码。
- 带 40 位加密和 MD5 消息验证的 RC2 密码。
- 不加密，只进行 MD5 消息验证。

- 带 56 位加密和 SHA 消息验证的 DES。
- 带 128 位加密和 MD5 消息验证的 RC4 密码。
- 带 168 位加密和 SHA 消息验证的三元 DES。
- 带 56 位加密和 SHA 消息验证的 FIPS DES。此加密系列符合 FIPS 140-1 美国政府就加密模块的实施标准。
- 带 168 位加密和 SHA 消息验证的 FIPS 三元 DES。该密码符合 FIPS 140-1 美国政府就加密模块实施制订的标准。

要选择服务器所用的密码：

1. 确保服务器已启用 SSL。

   有关信息，请参阅第 355 页上的 “激活 SSL”。

2. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树最上端的条目。
3. 在右侧窗口中，选择 “加密”选项卡。

   此时出现当前服务器的加密设置。

4. 单击 “加密设置”。

   此时出现 “加密首选项”对话框。

5. 在 “加密首选项”对话框中，通过从列表中选择密码来指定服务器所要使用的密码，然后单击 “确定”。

   除非因安全原因而未使用特定的密码，否则应选择所有密码，但 `none,MD5` 除外。

6. 在 “加密”选项卡上，单击 “保存”。

警告 请勿选择 `none,MD5` 密码。这是因为如果客户机上无其它密码可用，则服务器将使用该选项。在这种情况下，由于没有使用加密，因此连接不安全。

为继续使用带 SSL 的 iPlanet Console，必须至少选择下列密码之一：

- 带 40 位加密和 MD5 消息验证的 RC4 密码。
- 不加密，只进行 MD5 消息验证。
- 带 56 位加密和 SHA 消息验证的 DES。
- 带 128 位加密和 MD5 消息验证的 RC4 密码。
- 带 168 位加密和 SHA 消息验证的三元 DES。

使用基于证书的验证

iPlanet Directory Server 允许将基于证书的验证与命令行工具 （LDAP 客户机）和复制通信配合使用。基于证书的验证可发生在下列对象之间：

- 连接 iPlanet Directory Server 的 LDAP 客户机
- 一台连接到另一台 iPlanet Directory Server （复制 (replication) 或链接 (chaining)）的 iPlanet Directory Server

设置基于证书的验证

要设置基于证书的验证，您必须：

1. 为客户机和服务器创建证书数据库，或者为复制过程所涉及的两台服务器创建证书数据库。

   在 iPlanet Directory Server 上，证书数据库是在安装证书时自动创建的。有关创建客户机证书数据库的信息，请参阅第 359 页上的 “配置 LDAP 客户机以使用 SSL”。

2. 在客户机和服务器上，或者在复制过程所涉及的两台服务器上获取并安装证书。
3. 在服务器上，或者在复制过程所涉及的两台服务器上启用 SSL。

   有关启用 SSL 的信息，请参阅第 355 页上的 “激活 SSL”。

注意 如果 iPlanet Console 通过 SSL 连接到 iPlanet Directory Server，则选择 “要求客户机验证”将禁用通讯功能。这是因为虽然 iPlanet Console 支持 SSL，但它没有用于客户机验证的证书。

---

4. 将证书的特异名称映射为目录已知的特异名称。

   这样，在使用该证书进行绑定的情况下，可以设置客户机的访问控制。该映射过程的说明见*通过 iPlanet Console 管理服务器*。

允许/请求客户机验证

如果已将 iPlanet Console 配置为使用 SSL 连接 iPlanet Directory Server，*且* iPlanet Directory Server *请求*进行客户机验证，则不能再使用 iPlanet Console 来管理任何 iPlanet 服务器。相反，此时必须使用相应的命令行实用程序。

但是，如果以后想将目录配置更改为不再*要求*而是*允许*进行客户机验证，从而可以使用 iPlanet Console，则必须执行以下步骤：

1. 停止 iPlanet Directory Server。

   有关从命令行停止和启动服务器的信息，请参阅第 35 页上的 “从命令行启动 / 停止服务器”。

2. 将 `nsSSLClientAuth` 属性的值从 `required` 更改为 `allowed`，从而对 `cn=encryption,cn=config` 条目进行修改。

   有关从命令行修改条目的信息，请参阅第 2 章 “创建目录项”。

3. 启动 iPlanet Directory Server。

   现在即可启动 iPlanet Console。

配置 LDAP 客户机以使用 SSL

如果希望 iPlanet Directory Server 的所有用户在利用 LDAP 客户机应用程序进行连接时都使用 SSL 或基于证书的验证，则确保他们执行以下任务：

- 创建证书数据库。
- 信任发放服务器证书的证书授机机构 (CA)。

如果想确保 LDAP 客户机能识别服务器的证书，则执行这些操作已足够。但是，如果还想让 LDAP 客户机使用自己的证书进行目录验证，则应确保所有目录用户都已获取并安装了个人证书。

注意 有些客户机应用程序不会验证服务器是否有信任的证书。

以下过程说明如何使用 Netscape Communicator 4.7 来执行该任务。

1. 要创建证书，则启动 Netscape Communicator 4.7 即可。

   如果尚未存在，则将创建证书数据库。

2. 使用 Communicator 连接到证书授权机构。

   如果是使用内部部署的 iPlanet 证书服务器，则将转到以下 URL：

   `https://`*hostname*`:444`

   某些证书授权机构所提供的链接允许下载 CA 证书。

3. 信任证书授权机构。

   该任务因 CA 而异。某些情况下 （例如连接 iPlanet 证书服务器时），Communicator 将自动提示您确定是否信任 CA。

这些步骤足以确保客户机应用程序接受与 iPlanet Directory Server 的连接，因为客户机可识别出 iPlanet Directory Server 证书系由信任的 CA 发放。

但是，如果还希望 iPlanet Directory Server 使用客户机证书来验证客户机，则必须执行以下附加步骤：

4. 在客户机系统上，从 CA 获取客户机证书。

5. 在客户机系统上，安装客户机证书。

   不管采用哪种方式接收证书 （电子邮件或 web 页），都应有供单击后用于安装证书的链接。单击该链接并按照 Communicator 所示的各种对话框进行操作。

   确保已将通过文件发送来的证书信息记录下来。特别是必须知道证书的主题 DN，因为您必须配置服务器以便将其映射到目录项。客户机证书将类似于：

```
-----BEGIN CERTIFICATE-----
MIICMjCCAZugAwIBAgICCEEwDQYJKoZIhvcNAQEFBQAwfDELMAkGA1UEBhMCVVMx
IzAhBgNVBAoTGlBhbG9va2FWaWxsZSBXaWRnZXRzLCBJbmMuMR0wGwYDVQQLExRX
aWRnZXQgTWFrZXJzICdSJyBVczEpMCcGA1UEAxMgVGVzdCBUZXN0IFRlc3QgVGVz
dCBUZXN0IFRlc3QgQ0EwHhcNOTgwMzEyMDIzMzU3WhcNOTgwMzI2MDIzMzU3WjBP
MQswCQYDVQQGEwJVUzEoMCYGA1UEChMfTmV0c2NhcGUgRGlyZWN0b3J5IFB1Ymxp
Y2F0aW9uczEWMBQGA1UEAxMNZHVgh49dq2itLmNvbTBaMA0GCSqGSIb3DQEBAQUA
A0kAMEYCQQCksMR/aLGdfp4m0OiGcgijG5KgOsyRNvwGYW7kfW+8mmijDtZRjYNj
jcgpF3VnlsbxbclX9LVjjNLC57u37XZdAgEDozYwNDARBglghkgBhvhCAQEEBAMC
APAwHwYDVR0jBBgwFoAU67URjwCaGqZuUpSpdLxlzweJKiMwDQYJKoZIhvcNAQEF
BQADgYEAJ+BVem3vBOP/BveNdLGfjlb9hucgmaMcQa98A/db8qimKT/ue9UGOJqL
bwbMKBBopsD56p2yV3PLJIsBgrcuSoBCuFFnxBnqSiTS/7YiYgCWqWaUAExJFmD6
6hBLseqkSWulk+hXHN7L/NrViO+7zNtKcaZLlFPf7d7j2MgX4Bo=
-----END CERTIFICATE-----
```

6. 必须使用 certutil 实用程序将客户机证书转换为二进制格式。为此：

   a. 从 http://www.iplanet.com 下载 certutil 实用程序。

      在 iPlanet 主页上，搜索 **certutil**。下载最新的 PKCS 软件包。它将包含 certutil 实用程序。

   b. 按如下方式运行 certutil：

   ```
   certutil -L -d cert7.db_path -n user_cert_name -r > user_cert.bin
   ```

      其中 *cert7.db_path* 指证书数据库的位置， *user_cert_name* 是安装证书数据库时赋予证书的名称，而 *user_cert.bin* 则指必须为包含二进制证书的输出文件指定的名称。

7. 在服务器上，通过编辑 certmap.conf 文件，将所获取证书的主题 DN 映射为相应的目录项。

   有关该过程的说明见*通过 iPlanet Console 管理服务器*。确保 certmap.conf 文件中的 verifyCert 参数设置为 **on**。

注意 注意：如果该参数未设置为 **on**，则 iPlanet Directory Server 将只搜索目录中与 certmap.conf 文件中的信息相匹配的条目。如果搜索成功，它将不实际检查 userCertificate 属性的值即授予访问权。

---

8. 在 iPlanet Directory Server 上，必须对那些拥有客户机证书的用户的目录项进行修改，以便添加 userCertificate 属性。

   a. 选择 “目录”选项卡，然后导航到用户条目。

   b. 双击用户条目，然后使用属性编辑器按二进制子类型来添加 userCertificate 属性。

      添加该属性时，服务器将提供 “设置值”按钮，而非可编辑的字段。

   c. 单击 “设置值”。

      此时显示文件选择器。使用该选择器可以选择步骤 6 中所创建的二进制文件。

   有关使用 iPlanet Directory Server Console 编辑条目的信息，请参阅第 45 页上的 “修改目录项”。

第 12 章

监控服务器和数据库活动

本章介绍监控数据库及服务器日志。本章包含以下几部分：

- 查看和配置日志文件
- 手动日志文件循环
- 监控服务器活动
- 监控数据库活动
- 监控数据库链接活动

有关利用 SNMP 监控服务器的信息，请参阅第 13 章 “使用 SNMP 监控 iPlanet Directory Server”。

查看和配置日志文件

iPlanet Directory Server 提供了三种类型的日志，用于帮助您更好地管理目录和调整性能。这些日志包括：

- 访问日志
- 错误日志
- 审计日志

以下各项对所有类型的日志配置都通用：

- 定义日志文件创建策略
- 定义日志文件删除策略

下面各部分介绍如何定义日志文件创建和删除策略，以及如何查看和配置各种类型的日志。

定义日志文件循环策略

如果需要目录定期对当前日志进行归档并开始新的日志，可以利用 iPlanet Directory Server Console 定义日志文件循环策略。您可以配置以下参数：

- 希望目录保留的日志总数。目录达到该日志数后，将在创建新日志之前删除文件夹中最早的日志文件。缺省设置为 10 个日志。切勿将该值设为 1。否则，目录将不会进行日志循环，从而使日志无限增大。
- 每个日志文件所占的最大空间 (MB)。如果不想设置最大空间限制，则在该字段中键入 -1。缺省设置为 100 MB。一旦日志文件达到最大空间限制 （或者是下一步中定义的最长存在周期），该目录就会将文件归档并开始新的日志文件。如果将最大日志数设置为 1，目录就将忽略该属性。
- 输入分钟数、小时数、天数、周数或月数，从而确定目录归档当前日志文件并创建新日志文件的时间间隔。缺省设置为每天。如果将最大日志数设置为 1，目录就将忽略该属性。

定义日志文件删除策略

如果需要目录自动删除旧的归档日志，可以从 iPlanet Directory Server Console 中定义日志文件删除策略。

注意 只有在以前已定义日志文件循环策略的情况下，日志删除策略才有意义。如果仅有一个日志文件，日志文件删除策略将无效。

服务器将在日志循环时评估日志文件删除策略。

您可以配置以下参数：

- 组合归档日志的最大空间大小。达到最大空间值时，系统将会自动删除最早的归档日志。如果不想设置最大空间限制，则在该字段中键入 -1。缺省设置为 500 MB。日志文件数设置为 1 时，将忽略该参数。
- 最小可用磁盘空间。可用磁盘空间达到该最小值时，系统将会自动删除最早的归档日志。缺省设置为 5 MB。日志文件数设置为 1 时，将忽略该参数。
- 日志文件的最长存在周期。日志文件达到该最长存在周期时，将被自动删除。缺省设置为 1 个月。日志文件数设置为 1 时，将忽略该参数。

访问日志

访问日志中包括有关到目录的客户机连接的详细信息。

本部分包含下列步骤：

- 第 365 页上的 “查看访问日志”
- 第 365 页上的 “配置访问日志”

查看访问日志

要查看访问日志：

1. 在 iPlanet Directory Server Console 上，选择 “状态”选项卡，然后在导航树中展开日志文件夹并选择访问日志图标。

   此时出现一表格显示访问日志中最后 25 个条目的列表。

2. 要刷新当前显示，请单击 “刷新”。如果需要每十秒钟自动刷新一次显示，请选中 “持续”复选框。

3. 要查看某个归档的访问日志，请从 “选择日志”下拉菜单中选择该日志。

4. 要显示不同数目的消息，请在 “要显示的行”文本框中输入需要查看的数目，然后单击 “刷新”。

5. 可以显示包含指定字符串的消息。为此，请在 “所显示的行中须包含”文本框中输入字符串，然后单击 “刷新”。

配置访问日志

您可以配置一些设置以定制访问日志，其中包括目录用于存储访问日志及创建策略和删除策略的位置。

也可以禁用目录的访问记录功能。这样做的原因，可能是因为访问日志增长非常快（每访问目录 2,000 次，将使访问日志增加大约 1 MB）。但在关闭访问记录功能前，请慎重考虑，因为访问日志会提供有用的故障排除信息。

要配置目录的访问日志：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。然后，展开导航树中的日志文件夹并选择访问日志图标。

   访问日志配置属性将显示在右侧窗口中。

2. 要启用访问记录功能，请选中 “启用记录”复选框。

   如果不希望目录保留访问日志，请清除该复选框。

   缺省设置为启用访问记录。

3. 在 “日志文件”字段中，输入希望目录用于存储访问日志的完整路径和文件名。默认的文件是：

   `/var/ds5/slapd-`*`serverID`*`/logs/access`

4. 设置日志数、日志大小和存档周期的最大值。

   有关这些参数的信息，请参阅第 364 页上的 “定义日志文件循环策略”。

5. 设置组合存档日志的最大空间大小、可用磁盘空间的最小值及日志文件的最长存在周期。

   有关这些参数的信息，请参阅第 364 页上的 “定义日志文件删除策略”。

6. 完成更改后，单击 “保存”。

错误日志

错误日志中包含目录在正常操作过程中所遇事件和错误的详细信息。

本部分包含下列步骤：

- 第 367 页上的 “查看错误日志”
- 第 367 页上的 “配置错误日志”

查看错误日志

要查看错误日志：

1. 在 iPlanet Directory Server Console 上，选择 “状态”选项卡，然后在导航树中展开日志文件夹并选择错误日志图标。

   此时出现一表格显示错误日志中最后 25 个条目的列表。

2. 要刷新当前显示，请单击 “刷新”。如果需要每十秒钟自动刷新一次显示，请选中 “持续”复选框。

3. 要查看某个归档的错误日志，请从 “选择日志”下拉菜单中选择该日志。

4. 要指定不同数目的消息，请在 “要显示的行”文本框中输入所要查看的数目，然后单击 “刷新”。

5. 可以显示包含指定字符串的消息。为此，请在 “所显示的行中须包含”文本框中输入字符串，然后单击 “刷新”。

配置错误日志

您可以更改错误日志的若干设置，包括目录用于存储日志的位置及希望目录日志中包含的内容。

要配置错误日志：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。然后，展开导航树中的日志文件夹并选择错误日志图标。

   错误日志配置属性将显示在右侧窗口中。

2. 在右窗口中，选择 “错误日志”选项卡。

3. 要启用错误记录功能，请选中 “启用记录”复选框。

   如果不希望目录保留错误日志，请清除该复选框。

   缺省设置为启用错误记录。

4. 在 “日志文件”字段中，输入希望目录用于存储错误日志的完整路径和文件名。默认的文件是：

   `/var/ds5/slapd-`*`serverID`*`/logs/error`

5. 设置日志数、日志大小和存档周期的最大值。

   有关这些参数的信息，请参阅第 364 页上的 “定义日志文件循环策略”。

6. 设置组合存档日志的最大空间大小、可用磁盘空间的最小值及日志文件的最长存在周期。

   有关这些参数的信息，请参阅第 364 页上的 “定义日志文件删除策略”。

7. 如果要设置日志级别，请按住 Ctrl 键并单击希望目录在 “日志级别”列表框中包含的选项。

   有关日志级别选项的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*中的 “日志级别”。

   更改这些值的缺省值可能会导致错误日志快速增大。因此，除非 iPlanet 客户支持部门有要求，否则建议不要更改日志级别。

8. 完成更改后，单击 “保存”。

审计日志

审计日志中包含对每个数据库和服务器配置所做更改的详细信息。

本部分包含下列步骤：

- 第 368 页上的 “查看审计日志”
- 第 369 页上的 “配置审计日志”

查看审计日志

查看审计日志之前，必须启用目录的审计记录功能。有关信息，请参阅第 369 页上的 “配置审计日志”。

要查看审计日志：

1. 在 iPlanet Directory Server Console 上，选择 “状态”选项卡。然后，展开导航树中的日志文件夹并选择审计日志图标。

   此时出现一表格显示审计日志中最后 25 个条目的列表。

2. 要刷新当前显示，请单击 “刷新”。如果需要每十秒钟自动刷新一次显示，请选中 “持续”复选框。

3. 要查看某个归档的审计日志，请从 “选择日志”下拉菜单中选择该日志。

4. 要显示不同数目的消息，请在 “要显示的行”文本框中输入需要查看的数目并单击 “刷新”。

5. 可以显示包含指定字符串的消息。为此，请在 “所显示的行中须包含”文本框中输入字符串，然后单击 “刷新”。

配置审计日志

使用 iPlanet Directory Server Console 可以启用和禁用审计记录功能，同时可以指定审计日志的存储位置。

要配置审计记录功能：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。然后，展开导航树中的日志文件夹并选择审计日志图标。

   审计日志配置属性将显示在右侧窗口中。

2. 要启用审计记录，请选中 “启用记录”复选框。

   要禁用审计记录，请清除该复选框。缺省情况下将禁用审计记录功能。

3. 在 “日志文件”字段中，输入希望目录用于存储审计日志的完整路径和文件名。默认的文件是：

   `/var/ds5/slapd-`*`serverID`*`/logs/audit`

4. 设置日志数、日志大小和存档周期的最大值。

   有关这些参数的信息，请参阅第 364 页上的 “定义日志文件循环策略”。

5. 设置组合存档日志的最大空间大小、可用磁盘空间的最小值及日志文件的最长存在周期。

   有关这些参数的信息，请参阅第 364 页上的 “定义日志文件删除策略”。

6. 完成更改后，单击 “保存”。

手动日志文件循环

目录服务器支持所有三种日志的自动日志文件循环。但是，如果未设置自动日志文件创建或删除策略，则可手动循环日志文件。缺省情况下，可在以下目录位置找到访问日志、错误日志和审计日志文件：

`/var/ds5/slapd-`*`serverID`*`/logs`

要手动循环日志文件：

1. 关闭服务器。有关信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

2. 移动或重命名所要循环的日志文件，以备旧日志文件用于以后参考。

3. 重新启动服务器。有关信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

监控服务器活动

从 Directory 服务器控制台或命令行可以监控目录服务器的当前活动。也可监控所有数据库高速缓存的活动。本部分包含下列信息：

- 第 370 页上的 “从 iPlanet Directory Server Console 监控服务器”
- 第 374 页上的 “从命令行监控服务器”

从 iPlanet Directory Server Console 监控服务器

本部分包含有关使用 Directory Server Console 来监控服务器的信息，同时还包含性能监控中的可用信息。

查看服务器性能监控

要使用 Directory 服务器控制台来监控服务器的活动：

1. 在 iPlanet Directory Server Console 上，选择 “状态”选项卡。在导航树中，选择 “性能计数器”。

   右侧窗口中的 “状态”选项卡显示有关服务器活动的当前信息。如果服务器当前未运行，则该选项卡将不提供性能监控信息。

2. 单击 “刷新”可刷新当前显示内容。如果希望服务器连续更新所显示的信息，则选中 “持续”复选框。

服务器性能监控信息概述

服务器所提供的监控信息如下列各部分所述：

- 常规信息 （服务器）
- 资源概要
- 当前资源使用情况
- 连接状态
- 全局数据库缓存信息

常规信息 （服务器）

服务器可提供以下常规信息：

- 服务器版本

  标识当前的服务器版本。

- 配置 DN

  标识必须用作搜索基，从而利用 `ldapsearch` 命令行实用程序来获取这些结果的特异名称。该字段应为 `cn=monitor`。

- 数据版本

  提供服务器数据的标识信息。通常，仅在服务器将副本提供给客户服务器的情况下，此处显示的信息才相关。数据版本信息按如下所示提供：

  - 服务器主机名。
  - 服务器端口号。
  - 数据库生成号。可能废弃：仅当创建目录数据库且 LDIF 文件中不含计算机数据条目时，才会创建该唯一标识符。
  - 当前更改日志号。该编号对应于目录的上一次更改。它从 1 开始，且数据库每更改一次即增加 1。

- 服务器启动时间

  服务器启动的日期和时间。

- 服务器上的当前时间

  显示服务器上的当前日期和时间。

资源概要

由控制台显示的 “资源概要”表可提供以下特定资源的信息：

表 12-1 服务器性能监控 — 资源概要表

资源	自启动以来的使用情况	每分钟的平均值
连接	自服务器启动以来连接到服务器的连接总数。	自服务器启动以来每分钟的平均连接数。
操作已启动	自服务器启动以来所启动的操作总数。操作包括任何客户机对服务器的操作请求，例如搜索、添加和修改。通常情况下，每次连接会启动多个操作。	自服务器启动以来每分钟的平均操作数。

表 12-1 服务器性能监控 — 资源概要表 *（续）*

资源	自启动以来的使用情况	每分钟的平均值
操作已完成	自服务器启动以来服务器完成的操作总数。	自服务器启动以来每分钟的平均操作数。
发送到客户机的项目	自服务器启动以来发送到客户机的条目总数。条目作为搜索请求的结果被发送给客户机。	自服务器启动以来每分钟发送到客户机的平均条目数。
发送到客户机的字节数	自服务器启动以来发送到客户机的字节总数。	自服务器启动以来每分钟发送到客户机的平均字节数。

当前资源使用情况

Directory Server Console 中的 “资源概要”表提供以下特定资源的信息

表 12-2 服务器性能监控 — 当前资源使用情况表

资源	当前总计
活动线程	当前用于处理请求的活动线程数。其它线程可由内部服务器任务 （例如复制或链接）来创建。
打开连接	打开的连接总数。每个连接可使用多个操作，从而会占用多个线程。
剩余的可用连接	服务器可并行打开的剩余连接总数。该数目基于当前打开的连接数和服务器所允许打开的并行连接总数。大多数情况下，后面一个值由操作系统确定，表示为可用于任务的文件描述符数。
线程等待将信息写入客户机	等待写入客户机的线程总数。将数据发送到客户机的过程中，如果必须暂停服务器，则可能无法立即写入线程。暂停的原因包括网络速度慢、客户机速度慢或发送给客户机的信息量过大。
线程等待从客户机读取信息	等待从客户机中读取信息的线程总数。如果服务器开始从客户机中接收请求，之后在请求的传输过程中因某种原因而中止，则可能无法立即读取线程。一般而言，线程等待读取是网络或客户机速度慢的征兆。
并行线程	指示线程并行性的级别。
数据库正在使用	正在接受服务器服务的数据库总数。

连接状态

Directory Server console 中的 “连接状态”表可提供下列有关当前打开连接所用资源量的信息：

表 12-3 服务器性能监控 — 连接状态表

表格标题	说明
打开的时间	指示最初打开连接时的服务器时间。
已启动	指示该连接所启动的操作数。
已完成	指示服务器对该连接所完成的操作数。
绑定为	指示客户机绑定到服务器时所用的特异名称。如果客户机未通过到服务器的身份验证，则服务器将在该字段中显示未绑定。
读取 / 写入	指示当前是否阻塞服务器对客户机的读写访问。可能的值包括： • 未阻塞。指示服务器空闲，正在顺利地将数据发送到客户机中，或者正在顺利地从客户机中读取数据。 • 阻塞。指示服务器尝试将数据发送到客户机，或尝试从客户机中读取数据，但无法完成操作。可能的原因是：网络或客户机速度慢。

全局数据库缓存信息

Directory Server Console 中的 “全局数据库缓存信息”表包含以下信息：

表 12-4 服务器性能监控 — 全局数据库缓存表

表格标题	说明
命中次数	指示服务器可通过从高速缓存 （而非转到磁盘）获取数据而处理搜索请求的次数。
尝试次数	自服务器启动以来对目录所执行的请求总数。
命中率	缓存尝试次数与成功的缓存命中次数之比。该数越接近 100% 越好。
读入页	指示从磁盘读入缓存中的页数。
写出页	代表从高速缓存写出到磁盘的页数。
排除只读页	指示从缓存中丢弃的只读页数 （旨在为新页腾出空间）。从缓存中丢弃的页必须写到磁盘中，因此可能会影响服务器的性能。排除的页数越少越好。
读写页排除	指示从缓存中丢弃的读写页数 （旨在为新页腾出空间）。该值与 “写出页”的不同之处在于：这些页是未经修改的被丢弃读写页。 从缓存中丢弃的页必须写到磁盘中，因此可能会影响服务器的性能。排除的页数越少越好。

从命令行监控服务器

执行具有以下特性的搜索操作，即可通过任何 LDAP 客户机来监控目录服务器的当前活动：

- 搜索属性 `objectClass=*`
- 搜索基：`cn=monitor`
- 搜索范围：`base`

例如：

```
ldapsearch -h directory.siroe.com -s base -b "cn=monitor"
"(objectclass=*)"
```

有关搜索 iPlanet Directory Server 的信息，请参阅第 458 页上的 “使用 ldapsearch”。

服务器的监控属性可在 `cn=monitor,cn=config` 条目中找到。

使用 `ldapsearch` 监控服务器的活动时，可查看到以下信息：

- `version`：标识目录的当前版本号。
- `threads`：用于处理请求的当前活动线程数。其它线程可由内部服务器任务（例如复制或链接）来创建。
- `connection`:*fd*:*opentime*:*opsinitiated*:*opscompleted*:*binddn*:`[rw]`：提供每个打开连接的以下概要信息（仅当您作为目录管理员而绑定到目录时可用）：
  - `fd` — 该连接使用的文件描述符。
  - `opentime` — 打开该连接的时间。
  - `opsinitiated` — 该连接所启动的操作数。
  - `opscompleted` — 已完成的操作数。
  - `binddn` — 该连接用于连接到目录的特异名称。
  - `rw` — 连接被阻塞读写操作时所显示的字段。

  默认情况下，仅当您作为目录管理员而绑定到目录时，这些信息才可用。但可以更改与这些信息相关的 ACI，从而允许其它用户访问这些信息。
- `currentconnections` — 标识正处于服务状态的目录连接数。
- `totalconnections` — 标识自目录启动以来所处理的连接数。

- `dtablesize` — 显示该目录可用的文件描述符数。每个连接需要一个文件描述符：一个用于每个打开的索引、一个用于日志文件管理、一个用于 `ns-slapd` 本身。实际上，该值可告知目录还能为多少个并行连接提供服务。有关文件描述符的详细信息，请参阅操作系统文档。
- `readwaiters` — 标识等待从客户机中读取数据的线程数。
- `opsinitiated` — 标识自服务器启动以来已启动的操作数。
- `opscompleted` — 标识自服务器启动以来已完成的操作数。
- `entriessent` — 标识自服务器启动以来发送到客户机的条目数。
- `bytessent` — 标识自服务器启动以来发送到客户机的字节数。
- `currentime` — 标识服务器拍摄该快照的时间。该时间按 “格林尼治时间” (GMT) UTC 格式显示。
- `starttime` — 标识服务器启动的时间。该时间按 “格林尼治时间” (GMT) UTC 格式显示。
- `nbackends` — 标识服务器为其提供服务的后端 （数据库）数。
- `concurrency` — 指示当前的线程并行性级别。
- `backendmonitordn` — 标识每个目录数据库的 DN。

监控数据库活动

可从 Directory 服务器控制台或命令行来监控数据库的当前活动。本部分包含下列信息：

- 第 375 页上的 “从服务器控制台监控数据库活动”
- 第 379 页上的 “从命令行监控数据库”

从服务器控制台监控数据库活动

本部分介绍使用 Directory Server Console 查看数据库性能监控结果的方法及性能监控所提供的信息类型。

查看数据库性能监控结果

要监控数据库的活动：

1. 在 iPlanet Directory Server Console 上，选择 “状态”选项卡。在导航树中，展开 “性能计数器”文件夹，然后选择要监控的数据库。

   该选项卡显示有关数据库活动的当前信息。如果服务器当前未运行，则该选项卡将不提供性能监控信息。

2. 单击 “刷新”可刷新当前显示的信息。如果希望目录连续更新所显示的信息，则选中 “持续”复选框，然后单击 “刷新”。

数据库性能监控信息概述

目录可提供数据库监控信息，如下列各部分所述：

- 常规信息 （数据库）
- 概要信息表
- 数据库缓存信息表
- 面向特定数据库文件的表

常规信息 （数据库）

目录可提供下列常规数据库信息：

- 数据库

  标识所要监控的数据库类型。

- 配置 DN

  标识必须用作搜索基，从而利用 `ldapsearch` 命令行实用程序来获取这些结果的特异名称。

概要信息表

“概要信息”表提供以下信息：

表 12-5 数据库性能监控 — 概要信息

性能度量	当前总计
只读状态	指示数据库当前是否处于只读模式下。当 `readonly` 属性设置为 `on` 时，数据库将处于只读模式下。
条目缓存命中次数	指示条目高速缓存中成功查询的总数。即：服务器可通过从高速缓存 （而非转到磁盘）获取数据而处理搜索请求的次数。
条目缓存尝试次数	指示自目录上次启动以来条目高速缓存的查询总次数。即：自服务器启动以来，在服务器上执行的搜索操作总次数。
条目缓存命中率	指示条目缓存尝试次数与条目缓存查询成功次数的比率。该数字基于自目录上次启动以来的总查询数和总命中数。该值越接近 100% 越好。搜索操作尝试查找条目缓存中不存在的项目时，目录必须执行磁盘访问以获取该条目。因此，当该比率趋向零时，磁盘访问次数将增加，而目录搜索性能将下降。 要提高该比率，可加大缓存中的最大条目数属性的值，从而增加目录保留在条目高速缓存中的条目数。有关使用服务器控制台更改该值的信息，请参阅第 392 页上的 “调整数据库性能”。
当前条目缓存大小（字节）	指示当前条目缓存中的目录条目总计大小。
最大条目缓存大小（字节）	指示目录保留的条目缓存大小。该值通过最大缓存大小属性来管理。有关使用服务器控制台更改该值的信息，请参阅第 392 页上的 “调整数据库性能”。
当前条目缓存大小（条目数）	指示当前条目缓存中的目录条目总数。
最大条目缓存大小（条目数）	指示可保留在条目缓存中的目录条目数的最大值。该值通过缓存中的最大条目数属性来管理。有关使用服务器控制台更改该值的信息，请参阅第 392 页上的 “调整数据库性能”。

数据库缓存信息表

“数据库缓存信息”表可提供以下缓存信息：

表 12-6 数据库性能监控 — 数据库缓存信息

性能度量	当前总计
命中次数	指示数据库缓存成功提供请求页的次数。一页是大小为 2K 的缓冲区。
尝试次数	指示向数据库缓存请求某一页面的次数。
命中率	指示数据库缓存命中次数与数据库缓存尝试次数的比率。该值越接近 100% 越好。目录操作尝试查找数据库高速缓存中不存在的数据库部分时，目录必须执行磁盘访问以获取相应的数据库页。因此，当该比率趋于零时，磁盘访问次数将增加，而目录性能将下降。 要提高该比率，可提高最大缓存大小属性的值，从而增加目录保留在数据库缓存中的数据量。有关使用服务器控制台更改该值的信息，请参阅第 392 页上的 “调整数据库性能”。
读入页	指示从磁盘读入数据库缓存的页数。
写出页	代表从高速缓存写出到磁盘的页数。修改读写页面后，数据库页面被写入磁盘中，随后从缓存中被删除。当缓存已满且目录操作需要当前未存储在缓存中的数据库页面时，页面将从数据库缓存中被删除。
排除只读页	指示从缓存中丢弃的只读页数 （旨在为新页腾出空间）。
读写页排除	指示从缓存中丢弃的读写页数 （旨在为新页腾出空间）。该值与 “写出页”的不同之处在于：这些页是未经修改的被丢弃读写页。

面向特定数据库文件的表

目录将为构成数据库的每个索引文件显示一个表。每个表都提供以下信息：

表 12-7 数据库性能监控 — 面向特定数据库文件的表

性能度量	当前总计
高速缓存命中次数	搜索结果命中该特定文件之缓存的次数。即：客户机执行搜索时需要该文件的数据，且目录从缓存中获得了所需的数据。
高速缓存未命中次数	搜索结果未命中该特定文件之缓存的次数。即：执行搜索时需要该文件的数据，但在缓存中找不到所需的数据。
读入页	指示从该文件进入高速缓存的页数。
写出页	指示该文件中从高速缓存写出到该文件的页数。

从命令行监控数据库

执行具有以下特性的搜索操作，即可通过任何 LDAP 客户机来监控目录的数据库活动：

- 搜索属性 `objectClass=*`
- 搜索基：`cn=monitor`、`cn=database_instance`、`cn=ldbm database`、`cn=plugins` 和 `cn=config`，其中 database 是所要监控的数据库的名称
- 搜索范围：`base`

例如：

```
ldapsearch -h directory.siroe.com -s base -b
"cn=monitor,cn=Siroe,cn=ldbm database,cn=plugins, cn=config"
"objectclass=*"
```

本例中， ldapsearch 操作将查找 Siroe 数据库。有关搜索目录的信息，请参阅第 458 页上的 “使用 ldapsearch”。

监控服务器的活动时，可查看以下信息：

- `database` — 标识当前监控的数据库的类型。
- `readonly` — 标识数据库是否处于只读模式。 `0` 表示服务器未处于只读模式，而 `1` 则表示服务器处于只读模式。
- `entrycachehits` — 提供的信息与第 377 页的表 12-5 中 “条目缓存命中次数”所描述的信息相同。
- `entrycachetries` — 提供的信息与第 377 页的表 12-5 中 “条目缓存尝试次数”所描述的信息相同。
- `entrycachehitratio` — 提供的信息与第 377 页的表 12-5 中 “条目缓存命中率”所描述的信息相同。
- `currententrycachesize` — 提供的信息与第 377 页的表 12-5 中 “当前条目缓存大小 （条目数）”所描述的信息相同。
- `maxentrycachesize` — 提供的信息与第 377 页的表 12-5 中 “最大条目缓存大小 （条目数）”所描述的信息相同。
- `dbchehits` — 提供的信息与第 378 页的表 12-6 中 “命中次数”所描述的信息相同。
- `dbcachetries` — 提供的信息与第 378 页的表 12-6 中 “尝试次数”所描述的信息相同。
- `dbcachehitratio` — 提供的信息与第 378 页的表 12-6 中 “命中率”所描述的信息相同。

- dbcachepagein — 提供的信息与第 378 页的表 12-6 中 “读入页”所描述的信息相同。
- dbcachepageout — 提供的信息与第 378 页的表 12-6 中 “写出页”所描述的信息相同。
- dbcacheroevict — 提供的信息与第 378 页的表 12-6 中 “排除只读页”所描述的信息相同。
- dbcacherwevict — 提供的信息与第 378 页的表 12-6 中 “读写页排除”所描述的信息相同。

随后显示构成数据库的各个文件的以下信息：

- dbfilename-*number* — 指示文件名。 *number* 提供该文件的连续整数标识符（从 0 开始）。该文件的所有相关统计信息都使用相同的数字标识符。
- dbfilecachehit-*number* — 提供的信息与第 378 页的表 12-7 中 “高速缓存命中次数”所描述的信息相同。
- dbfilecachemiss-*number* — 提供的信息与第 378 页的表 12-7 中 “高速缓存未命中次数”所描述的信息相同。
- dbfilepagein-*number* — 提供的信息与第 378 页的表 12-7 中 “读入页”所描述的信息相同。
- dbfilepageout-*number* — 提供的信息与第 378 页的表 12-7 中 “写出页”所描述的信息相同。

监控数据库链接活动

可以从命令行上使用监控属性来监控数据库链接的活动。使用 ldapsearch 命令行实用程序可以返回自己感兴趣的属性值。监控属性存储在以下条目中：
cn=monitor,cn=database_link_name,cn=chaining database, cn=plugins, cn=config。

例如，可使用 ldapsearch 命令行实用程序来检索由名为 DBLink1 的特定数据库链接所接收的添加操作数。按如下所示运行 ldapsearch：

```
ldapsearch -h server.siroe.com -p 389 \
-D "cn=Directory Manager" -w password -s sub -b \
"cn=monitor,cn=DBLink1,cn=chaining database,cn=plugins,cn=config" \
"(objectclass=*)" nsAddCount
```

注意 以上命令应在一行内键入。此处，由于页面大小的限制，该命令未显示为一行。

可以搜索以下数据库链接监控属性：

表 12-8 数据库链接监控属性

属性名	说明
nsAddCount	已接收的添加操作数。
nsDeleteCount	已接收的删除操作数。
nsModifyCount	已接收的修改操作数。
nsRenameCount	已接收的重命名操作数。
nsSearchBaseCount	已接收的基本级别搜索数。
nsSearchOneLevelCount	已接收的一级搜索数。
nsSearchSubtreeCount	已接收的子树搜索数。
nsAbandonCount	已接收的放弃操作数。
nsBindCount	已接收的绑定请求数。
nsUnbindCount	已接收的解除绑定请求数。
nsCompareCount	已接收的比较操作数。
nsOperationConnectionCount	LDAP 操作的打开连接数。
nsBindConnectionCount	绑定操作的打开连接数。

第 13 章

使用 SNMP 监控 iPlanet Directory Server

在第 12 章 “监控服务器和数据库活动”中所述的服务器和数据库活动监视日志设置是 iPlanet Directory Server 所特有的。也可以使用简单网络管理协议 (Simple Network Management Protocol) (SNMP) 来监控 iPlanet Directory Server。 SNMP 是一个用于实时监控多种设备的网络活动的管理协议。

SNMP 是全局网络控制和监控的理想标准机制。它允许网络管理员将所有网络监控行为都统一到一起，而 iPlanet Directory Server 则只监控其中的一部分。

本章包括以下主题：

- 关于 SNMP
- iPlanet Directory Server 管理信息库概述
- 设置 SNMP
- 为 iPlanet Directory Server 配置 SNMP

关于 SNMP

SNMP 是用于交换网络活动数据的协议。通过使用 SNMP，数据可在受管理设备和网络管理工作站 (network management station) (NMS) 之间进行传输，这样用户即可远程管理网络。受管理的设备可以是运行 SNMP 的所有设备，如主机、路由器及 iPlanet Directory Server。 NMS 通常是装有一个或多个网络管理应用程序的强大工作站。网络管理应用程序以图形方式显示有关受管理设备的信息 （设备的开关状态，接收的错误信息类型及数量等）。

NMS 和受管理设备之间的信息传输是通过两种类型的代理完成的：子代理和主代理 (master agent)。子代理收集关于受管理设备的信息，并把这些信息传送给主代理。

iPlanet Directory Server 有一个子代理。主代理负责在各种子代理和 NMS 之间交换信息。主代理与所要通话的子代理在相同的主机上运行。

一台主机上可以同时安装多个子代理。例如，如果某个主机上同时安装了 iPlanet Directory Server、 Enterprise Server 和 Messaging Server，则这些服务器中每个服务器的子代理都将与同一主代理通讯。在 UNIX 环境下，主代理随 iPlanet Administration Server 一起安装。

可查询到的 SNMP 属性值 （否则称做变量）保存于受管理设备中，并在必要时向 NMS 报告。每个变量都称为一个受管理的对象 (managed object)，也就是代理所能访问并发送给 NMS 的任何对象。所有受管理对象均在管理信息库 (management information base) (MIB) 中定义，后者是一个树形分层结构的数据库。分层结构的最上层包含有关网络的最常规信息。下面的每个分支则更加具体，分别处理独立的网络区域。

SNMP 概述

SNMP 以*协议数据单元 (protocol data unit)* (PDU) 的形式交换网络信息。 PDU 包含有关储存在受管理设备中的变量的信息。这些变量也称作受管理对象，其值和标题在必要时会报告给 NMS。 NMS 和受管理设备之间的通讯是通过以下两种途径之一实现的：

- 启动 NMS 的通讯
- 启动受管理设备的通讯

启动 NMS 的通讯

启动 NMS 的通讯是 NMS 和受控设备之间通讯的最常见形式。在这种通讯形式中，NMS 或者向受管理设备请求信息，或者更改储存于受管理设备上的变量值。

下列步骤构成了启动 NMS 通讯的 SNMP 对话：

1. NMS 决定哪些受管理设备和对象需要进行监测。
2. NMS 通过主代理向受管理设备的子代理发送协议数据单元。该协议数据单元或者向受管理设备请求信息，或者通知子代理更改储存在受管理设备上的变量的值。
3. 受管理设备的子代理从主代理那接收协议数据单元。
4. 如果来自 NMS 的协议数据单元请求变量信息，则子代理将把信息发给主代理，然后主代理以另一个协议数据单元的形式将其发回 NMS。 NMS 随即以文字或图形的方式显示相应的信息。

   如果来自 NMS 的协议数据单元请求子代理设置变量值，子代理就会进行设置。

启动受管理设备的通讯

这种通讯出现在当有事件发生时，受管理设备需要通知 NMS 的情况下。受管理设备启动与 NMS 的通讯，从而告知 NMS 有关的关闭或开启操作。由受管理设备所启动的通讯也称为*陷阱*。 iPlanet Directory Server 在 iPlanet Directory Server 启动或停止时都会向 NMS 发送陷阱。

下面是构成启动受管理设备的 SNMP 对话的步骤：

1. 假设受管理设备上发生某一事件。
2. 子代理就会将该事件告知主代理。
3. 主代理向 NMS 发送 PDU，从而将事件告知 NMS。
4. NMS 以文字或图形方式显示相应的信息。

iPlanet Directory Server 管理信息库概述

每个 iPlanet 服务器都有自己的 MIB。 iPlanet Directory Server 的 MIB 在以下文件中定义：

```
/usr/iplanet/ds5/plugins/snmp/netscape-ldap.mib
```

该 MIB 中包含与目录的网络管理有关的变量定义。这些变量也称为受管理对象。利用目录 MIB 和网络管理软件 （例如 Sun Net Manager），即可象监控网络上其它受管理设备那样对目录进行监控。

目录 MIB 具有下列对象标识符：

```
iso.org.dod.internet.private.enterprises.netscape.nsldap
(nsldapd OBJECT IDENTIFIER ::= { 1.3.6.1.4.1.1450.7 })
```

利用目录 MIB，您可以看到有关目录的管理信息，并可进行实时服务器监控。目录 MIB 被分为三个不同的受管理对象表：

- 操作表
- 条目表
- 交互作用表

在使用目录 MIB 之前，必须将它与在下面目录中找到的 MIB 一起进行编译：

```
/usr/iplanet/ds5/plugins/snmp/mibs
```

有关如何编译 MIB 的信息，请参阅 SNMP 产品文档。下列部分将对每个表做详细说明。

关于操作表

操作表提供有关 iPlanet Directory Server 访问、操作和错误的统计信息。下表介绍了 `netscape-ldap.mib` 文件操作表中所存储的受管理对象。

表 13-1 操作表 — 受管理的对象和说明

受管理的对象	说明
`dsAnonymousBinds`	自服务器启动后到目录的匿名绑定数。
`dsUnauthBinds`	自服务器启动后到目录的未经验证的绑定数。
`dsSimpleAuthBinds`	自服务器启动后，利用简单验证方法 （例如口令保护）建立的绑定数。
`dsStrongAuthBinds`	自服务器启动后，利用严格验证方法 （例如 SSL 或诸如 Kerberos 等 SASL 机制）建立的绑定数。
`dsBindSecurityErrors`	自服务器启动后，因验证失败或凭证无效而被服务器拒绝的绑定请求数。
`dsInOps`	自服务器启动后，从其它目录转移到本目录中的操作数。
`dsReadOps`	自应用程序启动后，本目录执行的读取操作数。因为 LDAP 通过搜索操作间接执行读取操作，所以该对象的值始终是 0。
`dsCompareOps`	自服务器启动后，本目录执行的比较操作数。
`dsAddEntryOps`	自服务器启动后，本目录执行的添加操作数。
`dsRemoveEntryOps`	自服务器启动后，本目录执行的删除操作数。
`dsModifyEntryOps`	自服务器启动后，本目录执行的修改操作数。
`dsModifyRDNOps`	自服务器启动后，本目录执行的 “修改 RND”操作数。
`dsListOps`	自服务器启动后，本目录执行的列表操作数。因为 LDAP 通过搜索操作间接执行列表操作，所以这个对象的值始终是 0。
`dsSearchOps`	自服务器启动后，本目录执行的搜索操作数。
`dsOneLevelSearchOps`	自服务器启动后，本目录执行的一级搜索操作数。
`dsWholeSubtreeSearchOps`	自服务器启动后，本目录执行的对整个子树进行搜索的操作数。
`dsReferrals`	自服务器启动后，因响应客户机请求而被本目录退回的引荐数。
`dsSecurityErrors`	不符合安全要求而转移到本目录中的操作数。
`dsErrors`	由于错误 （非安全和引荐错误）而不能实现的请求数。这些错误包括名称错误、更新错误、属性错误和服务错误。部分执行的请求将不计为错误。

条目表

条目表提供有关目录项的内容信息。表 13-2 介绍了 netscape-ldap.mib 文件条目表中所储存的受管理对象信息。

表 13-2 条目表 — 受管理的对象和说明

受管理的对象	说明
dsMasterEntries	该目录中包含其原版条目的目录项数量。该对象的值始终是 0 （因为当前没有执行更新）。
dsCopyEntries	该目录中包含其从属副本的目录项数量。该对象的值始终是 0 （因为当前没有执行更新）。
dsCacheEntries	缓存于目录中的条目数。
dsCacheHits	从应用程序启动后，从本地缓存执行的操作数。
dsSlaveHits	从本地复本 （影子项）执行的操作数。该对象的值始终是 0。

设置 SNMP

要为 iPlanet Directory Server 设置 SNMP 支持：

1. 利用 Administration Server Console 配置并启动主代理。

注意 如果想使用默认的端口设置 （对于 SNMP 为 161，对于 SMUX 则为 199），则必须成为 root 用户。如果重新配置主代理并使端口的值超过 1000，则不必成为 root 用户。

有关设置主代理的信息，请参阅*通过 iPlanet Console 管理服务器*。

2. 启用目录子代理。

有关信息，请参阅第 388 页上的 “为 iPlanet Directory Server 配置 SNMP”。

3. 启动目录子代理。

有关信息，请参阅第 388 页上的 “启动和停止 SNMP 子代理”。

启动和停止 SNMP 子代理

要启动、停止和重新启动 SNMP 子代理 (SNMP subagent)：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树中最上面的条目。
2. 在右侧窗口中，选择 SNMP 选项卡。
3. 单击 “启动”可启动子代理 (subagent) ；单击 “停止”可停止子代理，而单击 “重新启动”则重新启动子代理。

   停止目录不会停止目录子代理。如果确实要停止子代理，则必须利用该选项卡来完成。

注意 如果想添加其它服务器实例并使其成为 SNMP 网络的一部分，则必须重新启动子代理。

为 iPlanet Directory Server 配置 SNMP

要从 iPlanet Directory Server Console 配置 SNMP 的设置：

1. 确保 iPlanet Directory Server 处于运行状态。
2. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树最上端的条目。
3. 在右侧窗口中，选择 SNMP 选项卡。
4. 选中 “启用统计数据集合”复选框以启用 iPlanet Directory Server 统计数据集合。清除复选框则会禁用统计数据集合。
5. 在 “主控主机”和 “主端口”文本框中分别输入主代理所在的主机名和用于与主代理进行通讯的端口号。

注意 主机名和端口数为必需项。

   默认值分别是 `localhost` 和 `199`。
6. 在 “说明”文本框中输入唯一描述目录实例的说明。
7. 在 “组织”文本框中输入目录所属的公司或组织名。

8. 在 “地址”文本框中输入目录所在公司或组织的地址。
9. 在 “联系”文本框中输入负责维护目录者的电子邮件地址。
10. 单击 “保存”。
11. 重新启动子代理。

    请参阅第 388 页上的 “为 iPlanet Directory Server 配置 SNMP”。

第 14 章

调整 Directory Server 的性能

本章介绍 iPlanet Directory Server 随带的、用于优化性能的工具。同时，文中还提供若干用于改进目录性能的技巧。

本章包含以下几部分：

- 调整服务器性能
- 调整数据库性能
- 其它调整提示

调整服务器性能

通过限制服务器用于处理客户机搜索请求所需的资源量，可以对服务器的性能加以管理。您可以定义：

- 在响应搜索操作时，服务器返回给客户机的最大条目数 （大小限制属性）
- 执行搜索请求时，希望服务器所用的最大实时时间 （时间限制属性，以秒计）
- 服务器保持空闲连接的时长 （空闲超时属性，以秒计）
- 目录服务器可用的最大文件描述符数 （文件描述符的最大数属性）

要对 iPlanet Directory Server 进行配置以优化性能：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后选择左侧窗口导航树最上端的条目。

   右侧窗口中所显示的选项卡控制面向整个服务器的配置属性。

2. 选择右侧窗口中的 “性能”选项卡。

   此时显示当前服务器的性能设置。

3. 在 “大小限制”文本框中输入新的值，从而设置在响应搜索操作时，服务器返回给客户机的最大条目数。

   如果不想设置限定值，请在该文本框中键入 `-1`。

4. 在 “时间限制”文本框中输入执行搜索请求时，希望服务器所用的实时时间。

   如果不想设置限定值，请在该文本框中键入 `-1`。

5. 在 “空闲超时”文本框中，输入您希望服务器保持空闲连接的时长。

   如果不想设置限定值，请在该文本框中键入零 (`0`)。

6. 在 “文件描述符的最大数”文本框中，设置目录服务器可用的最大文件描述符数。

要想更好地理解这些参数对服务器搜索性能所带来的影响，请参阅第 321 页上的 “关于索引”。

调整数据库性能

本节分以下几个部分说明调整数据库性能的方法：

优化搜索性能

通过调整数据库的设置，可以改进服务器的搜索性能。影响性能的数据库属性主要限定了服务器的可用内存量。

为提高搜索操作的缓存命中率，可以增加目录服务器在数据库缓存中保留的数据量。为此，可以增加缓存中所存储的条目数，或者增加缓存的大小。为这些属性所能设置的最大值与计算机中实际的内存量有关。大致而言，计算机上的可用内存量应始终大于：

(缓存中的最大条目数 + 最大缓存大小) × 平均条目大小

更改这两个属性时，请务必小心。能否用这些属性改进服务器的性能与数据库大小、计算机上实际可用内存量及是否进行随机搜索 （即目录客户机是否搜索随机且广泛分布的目录数据）有关。

如果数据库无法适应内存且搜索过程为随机搜索，即使试图增大这些属性的设置值，也不会改进目录的性能。实际上，更改这些属性可能会有损于整体性能。

您可以调整下列属性：

- 用于管理其它所有数据库实例的数据库属性。在 iPlanet Directory Server Console 中只能查看包含目录数据的数据库及 `NetscapeRoot` 数据库。但服务器使用另一个数据库进行管理。在该数据库上，可以更改下列属性以改进性能：
  - 可用于所有数据库的内存量 （最大缓存大小属性）
  - 响应搜索请求时希望服务器核查的最大条目数 （审核限制属性）
- 用于存储目录数据 （包括 `NetscapeRoot` 数据库中的服务器配置数据）的各个数据库的属性。在这些数据库上，可以更改下列属性以改进性能：
  - 希望服务器在内存中保留的最大条目数 （缓存中的最大条目数属性）
  - 可用于缓存条目的内存量 （缓存的可用内存属性）

要配置应用于其它所有数据库实例的默认数据库属性：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后展开导航树中的 “数据图标”，同时突出显示 “数据库设置”节点。

   此时将在右侧窗口中显示 “数据库”选项卡。

2. 选择右侧窗口中的 “LDBM 插件设置”选项卡。

   该选项卡中包含用于该服务器上所存所有数据库的属性。

3. 在 “高速缓存最大空间”字段中，输入对应于所有数据库可用内存量的值。
4. 在 “审核限制”字段中，输入希望服务器在响应搜索请求时所检查的最大条目数。

   如果不想设置限定值，请在该文本框中键入 `-1`。如果是作为目录管理器绑定到目录上，则默认情况下将不对审核限制予以约束，并将忽略此处指定的任何设置。

要配置存储目录数据的各个数据库的属性：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后展开导航树中的 “数据图标”。展开要调整的数据库后缀并突出显示该数据库。

   右侧窗口中所显示的选项卡控制该数据库的参数设置。
2. 选择右侧窗口中的 “数据库设置”选项卡。
3. 在 “高速缓存中的最大项目数”字段中，输入希望服务器在内存中保留的条目数。
4. 在 “缓存的可用内存”字段中，输入可用于缓存条目的内存量。

   如果是利用 LDIF 创建非常大的数据库，请将该属性设置得尽可能大，而这将由计算机上的可用内存决定。该参数值越大，创建数据库的速度也就越快。

   创建完数据库后，在某个产品环境中运行服务器前，请务必将该参数重新设置为较小的值。

调整事务记录

每个 iPlanet Directory Server 中均包含事务日志，用于记录它所管理的所有数据库的操作。执行目录数据库操作 （例如写入）时，服务器就会将该操作记录到事务日志中。为确保最佳性能，目录不会立即执行操作。相反，操作将被存储到目录服务器的临时内存缓存中，直到完成该操作。

如果服务器出现故障 （例如电能耗尽及异常关闭），则缓存中存储的有关近期目录更改的信息将会丢失。但重新启动服务器时，目录将自动检测错误条件，然后利用数据库事务日志来恢复数据库。

尽管数据库事务日志和数据库恢复均为自动执行的过程，不需要任何干预，但用户可以调整数据库事务日志的某些属性，从而优化其性能。

警告 事务日志属性仅用于系统修改和诊断。这些设置应在 iPlanet 专业服务或 iPlanet 技术支持的指导下进行更改。

如果这些属性及其它配置属性的设置不一致，就可能导致目录不稳定。

更改数据库事务日志的位置

默认情况下，数据库事务日志文件及数据库文件自身都存储在以下目录中：

```
/var/ds5/slapd-serverID/db
```

由于事务日志的用途在于协助恢复被异常关闭的目录数据库，因此建议将数据库事务日志存储到不同于包含目录数据库的磁盘上。在独立的物理磁盘上存储数据库事务日志还可能会改进目录的性能。

要更改数据库事务日志文件的位置，请执行以下过程：

1. 停止 iPlanet Directory Server。

   有关说明，请参阅第 35 页上的 “从控制台启动 / 停止服务器”。

2. 使用 `ldapmodify` 命令行实用程序，将 `nsslapd-db-logdirectory` 属性添加到 `cn=config,cn=ldbm database,cn=plugins,cn=config` 条目中。在该属性中提供到日志目录的完整路径。

   有关 `nsslapd-db-logdirectory` 属性语法的信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。有关使用 `ldapmodify` 的说明，请参阅第 53 页上的 “使用 ldapmodify 添加和修改条目”。

3. 重新启动 iPlanet Directory Server。

更改数据库检查点间隔

目录服务器会定期将事务日志中所记录的操作写入磁盘中，并在数据库事务日志中记录检查点条目。通过指示写入目录的更改内容，检查点条目可以在事务日志中指示开始恢复的位置，从而加速恢复过程。

默认情况下，目录服务器被设置为每 60 秒发送检查点条目给数据库事务日志。增加检查点间隔可能会提高目录写入操作的性能。但增加检查点间隔也同时可能会增加在异常关闭的情况下进行目录数据库恢复所需的时间。同时，由于数据库事务日志文件过大，还会需要更多的磁盘空间。因此，只有在熟悉数据库的优化性并可充分评估更改效果的情况下，才应修改该属性。

要在服务器运行时修改检查点间隔，请执行以下过程：

1. 使用 `ldapmodify` 命令行实用程序将 `nsslapd-db-checkpoint-interval` 属性添加到 `cn=config,cn=ldbm database,cn=plugins,cn=config` 条目中。

   有关 `nsslapd-db-checkpoint-interval` 属性之语法的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。有关使用 `ldapmodify` 的说明，请参阅第 53 页上的 “使用 ldapmodify 添加和修改条目”。

禁用持久性事务

持久性事务记录的含义是：临时数据库事务日志实际上将被真实地写入磁盘中。

禁用持久性事务记录后，每个目录数据库操作都将写入数据库事务日志文件中，但可能不会立即写入磁盘中。如果目录更改已写入逻辑数据库事务日志文件中，但在系统崩溃时并未真正写入磁盘中，就将无法恢复所做的更改。禁用持久性事务记录后，恢复的数据库将保持一致，但无法反映系统崩溃前刚刚完成的任何 LDAP 写入操作的结果。

默认情况下将启用持久性数据库事务记录功能。要禁用持久性事务记录功能，请执行以下操作过程：

1. 停止 iPlanet Directory Server。

   有关说明，请参阅第 35 页上的 “从命令行启动 / 停止服务器”。

2. 使用 `ldapmodify` 命令行实用程序将 `nsslapd-db-durable-transactions` 属性添加到 `cn=config,cn=ldbm database,cn=plugins,cn=config` 条目中，然后将该属性的值设置为 `off`。

   有关 `nsslapd-db-durable-transactions` 属性之语法的信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。有关使用 `ldapmodify` 的说明，请参阅第 53 页上的 “使用 ldapmodify 添加和修改条目”。

3. 重新启动 iPlanet Directory Server。

指定事务批处理

在不需要随时进行事务处理时，为提高更新性能，可以使用 `nsslapd-db-transaction-batch-val` 属性来指定一次性提交多少个事务给事务日志。设置该属性为大于 0 的值将导致服务器推迟提交事务，直至排队的事务数等于该属性值。要使事务批处理有效， `nsslapd-db-durable-transaction` 属性必须设置为 `on`。

要在服务器运行时指定或修改事务批处理，请使用 `ldapmodify` 命令行实用程序将 `nsslapd-db-transaction-batch-val` 属性添加到 `cn=config,cn=ldbm database,cn=plugins,cn=config` 条目中。

有关 `nsslapd-db-transaction-batch-val` 属性的语法和值的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。有关使用 `ldapmodify` 的说明，请参阅第 53 页上的 “使用 ldapmodify 添加和修改条目”。

其它调整提示

本节提供一些与性能有关的提示和应了解的概念。

在 cn=config 下创建条目

cn=config 条目存储在简单的、平面化的 dse.ldif 配置文件中，而不象一般条目那样存储在同一个、具有高度伸缩性的数据库中。因此，如果有许多条目，尤其是可能要经常更新的条目储存在 cn=config 下面，则性能将会受到严重影响。

然而，尽管由于性能的原因不推荐在 cn=config 下存储简单的用户条目，但是将诸如目录管理员条目或复制管理器 （供给器绑定 DN）等特殊的用户条目储存在 cn=config 下很有用，因为这可以将配置信息集中起来。

第 15 章

管理 iPlanet Directory Server 插件

iPlanet Directory Server 插件可以扩展服务器的功能。iPlanet Directory Server 随带有多个插件，可以帮助进行目录的管理。本章提供有关可用插件的类型及如何启用、禁用插件的一般信息。本章分为以下几节：

- 服务器插件功能参考
- 从服务器控制台启用和禁用插件

服务器插件功能参考

下表概述了 iPlanet Directory Server 5.1 随带的插件及其可配置的选项、可配置的参数、缺省设置、相关性、与性能有关的一般信息及其它相关读物。这些表可用于权衡插件性能的利弊，从而为部署方案选择最佳的设置。标题为 “详细信息”的部分是对相关参考读物的交叉引用。

7 位检查插件

插件名	7 位检查 (NS7bitAtt)
配置条目的 DN	`cn=7-bit check,cn=plugins,cn=config`
说明	检查某些属性是否为纯 7 位属性
可配置的选项	on | off
缺省设置	`on`
可配置的参数	属性 (`uid mail userpassword`) 的列表，后跟 “,”，然后是所要检查的后缀
相关性	无
与性能有关的信息	无
详细信息	如果 iPlanet Directory Server 使用的不是 ASCII 字符 （例如日语），则关闭该插件。

ACL 插件

插件名	ACL 插件
配置条目的 DN	`cn=ACL Plugin,cn=plugins,cn=config`
说明	ACL 访问检查插件
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	N/A
详细信息	第 6 章 “管理访问控制”。

ACL 预处理插件

插件名	ACL 预处理
配置条目的 DN	`cn=ACL preoperation,cn=plugins,cn=config`
说明	ACL 访问检查插件
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	数据库
与性能有关的信息	无
详细信息	第 6 章 “管理访问控制”。

二进制语法插件

插件名	二进制语法
配置条目的 DN	`cn=Binary Syntax,cn=plugins,cn=config`
说明	用于处理二进制数据的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

布尔语法插件

插件名	布尔语法
配置条目的 DN	`cn=Boolean Syntax,cn=plugins,cn=config`
说明	用于处理布尔值的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

大小写完全匹配的字符串语法插件

插件名	大小写完全匹配的字符串语法
配置条目的 DN	`cn=Case Exact String Syntax,cn=plugins,cn=config`
说明	用于处理区分大小写之字符串的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

忽略大小写的字符串语法插件

插件名	忽略大小写的字符串语法
配置条目的 DN	`cn=Case Ignore String Syntax,cn=plugins,cn=config`
说明	用于处理不区分大小写之字符串的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

链接数据库插件

插件名	链接数据库
配置条目的 DN	`cn=Chaining database,cn=plugins,cn=config`
说明	用于处理 DN 的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	第 3 章 “配置目录数据库”。

服务类插件

插件名	服务类
配置条目的 DN	`cn=Class of Service,cn=plugins,cn=config`
说明	允许在条目之间共享属性
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	第 5 章 “高级条目管理”。

国家字符串语法插件

插件名	国家字符串语法插件
配置条目的 DN	`cn=Country String Syntax,cn=plugins,cn=config`
说明	用于处理国家的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

特异名称语法插件

插件名	特异名称语法
配置条目的 DN	`cn=Distinguished Name Syntax,cn=plugins,cn=config`
说明	用于处理 DN 的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

通用化时间语法插件

插件名	通用化时间语法
配置条目的 DN	`cn=Generalized Time Syntax,cn=plugins,cn=config`
说明	用于处理日期、时间及时区的语法
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	“通用化时间字符串”包括下列内容： 四位数的年份、两位数的月份 （例如 01 代表一月）、两位数的日期、两位数的小时、两位数的的分钟、两位数的秒、可选的不足一秒的小数部分及时区指示。我们强烈建议使用 Z 时区指示，即 “格林尼治平均时”。

整数语法插件

插件名	整数语法
配置条目的 DN	`cn=Integer Syntax,cn=plugins,cn=config`
说明	用于处理整数的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

国际化插件

插件名	国际化插件
配置条目的 DN	`cn=Internationalization Plugin,cn=plugins,cn=config`
说明	用于处理 DN 的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	国际化插件有一个参数是不能修改的： `var/ds5/slapd-`*`serverID`*`/config/slapd-collations.conf` 该目录用于存储国际化插件所用的对照顺序和区域设置。
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	请参阅附录 D “国际化”。

ldbm 数据库插件

插件名	ldbm 数据库插件
配置条目的 DN	`cn=ldbm database plug-in,cn=plugins,cn=config`
说明	实现本地数据库
可配置的选项	N/A
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	有关数据库插件属性的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。
详细信息	第 3 章 “配置目录数据库”。

旧复制插件

插件名	旧复制插件
配置条目的 DN	`cn=Legacy Replication plug-in,cn=plugins,cn=config`
说明	允许 iPlanet Directory Server 5.1 用作 4.1 供给器的客户。
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无如果该服务器不是 （也永远不会是） 4.1 服务器的客户，则可禁用该插件。
相关性	数据库
与性能有关的信息	无
详细信息	第 8 章 “管理复制”。

多原版复制插件

插件名	多原版复制插件
配置条目的 DN	`cn=Multimaster Replication plugin,cn=plugins, cn=config`
说明	允许在两个 5.0 iPlanet Directory Server 之间复制
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	数据库
与性能有关的信息	N/A
详细信息	如果您只有一个不会进行复制的服务器，则可关闭该插件。另请参阅第 8 章 “管理复制”。

八位字节字符串语法插件

插件名	八位字节字符串语法
配置条目的 DN	`cn=Octet String Syntax,cn=plugins,cn=config`
说明	用于处理八位字节字符串的语法
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

CLEAR 口令存储插件

插件名	CLEAR
配置条目的 DN	`cn=CLEAR,cn=Password Storage Schemes,cn=plugins, cn=config`
说明	用于口令加密的 CLEAR 口令存储模式
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	第 7 章 “用户帐户管理”。

CRYPT 口令存储插件

插件名	CRYPT
配置条目的 DN	`cn=CRYPT,cn=Password Storage Schemes,cn=plugins, cn=config`
说明	用于口令加密的 CRYPT 口令存储模式
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	第 7 章 “用户帐户管理”。

NS-MTA-MD5 口令存储插件

插件名	NS-MTA-MD5
配置条目的 DN	`cn=NS-MTA-MD5,cn=Password Storage Schemes,cn=plugins,cn=config`
说明	用于口令加密的 NS-MTA-MD5 口令存储模式
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。 iPlanet 建议让该插件始终处于运行状态。
详细信息	不能选用 NS-MTA-MD5 口令存储模式来加密口令。该存储模式在 iPlanet Directory Server 5.1 中确实存在，但只是保持为了与 iPlanet Directory Server 的早期版本兼容。请参阅第 7 章 “用户帐户管理”。

SHA 口令存储插件

插件名	SHA
配置条目的 DN	`cn=SHA,cn=Password Storage Schemes,cn=plugins,cn=config`
说明	用于口令加密的 SHA 口令存储模式
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	如果目录中未含使用 SHA 口令存储模式加密的口令，则可关闭该插件。SSHA 相对于 SHA 更为安全，您应该首选 SSHA。
详细信息	第 7 章 “用户帐户管理”。

SSHA 口令存储插件

插件名	SSHA
配置条目的 DN	`cn=SSHA,cn=Password Storage Schemes,cn=plugins,cn=config`
说明	用于口令加密的 SSHA 口令存储模式
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	第 7 章 “用户帐户管理”。

邮政地址字符串语法插件

插件名	邮政地址语法
配置条目的 DN	`cn=Postal Address Syntax,cn=plugins,cn=config`
说明	用于处理邮政地址的语法
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

PTA 插件

插件名	传递验证插件
配置条目的 DN	`cn=Pass Through Authentication,cn=plugins,cn=config`
说明	允许进行传递验证。该机制允许一个目录对另一个目录进行查询，从而验证绑定请求。如果用户目录和配置目录使用同一服务器，则该插件将不在 iPlanet Directory Server Console 中列出。
可配置的选项	on | off
缺省设置	`off`
可配置的参数	`ldap://iplanet.com:389/o=iplanet`
相关性	无
与性能有关的信息	第 16 章 “使用传递验证插件”。
详细信息	第 16 章 “使用传递验证插件”。

Referential Integrity Postoperation 插件

插件名	Referential Integrity Postoperation
配置条目的 DN	`cn=Referential Integrity Postoperation,cn=plugins,cn=config`
说明	支持服务器确保实现参照完整性
可配置的选项	所有配置； on | off
缺省设置	off
可配置的参数	启用后， Referential Integrity Postoperation 插件将在删除或重命名操作后立即对 `member`、 `uniquemember`、 `owner` 和 `seeAlso` 属性执行完整性更新。您可以重新配置该插件，使之对其它所有属性都执行完整性检查。 可配置的参数如下所示： 1. 参照完整性检查 `-1` = 不进行参照完整性检查 `0` = 立即进行参照完整性检查 正整数 = 对参照完整性请求进行排队并在稍后处理此正整数用作线程处理请求时所用的唤醒调用，其时间间隔对应于指定的整数。 2. 存储更改用的日志文件，例如 `var/ds5/slapd-`*`serverID`*`/logs/referint` 3. 其它要进行参照完整性检查的属性名。
相关性	数据库
与性能有关的信息	在多原版复制环境下，只应启用一个原版上的参照完整性插件，以避免出现解决冲突的循环。在链接服务器上启用该插件后，必须对性能资源、所需时间及完整性需求进行分析。
详细信息	请参阅第 65 页上的 “保持参照完整性”。

回退更改日志插件

插件名	回退更改日志插件
配置条目的 DN	`cn=Retro Changelog Plugin,cn=plugins,cn=config`
说明	LDAP 客户机使用该插件来保持与 iPlanet Directory Server 4.x 版的应用程序兼容。保持 iPlanet Directory Server 中所做全部更改的日志。回退更改日志与 4.x 版 iPlanet Directory Server 中的更改日志功能相同。
可配置的选项	on | off
缺省设置	`off`
可配置的参数	有关回退更改日志插件中两个配置属性的详细信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南*。
相关性	无
与性能有关的信息	可能会降低 iPlanet Directory Server 的性能。
详细信息	第 8 章 “管理复制”。

角色插件

插件名	角色插件
配置条目的 DN	`cn=Roles Plugin,cn=plugins,cn=config`
说明	支持在 iPlanet Directory Server 中使用角色
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	第 5 章 “高级条目管理”。

电话语法插件

插件名	电话语法
配置条目的 DN	`cn=Telephone Syntax,cn=plugins,cn=config`
说明	用于处理电话号码的语法
可配置的选项	on | off
缺省设置	on
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

UID 唯一性插件

插件名	UID 唯一性插件
配置条目的 DN	`cn=UID Uniqueness,cn=plugins,cn=config`
说明	每次条目上有所更改时，都检查所指定属性的值是否唯一。
可配置的选项	on | off
缺省设置	`off`

可配置的参数	输入下列参数：
	`uid`
	`"DN"`
	`"DN"...`
	前提是您想检查所列所有子树的 `uid` 属性的唯一性。
	但有时则应输入下列参数：
	`attribute="uid"`
	`MarkerObjectclass = "ObjectClassName"`
	及可选的
	`requiredObjectClass = "ObjectClassName"`
	前提是想在用 `requiredObjectClass` 添加或更新条目时检查 uid 属性唯一性（从包含由 `MarkerObjectClass` 定义的 `ObjectClass` 的父项开始）。

相关性	N/A
与性能有关的信息	该插件可能会降低 iPlanet Directory Server 的性能。 在多原版复制环境中，UID 唯一性插件无法正常工作，因此请不要启用。 如果想将新条目添加到已启用 UID 唯一性插件且在子树中已创建引荐的服务器上，则 UID 唯一性插件将无法正常工作。原因是：如果该插件看到除 `noSuchObject`（意思是该条目不存在）之外的任何其它错误（创建引荐时会出现这种情况），它就会返回操作错误，从而阻止用户添加新条目。为防止被此类操作错误所阻挡，请在创建引荐的服务器上禁用该插件。但如果仍想进行 UID 唯一性检查，请确保仅激活最后所引荐的服务器上的插件，从而避免它阻挡引荐机制。
详细信息	第 17 章 “使用属性唯一性插件”。

URI 插件

插件名	URI 语法
配置条目的 DN	`cn=URI Syntax,cn=plugins,cn=config`
说明	用于处理包含 URL （唯一资源定位符）的 URI （唯一资源标识符）的语法
可配置的选项	on | off
缺省设置	`on`
可配置的参数	无
相关性	无
与性能有关的信息	不要修改该插件的配置。该插件应始终处于运行状态。
详细信息	

从服务器控制台启用和禁用插件

要使用 iPlanet Directory Server Console 通过 LDAP 启用和禁用插件：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡。
2. 双击导航树中的 Plugins 文件夹。
3. 从 “插件”列表中选择插件。
4. 要禁用插件，请清除 “启用”复选框。要启用插件，请选中该复选框。
5. 单击 “保存”。
6. 重新启动 Directory Server。

第 16 章

使用传递验证插件

传递验证 (PTA) 机制就是一个目录服务器向另一个目录服务器进行查询，从而验证绑定请求。 PTA 插件可提供该功能。对于未在本地数据库中存储的条目而言，该功能允许目录服务器接受其简单的绑定操作 （基于口令）。

利用 PTA，iPlanet Directory Server 5.1 允许在独立的 iPlanet Directory Server 实例上管理用户目录及配置目录。

本章将在以下各节中介绍 PTA 插件：

- Directory Server 5.1 如何使用 PTA
- PTA 插件的语法
- 配置 PTA 插件
- PTA 插件语法示例

Directory Server 5.1 如何使用 PTA

如果在独立的 iPlanet Directory Server 实例上安装配置目录和用户目录，安装程序就会自动将 PTA 设置为允许配置管理员用户 （通常为 `admin`）执行管理任务。

这种情况下将需要用到 PTA，因为 `admin` 用户项存储在配置目录的 `o=NetscapeRoot` 下。因此，试图作为 `admin` 绑定到用户目录时通常会失败。 PTA 允许用户目录将凭证传递给对其进行验证的配置目录。用户目录随即将 `admin` 用户绑定。

本例中的用户目录充当 PTA 目录服务器 (PTA directory server)，也就是将绑定请求传递给另一个目录服务器的服务器。配置目录则充当*验证目录*，也就是包含条目并验证请求客户机之绑定凭证的服务器。

本章中还用到*传递子树*一词。传递子树就是 PTA 目录中*不*出现的子树。当用户的绑定 DN 包含该子树时，用户凭证就会被传递给验证目录。

注意 当用户目录和配置目录使用同一服务器时， PTA 插件将不在 iPlanet Directory Server Console 中列出。

下面是传递验证的工作原理：

1. 假定在计算机 A 中安装配置目录服务器 （验证目录）。
   - 服务器名：**configdir.siroe.com**
   - 后缀：**o=NetscapeRoot**
2. 而在计算机 B 中安装用户目录服务器 （PTA 目录）。
   - 服务器名：**userdir.siroe.com**
   - 后缀：**dc=siroe,dc=com**
3. 在计算机 B 上安装用户目录时，系统将提示提供 LDAP URL。该 URL 指向计算机 A 的配置目录。
4. 安装程序将向启用 PTA 插件的用户目录中的 `dse.ldif` 文件添加条目。

   该条目中包含所提供的 LDAP URL。例如：

```
dn: cn=Pass Through Authentication,cn=plugins,
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: Pass Through Authentication
nsslapd-pluginPath: /usr/iplanet/ds5/lib/passthru-plugin.so
nsslapd-pluginInitfunc: passthruauth_init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: ldap://config.siroe.com/ou=NetscapeRoot
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: passthruauth
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: pass through authentication plugin
```

   用户目录现在即配置为将 DN 中包含 `o=NetscapeRoot` 的条目请求发送给配置目录 `configdir.siroe.com`。

5. 安装过程结束后， admin 用户将试图连接用户目录以开始添加用户。
6. 安装程序将把 admin 用户的条目作为 uid=admin,ou=TopologyManagement,o=NetscapeRoot 添加到目录中。因此，正如 PTA 插件配置所定义的那样，用户目录将把绑定请求传递给配置目录。
7. 配置目录将验证用户的凭证，然后将信息发送回用户目录。
8. 用户目录即允许 admin 用户进行绑定。

PTA 插件的语法

PTA 插件的配置信息是在 cn=Pass Through Authentication,cn=plugins,cn=config 条目 （位于 PTA 目录的 dse.ldif 文件中，其中 PTA 目录就是配置为将绑定请求传递给验证目录的用户目录）中指定的，使用的是本部分所述的语法。

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: Pass Through Authentication
nsslapd-pluginPath: /usr/iplanet/ds5lib/passthru-plugin.extension
nsslapd-pluginInitfunc: passthruauth_init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: state
nsslapd-pluginarg0: ldap|ldaps://authDS/subtree [maxconns,maxops,timeout,ldver,connlifetime]
```

有关 PTA 插件之变量组件的说明，请参阅表 16-1。

注意
- LDAP URL (ldap|ldaps://*authDS*/*subtree*) 必须用空格与可选参数（*maxconns*、 *maxops*、 *timeout*、 *ldver*、 *connlifetime*）分隔。
- 如果明确定义某些可选参数，则必须全部进行定义，即使是仅指定默认值。
- 通过每次使 nsslapd-pluginarg 属性后缀增 1，可以指定几个验证目录或子树。图示见第 429 页上的 “指定多个验证目录服务器”。

可选参数的说明见下表，取其在语法中的顺序。

表 16-1 PTA 插件参数

变量	定义
state	定义是否启用该插件。可接受的值为 **on** 或 **off**。有关详细信息，请参阅第 424 页上的 “打开或关闭插件”。
extension	插件的文件扩展名。扩展名为 .so。
ldap | ldaps	定义 SSL 是否用于两个目录服务器之间的通讯。有关详细信息，请参阅第 424 页上的 “将服务器配置为使用安全连接”。
authDS	验证目录的主机名。添加冒号，后跟端口号，即可指定目录服务器的端口号。例如： ldap://dirserver.siroe.com:390/ 如果不指定端口号。 PTA 服务器将尝试用下列端口进行连接： • 如果 URL 中指定 ldap://，则为 389。 • 如果 URL 中指定 ldaps://，则为 636。 有关详细信息，请参阅第 425 页上的 “指定验证目录服务器”。
subtree	传递子树 (pass-through subtree)。对于该子树中含有其 DN 的客户机， PTA 目录服务器将把其中的绑定请求传递给验证目录服务器。 有关详细信息，请参阅第 426 页上的 “指定传递子树”。
maxconns	可选。 PTA 目录可同时向验证目录打开的最大连接数。默认值为 3。 有关详细信息，请参阅第 426 页上的 “配置可选参数”。
maxops	可选。在单个连接中， PTA 目录可发送给验证目录的最大并行操作数 （通常为绑定请求）。默认值为 5。 有关详细信息，请参阅第 426 页上的 “配置可选参数”。
timeout	可选。 PTA 目录等待验证目录服务器进行响应时的时间限制 （秒）。如果超出该超时设置，服务器就会向客户机返回错误。 默认值为 300 秒 （5 分钟）。指定 0 时，指示没有强制的时间限制。 有关详细信息，请参阅第 426 页上的 “配置可选参数”。
ldver	可选。用于连接验证目录的 LDAP 协议的版本。 iPlanet Directory Server 支持 LDAP 版本 2 和 3。默认值为版本 3。 有关详细信息，请参阅第 426 页上的 “配置可选参数”。

表 16-1 PTA 插件参数 *（续）*

变量	定义
connlifetime	可选。可以使用连接的时间限制（秒）。如果绑定请求是在超出该时间限制后由客户机启动的，服务器就会关闭连接，然后打开到验证目录的新连接。除非已启动绑定请求且目录判定已超出连接的使用期，否则服务器将不会关闭该连接。 如果不指定该选项，或者仅列出一个主机，就不会有强制的连接使用期。如果列出了多个主机，则默认值为 300 秒（5 分钟）。 有关详细信息，请参阅第 426 页上的 “配置可选参数”。

配置 PTA 插件

配置 PTA 插件的唯一方法是修改条目 `cn=Pass Through Authentication, cn=plugins,cn=config`（位于 `dse.ldif` 文件中）。要修改 `dse.ldif` 文件，则必须进行如下操作：

1. 使用 `ldapmodify` 命令修改 `cn=Pass Through Authentication,cn=plugins, cn=config`
2. 重新启动 iPlanet Directory Server。

配置本部分所述的参数之前，`dse.ldif` 文件中必须存在 PTA 插件条目。如果该条目不存在，则必须用相应的语法予以创建，说明见第 421 页上的 “PTA 插件的语法”。

注意 如果已在不同的目录实例上分别安装用户目录和配置目录，PTA 插件条目就会自动添加到用户目录的 `dse.ldif` 文件中。如果是在同一目录实例上安装的用户目录和配置目录，则不会自动添加该语法。此时必须进行手动添加。

本部分将提供有关配置该插件的信息：

- 打开或关闭插件
- 将服务器配置为使用安全连接
- 指定验证目录服务器
- 指定传递子树
- 配置可选参数

打开或关闭插件

要从命令行上打开 PTA 插件：

1. 创建包含下列 LDIF 更新语句的 LDIF 文件：

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
cn: Pass Through Authentication
changetype: modify
replace: nsslapd-pluginenabled
nsslapd-pluginenabled: on
```

2. 使用 `ldapmodify` 命令可以将 LDIF 文件导入到目录中。

3. 启用插件时，还必须检查是否已正确定义插件的初始化功能。

   条目 `cn=Pass Through Authentication,cn=plugins,cn=config` 中应包含下列属性 – 值对：

```
nsslapd-pluginPath: /usr/iplanet/ds5/lib/passthru-plugin.so
nsslapd-pluginInitfunc: passthruauth_init
```

4. 重新启动服务器。

   有关重新启动服务器的信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

要禁用插件，请更改 LDIF 更新语句以删除 `nsslapd-pluginenabled: on` 语句，然后添加 `nsslapd-pluginenabled: off` 语句。从命令行上启用或禁用 PTA 插件时，必须重新启动服务器。

将服务器配置为使用安全连接

可以对 PTA 目录进行配置，使之通过 SSL 与验证目录进行通讯。为此，可指定 LDAPS（PTA 目录中的 LDAP URL）。

要将 PTA 目录和验证目录配置为使用 SSL：

1. 创建包含下列 LDIF 更新语句的 LDIF 文件：

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
cn: Pass Through Authentication
changetype: modify
replace: nsslapd-pluginarg0
nsslapd-pluginarg0: ldaps://authDS/subtree [optional_parameters]
```

   有关该语法中变量组件的信息，请参阅第 422 页上的 “PTA 插件参数”。

2. 使用 `ldapmodify` 命令可以将 LDIF 文件导入到目录中。
3. 重新启动服务器。

   有关重新启动服务器的信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

指定验证目录服务器

验证目录中包含客户机所要绑定的条目的绑定凭证。 PTA 目录将绑定请求传递给定义为验证目录的主机。您可以按下列方式指定验证目录服务器：用验证目录的主机名替换 PTA 目录之 LDAP URL 中的 *authDS*。

要指定 PTA 的验证目录：

1. 创建包含下列 LDIF 更新语句的 LDIF 文件：

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
cn: Pass Through Authentication
changetype: add
add: nsslapd-pluginarg0
nsslapd-pluginarg0: ldap://authDS/subtree [optional_parameters]
```

   也可以包含冒号，后面跟端口号。如果不指定端口号， PTA 目录服务器将尝试使用下列端口号进行连接：

   - 如果 URL 中指定 ldap://，则为 389。
   - 如果 URL 中指定 ldaps://，则为 636。

   例如，可以将 `nsslapd-pluginarg0` 属性的值更改为：

   `"ldap://dirserver.siroe.com:389/subtree [Parameters]"`

   有关该语法中变量组件的信息，请参阅第 422 页上的 “PTA 插件参数”。

2. 使用 `ldapmodify` 命令可以将 LDIF 文件导入到目录中。
3. 重新启动服务器。

   有关重新启动服务器的信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

指定传递子树

对于其 DN 已在传递子树 (pass-through subtree) 中定义的客户机而言， PTA 目录将从中将绑定请求传递给验证目录。指定子树时，应替换 PTA 目录之 LDAP URL 中的 *subtree* 参数。

传递子树不得在 PTA 目录中存在。如果存在传递子树， PTA 目录就会尝试使用自己的目录内容来解析请求，从而导致绑定操作失败。

要指定传递子树：

1. 创建包含下列 LDIF 更新语句的 LDIF 文件：

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
cn: Pass Through Authentication
changetype: add
add: nsslapd-pluginarg0
nsslapd-pluginarg0: ldap://authDS/subtree [optional_parameters]
```

   例如，可以将 `nsslapd-pluginarg0` 属性的值更改为：

   `"ldap://dirserver.siroe.com/o=NetscapeRoot [Parameters]"`

   有关该语法中变量组件的信息，请参阅第 422 页上的 “PTA 插件参数”。

2. 使用 `ldapmodify` 命令可以将 LDIF 文件导入到目录中。

3. 重新启动服务器。

   有关重新启动服务器的信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

配置可选参数

您可以对下列 PTA 插件的可选参数进行配置：

- PTA 目录服务器可同时向验证目录打开的最大连接数，在 PTA 语法中用 *maxconns* 表示。默认值为 `3`。
- 单个连接中， PTA 目录服务器可向验证目录服务器同时发送的最大绑定请求数。在 PTA 语法中，该参数表示为 *maxops*。默认值为 `5`。
- PTA 目录服务器等待验证目录服务器进行响应时的时间限制 （秒）。在 PTA 语法中，该参数表示为 *timeout*。默认值为 `300` 秒 （5 分钟）。
- 希望 PTA 目录服务器用于连接验证目录服务器的 LDAP 协议的版本。在 PTA 语法中，该参数表示为 *ldver*。默认值为 `LDAPv3`。

- 可以使用连接的时间限制 （秒）。如果绑定请求是在超出该时间限制后由客户机启动的，服务器就会关闭连接，然后打开到验证目录服务器的新连接。除非已启动绑定请求且服务器判定已超出超时设置，否则服务器将不会关闭该连接。如果不指定该选项，或者 *authDS* 参数中只列出了一个验证目录服务器，则不会有强制的时间限制。如果列出了多个主机，则默认值为 300 秒 （5 分钟）。在 PTA 语法中，该参数表示为 *connlifetime*。

注意 尽管这些参数为可选项，但如果指定其中的某个参数，则需要予以全部指定，即使使用的是默认值。

1. 创建包含下列 LDIF 更新语句的 LDIF 文件：

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
cn: Pass Through Authentication
changetype: add
add: nsslapd-pluginarg0
nsslapd-pluginarg0: ldap://authDS/subtree [maxconns,maxops,timeout,ldver,connlifetime]
```

   确保 *subtree* 参数和可选参数之前有一个空格。

   例如，可以将 `nsslapd-pluginarg0` 属性的值更改为：

   `"ldap://dirserver.siroe.com/o=NetscapeRoot 3,5,300,3,300"`

   本例中，每个可选参数均被设为其默认值。

2. 使用 `ldapmodify` 命令可以将 LDIF 文件导入到目录中。

3. 重新启动服务器。

   有关重新启动服务器的信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

PTA 插件语法示例

本部分在 dse.ldif 文件中提供 PTA 插件语法的以下示例：

- 指定一个验证目录服务器和一个子树
- 指定多个验证目录服务器
- 指定一个验证目录服务器和多个子树
- 使用非默认参数值
- 为不同的验证目录服务器指定不同的可选参数和子树

指定一个验证目录服务器和一个子树

本例将对 PTA 插件进行配置，使之接受可选变量的所有默认值。对于 o=NetscapeRoot 子树的绑定请求而言，该配置将导致 PTA 目录服务器连接到验证目录服务器。验证目录服务器的主机名为 config-dir.siroe.com。

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: Pass Through Authentication
nsslapd-pluginPath: /usr/iplanet/ds5/lib/passthru-plugin.so
nsslapd-pluginInitfunc: passthruauth_init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: ldap://config-dir.siroe.com/ou=NetscapeRoot
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: passthruauth
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: pass through authentication plugin
```

指定多个验证目录服务器

如果 PTA 目录服务器与验证目录服务器之间的连接被断开或无法打开， PTA 目录服务器就会将请求发送给所指定的下一服务器 （如果有）。可以根据需要指定任意多个验证目录服务器。

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: Pass Through Authentication
nsslapd-pluginPath: /usr/iplanet/ds5/lib/passthru-plugin.so
nsslapd-pluginInitfunc: passthruauth_init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: ldap://config-dir.siroe.com/ou=NetscapeRoot
nsslapd-pluginarg1: ldap://config2-dir.siroe.com/ou=NetscapeRoot
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: passthruauth
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: pass through authentication plugin
```

指定一个验证目录服务器和多个子树

下例将对 PTA 目录服务器进行配置，使之传递多个子树的绑定请求 （使用参数的默认值）：

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: Pass Through Authentication
nsslapd-pluginPath: /usr/iplanet/ds5/lib/passthru-plugin.so
nsslapd-pluginInitfunc: passthruauth_init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: ldap://config-dir.siroe.com/ou=NetscapeRoot
nsslapd-pluginarg1: ldap://config-dir.siroe.com/dc=siroe,dc=com
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: passthruauth
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: pass through authentication plugin
```

使用非默认参数值

本例仅对最大连接数参数 *maxconns* 使用非默认值 (10)。其它各个参数均设为其默认值。但由于已指定了一个参数，因此在语法中必须对所有参数进行明确定义。

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: Pass Through Authentication
nsslapd-pluginPath: /usr/iplanet/ds5/lib/passthru-plugin.so
nsslapd-pluginInitfunc: passthruauth_init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: ldap://config-dir.siroe.com/ou=NetscapeRoot 10,5,300,3,300
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: passthruauth
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: pass through authentication plugin
```

为不同的验证目录服务器指定不同的可选参数和子树

如果想为各个验证目录服务器指定不同的传递子树和可选参数，则必须指定多个 LDAP URL/可选参数对。如下所示， LDAP URL/可选参数对之间应使用单个空格进行分隔。

```
dn: cn=Pass Through Authentication,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: Pass Through Authentication
nsslapd-pluginPath: /usr/iplanet/ds5/lib/passthru-plugin.so
nsslapd-pluginInitfunc: passthruauth_init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: ldap://config-dir.siroe.com/ou=NetscapeRoot 7,7,300,3,300
nsslapd-pluginarg1: ldap://config2-dir.siroe.com/dc=siroe,dc=com 7,7,300,3,300
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: passthruauth
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: pass through authentication plugin
```

第 17 章

使用属性唯一性插件

属性唯一性插件可用于确保所指定的属性在目录中始终具有唯一值。对于要确保值具有唯一性的各个属性而言，必须创建插件的新实例。

iPlanet Directory Server 5.1 所提供的 UID 唯一性插件可用于管理 UID 属性的唯一性。

本章将在以下各节中介绍属性唯一性插件及 UID 唯一性插件：

- 属性唯一性插件概述
- UID 唯一性插件概述
- 属性唯一性插件的语法
- 创建属性唯一性插件的实例
- 配置属性唯一性插件
- 属性唯一性插件语法示例
- 复制和属性唯一性插件

属性唯一性插件概述

属性唯一性插件是一种预处理插件。这就意味着该插件将在服务器执行 LDAP 操作前检查所有更新操作。该插件将确定操作是否适用于已配置为监控对象的某个属性及后缀。

如果更新操作适用于该插件所监控的属性和后缀，且会使两个条目具有相同的属性值，服务器就会终止该操作，同时向客户机返回 `LDAP_CONSTRAINT_VIOLATION` 错误。

属性唯一性插件对下列对象执行检查：

- 单个属性
- 一个或多个子树

如果希望检查多个属性的唯一性，则必须为所要检查的各个属性创建独立的插件实例。

还可以对属性唯一性插件的操作进行配置：

- 它可以检查所指定的子树中的各个条目。

  例如，如果您所属的公司 siroe.com 有目录 Company333 和 Company999，则在添加诸如 `uid=jlittle,ou=people,o=Company333,dc=siroe,dc=com` 等条目时，则只需要在 `o=Company333,dc=siroe,dc=com` 子树中强制实施唯一性。这时，需要在 UID 唯一性插件配置中明确列出子树的 DN。

  有关该配置选项的详细说明，请参阅第 439 页上的 “指定后缀或子树”。

- 您可以在更新条目的 DN 中指定与条目有关的对象类，然后对下面的所有条目执行唯一性检查。

  该选项在主机环境中较为有用。例如，添加诸如 `uid=jlittle,ou=people,o=Company333,dc=siroe,dc=com` 等条目时，可以在 `o=Company333,dc=siroe,dc=com` 下强制实施唯一性，但在配置中不明确列出该子树，而是指示 *marker 对象类*。如果指定的 marker 对象类为 `organization`，则唯一性检查算法将找到 DN 中含有该对象类 (`o=Company333`) 的条目，然后对下面的所有条目执行检查。

  此外，还可以指定仅在更新条目中包含指定对象类的情况下执行唯一性检查。例如，可以指定仅在更新条目包含 `objectclass=inetorgperson` 时执行检查。

  有关该配置选项的详细说明，请参阅第 439 页上的 “使用 markerObjectClass 和 requiredObjectClass 关键字”。

如果打算在复制环境中使用属性唯一性插件，请参阅第 442 页上的 “复制和属性唯一性插件”。

UID 唯一性插件概述

iPlanet Directory Server 5.0 中提供了属性唯一性插件的一个实例：UID 唯一性插件。缺省情况下，该插件可确保赋予 UID 属性的值在为目录所配置的后缀中是唯一的 （后缀对应于 `userRoot` 数据库）。

您可以更改配置，从而指定其它后缀或子树，或者指定仅在包含所提供的对象类的条目下执行检查。 UID 唯一性插件的语法和配置与其它所有属性的相同。有关可以对配置进行何种更改的详细信息，请参阅第 437 页上的 “配置属性唯一性插件”。

由于 UID 唯一性插件会影响多原版复制操作，因此缺省情况下将处于禁用状态。有关在复制环境中使用属性唯一性插件的信息，请参阅第 442 页上的 “复制和属性唯一性插件”。

属性唯一性插件的语法

属性唯一性插件的配置信息是在 `cn=plugins,cn=config` 条目下的条目中指定的。可能的 `nsslapd-pluginarg` 属性语法结构有两种。区别之处已在以下部分突出显示。

使用下列语法可以在后缀或子树下执行唯一性检查：

```
dn: cn=descriptive_plugin_name,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: descriptive_plugin_namee
nsslapd-pluginPath: /usr/iplanet/ds5/lib/uid-plugin.so
nsslapd-pluginInitfunc: NSUniqueAttr_Init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: state
nsslapd-pluginarg0: attribute_name
nsslapd-pluginarg1: dn1
[ nsslapd-pluginarg2: dn2 ]
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: NSUniqueAttr
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: Enforce unique attribute values
```

注意：

- 可以在 cn 属性中指定任何名称来命名插件。名称应具有说明性。该属性*不*含要进行唯一性检查的属性的名称。
- 您只能指定一个要进行唯一性检查的属性。
- 对于要执行唯一性检查的后缀或子树而言，通过每次使 nsslapd-pluginarg 属性后缀递增 1，可以指定该后缀或子树的几个 DN。

有关属性唯一性插件语法的变量部分，请参阅表 17-1。

使用下列语法可以指定在包含指定对象类的条目下执行唯一性检查：

```
dn: cn=descriptive_plugin_name,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: descriptive_plugin_name
nsslapd-pluginPath: /usr/iplanet/ds5/lib/uid-plugin.so
nsslapd-pluginInitfunc: NSUniqueAttr_Init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: state
nsslapd-pluginarg0: attribute=attribute_name
nsslapd-pluginarg1: markerObjectClass=objectclass1
[ nsslapd-pluginarg2: requiredObjectClass=objectclass2 ]
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: NSUniqueAttr
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: Enforce unique attribute values
```

注意：

- 可以在 cn 属性中指定任何名称来命名插件。名称应具有说明性。该属性*不*含要进行唯一性检查的属性的名称。
- 您只能指定一个要进行唯一性检查的属性。
- 如果 nsslapd-pluginarg0 属性以 attribute= *attribute_name* 开头，则服务器就会希望 nsslapd-pluginarg1 属性中包含 markerObjectClass。

有关属性唯一性插件语法的变量部分，请参阅表 17-1。

表 17-1 属性唯一性插件变量

变量	定义
descriptive_plugin_name	指定该属性唯一性插件实例的名称。对于要确保唯一性的属性而言，并非必须包含其名称，但最好还是包含。例如 `cn=mail uniqueness,cn=plugins,cn=config`。
extension	插件的文件扩展名。 (`.so`)。
state	定义是否启用该插件。可接受的值为 **`on`** 或 **`off`**。有关详细信息，请参阅第 438 页上的 “打开或关闭插件”。
attribute_name	要确保其值具有唯一性的属性的名称。这里只能指定一个属性名。
dn	要确保其属性具有唯一性的后缀或子树的 DN。对于每个附加后缀或子树而言，通过将 `nsslapd-pluginarg` 属性的后缀的值递增 1，即可指定几个后缀或子树。
`attribute=`*attribute_name*	要确保其值具有唯一性的属性的名称。这里只能指定一个属性名。
`markerObjectClass=`*objectclass1*	对于具有在 `markerObjectClass` 关键字中指定的对象类的更新条目而言，如果某个条目属于该条目的 DN，系统就会在其中执行属性唯一性检查。 请勿在等号前后包含空格。
`requiredObjectClass=`*objectclass2*	可选。使用 `markerObjectClass` 关键字而非 DN 来指定唯一性检查的范围时，可以指定仅在更新条目中包含 `requiredObjectClass` 关键字中指定的对象类时才执行检查。 请勿在等号前后包含空格。

创建属性唯一性插件的实例

如果想确保目录中的某个特定属性始终具有唯一的值，则必须为要检查的属性创建属性唯一性插件的实例。例如，如果想确保目录中包含 mail 属性的各个条目都有该属性的唯一值，则必须创建 mail 唯一性插件。

要创建属性唯一性插件的实例，您必须修改 dse.ldif 文件，在 cn=plugins, cn=config 条目下为新插件添加一个条目。新条目的格式必须符合第 433 页上的“属性唯一性插件的语法”中所述的语法。

例如，要实例化 mail 属性的属性唯一性插件，您需要执行下列步骤：

1. 在 dse.ldif 文件中，查找 UID 唯一性插件条目 cn=uid uniqueness, cn=plugins,cn=config。
2. 在 UID 唯一性插件条目的前后，为 mail 唯一性插件条目添加以下各行：

```
dn: cn=mail uniqueness,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: mail uniqueness
nsslapd-pluginPath: /usr/iplanet/ds5/lib/uid-plugin.so
nsslapd-pluginInitfunc: NSUniqueAttr_Init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: mail
nsslapd-pluginarg1: dc=siroe,dc=com
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: NSUniqueAttr
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: Enforce unique attribute values
```

3. 重新启动 iPlanet Directory Server。

本例中，系统将对包含 mail 属性的 dc=siroe,dc=com 条目中的各个条目执行唯一性检查。

配置属性唯一性插件

本节介绍如何使用 iPlanet Directory Server Console 来查看为目录所配置的插件，同时说明如何修改属性唯一性插件的配置。

查看插件配置信息

在 iPlanet Directory Server Console 中，可以按如下所示显示属性唯一性插件的配置条目：

1. 在 iPlanet Directory Server Console 上，单击 “目录”选项卡。
2. 在左侧的导航树中，展开 config 文件夹，然后展开 Plugins 文件夹。

   插件列表将显示在右侧的导航窗口中。此时应看到 UID 唯一性插件及在第 436 页上的 “创建属性唯一性插件的实例”示例中所创建的其它所有属性唯一性插件。
3. 在右侧导航窗口中，双击要查看的插件条目。

   此时显示 “属性编辑器”。其中包含所有插件属性和值的列表。

从 iPlanet Directory Server Console 配置属性唯一性插件

在 iPlanet Directory Server Console 中可以通过几种方式来更新插件配置：

- 从 “属性编辑器”中

  按第 437 页上的 “查看插件配置信息”中所述显示属性编辑器，然后编辑属性值字段。
- 从 “配置”选项卡中

要从 iPlanet Directory Server Console “配置”选项卡中修改属性唯一性插件的配置：

1. 在 iPlanet Directory Server Console 上，选择 “配置”选项卡，然后在导航树中展开 Plugins 文件夹，之后选择要修改的属性唯一性插件。

   插件的配置参数将显示在左侧窗口中。
2. 要打开或关闭插件，请选中或清除 “启用插件”复选框。

3. 要添加后缀或子树，请单击 “添加”，然后在空的文本字段中键入 DN。

   如果不想添加 DN，可以使用 `markerObjectClass` 关键字。如果使用该语法，则可再次单击 “添加”，按第 433 页上的 “属性唯一性插件的语法”中所述指定 requiredObjectClass。

注意 此列表中不得添加属性名。如果想检查其它属性的唯一性，则必须为所要检查的属性创建属性唯一性插件的实例。有关信息，请参阅第 436 页上的 “创建属性唯一性插件的实例”。

4. 要从列表中删除某一项，请将光标置于要删除的文本字段中，然后单击 “删除”。
5. 单击 “保存”可保存更改结果。

从命令行配置属性唯一性插件

本节提供有关从命令行配置插件的信息。其中涉及以下任务：

- 打开或关闭插件
- 指定后缀或子树
- 使用 markerObjectClass 和 requiredObjectClass 关键字

打开或关闭插件

要从命令行打开插件，则必须创建包含下列 LDIF 更新语句的 LDIF 文件：

```
dn: cn=descriptive_plugin_name,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginenabled
nsslapd-pluginenabled: on
```

使用 `ldapmodify` 命令可以将 LDIF 文件导入到目录中。

要禁用插件，请更改 LDIF 更新语句：将 `nsslapd-pluginenabled: on` 语句替换为 `nsslapd-pluginenabled: off` 语句。

启用或禁用插件时，必须重新启动服务器。有关重新启动服务器的信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

指定后缀或子树

您可以指定希望插件确保其内容具有属性唯一性的后缀或子树，方法是在定义该插件的条目中使用 nsslapd-pluginarg 属性。

通过创建包含如下例中所示的更新语句的 LDIF 文件，即可指定子树：

```
dn: cn=mail uniqueness,cn=plugins,cn=config
changetype: add
nsslapd-pluginarg2: dc=iplanet,dc=sun,dc=com
nsslapd-pluginarg3: dc=iplanet,dc=netscape,dc=com
```

本 LDIF 文件示例将检查子树 dc=siroe,dc=com、dc=iplanet,dc=sun,dc=com 和 dc=iplanet,dc=netscape.com 下 mail 属性的唯一性。

使用 ldapmodify 命令可以将 LDIF 文件导入到目录中。

进行上述类型的配置更改时，必须重新启动服务器。有关重新启动服务器的信息，请参阅第 35 页上的 “启动和停止 iPlanet Directory Server”。

使用 markerObjectClass 和 requiredObjectClass 关键字

除了在属性唯一性插件的配置中指定后缀或子树外，还可以对属于更新条目的 DN，且更新条目中具有 markerObjectClass 关键字中所指定的对象类的条目执行检查。

要指定对包含组织单元 (ou) 对象类的更新条目 DN 中的条目进行唯一性检查，则可以创建如下例中所示的 LDIF 文件：

```
dn: cn=mail uniqueness,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: mail uniqueness
nsslapd-pluginPath: /usr/iplanet/ds5/lib/uid-plugin.so
nsslapd-pluginInitfunc: NSUniqueAttr_Init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: attribute=mail
nsslapd-pluginarg1: markerObjectClass=ou
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: NSUniqueAttr
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: Enforce unique attribute values
```

如果不希望服务器检查组织单元条目下的各个条目，则可指定仅在更新条目中包含指定对象类的情况下才执行检查，从而限定检查的范围。

例如，如果是检查 mail 属性的唯一性，则可能有必要仅在添加或修改包含 person 或 inetorgperson 对象类的条目时才执行检查。

通过使用 requiredObjectClass 关键字，可以限制检查范围，如下例所示：

```
dn: cn=mail uniqueness,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: mail uniqueness
nsslapd-pluginPath: /usr/iplanet/ds5/lib/uid-plugin.so
nsslapd-pluginInitfunc: NSUniqueAttr_Init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: attribute=mail
nsslapd-pluginarg1: markerObjectClass=ou
nsslapd-pluginarg2: requiredObjectClass=person
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: NSUniqueAttr
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: Enforce unique attribute values
```

通过递增 nsslapd-pluginarg 属性后缀的计数器并不能重复 markerObjectClass 或 requiredObjectClass 关键字。

注意 nsslapd-pluginarg0 属性始终包含要确保唯一性的属性的名称。

属性唯一性插件语法示例

本节在 dse.ldif 文件中含有属性唯一性插件语法的示例。所有示例所示的插件语法与其在 UNIX 系统上出现的语法都相同。

- 指定一个属性及一个子树
- 指定一个属性及多个子树

指定一个属性及一个子树

本例将对插件进行配置，确保 dc=siroe,dc=com 子树下 mail 属性的唯一性。

```
dn: cn=mail uniqueness,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: mail uniqueness
nsslapd-pluginPath: /usr/iplanet/ds5/lib/uid-plugin.so
nsslapd-pluginInitfunc: NSUniqueAttr_Init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: mail
nsslapd-pluginarg1: dc=siroe,dc=com
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: NSUniqueAttr
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: Enforce unique attribute values
```

指定一个属性及多个子树

本例将对插件进行配置，确保 l=Chicago,dc=siroe,dc=com 和 l=Boston, dc=siroe,dc=com 子树下 mail 属性的唯一性。

```
dn: cn=mail uniqueness,cn=plugins,cn=config
objectClass: top
objectClass: nsSlapdPlugin
objectClass: extensibleObject
cn: mail uniqueness
nsslapd-pluginPath: /usr/iplanet/ds5/lib/uid-plugin.so
nsslapd-pluginInitfunc: NSUniqueAttr_Init
nsslapd-pluginType: preoperation
nsslapd-pluginEnabled: on
nsslapd-pluginarg0: mail
nsslapd-pluginarg1: l=Chicago,dc=siroe,dc=com
nsslapd-pluginarg2: l=Boston,dc=siroe,dc=com
nsslapd-plugin-depends-on-type: database
nsslapd-pluginId: NSUniqueAttr
nsslapd-pluginVersion: 5.0
nsslapd-pluginVendor: Sun | Netscape Alliance
nsslapd-pluginDescription: Enforce unique attribute values
```

注意 nsslapd-pluginarg0 属性始终包含要确保唯一性的属性的名称。其它所有位置出现的 nsslapd-pluginarg （nsslapd-pluginarg1 至 nsslapd-pluginargx）都包含 DN。

利用该配置，插件允许 mail 属性值的实例分别在 l=Chicago,dc=siroe,dc=com 子树和 l=Boston,dc=siroe,dc=com 子树下各存在一次。例如，下列情况是允许的：

```
mail=bjensen,l=Chicago,dc=siroe,dc=com
```

```
mail=bjensen,l=Boston,dc=siroe,dc=com
```

如果想确保两个子树下只有值的一个实例，则需要对插件进行配置，以确保整个 dc=siroe,dc=com 子树的唯一性。

复制和属性唯一性插件

在涉及复制协议的 iPlanet Directory Server 上使用属性唯一性插件时，必须审慎考虑在各个服务器上如何配置插件。

应考虑下列情况：

- 简单复制：有一个供给器及一个或多个客户
- 复杂复制：有多个原版

当作为复制操作的一部分而执行更新时，属性唯一性插件不会对属性值执行任何检查。

简单复制环境

由于客户机应用程序所做的所有修改都是在供给服务器上进行的，因此应启用供给器上的属性唯一性插件。客户服务器上没必要启用该插件。

启用客户上的属性唯一性插件不会阻碍 iPlanet Directory Server 的正常运行，但可能会导致性能降低。

多原版复制环境

在多原版复制环境中，两个原版同时充当同一副本的供给器和客户。由于多原版复制使用松散的一致性复制模式，因此在一个服务器上启用属性唯一性插件尚不足以确保属性值在给定的时间内在两个原版上保持唯一性。因此，在一个服务器上启用属性唯一性插件可导致各个副本上所持数据的不一致。

但在满足下列所有条件的前提下，可以使用属性唯一性插件：

- 进行唯一性检查的属性为命名属性
- 两个原版上都已启用属性唯一性插件

在满足上述条件的情况下，复制时的属性唯一性冲突将报告为命名冲突。命名冲突需要进行手动解决。有关如何解决复制冲突的信息，请参阅第 306 页上的 “解决常见复制冲突”。

附录 A

LDAP 数据交换格式

iPlanet Directory Server 使用 LDAP 数据交换格式 (LDIF) 来描述文本格式的目录及目录项。LDIF 通常用于构建初始目录数据库或一次将大量的条目添加到目录中。此外，LDIF 还用于描述目录项的更改。因此，多数 iPlanet Directory Server 命令行实用程序都依赖于 LDIF 进行输入或输出。

由于 LDIF 是一种文本文件格式，因此实际上可以使用任何语言来创建 LDIF 文件。所有目录数据都将用 Unicode 的 UTF-8 编码方式进行存储。因此，所创建的 LDIF 文件也必须为 UTF-8 编码格式。

本章包含有关 LDIF 信息的以下部分：

有关使用 LDIF 来修改目录条目的信息，请参阅第 2 章 “创建目录项”。

LDIF 文件格式

LDIF 由一个或多个用空行分隔的目录条目组成。每个 LDIF 条目都包括可选的条目 ID、必需的特异名称、一个或多个对象类及多个属性定义。

LDIF 格式在 RFC 2849 *LDAP 数据交换格式 (LDIF)* 中定义。 iPlanet Directory Server 符合该标准。

以 LDIF 表示的目录条目基本格式如下所示：

```
dn: distinguished_name
objectClass: object_class
objectClass: object_class
...
attribute_type[;subtype]:attribute_value
attribute_type[;subtype]:attribute_value
...
```

您必须提供 DN 及至少一个对象类定义。此外，还须包括为该条目定义的对象类所需的所有属性。其它所有属性和对象类为可选项。指定对象类和属性的顺序可以任意。冒号后面的空格也是可选的。有关标准对象类和属性的信息，请参阅 *iPlanet Directory Server 模式参考指南*。

表 A-1 介绍上面定义中所述的 LDIF 各字段。

表 A-1 LDIF 字段

字段	定义
[*id*]	可选。代表条目 ID 的一个十进制正数。数据库创建工具将生成此 ID。请勿擅自添加或编辑该值。
`dn:` *distinguished_name*	指定条目的特异名称。有关特异名称的完整说明，请参阅 *iPlanet Directory Server 部署指南*。
`objectClass:` *object_class*	指定该条目所用的对象类。对象类标识条目所允许和需要的属性类型或模式。有关标准对象类的列表，请参阅 *iPlanet Directory Server 模式参考指南*；有关自定义模式的信息，请参阅第 9 章 “扩展目录模式”。
attribute_type	指定条目所用的说明性属性。该属性应在模式中定义。有关标准属性的列表，请参阅 *iPlanet Directory Server 模式参考指南*；有关自定义模式的信息，请参阅第 9 章 “扩展目录模式”。

表 A-1 LDIF 字段 *（续）*

字段	定义
[*subtype*]	可选。指定子类型：语言、二进制或发音。使用该标记可以标识表达相应属性值的语言，或者指定属性值是二进制还是属性值的发音。有关属性子类型的信息，请参阅第 49 页上的 “添加属性子类型”。有关受支持的子类型列表，请参阅第 483 页的表 D-2。
attribute_value	指定属性类型所用的属性值。

表示目录条目更改的 LDIF 语法与上述语法有所不同。有关利用 LDIF 来修改目录条目的信息，请参阅第 2 章 “创建目录项”。

LDIF 中的连续行

指定 LDIF 时，通过将行的续接部分缩进一个空格，可以断开行并换行继续 （即折行）。例如，下列两个语句相同：

```
dn: cn=Jake Lupinski,dc=siroe,dc=com
```

```
dn: cn=Jake Lup
 inski, dc=sir
 oe,dc=comcom
```

这里并不是必须断开 LDIF 行然后换行续接。但这样做却有可能提高 LDIF 文件的可读性。

表示二进制数据

使用 Base 64 编码，可以在 LDIF 中表示二进制数据 （例如 JPEG 图像）。

使用基本 64 位编码方式

基本 64 位编码数据的标识方式是使用 `::` 符号。例如：

`jpegPhoto::` *encoded_data*

除了二进制数据外，其它必须为基本 64 位编码的值包括：

- 任何以分号 (;) 或空格开头的值
- 任何包含非 ASCII 数据的值 （包含新行）

将 `directoryserver ldif` 命令行实用程序与 `-b` 参数配合使用，可以将二进制数据转换为 LDIF 格式：

```
# /usr/sbin/directoryserver ldif -b attributeName
```

其中 *attribute_name* 是要接受二进制数据的属性的名称。二进制数据将从标准输入中读入，而结果将写入标准输出中。因此，您应使用重定向运算符来选择输入和输出文件。

此命令将接收所有输入，然后设定正确的行续接格式及相应的属性信息。它还要判定是否需要基本 64 编码。例如：

```
/usr/sbin/directoryserver ldif -b jpegPhoto < mark.jpg > out.ldif
```

本例将接收一个包含 JPEG 格式图像的二进制文件，然后将其转换为 LDIF 格式，属性名为 `jpegPhoto`。输出将被保存到 `out.ldif` 中。

`-b` 选项指定实用程序应将整个输入解释为单个二进制值。如果未指定 `-b`，则各行将被视为独立的输入值。

然后可以编辑输出文件，以添加创建或修改包含该二进制值的目录条目所需的 LDIF 语句。例如，可以在文本编辑器中打开文件 `out.ldif`，然后在文件开头添加以下行（以粗体显示）：

dn: cn=Barney Fife,ou=People,dc=siroe,dc=com
changetype: modify
add: jpegPhoto
jpegPhoto:: *encoded_data*

在此示例中， *encoded_data* 代表由命令生成的 `out.ldif` 文件的内容。

使用 LDIF 指定目录项

您可以存储多种类型的目录条目。本部分主要介绍三种最常见的目录项：组织、组织单元及组织人员条目。

为条目定义的对象类用于指示该条目是代表组织、组织单元、组织人员还是其它类型的条目。有关在目录中可以创建的目录条目类型的一般说明，请参阅 *iPlanet Directory Server 部署指南*。有关目录中可以默认使用的对象类完整列表及最常用属性的列表，请参阅 *iPlanet Directory Server 模式参考指南*。

指定组织条目

目录中通常至少有一个组织条目。典型情况下是目录中第一个 （即最上端的）条目。组织条目通常对应目录的后缀设置。例如，如果将目录定义为使用后缀 dc=siroe,dc=com，则目录中即可有一个名为 dc=siroe,dc=com 的组织条目。

被指定用于定义组织条目的 LDIF 其形式应为：

```
dn: distinguished_name
objectClass: top
objectClass: organization
o: organization_name
list_of_optional_attributes
...
```

下面是 LDIF 格式的组织条目示例：

```
dn: dc=siroe,dc=com
objectclass: top
objectclass: organization
o: siroe.com Corporation
description: Fictional company for example purposes
telephonenumber: 555-5555
```

下例中的组织名称使用逗号：

```
dn: o="siroe.com Chile\\, S.A."
objectclass: top
objectclass: organization
o: "siroe.com Chile\\, S.A."
description: Fictional company for example purposes
telephonenumber: 555-5556
```

对于 LDIF 格式的组织条目而言，其各个元素的定义见表 A-2。

表 A-2 组织条目中的 LDIF 元素

LDIF 元素	说明
dn: *distinguished_name*	指定条目的特异名称 (distinguished name)。 DN 的说明见 *iPlanet Directory Server 部署指南*。 DN 为必需项。
objectClass: top	必需项。指定 top 对象类。
objectClass: organization	指定 organization 对象类。此行将条目定义为组织。有关该对象类可以使用的属性列表，请参阅 *iPlanet Directory Server 模式参考指南*。

表 A-2 组织条目中的 LDIF 元素 *（续）*

LDIF 元素	说明
`o:` *organization_name*	该属性指定组织的名称。如果组织名称中包含逗号，则必须用反斜杠进行转义，且整个组织变量都必须用引号引起来。但是，如果使用 UNIX shell，则此反斜杠也需要进行转义，也就是说需要使用两个反斜杠。例如，要将后缀设置为 siroe.com Bolivia, S.A. 您应该键入 `"o: siroe.com Bolivia\\, S.A."`。
list_of_attributes	指定要为条目维护的可选属性列表。有关该对象类可以使用的属性列表，请参阅 *iPlanet Directory Server 模式参考指南*。

指定组织单元条目

组织单元条目通常用于表示目录树中的主分支点，即子目录。它们对应于公司中的主要 （一般为静态）实体 （例如包含人员的子树或包含组的子树）。但条目中包含的组织单元属性也可代表公司中的主要组织，例如营销或工程部。

目录树中通常有多个组织单元，或称分支点。有关如何设计目录树的信息，请参阅 *iPlanet Directory Server 部署指南*。

被指定用于定义组织单元条目的 LDIF 必须显示为：

```
dn: distinguished_name
objectClass: top
objectClass: organizationalUnit
ou: organizational_unit_name
list_of_optional_attributes
...
```

下面是 LDIF 格式的组织单元条目示例：

```
dn: ou=people, dc=siroe,dc=com
objectclass: top
objectclass: organizationalUnit
ou: people
description: Fictional organizational unit for example purposes
```

表 A-3 定义了 LDIF 格式的组织单元条目的各个元素。

表 A-3 组织单元条目中的 LDIF 元素

LDIF 元素	说明
`dn:` *distinguished_name*	指定条目的特异名称 (distinguished name)。 DN 为必需项。如果 DN 中有逗号，则逗号必须用反斜杠 (\) 进行转义。例如： `dn: ou=people,o=siroe.com Bolivia\,S.A.`
`objectClass: top`	必需项。指定 `top` 对象类。
`objectClass: organizationalUnit`	指定 `organizationalUnit` 对象类。此行将条目定义为组织单元。有关该对象类可以使用的属性列表，请参阅 *iPlanet Directory Server 模式参考指南*。
`ou:` *organizational_unit_name*	该属性指定组织单元的名称。
list_of_attributes	指定要为条目维护的可选属性列表。有关该对象类可以使用的属性列表，请参阅 *iPlanet Directory Server 模式参考指南*。

指定组织人员条目

目录中的大多数条目表示组织人员。

在 LDIF 中，组织人员的定义如下所示：

```
dn: distinguished_name
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: common_name
sn: surname
list_of_optional_attributes
```

下面是 LDIF 格式的组织人员条目的示例：

```
dn: uid=bjensen,ou=people,dc=siroe,dc=com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: Babs Jensen
sn: Jensen
```

```
givenname: Babs
uid: bjensen
ou: Marketing
ou: people
description: Fictional person for example purposes
telephonenumber: 555-5557
userpassword: {sha}dkfljlk34r2kljdsfk9
```

表 A-4 定义了 LDIF 人员条目的各个元素。

表 A-4 人员条目中的 LDIF 元素

LDIF 元素	说明
`dn:` *distinguished_name*	指定条目的特异名称 (distinguished name)。 DN 为必需项。如果 DN 中有逗号，则逗号必须用反斜杠 (\) 进行转义。例如：`dn:uid=bjensen,ou=people,o=siroe.com` `Bolivia\,S.A.`
`objectClass: top`	必需项。指定 `top` 对象类。
`objectClass: person`	指定 `person` 对象类。进行人员或组织人员的搜索操作期间，由于许多 LDAP 客户机都要求指定该对象类，因此应包含该指定。
`objectClass:` `organizationalPerson`	指定 `organizationalPerson` 对象类。进行组织人员的搜索操作期间，由于许多 LDAP 客户机都要求指定该对象类，因此应包含该指定。
`objectClass:` *inetOrgPerson*	指定 `inetOrgPerson` 对象类。由于 `inetOrgPerson` 对象类所含的属性范围最广，因此建议在创建组织人员时使用该对象类。该对象类需要使用 `uid` 属性，而包含该对象类的条目的命名是以 `uid` 属性值为基础的。有关该对象类可以使用的属性列表，请参阅 *iPlanet Directory Server 模式参考指南*。
`cn:` *common_name*	指定人员的通用名，通常是此人常用的全名。例如：`cn: Bill Anderson`。至少需要一个通用名。
`sn:` *surname*	指定人员的姓氏。例如 `sn: Anderson`。姓氏为必需项。
list_of_attributes	指定为条目维护的可选属性列表。有关该对象类可以使用的属性列表，请参阅 *iPlanet Directory Server 模式参考指南*。

使用 LDIF 定义目录

可以使用 LDIF 来定义整个目录的内容。如果需要向目录中添加许多条目，则使用 LDIF 来创建目录是一种较为有效的方法。

要使用 LDIF 创建目录，请执行下列步骤：

1. 创建 ASCII 文件，其中包含要以 LDIF 格式添加的条目。

   确保各个条目之间由空行分隔。应只使用单行，且文件中的第一行不得为空行。否则， `ldapmodify` 实用程序将退出。有关详细信息，请参阅第 448 页上的“使用 LDIF 指定目录项”。

2. 各个文件的开头应为数据库中最上端的条目 （即根条目）。

   根条目必须代表数据库中所含的后缀或子后缀。例如，如果数据库的后缀为 `dc=siroe,dc=com`，则目录中的第一个条目必须为：

   ```
   dn: dc=siroe,dc=com
   ```

   有关后缀的信息，请参阅 *iPlanet Directory Server 配置、命令和文件参考指南* 中所述的后缀参数内容。

3. 确保 LDIF 文件中代表分支点的条目位于要在该分支下创建的条目之前。

   例如，如果想将条目放到人员或组子树中，则在这些子树中创建条目之前，应首先创建这些子树的分支点。

4. 使用下列方法之一，利用 LDIF 文件创建目录：

   - iPlanet Directory Server Console

     如果有小的数据库要导入 （小于 1000 个条目），则使用该方法。请参阅第 131 页上的 “从控制台执行导入”。

   - `directoryserver ldif2db` 命令

     如果有大的数据库要导入 （大于 1000 个条目），请使用该方法。请参阅第 134 页上的 “使用 ldif2db 命令进行导入”。

   - 带有 `-a` 参数的 `ldapmodify` 命令行实用程序

     如果当前有目录数据库，但想向数据库中添加新的子树，则使用该方法。与其它利用 LDIF 文件创建目录的方法不同，使用 `ldapmodify` 添加子树之前必须运行 iPlanet Directory Server。请参阅第 53 页上的 “使用 ldapmodify 添加和修改条目”。

LDIF 文件示例

下例所示的 LDIF 文件中包含一个组织、两个组织单元及三个组织人员条目：

```
dn: o=siroe.com Corp,dc=siroe,dc=com
objectclass: top
objectclass: organization
o: siroe.com Corp
description: Fictional organization for example purposes

dn: ou=People,o=siroe.com Corp,dc=siroe,dc=com
objectclass: top
objectclass: organizationalUnit
ou: People
description: Fictional organizational unit for example purposes
tel: 555-5559

dn: cn=June Rossi,ou=People,o=siroe.com Corp,dc=siroe,dc=com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: June Rossi
sn: Rossi
givenName: June
mail: rossi@siroe.com
userPassword: {sha}KDIE3AL9DK
ou: Accounting
ou: people
telephoneNumber: 2616
roomNumber: 220

dn: cn=Marc Chambers,ou=People,o=siroe.com Corp,dc=siroe,dc=com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Marc Chambers
sn: Chambers
givenName: Marc
mail: chambers@siroe.com
userPassword: {sha}jdl2alem87dlacz1
telephoneNumber: 2652
ou: Manufacturing
ou: People
roomNumber: 167
```

```
dn: cn=Robert Wong,ou=People,o=siroe.com Corp,dc=siroe,dc=com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Robert Wong
cn: Bob Wong
sn: Wong
givenName: Robert
givenName: Bob
mail: bwong@siroe.com
userPassword: {sha}nn2msx761
telephoneNumber: 2881
roomNumber: 211
ou: Manufacturing
ou: people

dn: ou=Groups,o=siroe.com Corp,dc=siroe,dc=com
objectclass: top
objectclass: organizationalUnit
ou: groups
description: Fictional organizational unit for example purposes
```

存储多语种信息

如果目录中只有一种语言，则向目录中添加新条目无须执行任何特殊的操作。但如果您的组织为跨国机构，就可能会觉得有必要存储多语种信息，从而使不同地区的用户都能以自己的语种查看目录信息。

当目录中的信息以多语种表示时，服务器就会建立语言标记与属性值的关联关系。添加新条目时，必须提供 RDN （相对特异名称）中所用的属性值，无需任何语言代码。

您甚至可以将多个语种存储到单个属性中。此时，属性类型将相同，但各个值都有不同的语言代码。

有关受 iPlanet Directory Server 支持的语言列表及其关联的语言标记，请参阅第 481 页上的 “识别受支持的区域设置”。

注意 语言标记不会影响目录中字符串的存储方式。所有对象类和属性字符串均以 UTF-8 进行编码。

例如，假设 siroe.com Corporation 在美国及法国都有办事处，且希望员工能以自己的本国语言查看目录信息。添加目录条目时，目录管理员会选择提供英语和法语的属性值。添加新员工 Babs Jensen 的目录项时，管理员会创建下列 LDIF 条目：

```
dn: uid=bjensen,ou=people,dc=siroe,dc=com
objectclass: top
objectclass: person
objectclass: organizationalPerson
name: Babs Jensen
cn: Babs Jensen
sn: Jensen
uid: bjensen
streetAddress: 1 University Street
streetAddress;lang-en: 1 University Street
streetAddress;lang-fr: 1\, rue de l'Université
preferredLanguage: fr
```

利用 LDAP 客户机，并且以英语作为语言首选项的客户在访问该目录项时将会看到地址 `1 University Street`。而利用 LDAP 客户机，并且以法语作为语言首选项的客户在访问该目录项时将会看到地址 `1 rue de l'Université`。

查找目录条目

利用任何 LDAP 客户机都可以查找目录中的条目。多数客户机都提供一些搜索界面，用于便捷地搜索目录及检索条目信息。

注意 您或您的管理员在目录中设置的访问控制确定了搜索的结果。普通用户一般“看”不到很多目录内容，但目录管理员拥有访问所有数据（包括配置）的完全访问权限。

有关在目录中设置访问控制的信息，请参阅第 177 页上的“管理访问控制”。

本章涉及以下主题：

- 使用服务器控制台查找条目
- LDAP 搜索过滤器
- 使用 ldapsearch
- 搜索国际化目录

使用服务器控制台查找条目

使用 iPlanet Directory Server Console 的“目录”选项卡可以浏览目录树的内容及搜索目录中的特定条目。

1. 确保 Directory Server 处于运行状态。
2. 启动 iPlanet Directory Server Console。

   有关具体说明，请参阅第 26 页上的“启动 iPlanet Directory Server Console”。

3. 在 iPlanet Directory Server Console 上，选择 “目录”选项卡。

   根据进行目录验证时所用的 DN，该选项卡将显示享有查看权的目录内容。您可以浏览目录树的内容，或者右键单击某个条目，然后从弹出菜单中选择 “搜索”。

4. “搜索”对话框提供在目录中查找名称的简单界面。调用该对话框时，它可以从所选目录的节点执行搜索。从目录最高级进行的搜索范围较大，而从较低级的子树进行的搜索速度较快。

   高级搜索可以将搜索精确到某些属性以及它们的值。如果想用自己的 LDAP 过滤字符串进行搜索，还可以使用过滤搜索。

   有关该功能的更多信息，请参阅在线帮助。

注意 用 “搜索”对话框执行的搜索*不*遵循引荐。

5. 如果要查看或编辑搜索返回的条目，请单击 “确定”关闭 “搜索”对话框。搜索结果将显示在另外一个窗口，在该窗口中双击任何名称将显示它的完整条目。如果访问控制允许，条目将显示在 “编辑”对话框中并且可以对其进行修改。

   否则，单击 “取消”关闭 “搜索”对话框而不显示更多的搜索结果。

使用 ldapsearch

您可以使用 `ldapsearch` 命令行实用程序查找和检索目录项。该实用程序将利用指定的特异名称和口令打开到指定服务器的连接，然后根据指定的搜索过滤器查找条目。搜索范围中可以包括单个条目、紧跟条目下一级的子树，或者是整个目录树或子树。

本节提供有关下列主题的信息：

- 使用特殊字符
- ldapsearch 命令行格式
- 常用 ldapsearch 选项
- ldapsearch 示例

使用特殊字符

使用 `ldapsearch` 命令行实用程序时，指定的值中可能需要包含对命令行解释程序有特殊含义的字符 （例如空格 [ ]、星号 [*]、反斜杠 [\] 等）。指定特殊字符时，应使用引号 (“”) 将其括起来。例如：

```
-D "cn=Barbara Jensen,ou=Product Development,dc=siroe,dc=com"
```

根据使用的命令行解释程序，有时使用单引号，而有时则使用双引号。有关详细信息，请参阅操作系统文档。

ldapsearch 命令行格式

使用 `ldapsearch` 时，必须输入下列格式的命令：

`ldapsearch` [*optional_options*] [*optional_search_filter*] [*optional_list_of_attributes*]

其中

- *optional_options* 表示一系列命令行选项。如果有，则必须在搜索过滤器之前指定。
- *optional_search_filter* 表示第 462 页上的 “LDAP 搜索过滤器”中所述的 LDAP 搜索过滤器。如果是在文件中利用 `-f` 选项提供搜索过滤器，则不要指定搜索过滤器。
- *optional_list_of_attributes* 表示由空格分隔的属性列表。指定属性列表可以减少搜索结果中返回的属性数。该属性列表必须位于搜索过滤器的后面。有关示例，请参阅第 462 页上的 “显示属性的子集”。如果不指定属性列表，搜索过程就会返回目录中设置的访问控制所允许的所有属性值 （操作属性除外）。

注意 如果希望操作属性也作为搜索操作的结果而予以返回，则必须在搜索命令中进行明确指定。除了明确指定的操作属性，如果还想检索常规属性，请在 `ldapsearch` 命令的属性列表中使用星号 (*)。

常用 ldapsearch 选项

下面列出了最常用的 ldapsearch 命令行选项。如果指定的值中包含空格 [ ]，则该值应使用双引号括起来 （例如 `-b "ou=groups, dc=siroe,dc=com"`）。

`-b` 指定搜索的起始点。这里指定的值必须为数据库中当前存在的特异名称。如果不指定值，则使用 `""`。

该选项中指定的值应括到双引号中。例如：

```
-b "cn=Barbara Jensen, ou=Product Development,
dc=siroe,dc=com".
```

如果希望搜索根 DSE 条目，请在这里指定空字符串。例如：

```
-b ""
```

`-D` 指定进行服务器验证时所用的特异名称。如果服务器支持匿名访问，则该选项将成为可选项。如果指定该选项，则值必须为可被 Directory Server 识别的 DN，且须有搜索该条目的授权。例如：

```
-D "uid=bjensen, dc=siroe,dc=com".
```

`-h` 指定安装 Directory Server 的计算机的主机名和 IP 地址。如果不指定主机， `ldapsearch` 就会使用 localhost。例如 `-h mozilla`。

`-l` 指定完成搜索请求前所等待的最长时间 （秒）。不管这里指定什么值，`ldapsearch` 的等待时间都不会超出服务器 `nsslapd-timelimit` 属性所允许的时间。例如 `-l 300`。`nsslapd-timelimit` 属性的默认值为 3,600 秒。

`-p` 指定 Directory Server 所用的 TCP 端口号。例如 `-p 1049`。默认值为 389。

`-s` 指定搜索范围。范围可以是下列之一：

- `base` — 仅搜索 `-b` 选项中指定的条目。
- `one` — 仅搜索 `-b` 选项中所指定条目的下一级子项。此时将仅搜索子项，而不会搜索 `-b` 选项中指定的实际条目。
- `sub` — 搜索 `-b` 选项中指定的条目及所有子项。也就是从 `-b` 选项所标识的点开始进行子树搜索。此为缺省值。

`-w` 指定与 `-D` 选项中所指定的特异名称相关联的口令。如果不指定该选项，则使用匿名访问。例如 `-w diner892`。

`-z` 指定对应于搜索请求所返回的最大条目数。例如 `-z 1000`。

正常情况下，不管这里指定什么值， `ldapsearch` 所返回的条目数都不会超出服务器 `nsslapd-sizelimit` 属性所允许的数目。但在使用该命令行变量时，如果绑定为根 DN，即可忽略此限制条件。绑定为根 DN 时，该选项默认为 0。 `nsslapd-sizelimit` 属性的默认值为 2,000 个条目。

ldapsearch 示例

在下一组示例中，假设下列条件为真：

- 要搜索目录中的全部条目。
- 目录已配置为支持匿名搜索和读取访问。执行搜索时，无须指定任何绑定信息。有关匿名访问的详细信息，请参阅第 195 页上的 “定义用户访问权 — userdn 关键字”。
- 服务器位于主机名为 **`mozilla`** 的主机上。
- 服务器使用端口号 **`389`**。由于这是默认端口，因此不必标识搜索请求上的端口号。
- 用于存储所有数据的后缀为 **`dc=siroe,dc=com`**。

返回所有条目

在给定上述信息的情况下，下列调用将返回目录中的所有条目：

```
ldapsearch -h mozilla -b "dc=siroe,dc=com" -s sub "objectclass=*"
```

`"objectclass=*"` 是匹配目录中所有条目的搜索过滤器。

在命令行上指定搜索过滤器

在命令行上可以直接指定搜索过滤器。此时，请务必将过滤器括在引号中 （“过滤器”）。同时，不要指定 `-f` 选项。例如：

```
ldapsearch -h mozilla -b "dc=siroe,dc=com" "cn=babs jensen"
```

搜索根 DSE 条目

根 DSE 是一个特殊的条目，其中包含所有受本地目录服务器支持的后缀列表。搜索该条目时，应提供搜索基 `""`。同时，还必须指定搜索范围为 `base`，过滤器为 `"objectclass=*"`。例如：

```
ldapsearch -h mozilla -b "" -s base "objectclass=*"
```

搜索模式条目

iPlanet Directory Server 将在特定的 cn=schema 条目中存储所有目录服务器模式。该条目中包含有关为目录服务器所定义各对象类和属性的信息。

可以按如下所示检查该条目的内容：

```
ldapsearch -h mozilla -b "cn=schema" -s base "objectclass=*"
```

显示属性的子集

ldapsearch 命令可以返回所有 LDIF 格式的搜索结果。默认情况下， ldapsearch 将返回条目的特异名称及所有允许读取的属性 （可以对目录的访问控制进行设置，从而允许以只读方式访问任何给定目录项上属性的子集）。只有操作属性不予以返回。如果希望操作属性也作为搜索操作的结果而予以返回，则必须在搜索命令中进行明确指定。

假设您不想看到搜索结果中返回的所有属性。这时可以将返回的属性限定为几个特定属性，方法是在命令行中紧跟在搜索过滤器的后面指定所需的属性。例如，要显示目录中每个条目的 cn 和 sn 属性，请使用下列命令行调用：

```
ldapsearch -h mozilla -b "dc=siroe,dc=com" "objectclass=*" sn cn
```

在搜索过滤器中指定包含逗号的 DN

当搜索过滤器中的 DN 值中包含逗号时，必须使用反斜杠 (\) 对逗号进行转义。例如，要查找 siroe.com Bolivia, S.A. 子树中的所有人员，请使用下列命令：

```
ldapsearch -h mozilla -s base -b "o=siroe.com Bolivia\, S.A.,dc=siroe,dc=com"
"objectclass=*"
```

LDAP 搜索过滤器

搜索过滤器将为搜索操作选择所要返回的条目。它们通常与 ldapsearch 命令行实用程序一起使用。使用 ldapsearch 时，可以将多个搜索过滤器放到一个文件中，每个过滤器占独立的一行，也可以在命令行上直接指定搜索过滤器。

例如，下列过滤器指定搜索通用名 Babs Jensen：

```
cn=babs jensen
```

该搜索过滤器将返回所有包含通用名 Babs Jensen 的条目。搜索通用名时并不区分大小写。

如果通用名属性的值与语言标记关联，则返回所有的值。因此，下列两个属性值均匹配该过滤器：

```
cn: babs jensen
cn;lang-fr: babs jensen
```

有关支持的语言列表，请参阅第 481 页的表 D-1。

搜索过滤器语法

搜索过滤器的基本语法为：

attribute operator value

例如：

```
buildingname>=alpha
```

本例中，`buildingname` 为属性，`>=` 为运算符，而 **`alpha`** 为属性值。也可将过滤器定义为使用由布尔运算符组合到一起的不同属性。

有关搜索过滤器的详细信息，请参阅以下各节：

- 在搜索过滤器中使用属性
- 在搜索过滤器中使用运算符
- 使用复合搜索过滤器
- 搜索过滤器示例

在搜索过滤器中使用属性

搜索条目时，可以指定与该条目相关联的属性。例如，搜索人员条目时，可以使用 `cn` 属性来搜索具有特定通用名的人。

人员条目中可能包含的属性示例：

- `cn`（人员的通用名）
- `sn`（人员的姓氏）
- `telephoneNumber`（人员的电话号码）
- `buildingName`（人员的住址）
- `l`（可以找到此人的地方）

有关与条目类型关联的属性列表，请参阅 *iPlanet Directory Server 模式参考指南*。

在搜索过滤器中使用运算符

搜索过滤器中可用的运算符列表，请参见表 B-1：

表 B-1 搜索过滤器运算符

搜索类型	运算符	说明
等同	=	返回所含属性值与指定的值完全匹配的条目。例如 `cn=Bob Johnson`

子字符串	=*string** *string*	返回所含属性中包含指定子字符串的条目。例如：
		`cn=Bob*`
		`cn=*Johnson`
		`cn=*John*`
		`cn=B*John`
		（星号 (*) 指示 0 个或更多字符。）

大于或等于	>=	返回所含属性大于或等于指定值的条目。例如： `buildingname >= alpha`
小于或等于	<=	返回所含属性小于或等于指定值的条目。例如： `buildingname <= alpha`

存在	=*	返回包含所指定属性的一个或多个值的条目。例如：
		`cn=*`
		`telephonenumber=*`
		`manager=*`

近似	~=	返回的条目中所含的指定属性的值近似等于搜索过滤器中指定的值。例如：
		`cn~=suret`
		`l~=san fransico`
		将返回
		`cn=sarette`
		`l=san francisco`

注意 除了上述搜索过滤器外，还可以指定特殊的过滤器来与首选语言对照顺序配合使用。有关如何用国际字符集来搜索目录的信息，请参阅第 466 页上的 “搜索国际化目录”。

使用复合搜索过滤器

如下所示，利用前缀记号中的布尔运算符可以将多个搜索过滤器组件组合到一起：

(*Boolean-operator* (*filter*) (*filter*) (*filter*) . . .)

其中 *Boolean-operator* 是表 B-2 中所列的任意布尔运算符。

布尔运算符可以组合和嵌套起来，从而构成复杂的表达式，例如：

(*Boolean-operator* (*filter*) ((*Boolean-operator* (*filter*) (*filter*)))

可用于搜索过滤器的的布尔运算符包括：

表 B-2 搜索过滤器布尔运算符

运算符	符号	说明
AND	&	所有指定的过滤器都必须为真时，该语句才为真。例如： `(&(filter)(filter)(filter)...)`
OR	|	至少指定的一个过滤器为真时，该语句即为真。例如： `(|(filter)(filter)(filter)...)`
NOT	!	指定的语句必须非真时，该语句才为真。只有一个过滤器受 NOT 运算符的影响。例如： `(!(filter))`

布尔表达式的评估顺序如下所示：

- 首先是最内侧的括号表达式，并依次向外
- 所有表达式都按从左向右的顺序进行

搜索过滤器示例

下列过滤器将搜索包含一个或多个 manager 属性值的条目。这也称为存在搜索：

```
manager=*
```

下列过滤器将搜索包含通用名 Ray Kultgen 的条目。这也称为等价搜索：

```
cn=Ray Kultgen
```

下列过滤器返回所有不包含通用名 Ray Kultgen 的条目：

```
(!(cn=Ray Kultgen))
```

下列过滤器返回的所有条目中都有包含子字符串 `X.500` 的说明属性：

```
description=*X.500*
```

下列过滤器返回所有组织单元为 Marketing 且说明字段中不包含子字符串 `X.500` 的条目：

```
(&(ou=Marketing)(!(description=*X.500*)))
```

下列过滤器返回所有组织单元为 Marketing 且 manager 为 Julie Fulmer 或 Cindy Zwaska 的条目：

```
(&(ou=Marketing)(|(manager=cn=Julie
Fulmer,ou=Marketing,dc=siroe,dc=com)(manager=cn=Cindy
Zwaska,ou=Marketing,dc=siroe,dc=com)))
```

下列过滤器返回所有不代表人员的条目：

```
(!(objectClass=person))
```

下列过滤器返回所有不代表人员且通用名近似于 `printer3b` 的条目：

```
(&(!(objectClass=person))(cn~=printer3b))
```

搜索国际化目录

执行搜索操作时，对于服务器具有支持的对照顺序的语言而言，可以请求目录根据该语言对结果进行排序。有关目录所支持的对照顺序的列表，请参阅第 481 页上的“识别受支持的区域设置”。

本节主要说明 `ldapsearch` 语法的匹配规则过滤器部分。有关一般 `ldapsearch` 语法的详细信息，请参阅第 462 页上的 “LDAP 搜索过滤器”。有关利用 iPlanet Console “用户和组”部分来搜索国际化目录的信息，请参阅在线帮助或*通过 iPlanet Console 管理服务器*。

该部分包含下列主题：

- 匹配规则过滤器语法
- 支持的搜索类型
- 国际搜索示例

匹配规则过滤器语法

匹配规则可在搜索操作期间为目录进行字符串比较提供特殊的标准。在国际化搜索中，匹配规则将告知系统在搜索过程中使用什么对照顺序和运算符。例如在国际搜索中，匹配规则可能会告知服务器搜索位于西班牙语对照顺序中 llama 或其后面的属性值。匹配规则过滤器的语法如下所示：

attr:*matchingRule*:=*value*

其中：

- *attr* 是属于所要搜索的条目的属性，例如 `cn` 或 `mail`
- *matchingRule* 是标识对照顺序或对照顺序与关系运算符的标识符，这与自己所喜欢的格式有关。有关匹配规则格式的说明，请参阅第 467 页上的 “匹配规则格式”。
- *value* 是所要搜索的属性值，或者是关系运算符和所要搜索的属性值。过滤器值部分的语法与所用的匹配规则格式有关。

匹配规则格式

搜索过滤器的匹配规则部分可以用几种方式来表示。具体使用哪种方式纯粹是个人喜好问题。表示匹配规则的方式包含：

- 对于要用作搜索基的区域设置而言，作为对照顺序的 OID。
- 对于要用作搜索基的对照顺序而言，作为与对照顺序关联的语言标记。
- 作为代表关系运算符的对照顺序和后缀的 OID。
- 对于代表关系运算符的对照顺序和后缀而言，作为与之关联的语言标记。

上述各种选项的语法将在以下各节进行讨论：

- 在匹配规则中使用 OID
- 在匹配规则中使用语言标记
- 在匹配规则中使用 OID 和后缀
- 在匹配规则中使用语言标记和后缀

在匹配规则中使用 OID

目录服务器所支持的各种区域设置都有关联的对照顺序 OID。有关目录服务器所指定的区域设置列表及其相关的 OID，请参阅第 481 页的表 D-1。

如下所示，可以使用匹配规则过滤器中匹配规则部分的对照顺序 OID：

attr:*OID*:=(*relational_operator value*)

关系运算符包含在字符串的值部分，与值之间用空格分隔。例如，要搜索按瑞典语对照顺序位于 N4709 或之后的所有 `departmentNumber` 属性，请使用下列过滤器：

```
departmentNumber:2.16.840.1.113730.3.3.2.46.1:=>= N4709
```

在匹配规则中使用语言标记

目录服务器所支持的各种区域设置都有关联的语言标记。有关目录服务器所支持的区域设置及相关的语言标记，请参阅第 481 页的表 D-1。

如下所示，可以在匹配规则过滤器的匹配规则部分使用语言标记：

attr:*language-tag*:=(*relational_operator value*)

关系运算符包含在字符串的值部分，与值之间用空格分隔。例如，要利用西班牙语对照顺序搜索目录中所有值为 `estudiante` 的说明属性，请使用下列过滤器：

```
cn:es:== estudiante
```

在匹配规则中使用 OID 和后缀

作为使用关系运算符 – 值对的替代方案，也可以将代表特定运算符的后缀追加到过滤器匹配规则部分 OID 上。将 OID 和后缀组合如下：

attr:*OID*+*suffix*:=*value*

例如，要按德语对照顺序来搜索值为 `softwareproduckte` 的 `businessCategory` 属性，请使用下列过滤器：

```
businessCategory:2.16.840.1.113730.3.3.2.7.1.3:=softwareprodukte
```

上例中的 `.3` 为等价后缀。

有关目录服务器所指定的区域设置列表及其相关的 OID，请参阅第 481 页的表 D-1。有关关系运算符及其等价后缀的列表，请参阅第 470 页的表 B-3。

在匹配规则中使用语言标记和后缀

作为使用关系运算符 – 值对的替代方案，也可以将代表特定运算符的后缀追加到过滤器匹配规则部分的语言标记上。将语言标记和后缀组合如下：

attr:*language-tag*+*suffix*:=*value*

例如，要搜索按法语对照顺序位于 `La Salle` 或之后的所有姓氏，请使用下列过滤器：

```
sn:fr.4:=La Salle
```

有关目录服务器所支持的区域设置及相关的语言标记，请参阅第 481 页的表 D-1。有关关系运算符及其等价后缀的列表，请参阅第 470 页的表 B-3。

在匹配规则过滤器中使用通配符

使用匹配规则过滤器执行子字符串搜索时，可以将星号 (*) 字符用作通配符，来代表 0 个或多个字符。

例如，要搜索以字母 `l` 开头，以字母 `n` 结尾的属性值，可在搜索过滤器的值部分输入 `l*n`。与此类似，要搜索所有以字母 `u` 开头的属性值，请在搜索过滤器的值部分输入值 `u*`。

要搜索包含星号 (*) 字符的值，则必须用指定的转义字符 (\) 对 * 进行转义。例如，要搜索所有属性值 `businessCategory` 为 `Siroe*Net product line` 的员工，请在搜索过滤器中输入下列值：

```
Siroe\*Net product line
```

支持的搜索类型

Directory Server 执行下列类型的国际搜索：

- 等式 (=)
- 子字符串 (*)
- 大于 (>)
- 大于或等于 (>=)
- 小于 (<)
- 小于或等于 (<=)

近似 （或语音）和存在搜索仅在英语中得到支持。

与常规 ldapsearch 搜索操作类似，国际搜索也使用运算符来定义搜索类型。但调用国际搜索功能时，既可在搜索字符串的值部分使用标准运算符 (=, >=, >, <, <=)，也可在过滤器的匹配规则部分使用特殊的运算符，称为后缀 （不要与目录后缀混淆）。表 B-3 总结了各类搜索操作、运算符及等价后缀。

表 B-3 搜索类型、运算符和后缀

搜索类型	运算符	后缀
小于	<	.1
小于或等于	<=	.2
等同	=	.3
大于或等于	>=	.4
大于	>	.5
子字符串	*	.6

国际搜索示例

下列各节给出如何对目录数据执行国际搜索的示例。每个示例都给出了可能的匹配规则过滤器格式，这样可以帮助您熟悉这些格式并选择最适合自己的格式。

小于示例

使用小于运算符 (<) 或后缀 (.1) 执行面向特定区域设置的搜索时，将搜索按特定对照顺序而言位于给定属性之前的所有属性值。

例如，要搜索按西班牙语对照顺序位于姓氏 Marquez 之前的所有姓氏，可以使用下列匹配规则过滤器之一：

```
sn:2.16.840.1.113730.3.3.2.15.1:=< Marquez
sn:es:=< Marquez
sn:2.16.840.1.113730.3.3.2.15.1.1:=Marquez
sn:es.1:=Marquez
```

小于或等于示例

使用小于或等于运算符 (<=) 或后缀 (.2) 执行面向特定区域设置的搜索时，将搜索按特定对照顺序而言位于给定属性处或之前的所有属性值。

例如，要搜索按匈牙利语对照顺序位于房间号 CZ422 或之前的所有房间号，请使用下列匹配规则过滤器之一：

```
roomNumber:2.16.840.1.113730.3.3.2.23.1:=<= CZ422
roomNumber:hu:=<= CZ422
roomNumber:2.16.840.1.113730.3.3.2.23.1.2:=CZ422
roomNumber:hu.2:=CZ422
```

等式示例

使用等于运算符 (=) 或后缀 (.3) 执行面向特定区域设置的搜索时，将搜索按特定对照顺序而言匹配给定属性的所有属性值。

例如，要搜索按德语对照顺序而言所有属性值为 `softwareprodukte` 的 `businessCategory` 属性，可以使用下列匹配规则过滤器之一：

```
businessCategory:2.16.840.1.113730.3.3.2.7.1:== softwareprodukte
businessCategory:de:== softwareprodukte
businessCategory:2.16.840.1.113730.3.3.2.7.1.3:=softwareprodukte
businessCategory:de.3:=softwareprodukte
```

大于或等于示例

使用大于或等于运算符 (>=) 或后缀 (.4) 执行面向特定区域设置的搜索时，将搜索按特定对照顺序而言位于给定属性处或之后的所有属性值。

例如，要搜索按法语对照顺序而言所有位于 Québec 或之后的属性，可以使用下列匹配规则过滤器之一：

```
locality:2.16.840.1.113730.3.3.2.18.1:=>= Québec
locality:fr:=>= Québec
locality:2.16.840.1.113730.3.3.2.18.1.4:=Québec
locality:fr.4:=Québec
```

大于示例

使用大于运算符 (>) 或后缀 (.5) 执行面向特定区域设置的搜索时，将搜索按特定对照顺序而言位于给定属性之后的所有属性值。

例如，要搜索按捷克斯洛伐克语所有位于主机 schranka4 后面的邮件主机，可以使用下列匹配规则过滤器之一：

```
mailHost:2.16.840.1.113730.3.3.2.5.1:=> schranka4
mailHost:cs:=> schranka4
mailHost:2.16.840.1.113730.3.3.2.5.1.5:=schranka4
mailHost:cs.5:=schranka4
```

子字符串示例

执行国际子字符串搜索时，将搜索按特定对照顺序而言匹配给定形式的所有属性值。

例如，要搜索按汉语对照顺序所有以 ming 结尾的用户代号，可以使用下列匹配规则过滤器之一：

```
uid:2.16.840.1.113730.3.3.2.49.1:=* *ming
uid:zh:=* *ming
uid:2.16.840.1.113730.3.3.2.49.1.6:=* *ming
uid:zh.6:=* *ming
```

附录 C

LDAP URL

一种表示 LDAP 查询的方法，该方法使用 URL 指定目录服务器主机和搜索的 DN 或过滤器。 iPlanet Directory Server 将响应以 LDAP URL 方式发送的查询，并返回一个 HTML 页面说明查询结果。在允许匿名搜索的情况下，该方法可以让 web 浏览器执行目录搜索。

管理 iPlanet Directory Server 引荐或访问控制指令时，也可以使用 LDAP URL 来指定目标项。

本附录包含以下几部分：

- LDAP URL 的组件
- 对非安全字符进行转义
- LDAP URL 示例

LDAP URL 的组件

LDAP URL 的语法如下所示：

`ldap[s]://`*hostname*`:`*port*`/`*base_dn*`?`*attributes*`?`*scope*`?`*filter*

其中 `ldap://` 协议用于通过非安全连接来连接 LDAP 服务器，而 `ldaps://` 协议则用于通过 SSL 连接来连接 LDAP 服务器。表 C-1 列出 LDAP URL 的组件。

表 C-1 LDAP URL 组件

组件	说明
hostname	LDAP 服务器的名称 （或用点分隔的 IP 地址）。例如： `ldap.siroe.com` 或 `192.202.185.90`
port	LDAP 服务器的端口号 （例如 696）。 如果未指定端口，则使用标准 LDAP 端口 (389) 或 LDAPS 端口 (636)。
base_dn	目录中条目的特异名称 (DN)。此 DN 将标识作为搜索起始点的条目。 如果未指定基本 DN，搜索过程就会从目录树的根开始。
attributes	所要返回的属性。要指定多个属性，请使用逗号来分隔属性 （例如 `"cn,mail,telephoneNumber"`）。 如果 URL 中未指定属性，则返回所有属性。
scope	搜索范围，可以取下列值之一： • `base` 将仅检索有关 URL 中所指定的特异名称 (*base_dn*) 的信息。 • `one` 将检索有关 URL 中所指定的特异名称 (*base_dn*) 之下一级条目的信息。基本条目并不包含在该范围内。 • `sub` 将检索有关 URL 中所指定的特异名称 (*base_dn*) 下属各级条目的信息。基本条目包含在该范围内。 如果未指定范围，服务器就会执行 `base` 搜索。
filter	应用于指定搜索范围内条目的搜索过滤器。 如果未指定过滤器，服务器就会使用过滤器 `(objectClass=*)`。

属性、范围和过滤器组件分别由它们在 URL 中的位置进行标识。如果不想指定任何属性，您仍需要使用问号来分隔界定该字段。

例如，要指定从 `"dc=siroe,dc=com"` 开始、返回匹配 `"(sn=Jensen)"` 之条目所有属性的子树搜索，请使用下列 LDAP URL：

```
ldap://ldap.siroe.com/dc=siroe,dc=com??sub?(sn=Jensen)
```

两个连续的问号 `??` 指示未指定属性。由于 URL 中未指定具体的属性，因此将返回搜索到的所有属性。

对非安全字符进行转义

URL 中的所有 “非安全”字符都需要用特定的字符序列来表示。该过程称为 “非安全字符的转义”。

例如，空格就是一个非安全字符，在 URL 必须表示为 `%20`。这样，特异名称 `"o=siroe.com corporation"` 必须编码为 `"o=siroe.com%20corporation"`。

下表列出在 URL 中被视为非安全的字符，同时提供用于替代非安全字符的相关转义字符。

非安全字符	转义字符
空格	%20
<	%3c
>	%3e
"	%22
#	%23
%	%25
{	%7b
}	%7d
|	%7c
\	%5c
^	%5e
~	%7e
[	%5b
]	%5d
'	%60

LDAP URL 示例

- 下列 LDAP URL 指定对带有特异名称 dc=siroe,dc=com 的条目执行基本搜索。

```
ldap://ldap.siroe.com/dc=siroe,dc=com
```

  - 由于未指定端口号，因此将使用标准 LDAP 端口号 (389)。
  - 由于未指定属性，因此搜索过程将返回所有属性。
  - 由于未指定搜索范围，因此搜索过程将被限定在基本条目 dc=siroe, dc=com 上。
  - 由于未指定过滤器，因此目录将使用默认过滤器 (objectclass=*)。

- 下列 LDAP URL 将检索带有特异名称 dc=siroe,dc=com 的条目的 postalAddress 属性。

```
ldap://ldap.siroe.com/dc=siroe,dc=com?postalAddress
```

  - 由于未指定搜索范围，因此搜索过程将被限定在基本条目 dc=siroe, dc=com 上。
  - 由于未指定过滤器，因此目录将使用默认过滤器 (objectclass=*)。

- 下列 LDAP URL 将检索 Barbara Jensen 条目的 cn、mail 和 telephoneNumber 属性：

```
ldap://ldap.siroe.com/cn=Barbara%20Jensen,dc=siroe,dc=com?cn,mail,telephoneNumber
```

  - 由于未指定搜索范围，因此搜索过程将被限定在基本条目 cn=Barbara Jensen,dc=siroe,dc=com 上。
  - 由于未指定过滤器，因此目录将使用默认过滤器 (objectclass=*)。

- 下列 LDAP URL 指定搜索姓氏为 Jensen，且位于 dc=siroe,dc=com 下属某一级的条目：

```
ldap://ldap.siroe.com/dc=siroe,dc=com??sub?(sn=Jensen)
```

  - 由于未指定属性，因此搜索过程将返回所有属性。
  - 由于搜索范围是 sub，因此搜索过程将囊括基本条目 dc=siroe,dc=com 及基本条目下属各级上的条目。

- 下列 LDAP URL 指定搜索 dc=siroe,dc=com 下一级上所有条目的对象类：

```
ldap://ldap.siroe.com/dc=siroe,dc=com?objectClass?one
```

  - 由于搜索范围是 one，因此搜索过程将囊括基本条目 dc=siroe,dc=com 下一级上的所有条目。该搜索范围不包括基本条目。
  - 由于未指定过滤器，因此目录将使用默认过滤器 (objectclass=*)。

注意 LDAP URL 的语法不涉及指定凭证或口令。除非支持 LDAP URL 的 LDAP 客户机提供身份验证机制，否则由 LDAP URL 启动的搜索请求将不进行身份验证 （匿名）。

国际化

iPlanet Directory Server 允许用多种不同的语言来存储、管理和搜索条目及其关联的属性。国际化目录可以成为一笔无价的公司资源，允许员工和商业伙伴用自己所能理解的语言即时访问需要的信息。

由于目录数据是以 UTF-8 格式存储的，因此该目录默认支持所有国际字符集。此外， iPlanet Directory Server 允许根据语言首选项来指定搜索操作中的匹配规则及对照顺序。

注意 属性和对象类名称中必须使用 ASCII 字符。

本附录包含以下几部分：

- 关于区域设置
- 识别受支持的区域设置
- 受支持的语言子类型

关于区域设置

利用区域设置， iPlanet Directory Server 可支持多种语言。区域设置标识有关特定地区、文化和/或习惯的用户所希望的数据显示方式的特定语言信息，包括如何解释给定语言的数据及如何对数据进行排序或对照。

此外，区域设置信息还指示代表给定语言所用的代码页。代码页是一种供操作系统用来建立键盘按键与字符字体屏幕显示之间关系的内部表。

再具体的讲，区域设置指定以下内容：

- 对照顺序

  对照顺序提供有关如何对给定语言的字符进行排序的特定文化信息。它标识诸如字母表中的字母序列、如何对比带音质符号的字母与不带音质符号的字母，以及对比字符串时是否忽略某些字符等事宜。对照顺序还会考虑有关语言的特定文化信息，例如语言的阅读顺序（由左向右、由右向左还是由上向下）。

- 字符类型

  字符类型可将字母字符与数字或其它字符区别开来。此外，它还定义了大写字母向小写字母的映射。例如，在某些语言中，竖线 (|) 字符被视为标点符号，而在其它一些语言中则被视为字母。

- 货币格式

  货币格式指定了特定地区所用的货币符号，该符号是位于值的前面还是后面，以及如何表示货币单位。

- 时间和日期格式

  时间和日期格式指示当地时间和日期的惯用格式。时间格式指示当地是使用 12 小时时钟格式还是 24 小时时钟格式。日期格式包括短日期格式，例如 `MM/dd/yy`（月、日、年）或 `dd/MM/yy`（日、月、年），以及长日期格式（包含某种特定语言的月份和星期）。例如，日期“1996 年 1 月 10 日”在捷克语中表示为“10. leden 1996”，而在法语中则表示为“10 janvier 1996”。

由于区域设置除了机械的语言差异外还描述文化、习惯和区域性的差别，因此目录数据既可转换为用户所能理解的特定语言，也能用给定地区用户所希望的方式加以表示。

在 iPlanet Directory Server 安装过程中，区域设置信息将被自动复制到以下目录：

```
/usr/iplanet/ds5/lib/nls/locale31
```

识别受支持的区域设置

如果执行的目录操作 （例如搜索操作）要求指定区域设置，则可使用语言标记或对照顺序对象标识符 (OID)。

语言标记就是以两个标识语言的小写语言代码开头的字符串 （ISO 639 标准中定义）。必要时，为区分语言的区域差别，语言标记中还可能包括国家代码 – 两个字符的字符串 （ISO 3166 标准中定义）。语言代码和国家代码之间由破折号分隔。例如，用于标识英国英语区域设置的语言标记为 en-GB。

对象标识符 (OID) 则是用于唯一标识对象 （例如属性或对象类）的十进制数。搜索或索引国际化目录时所用的 OID 将标识 iPlanet Directory Server 所支持的特定对照顺序。例如， OID `2.16.840.1.113730.3.3.2.17.1` 代表芬兰语的对照顺序。

在目录中进行国际搜索时，请使用语言标记或 OID 来标识所要使用的对照顺序。但在设置国际索引时，必须使用 OID。有关索引的详细信息，请参阅第 10 章 “管理索引”。

下表列出 iPlanet Directory Server 所支持的各种区域设置，同时说明相关的语言标记及 OID。

表 D-1 受支持的区域设置

区域设置	语言标记	对照顺序对象标识符 (OID)
阿尔巴尼亚语	sq	2.16.840.1.113730.3.3.2.44.1
阿拉伯语	ar	2.16.840.1.113730.3.3.2.1.1
白俄罗斯语	be	2.16.840.1.113730.3.3.2.2.1
保加利亚语	bg	2.16.840.1.113730.3.3.2.3.1
加泰罗尼亚语	ca	2.16.840.1.113730.3.3.2.4.1
汉语 （简体）	zh	2.16.840.1.113730.3.3.2.49.1
汉语 （繁体）	zh-TW	2.16.840.1.113730.3.3.2.50.1
克罗地亚语	hr	2.16.840.1.113730.3.3.2.22.1
捷克斯洛伐克语	cs	2.16.840.1.113730.3.3.2.5.1
丹麦语	da	2.16.840.1.113730.3.3.2.6.1
英语 （美国）	en 或 en-US	2.16.840.1.113730.3.3.2.11.1
爱沙尼亚语	et	2.16.840.1.113730.3.3.2.16.1
芬兰语	fi	2.16.840.1.113730.3.3.2.17.1
法语	fr 或 fr-FR	2.16.840.1.113730.3.3.2.18.1
德语	de	2.16.840.1.113730.3.3.2.7.1

表 D-1 受支持的区域设置 *（续）*

区域设置	语言标记	对照顺序对象标识符 (OID)
希腊语	el	2.16.840.1.113730.3.3.2.10.1
希伯来语	iw	2.16.840.1.113730.3.3.2.27.1
匈牙利语	hu	2.16.840.1.113730.3.3.2.23.1
冰岛语	is	2.16.840.1.113730.3.3.2.24.1
日语	ja	2.16.840.1.113730.3.3.2.28.1
朝鲜语	ko	2.16.840.1.113730.3.3.2.29.1
拉脱维亚语、列托语	lv	2.16.840.1.113730.3.3.2.31.1
立陶宛语	lt	2.16.840.1.113730.3.3.2.30.1
马其顿语	mk	2.16.840.1.113730.3.3.2.32.1
挪威语	no	2.16.840.1.113730.3.3.2.35.1
波兰语	pl	2.16.840.1.113730.3.3.2.38.1
罗马尼亚语	ro	2.16.840.1.113730.3.3.2.39.1
俄语	ru	2.16.840.1.113730.3.3.2.40.1
塞尔维亚语（西里尔语）	sr	2.16.840.1.113730.3.3.2.45.1
塞尔维亚语（拉丁语）	sh	2.16.840.1.113730.3.3.2.41.1
斯洛伐克语	sk	2.16.840.1.113730.3.3.2.42.1
斯洛文尼亚语	sl	2.16.840.1.113730.3.3.2.43.1
西班牙语	es 或 es-ES	2.16.840.1.113730.3.3.2.15.1
瑞典语	sv	2.16.840.1.113730.3.3.2.46.1
土耳其语	tr	2.16.840.1.113730.3.3.2.47.1
乌克兰语	uk	2.16.840.1.113730.3.3.2.48.1

受支持的语言子类型

客户机可以使用语言子类型来决定所要搜索对象的特定值。有关使用语言子类型的详细信息，请参阅 第 49 页上的 “添加属性子类型”。

下表中列出了受支持的语言子类型。

表 D-2 受支持的语言子类型

语言标记	语言
af	南非荷兰语
be	白俄罗斯语
bg	保加利亚语
ca	加泰罗尼亚语
cs	捷克斯洛伐克语
da	丹麦语
de	德语
el	希腊语
en	英语
es	西班牙语
eu	巴斯克语
fi	芬兰语
fo	法罗语
fr	法语
ga	爱尔兰语
gl	加利西亚语
hr	克罗地亚语
hu	匈牙利语
id	印度尼西亚语
is	冰岛语
it	意大利语
ja	日语
ko	朝鲜语
nl	荷兰语

表 D-2 受支持的语言子类型 *（续）*

语言标记	语言
no	挪威语
pl	波兰语
pt	葡萄牙语
ro	罗马尼亚语
ru	俄语
sk	斯洛伐克语
sl	斯洛文尼亚语
sq	阿尔巴尼亚语
sr	塞尔维亚语
sv	瑞典语
tr	土耳其语
uk	乌克兰语
zh	汉语

术语表

ACI 访问控制指令 (Access Control Instruction)。授予或拒绝目录条目权限的指令。

ACL 访问控制列表 (Access control list)。用于控制目录访问的机制。

CA *请参阅“证书授权机构 (Certificate Authority)”*。

CIR *请参阅“客户启动的复制 (consumer-initiated replication)”*。

CoS 以应用程序不可见的方式在条目间共享属性的方法。

CoS 定义项 (CoS definition entry) 标识正在使用的 CoS 类型。在受它影响的分支下， CoS 定义项存储为 LDAP 子条目。

CoS 模板项 (CoS template entry) 包含共享属性值的列表。

DAP 目录访问协议 (Directory Access Protocol)。为客户机提供目录访问权的 ISO X.500 标准协议。

DIT *请参阅“目录树 (directory tree)”*。

DM *请参阅“目录管理员 (Directory Manager)”*。

DN *请参阅“特异名称 (distinguished name)”*。

DNS 域名系统 (Domain Name System)。供网络计算机将标准 IP 地址 （例如 198.93.93.10）与主机名 （例如 `www.iPlanet.com`）相关联的系统。计算机通常从 DNS 服务器获取特定主机名的 IP 地址，或在系统维护的表中查找该 IP 地址。

DNS 别名 (DNS alias) DNS 别名是 DNS 服务器所知道的、指向另一台主机的主机名，特别是 DNS CNAME 记录。计算机始终有一个真实名称，但可以有一个或多个别名。例如，别名 `www.[yourdomain].[domain]` 可能指向服务器当前所在的、名为 `realthing.[yourdomain].[domain]` 的真实计算机。

HTML 超文本标记语言 (Hypertext Markup Language)。用于万维网文档的格式语言。 HTML 文件是带格式码的纯文本文件。它告诉浏览器 （例如 Netscape Navigator）如何显示文本、如何确定图形位置和表单项，以及如何显示到其它页面的链接。

HTTP 超文本传输协议 (Hypertext Transfer Protocol)。 HTTP 服务器和客户机之间交换信息的方式。

HTTP-NG 新一代超文本传输协议。

HTTPD HTTP 守护程序或服务的缩写，是一种使用 HTTP 协议提供信息的程序。该守护程序或服务通常称为 httpd。

HTTPS HTTP 的安全版本，使用 SSL （安全套接层）来实现。

IP 地址 (IP address) Internet 协议地址。它是由圆点隔开的一组数字，指定计算机在 Internet 上的真实位置 （例如 198.93.93.10）。

ISO 国际标准化组织 (International Standards Organization)。

LDAP 轻型目录访问协议 (Lightweight Directory Access Protocol)。该目录服务协议设计运行于 TCP/IP，并可跨多个平台运行。

LDAP URL 提供了一种使用 DNS 来定位目录服务器，然后通过 LDAP 来完成查询的方法。 LDAP URL 的示例为 `ldap://ldap.iplanet.com`。

LDAP 客户机 (LDAP client) 用于从 LDAP 目录服务器请求和查看 LDAP 条目的软件。另请参阅 *“浏览器 (browser)”*。

LDAP 数据交换格式 (LDAP Data Interchange Format) *请参阅“LDAP 数据交换格式 (LDAP Data Interchange Format)”*。

LDAPv3 LDAP 协议的版本 3。iPlanet Directory Server 的模式格式即基于该协议版本。

LDBM 数据库 (LDBM database) 一种基于磁盘的高性能数据库，由一组包含分配到该数据库的所有数据的大文件组成。主要数据存储于 iPlanet Directory Server 中。

LDIF LDAP 数据交换格式 (LDAP Data Interchange Format)。用于以文本形式表示 Directory Server 条目的格式。

MD5 RSA Data Security,Inc. 的消息摘要算法，可用于生成短的数据摘要，这十分有可能是唯一的，且在数学上很难生成可产生相同消息摘要的数据块。

MD5 签名 (MD5 signature) 由 MD5 算法生成的消息摘要。

MIB 管理信息库 (Management Information Base)。与 SNMP 网络相关联的所有数据或其中任何一部分。可以将 MIB 看作是包含所有 SNMP 受管理对象定义的数据库。 MIB 具有树形分层结构，其顶级包含网络的最常用信息，以下各级则分别处理特定的独立网络区域。

MIB 名称空间 (MIB namespace) 管理信息名称空间。一种命名和引用目录数据的方法。也称为目录树。

n + 1 目录问题 (n + 1 directory problem) 也就是管理不同目录中相同信息的多个实例的问题。它会导致硬件成本和人力成本的增加。

NIS 网络信息服务 (Network Information Service)。程序和数据文件系统，供 Unix 计算机用于收集、比较和共享有关计算机网络中计算机、用户、文件系统和网络参数的特定信息。

NMS 网络管理工作站 (Network Management Station)。安装有一个或多个网络管理应用程序的强大工作站。

ns-slapd iPlanet 的 LDAP 目录服务器守护程序或服务，负责 Directory Server 的所有操作。另请参阅 “slapd”。

OID *请参阅 “对象标识符 (object identifier)”*。

PDU 协议数据单元 (Protocol Data Unit)。构成 SNMP 设备之间数据交换基础的编码消息。

PTA 传递验证 (Pass-through authentication)。一个目录服务器向另一个服务器进行查询，从而检查绑定凭证的机制。

PTA LDAP URL 在传递验证中，定义验证目录服务器 (authenticating directory server)、传递子树和可选参数的 URL。

PTA 目录服务器 (PTA directory server) 在传递验证 (PTA) 中， PTA 目录服务器就是将其收到的绑定请求发送（传递）到验证目录服务器 (authenticating directory server) 的服务器。

RAM 随机存储器 (Random access memory)。计算机中基于半导体的物理内存。关闭计算机时， RAM 中存储的信息会丢失。

RDN 相关特异名称 (Relative distinguished name)。在将原始条目添加到字符串中以构成完整的特异名称之前，实际条目本身的名称。

RFC 请求注解 (Request For Comments)。提交给 Internet 社区的程序或标准文档。在技术被接纳为标准以前，任何人都可以对该技术发表自己的看法。

root Unix 计算机上具有最高权限的用户。 root 用户对计算机上所有文件都具有完全访问权限。

SIE 服务器实例项 (Server Instance Entry)。

SIR *请参阅 “供给器启动的复制 (supplier-initiated replication)”*。

slapd LDAP 目录服务器守护程序或服务，负责除复制以外目录的大部分功能。另请参阅 *“ns-slapd”*。

SNMP 简单网络管理协议 (Simple Network Management Protocol)。通过交换有关网络活动的数据，从而用于监控和管理服务器上运行的应用程序进程。

SNMP 主代理 (SNMP master agent) 在各种子代理和 NMS 之间交换信息的软件。

SNMP 子代理 (SNMP subagent) 收集有关受管理设备的信息并将信息传递到主代理的软件。

SSL 安全套接层 (Secure Sockets Layer)。在双方 （客户机和服务器）之间建立安全连接的软件库，用于实现 HTTPS （HTTP 的安全版本）。

TCP/IP 传输控制协议 / INTERNET 协议。用于 Internet 和企业 （公司）网络的主要网络协议。

TLS 传输层安全协议 (Transport Layer Security)。安全套接层的新标准，是一种基于公共密匙的协议。

uid 与 Unix 系统上每个用户相关联的唯一编号。

URL 统一资源定位符 (Uniform Resource Locator)。服务器和客户机请求文档时所用的寻址系统。通常称为位置。 URL 的格式为 `[protocol]://[machine:port]/[document]`。端口号仅在选定服务器的情况下是必需项，通常由服务器指定，从而免除用户在 URL 中必须填写的麻烦。

X.500 标准 (X.500 standard) TISO/ITU-T 文档集，概要介绍推荐用于目录服务器的信息模型、对象类和属性。

安全套接层 (Secure Sockets Layer) *请参阅 “SSL ”*。

绑定 DN (bind DN) 在执行操作时，用于到 Directory Server 验证的特异名称。

绑定规则 (bind rule) 在访问控制环境下，绑定规则指定某特定用户或客户机为取得目录信息的访问权所必须具有的凭证和必须满足的条件。

绑定特异名称 (bind distinguished name) *请参阅 “绑定 DN (bind DN) ”*。

标准索引 (standard index) 默认维护的索引。

参照完整性 (referential integrity) 确保保持目录中相关条目关系的一种机制。

操作属性 (operational attribute) 操作属性包含目录用于在内部跟踪修改和子树属性的信息。除非明确请求，否则响应搜索时不会返回操作属性。

常规访问 (general access) 授予该权限时，表示所有已验证的用户都可以访问目录信息。

超级用户 (superuser) Unix 计算机上拥有最高权限的用户 （也称为 root 用户）。超级用户对计算机上的所有文件都有完全的访问权限。

传递验证 (pass-through authentication) *请参阅 “PTA ”*。

传递子树 (pass-through subtree) 在传递验证中，PTA 目录服务器 (PTA directory server) 将把绑定请求从该子树中包含其 DN 的所有客户机传递到验证目录服务器 (authenticating directory server)。

传输层安全协议 (Transport Layer Security) *请参阅 “TLS ”*。

存在索引 (presence index) 允许搜索包含特定索引属性的条目。

代理 DN (proxy DN) 与代理验证一同使用。对于客户机应用程序正试图在上面执行操作的目标而言，代理 DN 就是对该目标具有访问权限的条目的 DN。

代理验证 (proxy authorization) 一种特殊的验证方式，使用自身的标识绑定到目录的用户通过代理验证被授予其他用户的权限。这个 “其他用户” 是指代理用户，它的 DN 是代理 DN。

代码页 (code page) 由国际化插件环境中的区域设置使用的内部表。该国际化插件供操作系统用来建立键盘按键与字符字体屏幕显示之间的关系。

单原版复制 (single-master replication) 一种最基本的复制环境，其中两个服务器各自持有客户服务器的同一读写副本的拷贝。在单原版复制环境中，供给服务器维护更改日志。

等同索引 (equality index) 帮助用户有效地搜索包含特定属性值的条目。

典型 CoS (classic CoS) 典型 CoS 通过其 DN 和目标条目中某个属性的值来识别模板项。

定义项 (definition entry) *请参阅 “CoS 定义项 (CoS definition entry)”*。

对称加密 (symmetric encryption) 加密和解密都使用相同密匙的加密算法。对称加密算法的示例之一就是 DES。

对象标识符 (object identifier) 在面向对象的系统中唯一标识某个模式元素 （例如对象类或属性）的字符串，通常为十进制数字。对象标识符由 ANSI、 IETF 或类似机构分配。

对象类 (object class) 通过定义条目中所含的属性来定义目录中的条目类型。

对照顺序 (collation order) 提供有关如何对给定语言的字符进行排序的语言和特定文化信息。该信息可能包括字母表中的字母序列或如何对比带音质符号的字母与不带音质符号的字母。

多路复用器 (multiplexor) 包含用于与远程服务器进行通讯的数据库链接的服务器。

多原版复制 (multi-master replication) 一种高级的复制环境。在该环境中，两个服务器各有一套相同的读写副本的拷贝。每台服务器都会维护该副本的更改日志。一台服务器上所作的修改将被自动复制到另一台服务器上。如果出现冲突，则使用时间戳来确定持有最新版本的服务器。

访问控制列表 (access control list) *请参阅 “ACL”*。

访问控制指令 (access control instruction) *请参阅 “ACL”*。

访问权限 (access rights) 在访问控制环境中，指定授予或拒绝访问的级别。访问权限与可在目录上执行的操作类型有关。可以授予或拒绝下列权限：读取、写入、添加、删除、搜索、比较、自写、代理及全部权限。

分支项 (branch entry) 目录中表示子树顶部的条目。

服务类 (class of service) *请参阅 “CoS ”*。

服务器 root 目录 (server root) 服务器上的目录，专门用于保留服务器程序和配置、维护和信息文件。

服务器守护程序 (server daemon) 服务器守护程序是在运行后即侦听和接纳客户机请求的进程。

服务器选择器 (Server Selector) 允许利用浏览器来选择和配置服务器的接口。

副本 (replica) 参与复制的数据库。另请参阅 “客户副本 (consumer replica)”和“供给器副本 (supplier replica)”。

复制 (replication) 将目录树或子树从供给服务器拷贝到客户服务器的操作。

复制协议 (replication agreement) 存储在供给服务器上的配置参数组，可识别所要复制的数据库、要将数据发送到其中的客户服务器、复制发生的时间、将供给器绑定到客户服务器所用的 DN 和凭证，以及保证连接安全的方式等。

父访问权限 (parent access) 授予该权限时，表示用户具有访问目录树中自己条目的下级条目的权限 （如果绑定 DN 是目标条目的父项）。

根后缀 (root suffix) 一个或多个子后缀的父项。目录树中可包含多个根后缀。

更改日志 (change log) 更改日志是对发生在副本上的修改的记录。供给服务器随即会将这些修改内容在存储于客户服务器的副本上进行重现。在多原版复制的情况下，则在其它原版上进行重现。

供给服务器 (supplier server) 在复制环境中，如果某服务器持有被复制到另一服务器的副本，则称该服务器为此副本的供给器。

供给器 (supplier) 包含复制到客户服务器上的目录树或子树原版副本的服务器。

供给器启动的复制 (supplier-initiated replication) 供给器 (supplier) 服务器将目录数据复制到客户服务器的复制配置。

供给器副本 (supplier replica) 包含目录信息原版拷贝可进行更新的副本。一台服务器可持有任意多个原版副本。

管理信息库 (management information base) *请参阅 “MIB ”*。

国际标准化组织 (International Standards Organization) *请参阅“ISO”*。

国际索引 (international index) 加快在国际目录中搜索信息的速度。

过滤器 (filter) 用于目录查询的限制条件。它可以限制返回的信息。

后缀 (suffix) 目录树顶部的条目名，其下存储数据。同一目录中可有多个后缀。每个数据库仅有一个后缀。

货币格式 (monetary format) 指定特定地区所用的货币符号，指定该符号是位于值的前面还是后面，以及如何表示货币单位。

基本 DN (base DN) 基本特异名称。搜索操作的执行针对基本 DN、条目的 DN 及目录树中其下的所有条目。

基本特异名称 (base distinguished name) *请参阅“基本 DN (base DN)”*。

基于角色的属性 (role-based attributes) 因为在相关联的 CoS 模板内拥有特定角色而出现在条目上的属性。

级联复制 (cascading replication) 在级联复制环境中，一台服务器（通常称为中枢供给器）对于特定副本来说将同时用作客户服务器和供给服务器。该服务器持有只读副本并维护更改日志。对于持有数据原版副本的供给服务器而言，它将接收来自该供给服务器的更新，并继而将这些更新提供给客户。

间接 CoS (indirect CoS) 间接 CoS 使用目标项属性中某一属性的值来标识模板项。

简单网络管理协议 (Simple Network Management Protocol) *请参阅“SNMP”*。

角色 (role) 条目分组机制。每个角色都有自己的*成员*，即拥有角色的条目。

近似索引 (approximate index) 允许进行有效的近似或“大致”搜索。

客户 (consumer) 客户服务器，包含来自供给服务器的复制目录树或子树。

客户服务器 (consumer server) 在复制环境中，持有从另一服务器拷贝来的副本的服务器对该副本来说称为客户服务器。

客户副本 (consumer replica) 客户副本将所有的添加和修改操作指向原版副本。一台服务器可持有任意多个客户副本。

客户机 (client) *请参阅“LDAP 客户机 (LDAP client)”*。

客户启动的复制 (consumer-initiated replication) 一种复制配置，在该过程中，客户 (consumer) 服务器将从供给服务器中获取目录数据。

口令策略 (password policy) 一组控制口令在给定目录中使用方式的规则。

口令文件 (password file) Unix 计算机上的一个文件，用于存储 Unix 用户的登录名、口令和用户 ID 号。由于其保存的位置，也称为 `/etc/passwd`。

类别定义 (class definition) 指定创建特定对象实例所需的信息，同时决定该对象相对于目录中其它对象的工作方式。

链接 (chaining) 将请求传递到另一服务器的方法。请求的结果将被系统收集起来并进行编译，然后再返回到客户机。

密文 (ciphertext) 任何人在没有正确密钥以解密信息的情况下都无法读取的加密信息。

名称冲突 (name collisions) 具有相同特异名称的多个条目。

模板项 (template entry) *请参阅 “CoS 模板项 (CoS template entry)”*。

模式 (schema) 描述可存储为目录项的信息类型的定义。如果目录中存储的信息与模式不匹配，则试图访问该目录的客户机可能无法显示正确的结果。

模式检查 (schema checking) 确保目录中添加或修改的条目符合所定义的模式。默认情况下模式检查处于打开状态。如果用户试图保存不符合模式的条目，则会收到错误消息。

目标 (target) 在访问控制环境中，目标标识特定 ACI 所适用的目录信息。

目标项 (target entry) CoS 范围内的条目。

目录访问协议 (Directory Access Protocol) *请参阅 “DAP”*。

目录服务 (directory service) 一种数据库应用程序，设计用于管理一个组织的人员和资源的、基于属性的说明性信息。

目录服务器控制台 (Directory Server Console) 一个 LDAP 客户机应用程序，提供图形用户界面来浏览、配置和管理目录内容。它是 iPlanet Directory Server 产品的一个组件。

目录管理员 (Directory Manager) 有特权的数据库管理员，类似于 UNIX 上的 root 用户。访问控制对目录管理员不适用。

目录树 (directory tree) 存储在目录中的信息的逻辑表示。它反映大多数文件系统使用的目录树模式，树的根点出现在分层结构的顶部。也称为 “DIT”。

匿名访问 (anonymous access) 授予该权限时，允许任何人在无需提供凭证且不必考虑绑定条件的情况下访问目录信息。

匹配规则 (matching rule) 为服务器在搜索操作过程中如何比较字符串提供准则。在国际搜索中，匹配规则告知服务器所用的对照顺序及运算符。

嵌套的角色 (nested role) 允许创建包含其它角色的角色。

轻型目录访问协议 (Lightweight Directory Access Protocol) *请参阅 “LDAP”*。

区域设置 (locale) 标识对照顺序、字符类型、货币格式和时间/日期格式，用来为特定地区、特定文化和/或特定习惯的用户提供数据显示方式。这包括有关如何解释、存储或对照给定语言数据的信息。区域设置还指示用于代表给定语言的代码页。

权限 (permission) 在访问控制环境中，权限表明是否授予或拒绝目录信息的访问权限，以及授予或拒绝的访问权限级别。请参阅 “访问权限 (access rights)”。

缺省索引 (default index) 为每个数据库实例创建的默认索引集中的一个索引。缺省索引可以被删除，但删除前必须谨慎，因为可能有某些插件依赖于它们。

时间/日期格式 (time/date format) 表示特定区域内时间和日期的惯用格式。

守护程序 (daemon) Unix 计算机上负责特定系统任务的后台进程。守护程序进程无须人工干预即可连续工作。

受管理的对象 (managed object) SNMP 代理可访问并发送到 NMS 的标准值。每个受管理对象都用一个正式名称和一个以圆点记号表示的数字标识符来识别。

受管理的角色 (managed role) 用于创建明确枚举成员列表。

属性 (attribute) 含有关于条目的说明信息。属性具有标签和值。对于可存储为属性值的信息类型，每个属性还遵守其标准语法。

属性列表 (attribute list) 给定条目类型或对象类的必需和可选属性列表。

数据库链接 (database link) 链接的一种实现。数据库链接的行为方式类似数据库，但无永久存储。相反，它指向远程存储的数据。

索引关键字 (index key) 目录使用的每一索引由索引关键字和匹配条目 ID 列表组成。

所有 ID 令牌 (All IDs token) 使服务器认为所有目录项均与索引关键字匹配的机制。实际上，“所有 ID 令牌”使服务器表现为似乎没有索引可用于搜索请求。

所有 ID 阈值 (All IDs Threshold) 在全局范围内应用于受服务器管理的每个索引关键字的大小限制。当单个 ID 列表的大小达到该限制值时，服务器将用 “所有 ID 令牌”替换该 ID 列表。

特异名称 (distinguished name) 表示条目名称和在 LDAP 目录中位置的字符串。

条目 (entry) LDIF 文件中的若干行，其中包含关于对象的信息。

条目 ID 列表 (entry ID list) 目录使用的每一索引由索引关键字和匹配条目 ID 列表组成。目录使用条目 ID 列表来建立可能匹配客户机应用程序搜索请求的候选条目列表。

条目分布 (entry distribution) 将目录条目分配给多台服务器以支持大量条目的方法。

拓扑结构 (topology) 多台物理服务器分割一棵目录树的方式，以及这些服务器之间如何相互链接。

网络管理工作站 (network management station) *请参阅 “NMS ”*。

网络管理应用程序 (network management application) 网络管理工作站组件，以图形方式显示关于 SNMP 受管理设备的信息 （哪台设备处于打开或关闭状态，收到哪些错误信息、收到多少错误信息等）。

文件扩展名 (file extension) 文件名中位于句号或圆点 (.) 后面的部分，通常用来定义文件类型 （例如 .GIF 和 .HTML）。在文件名 `index.html` 中，文件扩展名为 `html`。

文件类型 (file type) 给定文件的格式。例如，图像文件通常保存为 GIF 格式，而文本文件通常保存为 ASCII 文本格式。文件类型通常由文件扩展名来标识 （例如 .GIF 或 .HTML）。

系统索引 (system index) 无法删除或修改，因为它对 iPlanet Directory Server 操作至关重要。

相关特异名称 (relative distinguished name) *请参阅 “RDN ”*。

协议 (protocol) 一组描述网络上设备如何交换信息的规则。

协议数据单元 (protocol data unit) *请参阅 “PDU”*。

虚拟列表视图索引 (virtual list view index) 也称为浏览索引，用于加速 Directory Server Console 中条目的显示。可在目录树中的任意分支点上创建虚拟列表视图索引，从而提高显示性能。

验证 (authentication) (1) 向 Directory Server 证实客户机用户身份的过程。为了获得目录访问权，用户必须提供绑定 DN 以及相应的口令或证书。基于目录管理员授予用户的权限， Directory Server 允许该用户执行某些功能或访问文件和目录。

(2) 允许客户机 (client) 确认它们已连接到安全服务器，从而防止其它计算机模拟该服务器，或者在不安全时试图显示为安全服务器。

验证目录服务器 (authenticating directory server) 在传递验证 (PTA) 中，验证目录服务器是指包含发出请求的客户机验证凭证的目录服务器。启用 PTA 的主机将从客户机中接收到的 PTA 请求发送给主机。

验证证书 (authentication certificate) 由第三方发放的、不可转让且不可伪造的数字文件。为核查和验证另一方，需要在服务器和客户机之间相互传送验证证书。

叶条目 (leaf entry) 其下无其它条目的条目。叶条目不能是目录树的分支点。

已过滤的角色 (filtered role) 允许用户根据每个条目包含的属性为条目分配角色。这是通过指定 LDAP 过滤器来实现的。与过滤器相匹配的条目即被视为拥有该角色。

引荐 (referral) (1) 当服务器从 LDAP 客户机收到其无法处理的搜索或更新请求时，通常会向客户机返回一个指向可以处理这些请求的 LDAP 服务器的指针。
(2) 在复制环境下，当客户副本收到更新请求时，它会将该请求转发给持有相应原版副本的服务器。该转发过程称为引荐。

映射树 (mapping tree) 一种将后缀 （子树）名称与数据库相关联的数据结构。

原版数据服务器 (data master) 作为特定数据块原版来源的服务器。

帐户去活 (account inactivation) 禁用用户帐户、组帐户或整个域，从而自动拒绝所有验证尝试。

证书 (certificate) 将网络用户的公共密钥与其目录 DN 相关联的数据集合。证书在目录中作为用户对象属性进行存储。

证书授权机构 (Certificate Authority) 销售和发放验证证书的公司或机构。您可以从自己信任的证书授权机构购买验证证书。也称为 CA。

知识参考 (knowledge reference) 指向存储在不同数据库中的目录信息的指针。

指针 CoS (pointer CoS) 指针 CoS 仅使用模板 DN 标识模板项。

中枢供给器 (hub supplier) 在复制环境中，持有从另一服务器拷贝的副本，并可反过来将该副本复制到第三方服务器中的服务器。另请参阅“级联复制 (cascading replication)”。

主代理 (master agent) *请参阅“SNMP 主代理 (SNMP master agent)”*。

主机名 (hostname) machine.domain.dom 格式的计算机名称，它将被转化为 IP 地址。例如，`www.iPlanet.com` 代表 com 域、 iPlanet 子域中的计算机 www。

子代理 (subagent) *请参阅“SNMP 子代理 (SNMP subagent)”*。

子后缀 (sub suffix) 根后缀下的分支。

子字符串索引 (substring index) 允许按条目中的子字符串进行有效搜索。对于每个条目而言，子字符串索引被限制为最少两个字符。

自访问 (self access) 授予该权限时，表示用户有权访问自己的条目 （如果绑定 DN 与目标条目相匹配）。

字符类型 (character type) 将字母字符与数字或其它字符以及从大写字母到小写字母的映射区分开来。

浏览器 (browser) 诸如 Netscape Navigator 的软件，用于请求和查看万维网上作为 HTML 文件存储的资料。浏览器使用 HTTP 协议与主机服务器通信。

浏览索引 (browsing index) 也称虚拟视图索引，用于加速显示 Directory Server Console 中的条目。可在目录树的任意分支点上创建浏览索引，从而提高显示性能。

索引

字母

A

B

C

D

E

F

G

H

J

K

L

M

N

P

Q

R

S

T

W

X

Y

Z